Yohei Murakami
Kosaku Kimura *Editors*

Human-Centered Services Computing for Smart Cities

IEICE Monograph

EiC

OPEN ACCESS

Springer

Human-Centered Services Computing for Smart Cities

Yohei Murakami · Kosaku Kimura
Editors

Human-Centered Services Computing for Smart Cities

IEICE Monograph

Springer

Editors
Yohei Murakami
Faculty of Information Science
and Engineering
Ritsumeikan University
Kusatsu, Shiga, Japan

Kosaku Kimura
Artificial Intelligence Laboratory
Fujitsu Research
Fujitsu Limited
Kawasaki, Kanagawa, Japan

ISBN 978-981-97-0778-2 ISBN 978-981-97-0779-9 (eBook)
https://doi.org/10.1007/978-981-97-0779-9

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Paper in this product is recyclable.

Preface

In a smart city, where an urban area is enhanced by cyberspace, services computing is expected to be a key technology that connects components in real space and cyberspace. Traditionally focused on functional requirements, services computing must now embrace the demands of smart cities, where a seamless combination of real-world sensing and responsive feedback under an uncertain environment are crucial. This requires an equal emphasis on non-functional requirements, such as response time and data transfer size, considering the locations and situations where the services are deployed. Furthermore, services need to facilitate smooth and safe interaction with the humans who may be users or providers of the services in smart cities.

This book compiles seven monographs from researchers at the forefront of services computing and artificial intelligence for smart cities, covering service composition, big data analysis, privacy-preserving data processing, human-in-the-loop, and service integrations. This is structured into three thematic parts: service composition, big data analysis, and service integration for smart cities.

The first part describes service compositions for smart cities, where interaction between services and the physical world, including humans, is paramount, unlike services on the Web and clouds. Service compositions for smart cities should consider the physical effect of services shared among users in the common physical space and also the uncertainty of human services. This part introduces a novel framework for verifying service consistency of the physical effects, and a theoretical analysis model and an iterative service design methodology for optimizing the Quality of Service (QoS) of human-in-the-loop service compositions prior to their implementation and deployment in the real world. Furthermore, for advanced intelligent applications in smart cities, this part explores the collaboration of artificial intelligence-based services, called AI services, and human services, presenting a reliable crowdsourced framework that can efficiently aggregate correct feedback from a few experts in low-reliable crowds.

The second part addresses the challenges of big data analytics in smart cities, with a focus on privacy-preserving methodologies. This part proposes a novel architecture that automates the big data analysis process using automatic service composition, and a framework for privacy-preserving data collection and analysis. The former allows users to manipulate data with a standard data analysis process, enhancing scalability, while the latter can deal with sensing errors and interpersonal interactions inherent in smart cities.

The Third part reports human-centered service integration for applications in smart cities. Advanced intelligent applications in smart cities need to non-intrusively coexist with users and interact with multiple individual users. This part discusses the potential of autonomous agents and multi-agent systems, focusing on automated negotiation protocols to build a consensus among various agents and the development of virtual agents that can engage with users, such as the elderly, in a non-intrusive manner.

As we mark a decade since the establishment of the IEICE Technical Committee on Services Computing, we reflect on the collaborative journey with the IEICE Technical Committee on Artificial Intelligence and Knowledge-Based Processing. This book is a testament to that collaboration, and we would like to express our deepest gratitude to IEICE for the opportunity to share these collective insights.

Kusatsu, Japan — Yohei Murakami
Kawasaki, Japan — Kosaku Kimura
November 2023

Contents

Contributors

Katsuhide Fujita Tokyo University of Agriculture and Technology, Tokyo, Japan

Fuyuki Ishikawa National Institute of Informatics, Chiyoda City, Japan

Donghui Lin Faculty of Environmental, Life, Natural Science and Technology, Okayama University, Kita-ku, Okayama, Japan

Yohei Murakami Ritsumeikan University, Kusatsu, Japan

Masahide Nakamura Center of Mathematical and Data Sciences, Kobe University, Kobe, Japan

Incheon Paik School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu, Japan

Yuichi Sei The University of Electro-Communications, Chofu, Japan

Service Composition for Smart Cities

Consistent and Quality-Aware Service Composition in Smart Cities

Fuyuki Ishikawa

Abstract In this chapter, we review our research for dependable service composition for smart cities in both cyber and physical spaces. For the cyber space, given the active investigation on web services or web APIs, we intensively worked on the problem of service composition that explores the "best" combination of available services from different providers. The key point was efficient exploration of enormous combinations both in terms of functional consistency and QoS. For the physical space, we worked on compositions of physical services given the trend of Internet-of-Things (IoT). This direction focuses on consistency of composition as different services make physical effects on multiple users and shared spaces. At the end of this chapter, we discuss the prospect after these past research studies.

1 Introduction

Services computing or service-oriented computing is a paradigm that emerged in the 2000s [1]. There might be different definitions, but the common essence is to make use of "services," that is, components that can be accessed via network by using published API. This allows for the rapid development of new applications by combining existing services, thus focusing more on application requirements rather than implementation details. Such a principle, to bridge the gap between business and IT, had been common in software engineering. The emergence of web-based services enlarged the potentials with easier access methods and a notable number of publicly available services. Nowadays, the use of services has become common even in the closed contexts, e.g., business applications built with the micro-service architecture, and smart home applications built with services provided by Internet-of-Things (IoT) devices. Cloud computing is the most successful application of services computing. The principles of services have been thus serving as the essential foundation in the current and emerging computing paradigms.

F. Ishikawa (✉)
National Institute of Informatics, Chiyoda City, Japan
e-mail: f-ishikawa@nii.ac.jp

Y. Murakami and K. Kimura (eds.), *Human-Centered Services Computing for Smart Cities*, https://doi.org/10.1007/978-981-97-0779-9_1

In the research viewpoint, more challenging visions were investigated to realize automated service selection and composition. There has been enormous effort on automated techniques that focus on dependability aspects such as service price and reliability as well as compatibility of data exchanged. We still see active research in the services computing community in key conferences such as ICSOC (Int'l Conf. on Service-Oriented Computing) and ICWS (Int'l Conf. on Web Services).

This monograph describes and reviews our work in two directions, specifically, dependability of service composition both in the cyber space and the physical space. In the former case, i.e., web and cloud services, the primary challenge was the proper selection of service providers for achieving the best quality, assuming a non-trivial number of candidate providers or service plans. In the latter case, i.e., IoT services, the challenge is consistency as effects of the different services can affect each other or the same user.

The experience on these directions provided excellent opportunities to explore both a functional aspect and a quality aspect of services. Since the research work conducted around 2010–2015, there have been rapid changes in the research trends in the world and also for myself. Now I am focusing more on the software engineering aspect with the industry for automated driving systems and AI systems. However, the experience with services computing made the solid foundation for my research.

In the remainder of this monograph, the research work on web and cloud services will be described in Sect. 2. The work on physical services will then be described in Sect. 3. Finally, retrospective discussion will be given in Sect. 4.

2 Service Composition in Cyber Space

2.1 Background Around 2010

The initiative for web services was actively investigated since its emergence around 2000. It was driven by intensive effort on standard specifications for remote integration of program components via Internet protocols and XML-based formats. Besides the detailed specifications such as SOAP and WSDL, the essential vision was to enable easy, rapid, and flexible realization of application goals by combining services published in the network, especially the web into a composite service, called service-oriented computing or services computing [1].

Although there were some efforts to fully automatically compose a service, given the input and the designated output and effect, this problem was too difficult, especially in terms of feasibility. It was because this full service composition requires that candidate component services have formal description of their functions to allow the planning task, i.e., description of input, precondition, output, and effect in a logical language with shared ontology.

Therefore, the most common problem setting was service selection in a workflow or business process of service composition. The assumption is that the workflow is

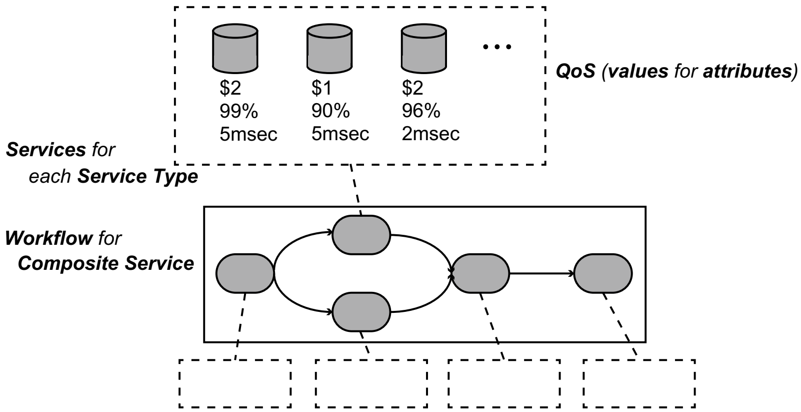


Fig. 1 Quality-aware service selection

given by human users, e.g., get an article from a news retrieval function and send it to a translation function. But there are many candidate services for the involved functions, then there is a problem of how to select from them. This problem is computationally hard when we assume there are an enormous number of possible combinations of candidate services. There have been very active research studies since the first representative work of QoS-aware service selection [2] in 2003.

The standard problem of quality-aware service selection is described in Fig. 1. In the figure, a workflow is shown with sequential and parallel execution. For each service type, or a task in the workflow, there are multiple services as candidates. We distinguish these services by quality of service (QoS) such as price, availability, and response time. We can use the term SLA, service-level agreement, to refer to the QoS values that should be ensured by the providers. We consider the aggregated QoS of the workflow. For example, the aggregated price for the composite service can be calculated as the sum of the price values of the involved services, assuming all are executed in each invocation. Similarly, the aggregated reliability for the composite service can be calculated as the product of the reliability values of the involved services.

As the most simple form, the baseline problem of quality-aware service description can be described. Note that we use simplified formalization for the illustration purpose in this chapter and the definitions may differ from those in the original papers.

Definition 1. Quality-Aware Service Composition

Given a set of service candidates for each task or service type required in the workflow, we choose one of the candidates to maximize the overall quality of the workflow.

$$\texttt{max OverallQuality}(services)$$

where $services = [s_1, s_2, \cdots, s_N]$ with $s_i \in SC(i)$, N is the number of service types or tasks necessary in the workflow, and $SC(i)$ is the given service candidates for each service type.

`OverallQuality` of the workflow is obtained by integrating each of the quality aspects $q \in Q$ by a weighted sum with the weights of each q as $w(q)$, as $\sum w(q) = 1$:

$$\texttt{OverallQuality}(services) = \sum_{q \in Q} w(q)\texttt{Aggregate}(services, q)$$

The `Aggregate` function depends on the quality aspect. For the price, it is a sum of the price values for each selected service, made negative (as we "maximize" the quality):

$$\texttt{Aggregate}(services, \texttt{price}) = -\sum_{t \in ST} \texttt{price}(services(t))$$

where ST is the set of service types in the workflow and $services(t)$ is the selected service for a service type t.

As another example, for the availability, the overall availability of the workflow is a product of that value for each service when we use all of the services, i.e., if we do not involve alternative services:

$$\texttt{Aggregate}(services, \texttt{availability}) = \prod_{t \in ST} \texttt{availability}(services(t))$$

2.2 Different Quality Aspects in Service Selection

We had intensive research studies on quality-aware service selection on the web. The direction was to involve practical aspects into the standard problem of quality-aware service selection and also investigate technical solutions for the extended problems, which are more computationally intensive. Below we overview how the baseline problem was extended.

2.2.1 Probabilistic Selection

The work in [3] considered conditional contracts and usage patterns during the service selection. For example, the SLA may declare the ensured response time differs during the working hours, e.g., 9am–5pm weekdays. On the other hand, the client side, who is going to make the service selection, also has usage patterns, e.g., often use the services during the night time for batch processing.

The baseline problem in Sect. 2.1 is extended so that the atomic quality values of each service, such as `price(service(t))`, is now not a static constant but an expected value, which may be obtained by simulation for example.

2.2.2 Combined Use of Functionally Equivalent Services

The work in [4] considered using multiple services for one service type. For example, we may keep two service candidates for one service type, and invoke the second one when the first one does not respond. Or, we may invoke multiple services and adopt the fastest response. By considering such combined usages, we can make additional virtual service candidates for each service type.

The baseline problem in Sect. 2.1 is extended by changing the way of making the sets of candidate services. Given the original service candidates $SC(i)$ for the service type i, we can extend the candidates with combined services:

$$SC'(i) = \bigcup_{ss \subseteq SC(i)} \texttt{combine}(ss)$$

where `combine` makes different ways of aggregation of functionally equivalent services. The quality functions such as `price` are extended as well to handle the combined services, e.g., sum of price and minimum of response time in the case of parallel composition.

2.2.3 Different Granularity

The work in [5] considered with different granularity of service functions. For example, suppose there are two successive service types of "English newspaper download" and "translation to Japanese." One service may work for the two service types if it provides "Japanese version download of English newspaper." The mathematical representation changes from the baseline one in Sect. 2.1 to select a service sequence for the whole workflow, not a service for each service type.

2.2.4 Network Quality and Location Awareness

The studies in [6–8] considered network quality and location awareness. One aspect is the latency between services that may matter in data-intensive workflow. The other aspect is the location diversity for higher availability when we consider backup scenarios when some of the best services are unavailable.

For the network latency aspect, we can extend the baseline problem in Sect. 2.1 by including the network quality in the optimization target:

$$\texttt{max OverallQuality}(services) + w_{NET}\texttt{OverallNetworkQuality}(services)$$

where w_{NET} refers to a weight to decide the balance of the service quality and network quality.

Here, for $services = [s_1, s_2, \cdots, s_N]$,

$$\texttt{OverallNetworkQuality}(services) = \sum_{i \in [0,N)} \texttt{Latency}(s_i, s_{i+1})$$

The second setting of location diversity will be discussed in 2.4.

2.3 Self-Adaptive Network-Aware Service Selection

As the concrete work, the work in [7] is described briefly. This work considered network awareness or location awareness by integrating the network latency and transfer rate into the service composition. Although the standard QoS of each service includes the execution time, the actual response time is affected by the network latency, especially for data-intensive applications. It is therefore essential to consider this aspect in service selection, i.e., sometimes it can make sense to choose services nearby.

We employed a network model from the network research but also made a custom genetic algorithm for service selection. Specifically,

- A mutation operator is used to make a random change in the current solution candidate in the evolutionary process of genetic algorithms. We made a custom mutation operator that replaces a service candidate selected in a current solution by another candidate nearby.
- A crossover operator is used to make a new solution from two parent solutions in the evolutionary process of genetic algorithms. We made a custom crossover operator that tries to "smoothen" the network flow. Figure 2 shows this process. The service locations are mapped on the two-dimensional coordinates and we start with the two parents (the leftmost) that were chosen to create a new solution, called offspring. To select the service for the i-th task, we look at the middle location of the services of the $i + 1$-th task from the two parents.
- These custom operators and standard ones are used in an adaptive way by updating the probabilities of each operator during the evolutionary process.
- Specific data structures were used to efficiently make the above query on locations such as a K-D tree.

Figure 3 shows an example of the evaluation results of the technique for network-aware service selection. If we put extremely a high weight on the network latency,

Fig. 2 Custom crossover operator in network-aware service selection (cited from [7])

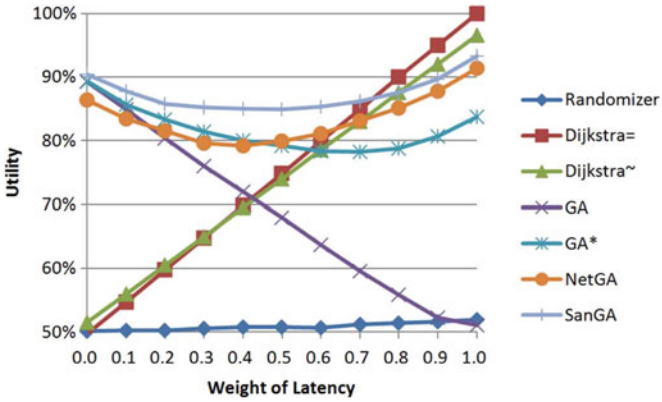


Fig. 3 Example of evaluation on network-aware service selection (cited from [7])

on the right side of the graph, the optimal path selection by Diikstra algorithms, not considering QoS, is slightly better. But otherwise, our technique of SanGA outperforms the other approaches, including a genetic algorithm with straightforward network-awareness (GA*).[1]

2.4 Consistency in Service Selection

2.4.1 Problem Setting

Although there were a large amount of studies of quality-aware service selection, the limitation was assumption on the exactly identical functions of candidate services, i.e., all are compatible if the target task is the same. It is necessary to consider the consistency or compatibility of slightly different output-input connection.

In addition, the typical setting of service selection did not consider the failure. It is of course possible to employ an adaptive mechanism at runtime to search for an alternative service after detecting a service failure. However, this may not be optimal, for example, when a service with no good alternative was selected. This

[1] GA is a genetic algorithm without considering network quality, NetGA is from our previous work.

makes a similar extension to the combined use of functionally equivalent services in 2.2.2, but now we think of functionally *compatible* services as well.

These aspects were handled in the work in [8, 9]. We select a list of service candidates for each service type so that at runtime we can switch between them when the primary one is unavailable and the quality of such backup plans can be explored in a probabilistic way during the selection procedure.

The baseline problem in Sect. 2.1 is now extended to select a list of candidate services for each service type:

$$\texttt{max ExpectedOverallQuality}(services_{backup})$$

where $services_{backup} = [S_1, S_2, \cdots, S_n]$ and S_i refers to a list of service candidates for i-th service type.

We consider the compatibility constraint or the possibility that available services for the same service type may have slightly different interfaces. The selected service candidates $[S_1, S_2, \cdots, S_N]$ must satisfy $\forall s_i \in S_i, s_{i+1} \in S_{i+1}\,.\,\texttt{Compatible}(s_i, s_{i+1})$. The compatibly may be defined with the semantic web technique that uses formal ontology, or at the minimum with common semantics of programming languages, e.g., we can pass an integer output to a float input.

The quality is now considered as an expected value by considering the availability as the success probability of each service. For example, given a service candidate list $[s_{i1}, s_{i2}]$, the expected price for this service type is $p_{i1}\texttt{PRICE}(s_{i1}) + (1 - p_{i1})p_{i2}\texttt{PRICE}(s_{i2})$ where p_{i1}, $pi2$ are the success probabilities for the candidate services. Note that this is a simplified version as we also employed a location-aware availability model, e.g., consider the fact that services in the same datacenter are likely to become unavailable at the same time.

2.4.2 Proposed Methods

To effectively deal with the compatibility aspect, we employed a clustering approach to efficiently traverse compatible services [10]. Figure 4 shows how the selection problem is modified to deal with the functional consistency problem. For each service type, candidate services are organized in clusters with the compatibility relations between services. For example, $S6$ and $S7$ can be used as alternatives of the currently chosen one, $S5$, which intuitively means they require the same or less input and produce the same or more output.

We also developed a custom genetic algorithm with the following features:

- The QoS values are calculated in a probabilistic way, i.e., as the expected value, by considering the reliability of each service candidate.
- In order to assess the reliability, locations of service candidates are considered, i.e., service candidates in the same region can fail at the same time.
- Custom mutation and crossover operators are used to prioritize service candidates with more location diversity.

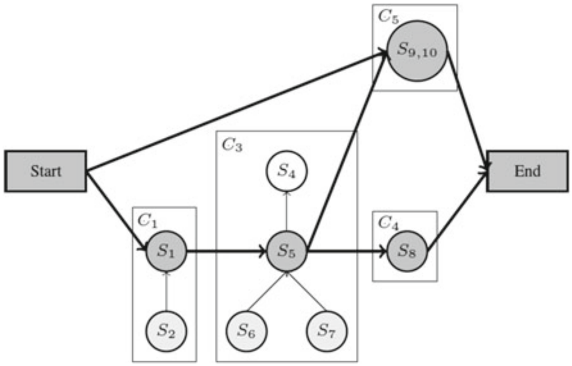


Fig. 4 Service selection with functional consistency (cited from [10])

- A custom step is added in the evolutionary process in which incompatible combinations of services are sometimes replaced with compatible ones. This computation is efficiently done with the cluster structure as shown in Fig. 4.

Figure 5 shows the tool interface for this extended service selection. QoS values are shown with backup plans and location diversity is explored. The proposed algorithm uses multi-objective optimization to allow for producing the Pareto-front solutions, i.e., solutions with different prioritization over multiple evaluation criteria. Users can choose among the solutions such as "the best quality in the normal plan but poor in backup plans" or "so-so quality in either of normal or backup plans."

Figure 6 shows an example of evaluation result of the custom algorithm (SHUURI and SHUURI_2). The problem becomes more difficult when the service compatibility is more limited (the horizontal axis) and the proposed algorithm, SHUURI_2, outperforms in the optimization performance measured by hypervolume, a common criterion to evaluate Pareto-front solutions.

2.5 Service Selection in Cloud Computing

Cloud computing emerged as the new paradigm after the trend of services computing. The problem of selecting infrastructure services for computational resources also emerged as the central problem as practical cloud services offer many plans with different qualities such as CPU speed and memory size even inside one service provider. We also investigated algorithms for selecting cloud services. The work in [11] considered cloud service selection for workflow applications with deadline constraints by extending ant colony optimization algorithms. We also worked on consolidation of virtual machines [12].

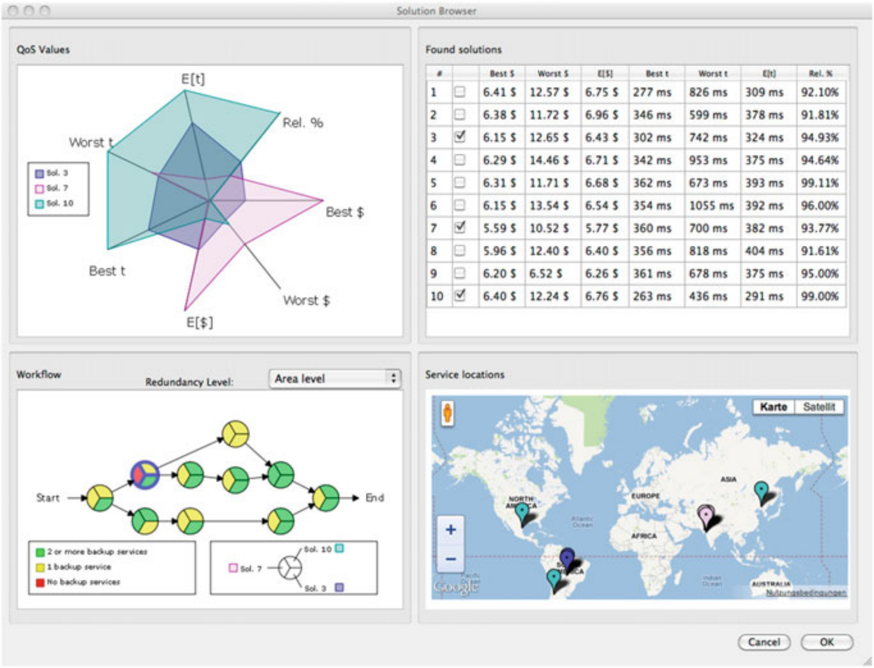


Fig. 5 Tool interface for QoS-aware service selection with backup plans and location diversity (cited from [8])

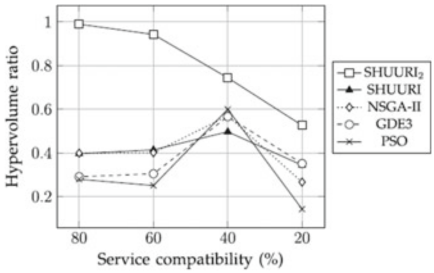


Fig. 6 Example of evaluation on robust and consistent service selection (cited from [8])

3 Service Composition in Physical Space

3.1 Background Around 2015

Besides the intensive work on the web and cloud services, Internet-of-Things (IoT) and smart cities, including smart home, smart office, etc., attracted wide attention in the 2010s. Given the increasing capability of sensors and actuators, more and more applications were investigated as a combination of functions provided by such devices, which can be said service composition in the physical world.

Similar to web service composition, the workflow to combine multiple services is described in a high-level language, e.g., Node-RED.[2] In the case of physical services, the length of the workflow is rather limited and the key characteristic is the event-driven behavior to respond to environmental events, e.g., user movement. Event-based behavior description is also used, rather than workflow-based one, such as sensiNact [13]. With sensiNact, service composition can be specified as ECA rules in the form of "ON *event* IF *condition* DO *action*." Such rules are also called trigger-action programming [14].

3.2 EU-Japan Smart City Projects

We worked in the context of two EU-Japan projects, ClouT and BigClouT.[3] The projects aimed at providing reference architecture and its implementation for making use of web, cloud, and physical services in smart cities. The architecture and its implementation were holistic, covering infrastructure-level, platform-level, and software-level as in the common layers of cloud computing, i.e., we had smart-city versions of IaaS, PaaS, and SaaS integrating not only cloud resources but also sensor and actuator devices as well as human acting as sensors and actuators.

Service composition was one of the key aspects of the City-PaaS in the projects. In addition to the web and cloud service composition mechanisms presented in 2, we investigated supporting tools for physical service composition at development time and runtime.

3.3 Consistency in Physical Service Composition

The essential difference of physical services from web and cloud services are interactions among multiple users and multiple composite applications. In other words, the effect of services can be shared among different users in the same physical place, thus potentially leading to inconsistency or undesirable situations. It is thus necessary to deal with a different type of consistency from that for the web and cloud services.

As a simple scenario, consider a smart office system that supports presentation of slides and electronic posters, demonstration of tools, and discussion in a room (Fig. 7). This system is expected to support both presenters and audiences, often without explicit commands from them while preventing undesirable situations. In this section, a very small part is discussed to quickly illustrate the difficulties with ECA rules.

---

[2] https://nodered.org/.

[3] https://clout-project.eu/, http://bigclout.eu/.

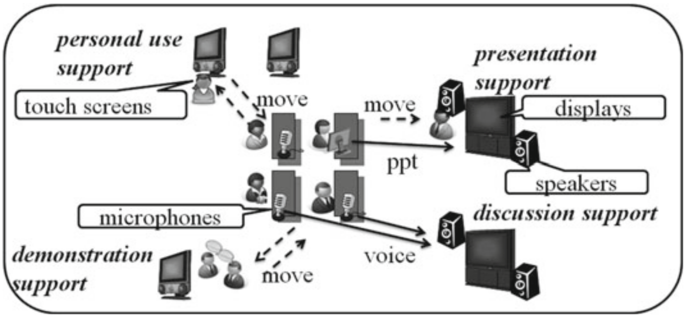


Fig. 7 Example scenario of smart office system

R1 An authorized user can use a display nearby to browse shared business information such as a shared calendar. **B1** Given a request by an authorized user nearby (who is allowed to access the information and *is able to see the display*), the system starts to show the requested information on the display. **R2** Shared business information is never seen by an unauthorized user. **B2** The system stops showing the information when an unauthorized user comes nearby (*and can see the information on the display*).

Fig. 8 Example specifications with potential conflict

An example of specifications of this system is shown in Fig. 8, regarding the simple usage of shared displays. It includes requirements on the system **R1** and **R2**, as well as behavior specification (ECA rules) to meet the requirements, **B1** and **B2**.

The example specifications are not satisfactory in the sense that the set of behavior specifications **B1** and **B2** does not meet requirement **R2**. In fact, behavior **B1** can start to show the information on the display even when there is already an unauthorized user there. This situation means that there is a conflict between **R1** and **R2**, i.e., they cannot be met as they are (without any restrictions). If a decision is made to put higher priority on **R2**, **B1** and **R1** are then modified by adding a constraint: "*only if there is no unauthorized user nearby*".

This conflict is only detected by considering specific test scenarios, either executed in the physical environment, in a simulation model, or even in the engineer's mind. It may be thus overlooked by engineers, and it is essential to have automated, systematic support to detect such scenarios or potentials of conflicts.

3.4 Verification Framework

Our work investigated modeling of physical effects and verification to detect potential conflicts [15–17]. Figure 9 describes the framework. The left side shows three

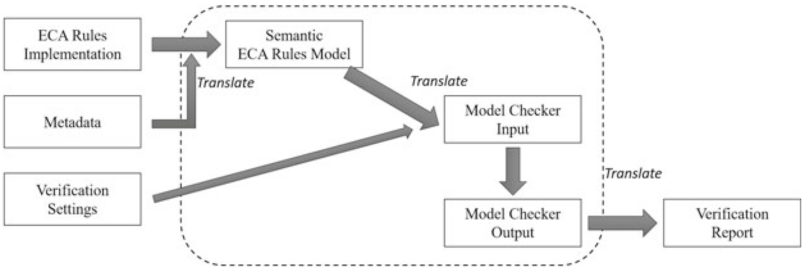


Fig. 9 Overview of proposed framework for consistency verification of physical service composition

elements of the input and the right side shows one element of the output. The dashed rectangle denotes the boundary of the tool: users of the framework do not need to look at its inside. This architecture is defined to bridge the gaps between practical domain-specific representations for smart space applications and required formal inputs for model checkers.

3.4.1 Underlying Formal Modeling

We developed a formal modeling framework to capture the essence of smart space services by abstracting away the implementation details. The core idea is to model the physical effects of services on users, such as "see" and "hear." Such effects may or may not be active for a user depending on whether the user is inside the "scope" of the service, i.e., enough nearby the device.

Figure 10 illustrates a few examples of the abstract formal models as described below.

- The left figure denotes the situation of the example scenario, where a user comes near and is able to see the display that has been activated for another user. This is explained by inclusion of the two users in the scope for visual interaction with the display device.
- The middle figure denotes a situation of sound conflicts, where a user hears different sounds from different audio devices and becomes uncomfortable, e.g., when a movie player is automatically activated while a recipe reader is running in a smart home application. This is explained by inclusion of the user in the overlapping two scopes for audio interaction with the two devices.
- The right figure denotes a situation in which a user sees different direction instructions in a smart museum application. This is explained similarly by inclusion of the user in the two overlapping scopes for two visual devices.

These examples include conflicts that can occur depending on the relationships between users and scopes, i.e., user inclusion, or between scopes, i.e., scope overlap.

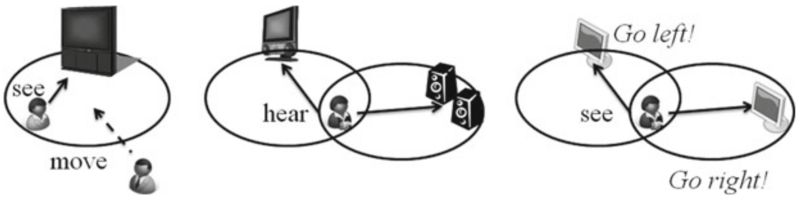


Fig. 10 Scope-based modeling of physical services

By modeling and examining such relationships explicitly, implicit assumptions on device layout or potential conflicts can be clarified.

3.4.2 Verification via Model Checking

Once we have the formal model of the smart space and its services, we can explore the possible state transitions. The state transitions are represented in the abstract form, for example: a user enters a scope of one service; then, the physical effect of the service becomes active; after that another user enters the same scope; finally, the physical effect of the original service is overridden by the newly activated one.

Model checking is an approach to have exhaustive exploration of the possible state transitions for verification [18]. SPIN is one of the popular tools for model checking [19]. The primary input of the SPIN is state transitions to explore and specified by a dedicated language called Promela. The other key input is what we want to verify. This can be given by a command, e.g., we want to detect deadlocks, or by properties specified in temporal logic. Typical properties include safety to show some undesirable state is never reached and liveness to show some desirable states will be eventually reached.

It has been a common approach to prepare a translation mechanism from a language that engineers are familiar with, such as UML or domain-specific languages, into a language used in a model checker, such as Promela. This approach is effective in our context as well. Engineers prefer to describe ECA rules in domain-specific languages and we can support model checking by providing a translation mechanism. We can also provide support typical properties to verify such as conflicts of sounds in the same space.

3.4.3 Integration with sensiNact

We implemented the architecture in Fig. 9 including the transformation function from ECA rules in the sensiNact platform to Promela for the SPIN model checker. In the sensiNact platform, ECA rules are specified with REST APIs. For example, an action part of an ECA rule may refer to invocation of the speaker service as `speakerService1.play.act()`. We need a mapping from this implementation-level description to the formal model. Specifically, we need metadata including the effect of each API, e.g., *AUDIO* as well as scope of the effect, e.g., *Room1*.

3.5 Runtime Adaptation

The verification framework allows for detecting potential inconsistency in applications of physical service composition specified with ECA rules. This task is expected to be conducted at development time by software engineers. As a more advanced use case, we also worked on runtime mechanisms for automated self-adaptation to detect and resolve potential inconsistencies when a new application of physical service composition is deployed by end users.

This runtime adaptation is implemented with the models@run.time approach [20]. In the models@run.time approach, the system makes use of its models used in the development time for monitoring and adaptation. This approach is significant as more and more systems are facing with increasing uncertainty, i.e., we cannot precisely predict all that occurs in the operation in the physical environment, user behavior, or black-box AI behavior.

In our case, we already had a framework for formal modeling and verification that aimed at support engineers at development time. This mechanism can be explored at runtime, for example:

1. The user installs a new application, which is written in the implementation language, e.g., sensiNact, but also accompanies the metadata.
2. The formal model of installed applications and the environment is updated with the new application.
3. Model checking is conducted and a scenario for conflict is detected.
4. The user is asked to fix it by providing priorities on the conflicting applications. We may iterate by going back to Step 3 until all the conflicts are resolved.

The critical difficulty here is the tasks imposed on the end user. One implementation we chose was use of priorities between applications or ECA rules. We can prepare a mechanism to rewrite the ECA rules according to the priority configuration. For example, we can make a modified rule "close the window if it is raining only if the CO2 density of the room is not too high" if the safety app, monitoring the CO2 density, has a higher priority than the comfort app, monitoring the weather.

There can be a variety in how to implement such an adaptation mechanism. For example, we may deploy simple conflict detection that only checks the device state, e.g., open versus close, not looking at the state transitions. This is much more lightweight but may cause too strict check such as reporting "open window" in the morning and "close window" at night as a conflict.

For supporting such variability, we have implemented the adaptation mechanism in a generic way via API. Specifically, the adaptation mechanism is separated as a component and it works with API provided by a platform such as *getCurrentModel*, *addNewRule*, and *checkConsistency*.

4 Retrospective Discussion

4.1 Services Computing

In this monograph, we reviewed our research in the services computing area. One direction was service composition in the web, and it focused on (constrained) optimization problems by assuming a large number of services with different QoS values. The other direction was service composition in smart spaces and it focused on the consistency problem.

Even though both directions focused on the same concept of services, the underlying technical assumption and thus the applied techniques were different. The primary assumption in web services is that services executed by different users do not affect each other. On the other hand, the essential common characteristic is the focus on the application-level goals by abstracting away the implementation detail. QoS aspects are absolutely essential in both types of services though we didn't work on QoS optimization problems in IoT or fog computing [21].

The initial vision of services computing, flexibly combining services provided by various providers in the open network, turned out to be some or less impractical. This is because people did not choose to give rich annotations, even machine-readable description of API for fully automated service selection and composition. However, the vision was successfully employed for cloud computing where the services are simple and standardized or virtualized. In addition, the technical approaches of modeling quality and problem formulation have been leveraged even if we do not consider millions of candidate services. In this sense, contributions are essential from the 20 years of services computing.

4.2 Impact on the Author

The experiences with these two different directions have established the solid research foundation for the author, that is, investigation of application-level dependability goals with different types of automated techniques. The insights obtained in the experiences have helped the author tackle challenges in different domains such as automated driving systems [22–25], automated delivery robots [26–28], and games-as-a-service [29]. We have been making use of optimization techniques as well as formal verification techniques to deal with various quality aspects though the systems are monolithic, and we focus more on the software engineering aspects such as optimization-based test generation. For example, in the problem of automated delivery robots, we are exploring different types of risk, cost, and value metrics by optimization techniques.

5 Concluding Remarks

In this monograph, we have reviewed our research in the services computing area around 2010s. The author believes the past work has contributed to establish the foundation of various current studies such as fog computing and microservices even if the proposed techniques may not fit perfectly with the current practical environments.

The communities of services computing are still very active in Japan and in the world on top of the accumulated insights for engineering of service composition as well as quality modeling and investigation. On the other hand, there have been different approaches to quickly realize application goals such as (monolithic) AI systems including deep learning approaches and large language model (LLM) approaches such as ChatGPT [30]. It is very attractive to discuss the roles and directions of services computing with these emerging approaches.

References

1. Huhns MN, Singh MP (2005) Service-oriented computing: key concepts and principles. IEEE Internet Comput 9(1):75–81
2. Zeng L, Benatallah B, Dumas M, Kalagnanam J, Sheng QZ (2003) Quality driven web services composition. In: The 12th international conference on World Wide Web (WWW 2003). ACM, New York, NY, USA, pp 411–421
3. Klein A, Ishikawa F, Bauer B (2009) A probabilistic approach to service selection with conditional contracts and usage patterns. In: The 7th international conference on service-oriented computing (ICSOC 2009), pp 253–268
4. Hiratsuka N, Ishikawa F, Honiden S (2011) Service selection with combinational use of functionally-equivalent services. In: The 9th IEEE international conference on web services (ICWS 2011), pp 89–96
5. Quanwang W, Ishikawa F, Zhu Q, Shin D-H (2016) QoS-aware multigranularity service composition: modeling and optimization. IEEE Trans Syst Man Cybern Syst 46(11):1565–1577
6. Klein A, Ishikawa F, Honiden S (2012) Towards network-aware service composition in the cloud. In: The 21th international World Wide Web conference (WWW 2012)
7. Klein A, Ishikawa F, Honiden S (2014) SanGA: a self-adaptive network-aware approach to service composition. IEEE Trans Serv Comput 7(3):452–464
8. Wagner F, Ishikawa F, Honiden S (2016) Robust service compositions with functional and location diversity. IEEE Trans Serv Comput 9(2):277–290
9. Wagner F, Kloepper B, Ishikawa F, Honiden S (2012) Towards robust service compositions in the context of functionally diverse services. In: The 21st international World Wide Web conference (WWW 2012) pp 968–978

10. Wagner F, Ishikawa F, Honiden S (2011) QoS-aware automatic service composition by applying functional clustering. In: The 9th IEEE international conference on web services (ICWS 2011), pp 89–96
11. Quanwang W, Ishikawa F, Zhu Q, Xia Y, Wen J (2017) Deadline-constrained cost optimization approaches for workflow scheduling in clouds. IEEE Trans Parallel Distrib Syst 28(12):3401–3412
12. Quanwang W, Ishikawa F, Zhu Q, Xia Y (2019) Energy and migration cost-aware dynamic virtual machine consolidation in heterogeneous cloud datacenters. IEEE Trans Serv Comput 12(4):550–563
13. Cano J, Rutten E, Delaval G, Benazzouz Y, Gürgen L (2014) ECA rules for IoT environment: a case study in safe design. In: Workshop on quality assurance for self-adaptive, self-organising systems (QA4SASO 2014) at SASO 2014, pp 116–121
14. Ur B, McManus E, Pak Yong Ho M, Littman ML (2014) Practical trigger-action programming in the smart home. In: Proceedings of the SIGCHI conference on human factors in computing systems, CHI '14. Association for Computing Machinery, New York, NY, USA, pp 803–812
15. Ishikawa F, Suleiman B, Yamamoto K, Honiden S (2009) Physical interaction in pervasive computing: formal modeling, analysis and verification. In: Proceedings of the 2009 international conference on pervasive services, ICPS '09. Association for Computing Machinery, New York, NY, USA, pp 133–140
16. Yagita M, Ishikawa F, Honiden S (2015) An application conflict detection and resolution system for smart homes. In: International workshop on software engineering for smart cyber-physical systems (SEsCPS 2015). IEEE Press, Piscataway, NJ, USA, pp 33–39
17. Ishikawa F, Honiden S (2016) Chapter 7 verification and configuration of smart space applications. In: Enablers for smart cities. Wiley
18. Clarke EM, Grumberg O, Kroening D, Peled D, Veith H (2018) Model checking, 2nd edn. MIT Press
19. Spin formal verification. https://spinroot.com/spin/whatispin.html
20. Blair G, Bencomo N, France RB (2009) Models@ run.time. IEEE Comput 42(10):22–27
21. Haghi Kashani M, Rahmani AM, Navimipour NJ (2020) Quality of service-aware approaches in fog computing. Int J Commun Syst 33(8)
22. Luo Y, Zhang XY, Arcaini P, Jin Z, Zhao H, Ishikawa F, Wu R, Xie T (2021) Targeting requirements violations of autonomous driving systems by dynamic evolutionary search. In: 2021 36th IEEE/ACM international conference on automated software engineering (ASE), pp 279–291
23. Castellano E, Zhang XY, Arcaini P, Takisaka T, Ishikawa F, Ikehata N, Iwakura K (2023) Explaining the behaviour of game agents using differential comparison. In: Proceedings of the 37th IEEE/ACM international conference on automated software engineering (ASE '22). Association for Computing Machinery, New York, NY, USA
24. Kobayashi T, Bondu M, Ishikawa F (2023) Formal modelling of safety architecture for responsibility-aware autonomous vehicle via Event-B refinement. In: Chechik M, Katoen JP, Leucker M (eds) Formal methods. Springer International Publishing, Cham, pp 533–549
25. Laurent T, Klikovits S, Arcaini P, Ishikawa F, Ventresque A (2023) Parameter coverage for testing of autonomous driving systems under uncertainty. ACM Trans Softw Eng Methodol 32(3)
26. Victorica MB, Arcaini P, Ishikawa F, Kawamoto H, Sawai K, Muramoto E (2023)Stability-aware exploration of design space of autonomous robots for goods delivery. In: The 27th international conference on engineering of complex computer systems (ICECCS 2023)
27. Arcaini P, Castellano E, Ishikawa F, Kawamoto H, Sawai K, Muramoto E (2023)Incremental search-based allocation of autonomous robots for goods delivery. In: IEEE 2023 congress on evolutionary computation (CEC 2023)
28. Bentley PJ, Lim SL, Arcaini P, Ishikawa F (2023) Using a variational autoencoder to learn valid search spaces of safely monitored autonomous robots for last-mile delivery. In: The genetic and evolutionary computation conference (GECCO 2023)

29. Castellano E, Zhang XY, Arcaini P, Takisaka T, Ishikawa F, Ikehata N, Iwakura K (2023) Explaining the behaviour of game agents using differential comparison. In: The 2nd international workshop on automated software engineering for computer games (ASE '22). Association for Computing Machinery, New York, NY, USA
30. Introducing ChatGPT. https://openai.com/blog/chatgpt. Access May 2023

Designing and Analyzing Human-in-the-Loop Service Compositions

Donghui Lin

Abstract To ensure the quality and performance of service compositions in a smart city, combining human services and automated services is expected to be a potential solution in various real-world scenarios. In this monograph, we summarize our research efforts on designing and analyzing human-in-the-loop service compositions, in the practical aspects and theoretical aspects as well. First, we describe how we design a practical human-in-the-loop translation service composition for supporting localization processes and real-world multilingual activities. Then, we propose theoretical crowdsourcing workflow models to study and analyze how human service workflows could achieve optimal performances in various situations.

1 Introduction

1.1 Background

The increasing availability of software and data services on the Internet has expanded the options for designing automated and semi-automated service compositions for application developers and users. When selecting and combining Web services, the quality of service (QoS) is considered a crucial factor. QoS-aware service composition involves defining general QoS attributes such as cost, response time, reputation, and availability [52], which are important for evaluating the non-functional quality of atomic and composite services. Since the early 2000s, QoS-aware service composition has been one of the most active research topics in service-oriented computing. In previous studies, various approaches have been proposed for computing QoS based on multiple attributes [1, 8, 14, 43, 52], focusing on the optimization of the overall non-functional quality of composite services.

D. Lin (✉)
Faculty of Environmental, Life, Natural Science and Technology, Okayama University, 3-1-1 Tsushima-naka, Kita-ku, Okayama 700-8530, Japan
e-mail: lindh@okayama-u.ac.jp

Y. Murakami and K. Kimura (eds.), *Human-Centered Services Computing for Smart Cities*, https://doi.org/10.1007/978-981-97-0779-9_2

On the other hand, application specific quality (functional QoS attributes) may also be crucial in many real-world services. For instance, when it comes to translation services, users are primarily concerned with the translation quality rather than general attributes. Hence, it is necessary to prioritize the optimization of translation quality while also considering non-functional QoS attributes. However, certain crucial functional QoS attributes may not consistently fulfill users' needs due to limitations specific to the application, e.g., it is not always feasible for a machine translation service to deliver flawless translation outcomes to users. Nowadays, this perspective becomes extremely important when considering various artificial intelligence services and machine learning applications in smart cities.

To address the above issue, the integration of Web services and human activities has emerged as a potential solution. While human activities have been extensively studied in the area of business process management, they have primarily been examined from an organizational or resource perspective [41, 56]. These studies have focused on situations where tasks cannot be automated and require human intervention. Since the late 2000s, the rise of crowdsourcing and cloud computing environments has sparked interest in combining human activities with existing services and applications [21, 22].

1.2 Approach

We aim at practicing and analyzing the effect of composing human activities and Web services in real-world scenarios. The human activities in this research involve both crowd workers and professionals. Specifically, we consider human activities from a perspective of QoS which was always neglected in the previous research.

We start by conducting empirical studies on designing and implementing human-in-the-loop service composition. Since 2006, we have been working on the Language Grid [16, 18, 39, 40], a service-oriented language infrastructure, which serves as the fundamentals for our research on service composition. A good example in the language service domain is that translation work can be done by composing various language services on the Language Grid, monolingual crowd workers, and bilingual professionals. In 2010, we conducted a small pilot experiment on translating a manual of digital camera and found that it was promising to combine Web services and human activities [27, 34]. In the following years, we increased the scale of the experiment, designed the human-in-the-loop composite services for supporting localization processes [31, 32], and implemented human-in-the-loop applications for real-world multilingual activities [28–30, 33].

On the other hand, we realize that it is necessary to provide theoretical foundations for designing and optimizing human-in-the-loop service composition. Therefore, we need to model the human activities and analyze how the composite services could achieve optimal performance with human activities. To achieve this goal, we propose theoretical crowdsourcing workflow models, use translation tasks to study human activities, and simulate the optimal service workflow under various situations [11–13].

This monograph reports our research efforts on designing and analyzing human-in-the-loop service compositions, in both practical aspects and theoretical aspects.

1.3 Structure of This Chapter

Section 2 introduces a motivating example of translation service design to illustrate the necessity of designing and implementing human-in-the-loop composite services in real-world applications. The section also defines various patterns of combining human activities and Web services.

Section 3 presents a large-scale experiment on the composition of human activities and Web services in the field of language translation. The study considers both the functional and non-functional QoS attributes. The experiment results demonstrate that the inclusion of human activities in service processes introduces diversity compared to traditional processes that only involve Web services. Additionally, the study analyzes the impact of human activities on the QoS of service processes. The findings also indicate that high-quality human activities can significantly enhance various QoS attributes of service processes, while low-quality human activities may have negative effects on these processes.

Section 4 focuses on the design of human-in-the-loop composite services, considering the uncertainties associated with real-world services and users' requirements. The section proposes a service design approach, which includes phases such as observation, modeling, implementation, and analysis. The section also presents a field study on the design of multi-language communication services to demonstrate the effectiveness of the proposed service design approach.

Section 5 proposes theoretical approaches to modeling and optimizing the crowdsourcing workflow. Experiments under various situations yield results consistent with existing studies in the research community of crowdsourcing.

Section 6 describes the related work on human activities in service composition, user-centered composite service design, and crowdsourcing workflow models.

Section 7 concludes this monograph by summarizing the contributions of our work on human-in-the-loop service composition and discussing future directions.

2 Human-in-the-Loop Service Composition

2.1 A Language Service Composition Example

To illustrate the research issue, we present a case study in the field of language translation. Specifically, we examine the two methods of achieving language translation: human translation and machine translation. To provide flexible language services, we have developed the Language Grid, a service-oriented intelligence platform [16, 17]. The Language Grid collects language resources from various sources such as the Internet, universities, research labs, and companies. These resources are then encapsulated as atomic Web services with standardized interfaces. We have also created a series of composite services using these atomic language services. Furthermore, it is also possible to encapsulate human activities as Web services on the Language Grid [31]. Within the Language Grid, multiple QoS attributes are managed for language services, including general attributes such as response time and cost, as well as application-specific attributes like translation quality [34]. In the domain of language services, the application-specific QoS attributes, particularly translation quality, are of utmost importance. Previous evaluations of translations have focused on the adequacy and fluency [34]. Adequacy refers to the extent to which the translation effectively conveys the information present in the original text, while fluency pertains to the degree to which the translation adheres to the grammar of the target language.

Given that users have varying QoS requirements for language services, it is necessary to provide different atomic services or composite services with different QoS for the same function. In the Language Grid, language services are categorized into several classes, with multiple atomic services or composite services provided for different QoS requirements within each class. For instance, the translation service class includes atomic machine translation service, two-hop machine translation service, machine translation service combined with a bilingual dictionary, and so on. By creating a composite machine translation service that incorporates services such as morphological analysis and dictionary, the functional QoS can be enhanced compared to using the atomic machine translation service alone. However, despite the availability of various types of services, there are still limitations in terms of functional QoS attributes. For example, machine translation services, even when combined with dictionaries or other services for QoS improvement, cannot achieve perfect fluency and adequacy. This means that service-based processes may not always meet users' requirements. While a composite translation service may be suitable for fulfilling QoS requirements in online multilingual chatting, it may be challenging to use a purely service-based process for writing business documents or translating product operation manuals.

To address both the functional and non-functional QoS of translation services, we conducted a preliminary experiment that aimed to integrate human activities and Web services [34]. However, we discovered that human resources can also become a bottleneck if they are not readily available. As a solution, we propose the incorporation of crowdsourcing into the service process.

2.2 Composition of Web Services and Human Activities

Given the presence of an established service process, it is feasible to incorporate human activities through various means such as replacing an atomic service or subprocess, establishing a selective control relationship with a service or subprocess, or improving the input or output of an atomic service or subprocess either fully or partially. This approach can also be applied to integrate human activities into a process that consists of both human activities and Web services. To enhance the QoS, we propose several fundamental patterns for introducing a human activity (or human service) into a service process. These fundamental patterns can also be combined to address more complicated scenarios.

- *Complete substitution*: a human activity h_i is used to substitute a service s_i (or a subprocess) completely.
- *Partial substitution*: a human activity h_i is used to form a selective control relationship with a service s_i (or a subprocess) under a certain condition.
- *Pre-processing*: a human activity h_i is used to pre-process the input of a service s_i (or a subprocess).
- *Partial pre-processing*: a human activity h_i is used to pre-process the input of a service s_i (or a subprocess) under a certain condition.
- *Post-processing*: a human activity h_i is used to post-process the output of a service s_i (or a subprocess).
- *Partial post-processing*: a human activity h_i is used to post-process the output of a service s_i (or a subprocess) under a certain condition.

In the context of machine translation services, the functional QoS attributes that are relevant are fluency and adequacy. In cases where the service process itself fails to meet the user's QoS requirement, there are several alternatives for introducing human activities. These alternatives include: (1) completely substituting the machine translation service process with human activity for translation, referred to as *complete substitution*; (2) incorporating a human activity for pre-editing the source sentence within the original service process, such as modifying long sentences or reordering words to facilitate easier translation, known as *pre-processing*; (3) introducing a human activity for post-editing the translation result, such as enhancing fluency by a monolingual user, when the original service process fails to satisfy the user's QoS requirement, referred to as *partial post-processing*; and (4) combining the human activities of pre-editing and post-editing to enhance the QoS of the original service process, which involves a combination of *pre-processing* and *post-processing* patterns.

3 Empirical Study on Human-in-the-Loop Translation Services

3.1 Experiment Design

To examine the impact of the composition of Web services and human activities on QoS, a comprehensive experiment is conducted focusing on language translation. The translation procedures employed in this experiment are constructed based on the patterns outlined in Sect. 2.2. Within the language service domain, QoS encompasses both non-functional attributes (such as cost and time) and functional attributes (specifically, the quality of translation, i.e., the adequacy of the translation result). To assess the effectiveness of combining human activities with Web services, a three-step experimental design is devised:

- *Step 1 (CMT)*: Use a composite machine translation service that integrates three atomic services (a machine translation service, a morphological analysis service, and a dictionary service).
- *Step 2 (CMT+Mono)*: Incorporate human activities involving *partial post-processing* into CMT. The human activities are conducted by monolingual users for post-editing a specific portion of the CMT-generated translation results, with the condition that monolingual users can understand the machine translation results.
- *Step 3 (CMT+Mono+Bi)*: Incorporate human activities of *post-processing* into CMT+Mono. The human activities are conducted by bilingual users to confirm the correctness of the post-editing results in *CMT+Mono* as well as translating the unmodified parts in *CMT+Mono*. The whole flow is shown in Fig. 1.

In this experiment, the Language Grid provides a range of essential Web services, such as machine translation services, morphological analysis services, and dictionary services. These Web services are constructed by wrapping language resources that are originally provided by various organizations.

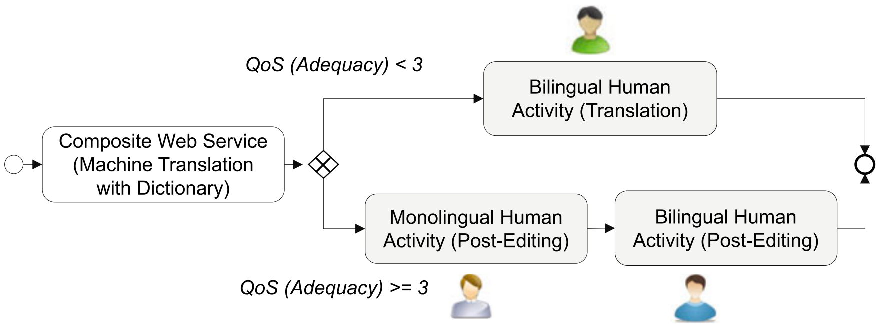


Fig. 1 Translation process composing by Web services and human activities (Step 3: CMT+Mono+Bi)

- *Machine translation services*: JServer service (language pairs used in the experiment: Japanese (ja) ↔ English (en), Japanese (ja) ↔ Korean (ko), Japanese (ja) ↔ Simplified Chinese (zh-CN) and Japanese (ja) ↔ Traditional Chinese (zh-TW)) provided by Kodensha Co., Ltd, GoogleTranslate service (language pairs used in the experiment: English (en) ↔ Traditional Chinese (zh-TW)) provided by Google, WebTranser service (language pairs used in the experiment: English (en) ↔ German (de), English (en) ↔ French (fr), English (en) ↔ Spanish (es), and English (en) ↔ Portuguese (pt)) provided by Cross Language Inc.
- *Morphological analysis services*: Mecab Japanese morphological analysis service provided by NTT Communication Science Laboratories, and TreeTagger English morphological analysis service provided by University of Stuttgart.
- *Dictionary services*: dictionary service for Business, University, and Temple provided by Kyoto Information Card System LLC, Ritsumeikan University, and the Kodaiji Temple.

The experiment incorporates two types of human activities. Monolingual users are involved in post-editing machine translation results, while bilingual users engage in translation and post-editing of results produced by monolingual users. To examine the impact of human activities on the QoS of service processes, we employ two distinct configurations of human activities as follows:

- *Crowd workers for monolingual human activities*: Crowd workers are selected from a list of numerous registered foreign student users at Kyoto University, Japan. The sole prerequisite is that the registered user is a native speaker of the language in which post-editing is needed. Consequently, the quality of human activities conducted by the monolingual crowd workers cannot be predicted during the experiment.
- *Professionals for bilingual human activities*: Since the translation/confirmation tasks have stringent criteria for participation, only registered users who possess expertise in two languages required for the tasks are eligible. Consequently, the experiment ensures the inclusion of bilingual users who can deliver high-quality translations.

Table 1 shows the 14 service processes employed in the translation experiment. Each process follows the three steps outlined in Sect. 3.1. For instance, **Process (1)** in Table 1 pertains to the translation of business-related documents from Japanese to English. The experiment consisted of a total of 551 process instances, with each instance representing the translation of a Japanese sentence to an English sentence. Consequently, there are 551 subtasks available for translation in **Process (1)**. The composite translation service used for **Process (1)** relies on three atomic services on the Language Grid: the JServer Japanese-English machine translation service, the business bilingual dictionary service, and the Mecab Japanese morphological analysis service. Human activities include post-editing tasks for English monolingual users and translation/post-editing tasks for Japanese-English bilingual users.

Table 1 Translation processes used in the experiments that combine Web services (MT: machine translation service; Dic: bilingual dictionary service; MA: morphological analysis service) and human activities (Mono: monolingual human activity; Bi: bilingual human activity)

Process ID	Instances	Web services and human activities				
		MT	Dic	MA	Mono	Bi
#1	551	JServer	Business	Mecab	en	ja, en
#2	551	JServer	Business	Mecab	zh-CN	ja, zh-CN
#3	551	JServer	Business	Mecab	ko	ja, ko
#4	551	WebTranser	Business	TreeTagger	de	en, de
#5	551	GoogleTranslate	Business	TreeTagger	zh-TW	en, zh-TW
#6	551	WebTranser	Business	TreeTagger	pt	en, pt
#7	1,084	JServer	Univeristy	Mecab	en	ja, en
#8	1,084	JServer	University	Mecab	zh-CN	ja, zh-CN
#9	201	JServer	University	Mecab	ko	ja, ko
#10	179	JServer	Temple	Mecab	en	ja, en
#11	179	JServer	Temple	Mecab	zh-CN	ja, zh-CN
#12	179	JServer	Temple	Mecab	ko	ja, ko
#13	179	WebTranser	Temple	TreeTagger	de	en, de
#14	179	WebTranser	Temple	TreeTagger	fr	en, fr

3.2 Experiment Results

We perform a series of measurements to examine the impact of human activities on the QoS in service processes.

- Evaluation of the functional QoS in terms of translation adequacy, as well as the non-functional QoS attributes such as execution time and cost.
- Examination of the correlation between the functional and non-functional QoS.
- Analysis of the impact of variations in human activities on the QoS attributes.

To assess the quality of human activities, we establish three indices: submission rate, acceptance rate, and completion rate for monolingual users. The rationale behind defining the three indices exclusively for monolingual users is rooted in the assurance of the bilingual users' quality throughout the experiments, as outlined in Sect. 3.1. Consequently, the submission rate, acceptance rate, and completion rate can be considered as 100% for bilingual users in this experiment.

- **Monolingual Submission Rate (MSR)**: the proportion of post-edited results among all machine translation results for monolingual users in Step 2.
- **Monolingual Acceptance Rate (MAR)**: the proportion of successfully accepted post-edited results among all submitted results for monolingual users in Step 3.

Table 2 Measurements of the human-in-the-loop service processes

Process ID	MSR	MAR	MCR	MWT (min)	BWT (min)	TWT (min)	CWT (min)
#1	0.478	0.921	0.440	21.43	102.86	124.29	166.29
#2	0.460	0.484	0.223	23.60	128.57	152.17	116.57
#3	0.940	0.953	0.896	25.71	16.29	42.00	116.57
#4	1.000	0.635	0.635	27.60	53.14	80.74	116.57
#5	0.789	0.253	0.200	35.14	137.14	172.28	116.57
#6	0.996	0.541	0.539	34.86	53.14	88.00	116.57
#7	0.528	0.614	0.324	18.91	129.09	148.00	127.27
#8	0.718	0.245	0.176	17.03	101.82	118.85	78.18
#9	0.987	0.387	0.382	22.50	56.25	78.75	78.15
#10	0.456	0.273	0.125	19.44	213.33	232.77	166.67
#11	0.401	0.834	0.334	14.44	120.00	134.44	133.33
#12	0.753	0.876	0.660	22.78	60.00	82.74	133.33
#13	0.950	0.643	0.611	19.44	60.00	79.44	133.33
#14	0.908	0.785	0.713	26.67	60.00	86.67	133.33

- **Monolingual Completion Rate (MCR)**: the proportion of completed post-edited (submitted and accepted) results among all the machine translation results for monolingual users in Step 3, which is determined by $MCR = MSR \times MAR$.

To investigate the impact of human activities on the execution time (duration) of the service process, we assess the following items:

- **Monolingual Work Time (MWT)**: execution time of the monolingual human activities.
- **Bilingual Work Time (BWT)**: execution time of the bilingual human activities.
- **Total Work Time (TWT)**: summation of monolingual work time (MWT) and bilingual work time (BWT), which is determined by $TWT = MWT + BWT$.
- **Common Work Time (CWT)**: execution time when the process is a purely human translation process.
- **Time Reduction Rate (TRR)**: the extent to which the execution time is reduced in comparison to the conventional human translation process, which is determined by $TRR = 1 - \frac{TWT}{CWT}$.

Table 2 presents the results of above the indices for all 14 processes conducted in the experiments. The results indicate significant variations in the quality of monolingual human activities and execution time across the different processes.

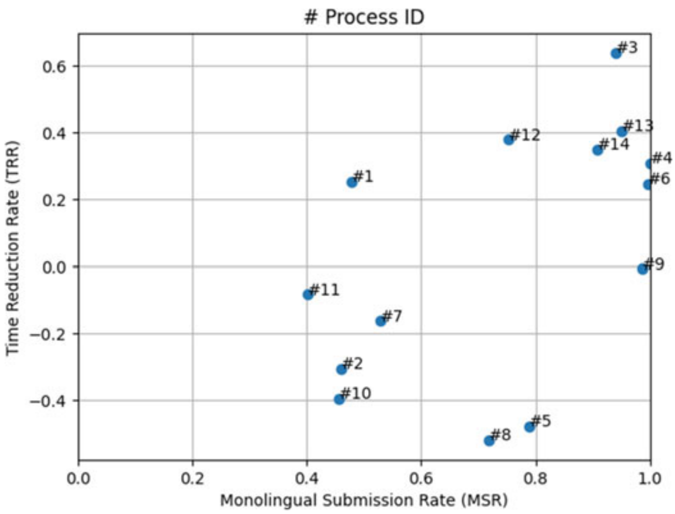


Fig. 2 Relationship between time reduction rate (TRR) and monolingual submission rate (MSR)

3.2.1 Effects of Human Activities on Execution Time

Figures 2 and 3 provide an analysis of the correlation between time reduction rate (TRR), monolingual submission rate (MSR), and monolingual completion rate (MCR). The data presented is based on the translation task of an average calculation of a single A4-size page, which is approximately 700 Japanese characters or 400 English words. The involvement of human activities in the translation process results in a reduction in execution time for half of the 14 processes, while the other half experiences an increase in execution time compared to a purely human translation process. The findings also indicate that a high monolingual submission rate (MSR) does not necessarily lead to a high time reduction rate (TRR). However, there is a trend suggesting that a higher monolingual completion rate (MCR) is associated with a greater time reduction rate (TRR). Additionally, it appears challenging to reduce execution time when the monolingual submission rate (MSR) is relatively high, but the monolingual completion rate (MCR) is low (e.g., **Process (5)**, **Process (8)** and **Process (9)**). This difficulty arises from the significant time wasted in dealing with low-quality submissions by monolingual users that are not accepted.

3.2.2 Effects of Human Activities on Cost

To investigate the impact of human activities on the cost of executing the service process, a series of measurements are conducted. In this experiment, bilingual users and monolingual users are paid at rates of US$ 50.00 and US$ 5.00 per A4-size page,

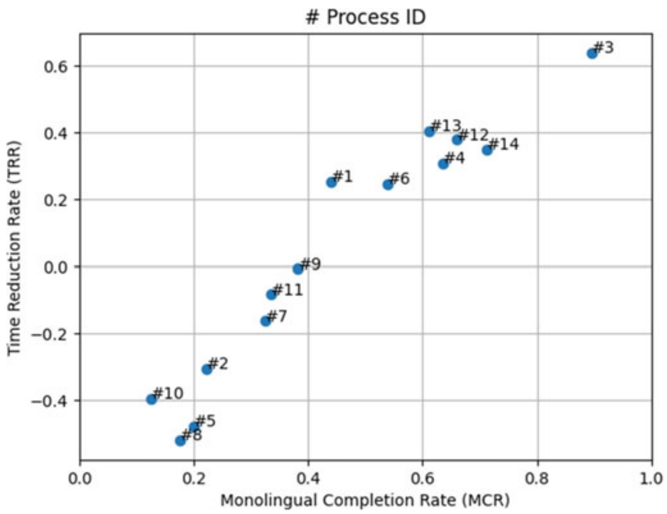


Fig. 3 Relationship between time reduction rate (TRR) and monolingual completion rate (MCR)

respectively. However, in cases where the results were not accepted, the payment to the monolingual users was reduced by half.

- **Monolingual Work Cost (MWC)**: cost of monolingual human activities, which is calculated by $MWC = 5.00 \times (MCR + \frac{1}{2}(MSR - MCR))$.
- **Bilingual Work Cost (BWC)**: cost of bilingual human activities, which is calculated by $BWC = 50.00 \times (1 - MCR)$.
- **Total Work Cost (TWC)**: summation of the cost of monolingual human activities and bilingual human activities, which is determined by $TWC = MWC + BWC$.
- **Common Work Cost (CWC)**: cost when the process is a purely human translation process, and $CWC = 50.00$.
- **Cost Reduction Rate (CRR)**: the cost reduction percentage in comparison to a purely human translation process, which is calculated by $CRR = 1 - \frac{TWC}{CWC}$.

Figure 4 illustrates the correlation between the cost (monolingual work cost (MWC), bilingual work cost (BWC), total work cost (TWC)) and monolingual completion rate (MCR). The findings indicate that employing a composite process involving both human activities and Web services can effectively reduce translation costs compared to relying solely on human translation. This supports the analysis conducted in our previous preliminary experiments [34]. The reason lies in that a part of the work in a purely human translation process is substituted with Web services and monolingual users with lower cost. Additionally, the results demonstrate that the cost reduction rate (CRR) increases as the monolingual completion rate (MCR) rises. An extremely successful example is **Process (3)**, which achieves a cost reduction rate (CRR) of 80.41% due to the high quality of monolingual human activity with the monolingual completion rate (MCR) of 89.59%.

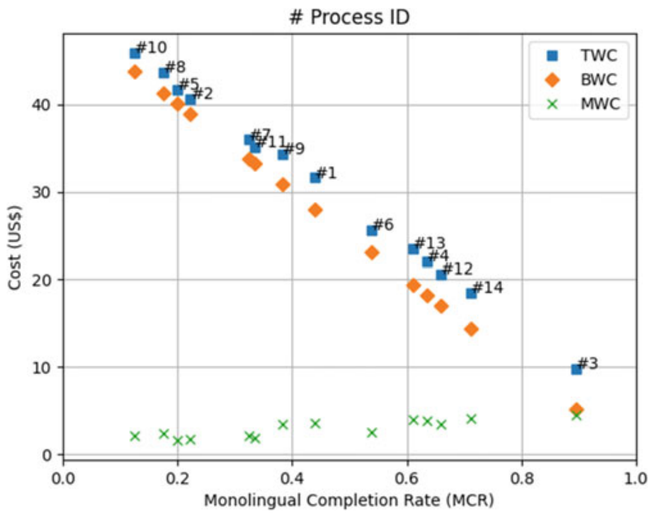


Fig. 4 Relationship between execution cost (monolingual work cost (MWC), bilingual work cost (BWC), total work cost (TWC)) and monolingual completion rate (MCR)

3.2.3 Effects of Human Activities on Relations of QoS Attributes

To analyze the impact of variations in human activities on the QoS attributes, we have classified the 14 processes into three groups according to their monolingual completion rate (MCR). This metric serves as a direct indicator of the quality of monolingual human activities.

- **Low-quality monolingual activity group**: Process (2), (5), (8), (10).
- **Medium-quality monolingual activity group**: Process (1), (6), (7), (9), (11).
- **High-quality monolingual activity group**: Process (3), (4), (12), (13), (14).

Figures 5 and 6 examine the correlation between functional QoS attributes, specifically translation quality, and non-functional QoS attributes, namely execution time and cost. The analysis compares different steps (Step 1 to Step 3 from left to right in each subgraph of Figs. 5 and 6) for all 14 processes in the experiment. The findings indicate that both execution time and cost increase as the steps progress from Step 1 to Step 3, indicating that achieving higher functional QoS requires more time and cost. Step 1, which solely involves Web services, incurs negligible cost and execution time compared to other steps. However, the functional QoS achieved in Step 1 is also limited. In contrast, Step 2, and Step 3, which prioritize high functional QoS, entail significantly higher cost and execution time.

The results in Figs. 5 and 6 also demonstrate that the quality of human activities has varying effects on the QoS attributes of composite services. Specifically, composite services characterized by low-quality monolingual activity group incur significant costs in improving functional quality from Step 2 to Step 3, resulting

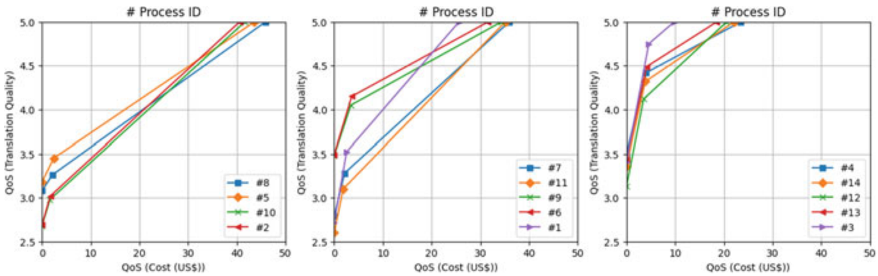


Fig. 5 Relationship between cost and translation quality

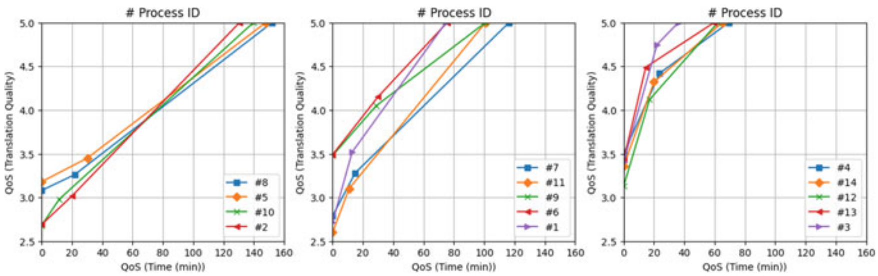


Fig. 6 Relationship between execution time and translation quality

in only marginal cost savings compared to purely human processes (valued at US$ 50). Furthermore, these composite services require more execution time in Step 3 compared to purely human processes (100 min). Conversely, composite services with high-quality monolingual activity groups can enhance functional QoS with minimal cost and execution time from Step 2 to Step 3. Consequently, the variations in the quality of human activities significantly influence QoS attributes. These results suggest the need for the development of quality control models for human activities to ensure high QoS in composite services.

3.3 Discussion

Although the example used in this study falls into the language service domain, it is important to note that the issue of service-based processes not always meeting users' requirements due to limitations in functional QoS attributes is prevalent in other domains, such as various artificial intelligence (AI) services in smart cities, ranging from object detection to voice recognition. To address both functional QoS and non-functional QoS attributes in such service processes, the integration of human activities and Web services can be considered a promising approach. By combining human activities and Web services, the variety of service implementation can be expanded. In cases where Web service-based processes exhibit limited functional QoS, the introduction of human activities can enhance functional QoS to varying degrees based on users' requirements. Similarly, in purely human processes, the incorporation of Web services, even with limited functional QoS, can enhance efficiency and improve non-functional QoS.

In this empirical investigation, our primary objective is to examine the impact of human activities on both functional and non-functional QoS. Consequently, we have chosen to utilize only a limited number of service composition patterns as defined in Sect. 2.2. Nevertheless, it is crucial to consider the appropriate application of various patterns for inducing human activities in different situations, considering users' requirements. This is because the effect of human activities on the QoS of service processes may vary depending on the specific pattern employed. In the language translation example, the analysis of QoS effects of different patterns can be used for service design of field-based multi-language communication [28].

4 Human-in-the-Loop Service Design for Supporting Real-World Multilingual Activities

In the previous section, we described our research efforts on analyzing non-functional and functional QoS in human-in-the-loop service composition by using a pre-designed language translation service process. In this section, we will report our study of designing human-in-the-loop composite services for real-world applications, where there are numerous variations of combining human activities and Web services.

4.1 Designing Composite Services for Real-World Applications

To design human-in-the-loop composite services in the real world, there are several significant issues that need to be addressed. Firstly, the performance of services may vary due to the dynamic nature of service environments [31], resulting in inherent uncertainty in QoS [49]. This uncertainty poses challenges in designing composite services based on QoS. This issue becomes even more challenging when considering the combination of human activities and Web services. Secondly, when multiple QoS attributes are associated with services, it is often difficult to optimize all of these attributes simultaneously due to the presence of anti-correlated relationships among them [2]. For example, improving the quality of translation in a multi-language communication service might result in a significant increase in cost. Therefore, it is necessary to design composite services based on users' requirements.

We present an example of a multi-language communication service design project, the YMC (Youth Mediated Communication)-Viet project, which aims to assist Vietnamese farmers in accessing agricultural knowledge from Japanese experts [28–30, 33]. The YMC-Viet project was conducted in collaboration with the Ministry of Agriculture and Rural Development of Vietnam (MARD) as a model initiative for providing ICT assistance to developing nations. Due to the low literacy rate among farmers in rural areas, literate youths, who are the children of these farmers, serve as intermediaries between the Japanese experts and the Vietnamese farmers. This project was implemented in Thien My Commune and Tra On District of Vinh Long Province, Vietnam, over four seasons from 2011 to 2014, involving 15–30 families of farmers in each season. The YMC-Viet project facilitates communication between Japanese experts and Vietnamese youths through an online tool called the YMC system [44, 45], where human-in-the-loop composite services are embedded. This system supports multiple languages and allows Vietnamese youths to send field data and questions. The Japanese experts receive these data and questions and respond in Japanese, which is then translated into Vietnamese by the system and delivered back to the youths. The key challenge is to design a multi-language communication service that maximizes the effectiveness of the YMC system.

To design the multi-language communication service, we utilize the Language Grid as the platform for language service composition. Figure 7 illustrates a part of available services for the YMC-Viet project. With the availability of various language resources on the Internet, such as machine translators, multi-language dictionaries, and parallel texts, it has become possible for users to design language services to suit their own requirements [34, 39]. However, challenges arise when dealing with the uncertain quality of different language services. For instance, estimating the quality of a machine translation service is always a difficult task. Therefore, it is crucial to develop an approach for designing composite services that can effectively handle the QoS uncertainty.

Based on the available services depicted in Fig. 7, several alternative composite services can be employed to support multi-language communication between Japanese and Vietnamese. These alternatives include: (1) a composite machine translation service that integrates Japanese-English machine translation and English-Vietnamese machine translation, (2) a composite Japanese-Vietnamese machine translation service that incorporates an agriculture dictionary, (3) a composite translation service that combines Japanese-Vietnamese machine translation with Vietnamese post-editing by human translators, and so on. However, determining the optimal composite service is challenging due to the uncertain quality of translation services, as previously discussed. Consequently, it is imperative to consider how to design an appropriate composite service that meets users' requirements. Furthermore, it is likely that a combination of human activities and Web services will be necessary, thereby further complicating the service design process.

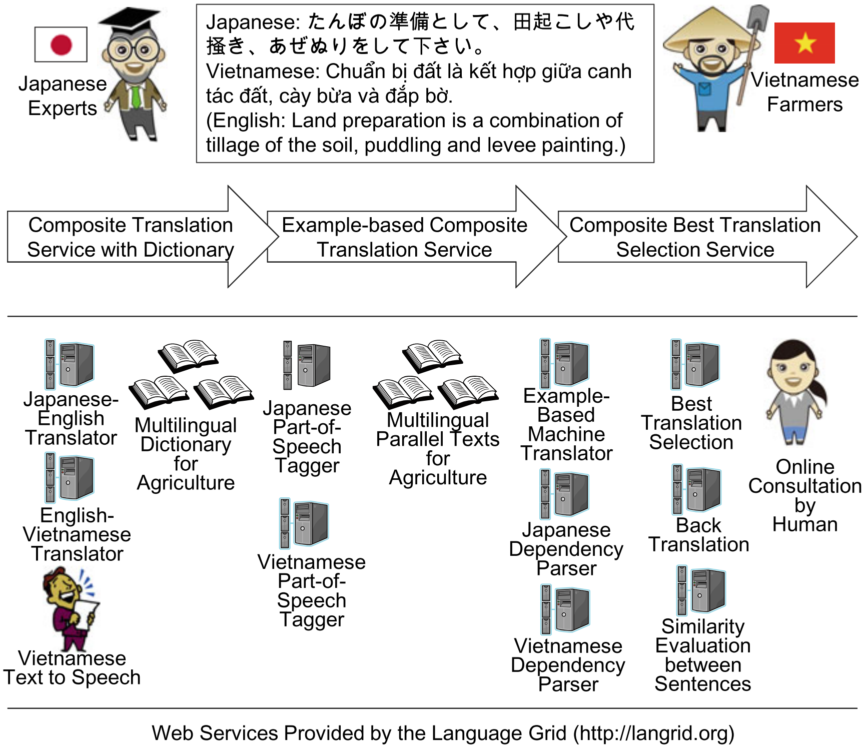


Fig. 7 Available language services for multi-language agricultural support (cited from [32])

4.2 *Service Design Process*

To address the complex challenges posed by factors such as the QoS uncertainty, the composition of human activities and Web services, and the diverse requirements of users, it is imperative to adopt an iterative service design methodology for composite services prior to their implementation and deployment in the real world. In this regard, it is natural to assess the QoS of the composite services and users' satisfaction throughout the entire design process.

In this study, we propose a user-centered participatory service design approach to address these challenges. While participatory design has been previously suggested for community informatics [9] and multi-agent systems [19], its application in service-oriented computing, particularly in the context of user-centered design for service composition, is also expected to be effective to address the aforementioned challenging issues. The proposed service design process includes the following phases:

- *Observation*: Investigate and/or update the information of available Web services and human services, establish QoS criteria, and understand users' QoS requirements for service design.
- *Modeling*: Utilize a user-centered approach to identify the most suitable candidate human-in-the-loop composite service that can effectively meet the QoS requirements of users [30].
- *Implementation*: Implement the composite service model defined in the previous phase. To facilitate the improvement of system implementations, participatory simulations are conducted prior to their deployment in real-world settings [28].
- *Analysis*: Evaluate the implemented service by analyzing the log data of QoS based on the defined evaluation criteria. The findings from this analysis will offer valuable insights and knowledge that can be applied to refine the composite service in subsequent design iterations.

4.3 Experiment, Result and Analysis

We use the YMC-Viet project to illustrate the effectiveness of our proposed approach for human-in-the-loop composite service design [30]. Key elements during the service design process in the YMC-Viet project are as follows.

- **Services for composition.** To implement the multi-language communication service, a range of atomic services and composite services are utilized. Table 3 shows a list of Web services provided by the Language Grid and human services used.
- **QoS attributes and QoS data.** As previously discussed, QoS within the language service domain encompasses both non-functional attributes, such as translation cost and execution time, as well as functional attributes, such as translation quality. In this study, we have also focused on cost, execution time, and translation quality as the primary QoS attributes. Given the absence of QoS data prior to conducting field experiments, we estimated the QoS ranges for various composite services by simulations.
- **Users' requirements.** The user requires that the translation quality should exceed 4.0 and the cost should be reduced to below 50% of a purely human translation service.

The user-centered participatory service design approach was employed in the design of the multi-language communication service during the first two seasons' experiments. The iterative participatory design result, ranging from process P1 to P5, is presented in Table 4. The parallel text service, which was utilized from process P2 to P5, is omitted from Table 4 for simplicity. Figure 8 provides an overview of the QoS values associated with each process outlined in Table 4. Moreover, the refinement of composite service design is depicted, with four iterations observed throughout the experiment: from P1 to P2, from P2 to P3, from P3 to P4, and from P4 to P5. Composite service P5 successfully met the users' requirements and was adopted

Table 3 List of web services and human services for multi-language communication service design (cited from [30])

Service	Service type	Description
s_1	Composite web service	Composite Japanese-Vietnamese machine translation service combined with agriculture dictionary
s_2	Composite web service	Composite Japanese-English machine translation service combined with agriculture dictionary
s_3	Composite web service	Composite English-Vietnamese machine translation service combined with agriculture dictionary
s_4	Atomic web service	Japanese-Vietnamese parallel text service for agriculture
h_1	Human service	Japanese pre-editing service
h_2	Human service	English post-editing service
h_3	Human service	Vietnamese post-editing service
h_4	Human service	Japanese-English human translation service
h_5	Human service	Japanese-Vietnamese human translation service

Table 4 Composite service processes designed in the YMC-Viet project

Process ID	Service workflow	Description
P1	s_1	Initial process in the first season of YMC
P2	$h_1 \rightarrow s_1$	Refined process 1 in the first season of YMC
P3	$h_1 \rightarrow s_1 \rightarrow h_3$	Refined process 2 in the first season of YMC
P4	$h_4 \rightarrow s_3 \rightarrow h_3$	Final process used in the first season of YMC
P5	$h_1 \rightarrow s_2 \rightarrow h_2 \rightarrow s_3 \rightarrow h_3$	Final process used in the second season of YMC
P6	h_5	Used for QoS comparison

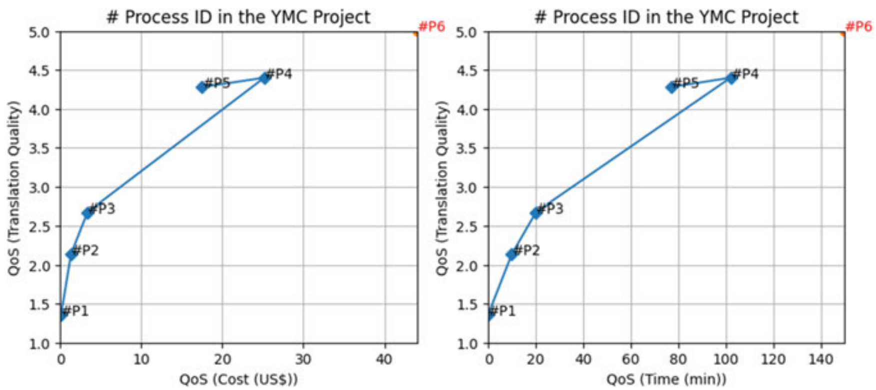


Fig. 8 Change of service processes and QoS values with participatory service design in the YMC-Viet project

as the optimal composite service. P5 combines several human-in-the-loop patterns defined in Sect. 2, including *pre-processing* and *post-processing*. As a result, P5 was selected as the composite service model for the implementation of the multi-language communication tool (YMC system) during the field experiment following the second season in 2012. The details of the composite service refinements are described in [29, 30].

The service process implemented in the YMC-Viet project yielded positive outcomes. There are two possible reasons. Firstly, the service process employed in this project was relatively straightforward and not overly complex. Secondly, we were able to leverage valuable insights gained from a prior study that focused on analyzing the QoS in human-in-the-loop language services. These valuable insights played a crucial role in reducing the number of potential composite service models at the initial stage of the project. On the other hand, it is imperative to devise effective techniques for optimizing human-in-the-loop services in situations where the service composition is intricate or when a novel application domain is introduced.

5 Analyzing Crowdsourcing Workflow Models

5.1 Crowdsourcing Workflows

In the previous section, we described the design and implementation of the human-in-the-loop service workflow for multi-language activities. In such workflows, human services performed through crowdsourcing are an attractive source of language services. Since the early 2010s, crowdsourcing has been utilized for a range of open-ended tasks, including writing, design, and translation. One of the advantages of crowdsourcing is its flexibility compared to machine services. However, when it comes to open-ended tasks like translation, the quality of the output from an individual worker cannot be guaranteed due to the varying abilities of crowdsourcing workers. To ensure the desired level of quality, requesters often create a workflow in which the output of one crowd worker is refined incrementally by other workers. While the significance of crowdsourcing workflows has been acknowledged in previous research [24], a comprehensive understanding of the general characteristics of such workflows is still lacking.

Collaboration among workers in crowdsourcing has primarily relied on two processes: the iterative process and the parallel process. In an iterative process, one worker's task is improved upon by other workers in a continuous manner [24]. On the other hand, crowdsourcing is inherently a parallel process, where multiple workers execute the same task and the final result is determined through voting or other means [25, 37]. Studies on iterative and parallel processes in crowdsourcing workflows have revealed two main findings: (1) the diversity of crowd workers plays a significant role [37], and (2) prior results can negatively impact quality if subsequent workers are led astray in difficult iterative tasks [35]. Previous research has focused on analyzing workflows for specific tasks and has not provided a comprehensive understanding of crowdsourcing workflows. While there have been studies on optimizing workflows, these works have mainly concentrated on optimizing fixed workflow structures such as the number of iterations or degree of parallelism.

To optimize the utilization of crowdsourcing, it is imperative to address two key challenges. Firstly, it is crucial to develop a mechanism that allows task requesters to obtain an accurate estimation of the utility of crowdsourcing. This estimation of utility would assist them in making decisions on whether they use crowdsourcing or not prior to submitting an actual request. Secondly, an intuitive interface needs to be designed that enables users to request tasks easily.

Consider the scenario in which a requester intends to utilize crowdsourcing for a translation task. The crowdsourcing platform offers a pool of available workers, but the requester cannot determine the suitability of a worker until the task is completed. Since relying on a single worker may not guarantee translation quality, it is important to establish a translation workflow that involves multiple workers performing improvement tasks. In each iteration, a worker enhances the best result from the previous iteration. However, the requester aims to achieve the best outcome while considering the trade-off between cost and quality. Furthermore, the requester needs to decide whether to request a task based on the predicted cost and quality before posting it. Therefore, it is crucial to develop a model that encompasses crowd workers, tasks, and requester utility to gain a comprehensive understanding of crowdsourcing performance in general.

5.2 *Modeling Iterative and Parallel Processes*

To gain a comprehensive understanding of the crowdsourcing workflow, it is necessary to construct a model that can effectively estimate the utility of the workflow composed of iterative and parallel processes. This model is defined by several key factors, including the distribution of abilities among crowd workers, the level of difficulty associated with the task, and the preferences of the requester.

5.2.1 Workers

It is expected that workers with high abilities will produce high-quality results. For the sake of simplicity, we assume that the quality of a task's execution is solely determined by the ability of the worker who performed the task. Given that the ability of a worker is not known prior to task execution, we employ a beta distribution to model the distribution of worker abilities. Probability density function $f(x|a, v)$ is given by Eq. (1).

$$f(x|a, v) = \text{Beta}\left(\frac{a}{\min(a, 1-a)v}, \frac{(1-a)}{\min(a, 1-a)v}\right) \tag{1}$$

Here $a \in (0, 1)$ is the normalized value of the average ability of the workers in the crowdsourcing platform. $v \in (0, 1)$ is a parameter that determines the variance in worker ability. When v is near 0, the variance approaches 0. When v is near 1, the variance approaches the highest variance with average worker ability of a. The model extends the previous work [10] by modifying a parameter that describes the variance of worker ability.

5.2.2 Workflows

An open-ended task consists of iterations of improvement tasks, thus referred to as an iterative process. In an iterative process, high-quality results are achieved by iteratively improving prior work by a new worker. However, it is worth noting that there are instances where multiple workers simultaneously improve the same task, known as a parallel process. Examples of improvement tasks implemented as iterative and parallel processes are reported by Little et al. [35].

We formally define a crowdsourcing workflow as $w = (p_1, \ldots, p_n)$, where n is the number of improvement tasks in the iterative process and $p_i (1 \leq i \leq n)$ is the number of workers that execute the ith improvement task in parallel. As a result, the total number of workers in the workflow is given by $m = \sum_{i=1}^{n} p_i$.

After each iteration, the best result will be automatically selected. In the case that none of the results have better quality than the input of the improvement task, the input will be designated as the best result.

5.2.3 Improvement Task

Various tasks possess varying levels of difficulty. We assign a parameter $d \in [0, 1]$ to quantify the improvement difficulty of a task. If the improvement difficulty, d, is 0, then the improvement task is extremely easy. In contrast, the quality of a task with $d = 1$ indicates that the task is extremely challenging to improve. For example, if the task involves adding a missing caption to an illustration, then d would be close to 0 as it is relatively simple for a new worker to improve the quality by providing additional information. Conversely, if the task involves improving the illustration itself, the value of d may approach 1 since it is always extremely difficult to improve the output of another designer. For most other types of tasks, such as translation improvement, the value of d lies between 0 and 1. Given the improvement difficulty d of a task, we use the function $q'(a, q)$ to define the quality of the outcome after executing the improvement task once, where a represents the worker's ability and q denotes the quality of the input result of the current improvement task.

$$q'(a, q) = q + (1 - q)a - q(1 - a)d \qquad (2)$$

The equation presented above represents the summation of three distinct components. The first component represents the original quality, denoted as q, of the input result for the current improvement task. The second component signifies the increase in quality that occurs following the execution of the improvement task. Lastly, the third component represents the penalty in quality that arises if the improvement fails. We will further explain the second and third components in more detail. If the original quality of the input result is q, then the remaining potential for quality improvement is $1 - q$. The second component, $(1 - q)a$, indicates that the extent of improvement is proportional to the worker's ability, denoted as a. Conversely, $q(1 - a)$ represents the likelihood of improvement failure. When the original quality is high or the worker's ability is low, the probability of improvement failure increases. The inclusion of the improvement difficulty d in the multiplication of the third component is justified by the fact that a larger value of d corresponds to a higher likelihood of quality deterioration. In other words, tasks with greater improvement difficulty are more prone to a decrease in quality. In the scenario where the improvement task is carried out by a single worker, the expected value of the quality after executing the improvement task is denoted as $q'(a, q)$, as the expected value of the worker's ability is a.

Next, we will elucidate the quality improvement through the incorporation of parallel processing. When multiple workers engage in the improvement task simultaneously, the outcome with the highest quality is selected as the assumed result. Consequently, the quality of the outcome is equivalent to that achieved by the worker with maximum ability during the iteration. We denote p as the number of workers involved in the improvement task in the current iteration. The maximum ability among these p workers ($a_p^{\max}$) is estimated as the average of the maximum distribution (Eq. (4)). Here, $F(x|a, v)$ represents the cumulative density function for $f(x|a, v)$, and $I_x(y, z)$ denotes the regularized beta function, which can be calculated using Eq. (3).

$$I_x(y, z) = \frac{\int_0^x t^{y-1}(1-t)^{z-1} dt}{\text{Beta}(y, z)} \tag{3}$$

$$\begin{aligned} a_p^{\max} &= \int_0^1 x F(x|a, v)^p dx \\ &= [x F(x|a, v)]_0^1 - \int_0^1 F(x|a, v)^p dx \\ &= 1 - \int_0^1 I_x\left(\frac{a}{\min(a, 1-a)v}, \frac{(1-a)}{\min(a, 1-a)v}\right)^p dx \end{aligned} \tag{4}$$

Taking $a_p^{\max}$ as a, the quality obtained by parallel processing with p workers will be $q'(a_p^{\max}, q)$.

5.2.4 Utility

The objective function for workflow optimization is determined by the utility of the requester in executing workflow, denoted as U. Previous studies have assessed the utility of a workflow by considering both the quality of the task and the cost of execution [10, 20]. In this study, utility is defined as the weighted sum of quality (Q) and cost (C) [48]. The preference of the requester is represented by the weight assigned to quality, denoted as β. Thus, the weight assigned to cost is equal to $1 - \beta$.

$$U = \beta Q + (1 - \beta)C \tag{5}$$

$Q \in [0, 1]$ can be obtained from the predicted quality of workflow w. The cost, $C \in [0, 1]$, is the normalized value given by Eq. (6), where m represents the number of workers and M represents the predefined maximum number of workers. It is important to note that the total cost is solely determined by the number of workers and is not affected by iterative or parallel processes.

$$C = \frac{M - m}{M} \tag{6}$$

5.3 *Workflow Optimization*

5.3.1 The Search Algorithm

Based on the process model presented above, it is possible to make predictions about the utility of a given workflow. Given a large number of potential workflows (specifically, if there are n workers, there are 2^n possible workflows), it is crucial to employ an efficient search strategy for workflow optimization. In this regard, we propose a search algorithm that identifies the maximum expected value of utility from a limited search space.

We assume that the cost of the workflow is proportional to the number of crowd workers involved. Therefore, when the quality is fixed, the utility of the workflow will monotonically decrease as the number of workers increases. On the other hand, the quality will monotonically improve with an increase in the number of workers. Although there may be occasional failures in the improvement tasks, it is assumed that the result with superior quality is selected when comparing the input and output of an improvement task. Therefore, an increase in the number of workers does not result in a decline in quality. Based on these assumptions, we can see that excessively increasing the number of workers will lead to a decrease in utility, as quality always has an upper limit. That is why there exists an optimal workflow that can maximize utility.

The proposed algorithm for identifying the optimal workflow is referred to as **Algorithm 1**. This algorithm operates within a state space composed of workflows, with each workflow being considered a state. The initial state, denoted as $w = (1)$, consists of a single improvement task performed by one crowd worker and is stored in the state set $OPEN$. The state space is searched by expanding the contents of state set $OPEN$. The expansion process $expand$ is outlined in **Algorithm 2**; it takes a workflow w as input and returns a set of workflows, denoted as W, which includes all possible workflows generated by adding one crowd worker to the original workflow w. The function $utility$ in **Algorithm 1** takes a workflow w as input and returns the predicted utility. The search algorithm stores only w' that is in the expanded set of w and has higher utility than workflow w in the $OPEN$ state set. This approach ensures that the search begins from the center of the crater and terminates at the crater rim, effectively avoiding the horizon effect in the state space where workflows are considered as states.

Algorithm 1 Searching Optimal Workflow *search*

```
w /* workflow */
utility(w) /* utility function for workflow w */
s /* current best workflow */
u /* utility value of the current best workflow */
Closed /* set of workflows already expanded */
Open /* set of workflows to be expanded */
s ← (1)
u ← utility(s)
Open ← {s}
Closed ← {}
while Open ≠ null do
   Select w ∈ Open
   Open ← Open − {w}
   Closed ← Closed ∪ {w}
   for all w′ ∈ expand(w) do
      if w′ ∉ Closed and utility(w′) ≥ utility(w) then
         Open ← Open ∪ {w′}
         if utility(w′) ≥ u then
            s ← w′
            u ← utility(w′)
         end if
      end if
   end for
end while
return s
```

5.3.2 Optimality

Here we will discuss the optimality of the workflow search algorithm (**Algorithm 1**) for crowdsourcing tasks. In a crowdsourcing workflow that consists of iterative and parallel processes, the search algorithm begins with an initial workflow state containing only one crowd worker and gradually expands the state space by adding

Algorithm 2 Expanding Workflow *expand*

Input: w
p_i /* number of workers that execute the ith improvement task in parallel */
n /* number of iteration */
$w = (p_1, \ldots, p_n)$ /* workflow */
$W = \{w_1, \ldots, w_m\}$ /* the set of created workflows by expansion of w */
m /* number of workflows created by expansion of w */
$W \leftarrow \{\}$
$W \leftarrow W \cup \{(1, p_1, \ldots, p_n)\}$
for $i = 1$ to n **do**
    $W \leftarrow W \cup \{(p_1, \ldots, (p_i + 1), \ldots, p_n)\}$
    $W \leftarrow W \cup \{(p_1, \ldots, p_i, 1, \ldots, p_n)\}$
end for
return W

one crowd worker at each epoch. The search algorithm terminates when the workflow state with the highest utility reaches the optimal workflow based on the given assumptions.

To prove the termination of our search algorithm, we show that the increase in utility created by adding a worker monotonically decreases with higher utility. Let the expected values of the quality and cost of workflow w with m crowd workers be q and c, respectively. First, we show that incremental quality monotonically decreases when one crowd worker is added with either iteration or parallelism. Assuming that the additional crowd worker is used to increase the iteration number, the incremental quality is given by $\Delta q = a(1-q) - (1-a)qd$. Here a and d are constants assuming the additional worker always has expected quality a. In each iteration, q monotonically increases. Therefore, $a(1-q)$ monotonically decreases and $(1-a)qd$ monotonically increases. As a result, incremental quality Δq monotonically decreases. On the other hand, when the additional crowd worker is used to increase parallelism, the quality increment Δq depends on the increment of the maximum ability of worker Δa. Since the maximum expected ability is calculated using the regularized beta function, which satisfies $I_x(y, z) \leq 1$, Δa monotonically decreases with the increase in m. Therefore, strengthening parallelism leads to a monotonic decrease in Δq. The increment of the maximum value of a beta distribution monotonically decreases as it approaches 1, so quality increment Δq monotonically decreases with the addition of a worker. Second, cost increment Δc remains constant when one crowd worker is added. This implies that the normalized cost C monotonically decreases. As utility is calculated by the weighted summation of quality and cost, the increase in the amount of utility decreases and turns negative.

In summary, the incremental utility monotonically decreases and eventually becomes negative at a certain point. Therefore, the search algorithm terminates under the given assumptions. Furthermore, since the expansion of the workflow state space stops when the incremental utility becomes negative, the workflow state with the maximum utility is obtained when the search algorithm terminates.

It should be noted that the above discussion does not guarantee an optimal solution when increasing crowd workers in real-world crowdsourcing tasks. Instead, the model can calculate the optimal workflow based on predetermined values. However, if the optimal solution search can be conducted efficiently, we can gain insights into the characteristics of crowdsourcing workflows and utilize this knowledge in the design of real-world crowdsourcing tasks.

5.3.3 Analysis of Optimal Workflows

Based on the established model and its optimization algorithm, it is possible to make estimations regarding the utility of each workflow under different parameter settings. In this monograph, we mainly report the experiment that examines the optimal workflows and their utility for different parameter settings. The details of the experiment that compares the performance of iterative and parallel processing methods are described in [11].

We use the proposed search algorithm to obtain optimal workflow w for various combinations of parameters. Furthermore, we calculate the utility of each workflow w. The specific parameter settings used in the experiments are as follows:

Average ability of workers $a \in (0, 1)$: varied from 0.1 to 0.9 in steps of 0.2.
Variance of worker ability $v \in (0, 1)$: varied from 0.1 to 0.9 in steps of 0.2.
Improvement difficulty $d \in [0,1]$: 0 (low), 0.5 (middle) and 1 (high).
Preference of the requester over quality β: 0.1, 0.5 and 0.9.

Table 5 and Fig. 9 present the findings of optimal workflows and their utilities under different settings of the variance of worker ability and improvement difficulty of tasks. The results indicate that as the variance of worker ability increases, optimal workflows tend to exhibit greater parallelism. Additionally, the parallelism of optimal workflows also tends to increase with higher levels of improvement difficulty. The utility of optimal workflows demonstrates an upward trend as improvement difficulty decreases. However, it is worth noting that the utility of optimal workflows can also increase with higher levels of worker ability variance, even in cases where improvement difficulty is high. This is because workflows with a high degree of parallelism are more likely to be optimal solutions, and worker ability becomes more influential when the variance of worker ability is high.

Table 5 Optimal workflows in different variations of v and d

v	$d = 1$	$d = 0.5$	$d = 0$
0.9	(2)	(2)	(1,1,1)
0.7	(2)	(2)	(1,1,1)
0.5	(2)	(2)	(1,1,1)
0.3	(2)	(1,1)	(1,1,1)
0.1	(1)	(1,1)	(1,1,1)

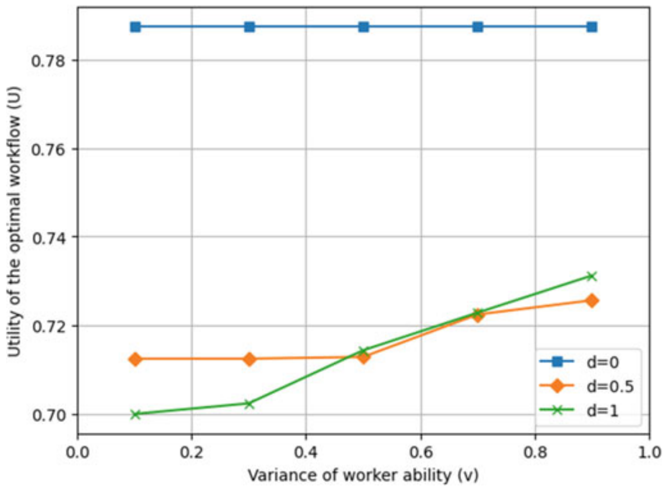


Fig. 9 Utilities of the optimal workflow in different variations of v and d

Table 6 Optimal workflows in different variations of a and β

a	$\beta = 0.9$	$\beta = 0.5$	$\beta = 0.1$
0.9	(1,2)	(1)	(1)
0.7	(1,4)	(1,1)	(1)
0.5	(8)	(2)	(1)
0.3	(3,6)	(1,1)	(1)
0.1	(2,5)	(1)	(1)

Table 6 and Fig. 10 present the outcomes of optimal workflows and their utilities under different variations of the average worker ability and quality preference of the requester. The findings indicate that the optimal workflows exhibit the highest level of parallelism when the average worker's ability is at the intermediate level (i.e., $a = 0.5$). Additionally, as the average worker's ability deviates from the intermediate level (either higher or lower), the degree of parallelism in the optimal workflows decreases and iterative improvement becomes more effective. Not surprisingly, optimal workflows involve a larger number of workers when the requester places a high emphasis on quality (i.e., cost has low importance). Furthermore, the utility of optimal workflows is more influenced by the average worker's ability when the requester prioritizes quality.

The above analysis can also provide an explanation for previous research findings. For instance, Kittur et al. demonstrated the significance of having a diverse pool of crowd workers in a parallel process [25]. Kamar et al. proposed that increasing the number of crowd workers is an effective strategy, particularly when the cost is relatively low [20]. Further, Little et al. revealed that prior work with poor quality

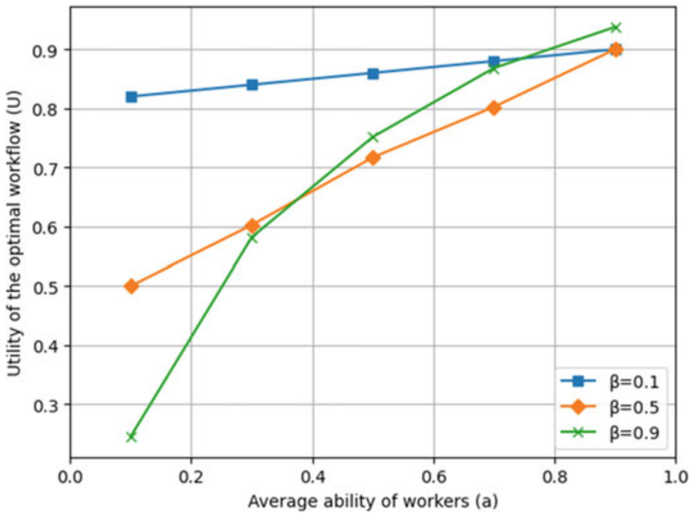


Fig. 10 Utilities of the optimal workflow in different variations of a and β

can have a negative effect on the overall quality of the workflow if the crowdsourcing task is difficult [35].

5.4 Implementing Crowdsourcing Workflow Models

Based on the proposed crowdsourcing workflow model and optimization method, we implement a system that facilitates the utilization of workflows for both task requesters and task interface developers [12].

The system consists of two modules: the workflow management module and the task interface module. The workflow management module calculates the optimal workflow by considering the average and variance of workers' abilities derived from past execution results, as well as an estimation of task difficulty. Requesters can select a workflow that they deem reasonable based on the predicted values of quality and cost. On the other hand, the task interface module is designed to cater to the needs of both requesters and workers.

While the implementation of this module may vary depending on the specific task, communication between requesters and workers remains a common feature across all tasks. The system receives input data through the task interface and communicates with the workflow management module. It is worth noting that the proposed system can be customized to suit typical translation tasks and other applications.

6 Related Work

6.1 Human Activities in Service Composition

Service composition has been a significant topic in the field of service-oriented computing for the past two decades. Various approaches, such as Petri nets, AI planning, formal models, and semantic approaches, have been proposed for service composition [14, 38, 43]. Zeng et al. introduce a multidimensional QoS model for service composition, considering attributes such as execution price, duration, reputation, successful execution rate, and availability [52]. In our work, we consider QoS attributes from both the non-functional aspects and functional aspects. Similarly, Canfora et al. consider application-specific QoS attributes along with general non-functional QoS [6]; they use an image processing workflow as an example, where resolution and color depth are considered application-specific QoS attributes. However, their work primarily focuses on overall QoS computing, while our work addresses the QoS optimization in human-in-the-loop service composition.

Human activities have been studied in the context of workflow management. Zhao et al. propose a formal model of human workflow based on BPEL4People specifications, which uses communicating sequential processes (CSP) to model a human workflow [54]. However, their model does not cover the composition of human activities and Web services. Other research has explored human workflow from the perspectives of organization management [56] and resource management [41]. Moreover, crowdsourcing has emerged as a promising approach for cost-effective task execution since the early 2010s. For instance, crowdsourcing translation has been proposed for building corpora in natural language processing, with a focus on quality management [3, 50]. While these studies discuss the possibility of replacing professional human translators with non-professional crowd workers, our research explores the integration of Web services and human activities to analyze the effects on QoS of composite services.

6.2 User-Centered Composite Service Design

Research on QoS-aware service composition has traditionally assumed that composite services are given in advance. The primary focus is then to select the most suitable set of atomic services based on QoS optimization [7, 15, 36, 46, 52, 53, 55]. Our research differs from previous studies in that we focus on designing composite services in real-world scenarios rather than selecting atomic services for given composite services.

Moreover, most of the previous work overlooks the challenges of handling QoS issues in real service composition environments. Firstly, there are situations where certain QoS attributes cannot be aggregated for composite services. For example, it is difficult to calculate the translation quality of a composite translation service by simply aggregating its component atomic services (e.g., machine translation service, morphological analysis service, dictionary service). Secondly, when multiple QoS attributes are present, maximizing all of them is challenging due to potential anti-correlated relations [2]. Thirdly, QoS values vary with the context of different service invocations, which is known as QoS uncertainty [49]. These issues become even more challenging in the human-in-the-loop composite service design. Therefore, a user-centered service design methodology is crucial when designing composite services, which is the focus of our work [28–30, 33].

6.3 Crowdsourcing Workflow Models

Crowdsourcing workflows are commonly employed to enhance the quality of challenging tasks. They were originally proposed to complete the tasks whose quality cannot be guaranteed by a single worker. Quality control of the classification or voting task by multiple workers can be regarded as the workflow of parallel processing [42]. On the other hand, the iterative process of improvement is proposed to deal with open-ended tasks. Several workflow processes have been proposed to address the issue of quality control in specific tasks. For example, Soylent utilizes the Find-Fix-Verify crowd programming pattern to improve worker quality by dividing word processing tasks into generation and review stages [5]. Zaidan and Callison-Burch propose a crowdsourcing translation workflow that achieves high-quality translations by aggregating multiple translations, redundantly editing them, and selecting the best results using machine learning [50].

Translation is used as a typical example throughout our work; it has also been a subject of study in the context of crowdsourcing. Zaidan et al. demonstrate the feasibility of crowdsourcing translation through a sequence of tasks, where workers create translation drafts, edit translated sentences, and vote to select the best translation [50]. Ambati et al. propose a combination of active learning and crowdsourcing translation to improve the quality of statistical machine translation [3]. Additionally, Aziz et al. develop and investigate a crowdsourcing-based tool for post-editing machine translations and evaluating their quality [4].

Moreover, various tools have been developed to manage the crowdsourcing of complex tasks. TurKit, for instance, is a toolkit designed for prototyping and exploring algorithmic human computation [35]. CrowdForge decomposes and recomposes complex crowdsourcing tasks based on the MapReduce algorithm [25]. Turkomatic supports task decomposition by crowd workers [26]. CrowdWeaver is a system that visually manages complex tasks and allows for task decomposition revision during execution [23]. The development of tools for modeling and managing workflows is of interest as it aligns with the objective of enhancing the understanding of crowdsourcing workflows. In contrast, our study provides a theoretical framework for the development of workflow design in crowdsourcing and offers valuable insights into the design of human-in-the-loop services as well.

7 Conclusion

This monograph summarized our research efforts on designing and analyzing human-in-the-loop service compositions. The main contributions are as follows:

- We studied composite services that compose human activities and Web services, considering both the functional and non-functional QoS attributes. To comprehensively analyze how human activities affect the QoS in such composite services, we conducted extensive experiments in the field of language translation services. Our findings indicated that the integration of human activities and Web services introduces diversity into conventional service processes. Our analysis also revealed that high-quality human activities can significantly enhance various QoS attributes of service processes, whereas low-quality human activities may have negative effects on service processes.
- We conducted an empirical study on designing human-in-the-loop composite services, considering the uncertain nature of real-world services and the need to satisfy users' QoS requirements. We proposed an iterative participatory service design process that consists of the phases of observation, modeling, implementation, and analysis. Then, we used a field study of multi-language communication service design to illustrate the effectiveness of our approach.
- We proposed theoretical approaches to understanding the crowdsourcing workflows by using an example of complex translation tasks. We modeled workers and tasks and calculated the optimal workflows. To confirm the feasibility of this model, we conducted computational experiments to calculate the optimal workflow under various parameter settings. The experiment results were also consistent with existing research. Although this study mainly focused on human activities, there is potential to incorporate the proposed crowdsourcing workflow optimization techniques into the human-in-the-loop service design.

The research presented in this monograph was carried out during the 2010s. In recent years, the emergence of cloud computing, edge computing, Internet of Things (IoT), artificial intelligence (AI), and machine learning (ML) has led to a substantial growth in the variety of service types and available services on the Internet. This development has had a significant impact on the research community of service composition.

On the other hand, the increasing demand for advanced intelligent applications in smart cities has highlighted the importance of the human-in-the-loop design methodology, particularly in the field of IoT, AI, and ML [47, 51]. We expect that the insights obtained from our previous research on human-in-the-loop service composition could contribute to these emerging fields.

Acknowledgements This research was partially supported by a Grant-in-Aid for Scientific Research (B) (21H03556, 2021–2024; 21H03561, 2021–2025) from the Japan Society for the Promotion of Science (JSPS).

References

1. Aggarwal R, Verma K, Miller J, Milnor W (2004) Constraint driven web service composition in METEOR-S. In: Proceedings of the 2004 IEEE international conference on services computing (SCC 2004), pp 23–30
2. Alrifai M, Skoutas D, Risse T (2010) Selecting skyline services for qos-based web service composition. In: Proceedings of the 19th international conference on World wide web. ACM, pp 11–20
3. Ambati V, Vogel S, Carbonell JG (2010) Active learning and crowd-sourcing for machine translation. In: Proceedings of the seventh conference on international language resources and evaluation (LREC'10), vol 11, pp 2169–2174
4. Aziz W, Castilho S, Specia L (2012) Pet: a tool for post-editing and assessing machine translation. In: Proceedings of the eighth international conference on language resources and evaluation (LREC'12), pp 3982–3987
5. Bernstein MS, Little G, Miller RC, Hartmann B, Ackerman MS, Karger DR, Crowell D, Panovich K (2010) Soylent: a word processor with a crowd inside. In: Proceedings of the 23nd annual ACM symposium on User interface software and technology. ACM, pp 313–322
6. Canfora G, Penta MD, Esposito R, Perfetto F, Villani ML (2006) Service composition (re)binding driven by application-specific qos. In: Proceedings of the 4th ICSOC, pp 141–152
7. Cao J, Huang J, Wang G, Gu J (2009) Qos and preference based web service evaluation approach. In: Eighth international conference on grid and cooperative computing (GCC 2009), pp 420–426
8. Cardoso J, Sheth A, Miller J, Arnold J, Kochut K (2004) Quality of service for workflows and web service processes. Web Seman Sci Serv Agents World Wide Web 1(3):281–308
9. Carroll JM, Rosson MB (2007) Participatory design in community informatics. Des Stud 28(3):243–261
10. Dai P, Lin CH, Weld DS (2013) Pomdp-based control of workflows for crowdsourcing. Artif Intell 202:52–85
11. Goto S, Ishida T, Lin D (2016) Understanding crowdsourcing workflow: modeling and optimizing iterative and parallel processes. In: Proceedings of the AAAI conference on human computation and crowdsourcing, vol 4, pp 52–58
12. Goto S, Ishida T, Lin D (2018) Optimizing crowdsourcing workflow for language services. In: Services computing for language resources, pp 75–89
13. Goto S, Lin D, Ishida T (2014) Crowdsourcing for evaluating machine translation quality. In: Proceedings of the ninth international conference on language resources and evaluation (LREC'14), pp 3456–3463
14. Hayyolalam V, Kazem AAP (2018) A systematic literature review on qos-aware service composition and selection in cloud environment. J Netw Comput Appl 110:52–74
15. Huang AFM, Lan CW, Yang SJH (2009) An optimal qos-based web service selection scheme. Inf Sci 179:3309–3322
16. Ishida T (2006) Language grid: an infrastructure for intercultural collaboration. In: IEEE/IPSJ symposium on applications and the internet (SAINT-06), pp 96–100
17. Ishida T (2008) Service-oriented collective intelligence for intercultural collaboration. In: IEEE/WIC/ACM international conference on web intelligence and intelligent agent technology (WI-IAT '08), vol 1, pp 4–8. https://doi.org/10.1109/WIIAT.2008.414
18. Ishida T, Murakami Y, Lin D, Nakaguchi T, Otani M (2018) Language service infrastructure on the web: the language grid. Computer 51(6):72–81
19. Ishida T, Nakajima Y, Murakami Y, Nakanishi H (2007) Augmented experiment: participatory design with multiagent simulation. In: Proceedings of the 20th international joint conference on Artifical intelligence. Morgan Kaufmann Publishers Inc., pp 1341–1346
20. Kamar E, Hacker S, Horvitz E (2012) Combining human and machine intelligence in large-scale crowdsourcing. In: Proceedings of the 11th international conference on autonomous agents and multiagent systems—vol. 1, AAMAS '12. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, pp 467–474

21. Kern R, Thies H, Satzger G (2010) Statistical quality control for human-based electronic services. In: Service-oriented computing, pp 243–257
22. Khazankin R, Psaier H, Schall D, Dustdar S (2011) Qos-based task scheduling in crowdsourcing environments. In: Service-oriented computing, pp 297–311
23. Kittur A, Khamkar S, André P, Kraut R (2012) Crowdweaver: visually managing complex crowd work. In: Proceedings of the ACM 2012 conference on computer supported cooperative work, CSCW '12. ACM, New York, NY, USA, pp 1033–1036
24. Kittur A, Nickerson JV, Bernstein M, Gerber E, Shaw A, Zimmerman J, Lease M, Horton J (2013) The future of crowd work. In: Proceedings of the 2013 conference on computer supported cooperative work, CSCW '13. ACM, New York, NY, USA, pp 1301–1318
25. Kittur A, Smus B, Khamkar S, Kraut RE (2011) Crowdforge: crowdsourcing complex work. In: Proceedings of the 24th annual ACM symposium on User interface software and technology. ACM, pp 43–52
26. Kulkarni A, Can M, Hartmann B (2012) Collaboratively crowdsourcing workflows with turkomatic. In: Proceedings of the ACM 2012 conference on computer supported cooperative work, CSCW '12. ACM, New York, NY, USA, pp 1003–1012
27. Lin D (2011) Humans in the loop of localization processes. In: The language grid: service-oriented collective intelligence for language resource interoperability, pp 201–213. Springer (2011)
28. Lin D, Ishida T (2013) Participatory service design based on user-centered qos. In: 2013 IEEE/WIC/ACM international joint conferences on web intelligence (WI) and intelligent agent technologies (IAT), vol 1. IEEE, pp 465–472
29. Lin D, Ishida T (2014) User-centered service design for multi-language knowledge communication. In: Serviceology for services: selected papers of the 1st international conference of serviceology. Springer, pp 309–317
30. Lin D, Ishida T, Murakami Y (2018) Language service design based on user-centered qos. In: Services computing for language resources, pp 125–137
31. Lin D, Ishida T, Murakami Y, Tanaka M (2011) Improving service processes with the crowds. In: Service-oriented computing-ICSOC 2011 workshops: ICSOC 2011, international workshops WESOA, NFPSLAM-SOC, and satellite events, Paphos, Cyprus, 5–8 Dec 2011. Revised Selected Papers 9. Springer, pp 295–306
32. Lin D, Ishida T, Murakami Y, Tanaka M (2014) Qos analysis for service composition by human and web services. IEICE Trans Inf Syst 97(4):762–769
33. Lin D, Ishida T, Otani M (2016) A value co-creation model for multi-language knowledge communication. In: Serviceology for designing the future: selected and edited papers of the 2nd international conference on serviceology. Springer, pp 435–447
34. Lin D, Murakami Y, Ishida T, Murakami Y, Tanaka M (2010) Composing human and machine translation services: language grid for improving localization processes. In: Proceedings of the seventh international conference on language resources and evaluation (LREC'10), pp 500–506
35. Little G, Chilton LB, Goldman M, Miller RC (2010) Exploring iterative and parallel human computation processes. In: Proceedings of the ACM SIGKDD workshop on human computation, HCOMP '10. ACM, New York, NY, USA, pp 68–76
36. Liu Y, Ngu AH, Zeng LZ (2004) Qos computation and policing in dynamic web service selection. In: Proceedings of the 13th international WWW Alt. ACM, New York, NY, USA, pp 66–73
37. Luther K, Hahn N, Dow SP, Kittur A (2015) Crowdlines: supporting synthesis of diverse information sources through crowdsourced outlines. In: Third AAAI conference on human computation and crowdsourcing, pp 110–119
38. Milanovic N, Malek M (2004) Current solutions for web service composition. IEEE Internet Comput 8(6):51–59
39. Murakami Y, Lin D, Ishida T (2014) Service-oriented architecture for interoperability of multilanguage services. In: Towards the multilingual semantic web: principles, methods and applications, pp 313–328
40. Murakami Y, Lin D, Ishida T (2018) Services computing for language resources. Springer

41. Russell N, Aalst W (2008) Work distribution and resource management in BPEL4People: capabilities and opportunities. In: Proceedings of the 20th international conference on advanced information systems engineering. Springer-Verlag Berlin, Heidelberg, pp 94–108
42. Sheng VS, Provost F, Ipeirotis PG (2008) Get another label? improving data quality and data mining using multiple, noisy labelers. In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp 614–622
43. Strunk A (2010) Qos-aware service composition: A survey. In: 2010 Eighth IEEE European conference on web services. IEEE, pp 67–74
44. Takasaki T, Mori Y, Ishida T, Otani M (2018) Youth mediated communication: knowledge transfer as intercultural communication. In: Services computing for language resources, pp 209–223
45. Takasaki T, Murakami Y, Mori Y, Ishida T (2015) Intercultural communication environment for youth and experts in agriculture support. In: 2015 International conference on culture and computing (culture computing). IEEE, pp 131–136
46. Wang X, Vitvar T, Kerrigan M, Toma I (2006) A qos-aware selection model for semantic web services. In: Proceedings of the 4th ICSOC. Springer, pp 390–401
47. Wu X, Xiao L, Sun Y, Zhang J, Ma T, He L (2022) A survey of human-in-the-loop for machine learning. Futur Gener Comput Syst 135:364–381
48. Yoon KP, Hwang CL (1995) Multiple attribute decision making: an introduction, vol 104. Sage Publications
49. Yu Q, Bouguettaya A (2010) Computing service skyline from uncertain qows. IEEE Trans Serv Comput 3(1):16–29
50. Zaidan O, Callison-Burch C (2011) Crowdsourcing translation: professional quality from non-professionals. In: ACL, pp 1220–1229
51. Zanzotto FM (2019) Human-in-the-loop artificial intelligence. J Artif Intell Res 64:243–252
52. Zeng L, Benatallah B, Ngu A, Dumas M, Kalagnanam J, Chang H, Center I, Yorktown Heights N (2004) QoS-aware middleware for web services composition. IEEE Trans Softw Eng 30(5):311–327
53. Zeng L, Lingenfelder C, Lei H, Chang H (2008) Event-driven quality of service prediction. In: Service-oriented computing-ICSOC 2008, pp 147–161
54. Zhao X, Qiu Z, Cai C, Yang H (2008) A formal model of human workflow. In: IEEE international conference on web services (ICWS'08), pp 195–202
55. Zheng Z, Ma H, Lyu MR, King I (2011) Qos-aware web service recommendation by collaborative filtering. IEEE Trans Serv Comput 4(02):140–152
56. Zur Muehlen M (2004) Organizational management in workflow applications-issues and perspectives. Inf Technol Manage 5(3):271–291

Human–Machine Collaboration for a Multilingual Service Platform

Yohei Murakami

Abstract Communication is a significant activity in a city. Especially, in a multi-ethnic country and intercultural city, multilingual communication is necessary for mutual understanding. Smart cities should provide a multilingual service platform for overcoming the language barrier. However, fewer language data causes fewer service components in low-resource languages. To augment the service components, the multilingual service platform requires effective collaboration between humans and artificial intelligence. In this chapter, we regard human activities and artificial intelligence as services and realize human–machine collaboration by composing the services dynamically. Firstly, we describe how to choose reliable human services among many unqualified ones. And then, we present a loop where data generated by human services is augmented by AI-based services and fed back to human services. Finally, we propose a planning technique that dynamically composes both human and AI services and report experimental results in Indonesia, one of the biggest multiethnic countries.

1 Introduction

1.1 Background

Globalization has caused large-scale human migration across borders and thus increased the demand for multilingual communication in an intercultural city. Although multilingual communication support is one of the most significant applications in smart cities, it is difficult to the application customized for each user activity because language resources that serve as service components are fragmented and distributed and do not provide a common access method. To address these challenges, multilingual service platforms have been constructed, such as the Language Grid and

Y. Murakami (✉)
Ritsumeikan University, 1-1-1 Nojihigashi, Kusatsu, Japan
e-mail: yohei@fc.ritsumei.ac.jp

Y. Murakami and K. Kimura (eds.), *Human-Centered Services Computing for Smart Cities*, https://doi.org/10.1007/978-981-97-0779-9_3

European Language Grid. The platforms allow users to share language resources, combine them, and integrate them into their applications.

However, the platforms mainly focus on official languages but do not support low-resourced languages sufficiently, especially ethnic languages. There are more than 7000 languages around the world, one-third of which are spoken in Asia [36]. The linguistic diversity in Asia is greater than that in Europe. Many multiethnic countries are located in Asia. For example, Indonesia, one of the typical multiethnic countries, is said to have almost 700 ethnic languages, and their ethnic languages lack digital language resources and face digital extinction. Therefore, a multilingual service platform is required as a unifying umbrella [29] and expected to support multilingual communication between ethnics in a local city as well as between global citizens in an intercultural city.

1.2 Approach

This monograph aims to construct comprehensive language resources in low-resource languages by combining crowdsourced human tasks and machine induction methods. The crowdsourced tasks create new language resources, while the machine induction augments the created language resources. To seamlessly integrate these two components, we regard each as a service and propose a dynamic service composition method that can address the uncertainties occurring in each service invocation. The existing service composition methods are classified into two types: one is a vertical service composition that achieves user's goal by combining functional requirements of services, and the other is a horizontal service composition that selects the best combination of services to execute a given plan while considering non-functional requirements. Different from these service composition methods, the proposed method optimizes the total cost by choosing the next service invocation from crowdsourced human services and machine induction services according to the results of the previous invocation as well as functional and non-functional requirements.

This section outlines the following steps for applying a collaboration of crowdsourced human services and machine induction services to the construction of multilingual services in low-resource languages.

Firstly, to improve the accuracy of the crowdsourced human services, we establish a crowdsourced workflow to make crowdsourcing services highly reliable. Quality assurance of crowdsourcing is a significant issue in an environment where a variable number of workers participate. Especially, in the creation task of language resources in low-resource languages, it is difficult to ensure enough highly reliable workers because there are fewer bilingual workers between low-resource languages. Therefore, a crowdsourced workflow is necessary, one that can identify a small number of highly reliable workers and preferentially allocate the creation tasks to these workers [4].

Secondly, to inductively create a new bilingual dictionary from two crowdsourced bilingual dictionaries, we adopt a pivot-based approach. This approach constructs a graph that connects the two bilingual dictionaries via a pivot language and identifies correct translation pairs from this graph. We formalized the identification task as a weighted max SAT problem. To ensure accuracy, we introduce semantic constraints based on language similarity. By solving this problem, we improve recall while maintaining the precision achieved by the existing inverse consultation method [21, 22]. Furthermore, to augment translation pairs in a bilingual dictionary, we employ a neural network-based approach to learn transformation rules between a source word and a target word [31, 32]. The learned rules are applied to translate a list of source words into the target language.

Finally, to achieve comprehensive coverage of bilingual dictionaries for closely related languages while minimizing total costs, it is essential to select the most suitable language pairs. This selection process entails a sequence of decisions, each with uncertainty. This uncertainty arises from the variability in dictionary induction accuracy and the size of the generated dictionary, both of which depend on language similarity and the size of pre-existing dictionaries. Therefore, we formalize the planning phase as a Markov Decision Process, enabling the generation of optimal plans [23–25].

1.3 Structure of This Chapter

Section 2 briefly introduces the background of multilingual service platforms for Asia and Europe. Also, this section discusses the requirements for each multilingual service platform by comparing Europe, which focuses on multilingualism and where people move across borders, and Asia, which has a large number of ethnic languages, most of which are digital extinctions. Asia targeted in this chapter requires a platform that involves various ethnics to collaboratively create language resources.

Section 3 designs human–machine service composition for multilingual service creation. The composite service creates translation pairs as seed data by iterating creation and evaluation tasks by crowdsourced services. Then, AI services are applied to induce new translation pairs, followed by evaluation with crowdsourced human services. This section also explains a crowdsourcing platform for collaboratively creating and evaluating translation pairs. This platform allows speakers of low-resource languages to collaboratively create and evaluate bilingual dictionaries between the low-resource languages, which are difficult to collect bilingual workers for.

Section 4 establishes a crowdsourced workflow to realize reliable crowdsourced human services. Even in an environment with a small number of highly reliable workers, this workflow can aggregate evaluation results more accurately by utilizing a hyper-question, a set of single questions. Moreover, by scoring the reliability of workers based on the evaluation results, this workflow preferentially assigns more tasks to reliable workers to improve the quality of the created language resources. We conduct experiments on simulated data to validate the workflow. The experimental results show the workflow achieves higher accuracy than other methods, regardless of the ratios of highly reliable workers.

Section 5 presents two types of AI-based services to augment language resources. One is a pivot-based method to induce translation pairs. This method creates a new bilingual dictionary between closely related languages by combining two bilingual dictionaries, which share a pivot language. To increase the recall rate compared to the existing pivot-based approach, the method is generalized to obtain translation pairs by relaxing constraints and implementing iterative induction, in which each cycle of induction is based on the previous induction results. The generalized method (64% average F-score) largely outperforms the existing method (41% average F-score). The other is a neural network-based method to acquire transformation rules of spelling between closely related languages. This method employs a two-layer Bi-LSTM encoder and LSTM decoder and compares character-based tokenization and BPE-based tokenization. Experimental results show both tokenizations achieve almost 80% precision in generating translation pairs between Indonesian and Minangkabau.

Section 6 proposes a dynamic service composition with Markov Decision Process. Manual creation of translation pairs needs to complement machine creation because low-resource languages do not have enough source dictionaries to perform the machine creation. To optimally combine AI-based creation services and crowdsourced human creation services, the composition process is modeled as Markov Decision Process (MDP) to minimize the total cost. We conducted a real experiment to create bilingual dictionaries with a minimum size threshold of 2,000 translation pairs between any combinations of 5 Indonesian ethnic languages: Indonesian, Malay, Minangkabau, Javanese, and Sundanese. The experiment result shows the proposed planning method achieves 42% cost reduction compared to an all-investment plan and is reliable: the actual total cost was 97% close to the estimated total cost.

Section 7 concludes this chapter by summarizing the results obtained through the Indonesia Language Sphere project. We also address the prospect of future research about multilingual service platforms.

2 Multilingual Service Platform for Smart Cities

In smart cities, multilingual communication support is one of the most significant applications, especially given the increased cross-border mobility resulting from globalization. To develop multilingual services, including multilingual communication support, we need a multilingual service platform that facilitates the integration of fragmented language resources. Therefore, we developed the Language Grid, a multilingual services platform for supporting intercultural collaboration [11]. This platform enables users to share various language services and combine them to create new language services customized for each user. In 2007, we initiated the operation of an experimental infrastructure to accumulate and share language resources as Web services.

In the 3-year operation of the Language Grid, we encountered difficulty in reaching service providers in other countries due to the barriers posed by geographical separation. This language locality motivated us to launch a new service grid in other countries. To address the language resource bias, we designed a federated operation of the Language Grid. In this operation model, grid operators, globally dispersed, operate local grids and facilitate service interoperability among them. Furthermore, we extended our grid architecture for interconnectivity between these local grids. This federated approach forms a network of operation centers that cover various Asian languages. Operation centers were opened in Bangkok in 2010, Jakarta in 2011, and Urumqi in 2014; they have connected themselves to us to share a variety of services in Asian languages [11]. For instance, through our federation with Bangkok, 14 Asian WordNets are now accessible. Meanwhile, Jakarta and Urumqi contribute language services for the Indonesian and Turkic language families, respectively. Currently, the Language Grid has 183 participating groups from 24 countries, collectively sharing 226 language services.

In Europe, the European Language Grid has also been constructed since 2019 [30]. The European Language Grid is a scalable cloud platform that provisions access to hundreds of commercial and non-commercial language resources for all European languages and aims to be the primary platform and marketplace for language resources in Europe. The European Language Grid harvests all relevant language resource repositories such as META-SHARE [27] and ELRC-SHARE [15, 28], collects metadata about resources and makes them available through the European Language Grid to increase in visibility of language resources. The European Language Grid now provides access to more than 14,000 commercial and non-commercial language resources.

These multilingual service platforms allow users to develop multilingual communication support services in smart cities. A multilingual medical reception support system called M^3 and SmartClassroom connecting classrooms in Japan and China were constructed with the Language Grid [18, 37], and a personal assistant named YouTwinDi that supports interaction with European citizens was developed with European Language Grid [40]. Provisioning a development environment for a new language tool that integrates the existing language resources fragmented among countries is one of the main purposes of the multilingual service platforms. On the other hand, in Asia, where more ethnic languages exist within a country than in Europe, the platforms are also required to sustainably create comprehensive language resources in various languages while involving citizens in the creation process. In a multi-ethnic country, how to support communication between different ethnic groups is a significant issue in local cities where ethnic languages are usually spoken.

Although the Language Grid and European Language Grid enhanced language service sharing and expanded language coverage, challenges persist in generating language services in low-resource languages. As per the data from LREMap in 2016, out of 5,758 entries, 1,999 resources related to English (approximately 34%, compared to 2% in 2012). This is followed by French (440 resources), German (403), Spanish (294), Chinese (218), and Japanese (196). In contrast, language resources in Indonesian are limited, with only 13 resources, and Malay has a mere 3 resources

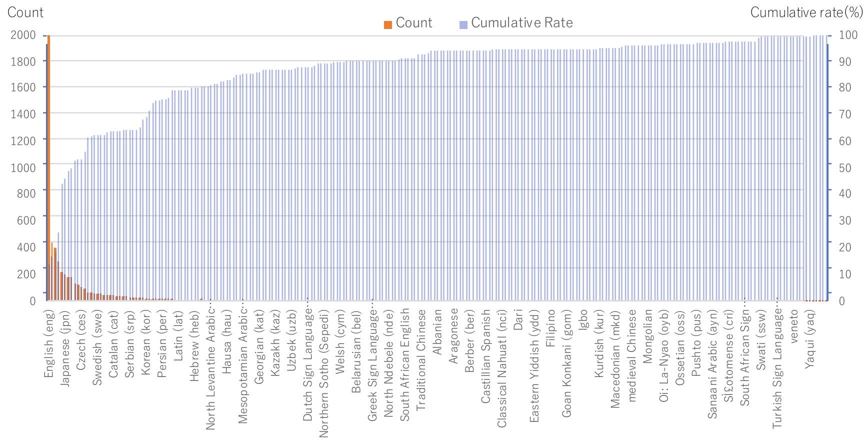


Fig. 1 LREMap: Statistics of language resources for 241 languages (from [19], licensed under CC-BY 3.0)

[2, 8]. Notably, Indonesian ethnic languages, even Javanese and Sundanese, each with over 30 million speakers, have seen no resources submitted to the top conferences related to language resources. Figure 1 shows the statistics of language resources in LREMap by language. The left vertical axis represents the number of language resources, while the right vertical axis indicates the cumulative percentage of speakers. Speakers of 11 languages, each with over 100 resources, occupy 54% of the world's population. This means the remaining speakers are not supported by adequate language resources. Therefore, we need technology to create language resources not limited to specific languages. Especially, to preserve and increase the use of Indonesia ethnic languages, we started the Indonesia Language Sphere project[1] in 2015. The purpose of this project is to develop comprehensive sets of bilingual dictionaries among Indonesian ethnic languages, which are closely related languages.

3 Human–Machine Service Composition

3.1 Collaborative Creation Workflow

Manual creation of language resources is essential to develop multilingual services in low-resource languages. To assure the quality of data, the created language resources need to be subsequently evaluated by other workers. In this manual creation process, reducing the total costs is a challenge while ensuring the quality of the language resources due to the high costs associated with manual creation and subsequent evaluation. Furthermore, augmenting manually created language resources with machine-

[1] https://langsphere.org.

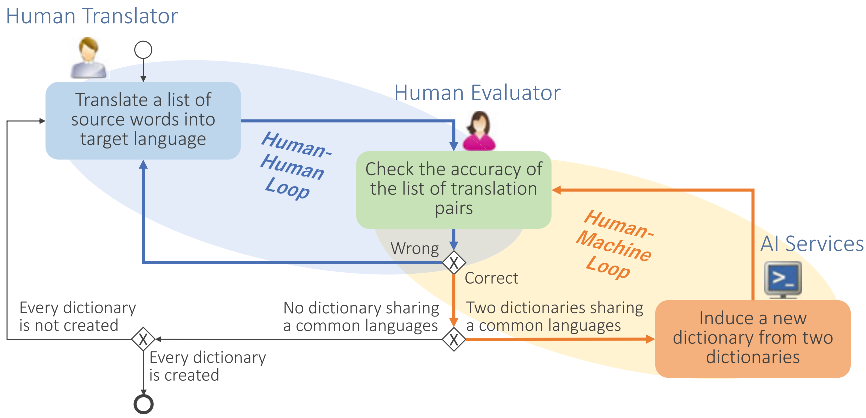


Fig. 2 Human–machine collaborative creation workflow

induced data becomes significant in increasing the size of language resources without proportionally increasing the total costs.

Therefore, we have constructed a human–machine collaboration workflow that combines a loop of manual creation and evaluation (called *human–human loop*) with a loop of machine induction and manual evaluation (called *human–machine loop*), as illustrated in Fig. 2. The human–human loop continues to modify mistranslations until sufficient seed data is obtained. Once it creates enough seed data, the data is utilized to induce a language resource. The induced data is manually evaluated, and any incorrect results are either manually modified or filtered out in the human–machine loop.

In the human–human loop, finding highly reliable workers is challenging because fewer bilingual speakers can create and evaluate translations. Although crowdsourcing, which allows us to request tasks from a variable number of workers on the Internet, is one possible solution, securing many highly reliable workers remains difficult. Therefore, we need a crowdsourced workflow that can create translation pairs at low costs, regardless of the ratio of highly reliable workers. To solve this problem, Sect. 4 proposes a crowdsourced workflow using hyper-questions, a technique designed to generate more informative responses from workers.

In the human–machine loop, machine induction methods cannot expect a large amount of training data as usual due to the nature of low-resource languages. Therefore, we need to augment language resources by using domain knowledge that the target languages are closely related and belong to the same language family. Specifically, Sect. 5 presents a pivot-based approach and a neural network approach. The former focuses on cognates originating from the same word in a proto-language, and the latter utilizes the similarity of spelling between the closely related languages to acquire transformation rules of spelling.

The total cost of this workflow varies according to which language pairs are crowdsourced and which language pairs are induced by the machine. For example, the cost of manual creation and evaluation depends on the number of highly reliable workers. Low language similarity decreases the accuracy of machine induction methods, which results in low cost-effectiveness due to the evaluation costs of mistranslations. Therefore, Sect. 6 describes a plan optimization method that selects which language pairs are targeted by crowdsourcing or by machine induction methods to minimize the total cost.

3.2 Crowdsourcing System for Language Resource Creation

To manually create and evaluate language resources in the human–human loop, we developed a crowdsourcing system [24]. This system enables a task requester to upload a list of headwords for creating a bilingual dictionary. The requester can assign translation creation and evaluation tasks to workers proficient in both languages for each headword. Each task progresses through eight states: pre-creation assignment, creation assignment, creation in progress, creation completion, pre-evaluation assignment, evaluation assignment, evaluation in progress, and evaluation completion, which are monitored by the task requester. After the creation of the requested translation pairs, the task requester can assign the evaluation task to the other workers. Once all of the evaluation results are collected, the task requester aggregates them to determine the final evaluation result. If incorrect, the task state reverts to pre-creation assignment, enabling re-assignment of the translation creation task.

Workers can manage their own assigned creation tasks and evaluation tasks on the system. When the tasks are assigned, they appear in the worker's management console, separated by task types such as creation and evaluation. As shown in Fig. 3, a headword (iklim (*climate*)) in a source language (Indonesian) is displayed in the creation task tab, and workers can register its translation (Cuaca (*weather*)) in a target language (Palembang). When the created translation pairs are accumulated, the task requester or the system generates their evaluation tasks and assigns them to workers different from the creators. As illustrated by Fig. 4, a translation pair (gelas (*glass*) and Cangkir (*cup*)) then appears in the evaluation task tab and is evaluated as correct (BENAR) or incorrect (SALAH) by the workers.

In addition to the individual tasks, the system facilitates collaborative tasks addressed by several workers collaboratively. This system also displays the meaning of the headword as a reference. This is particularly useful when creating and evaluating translations between two low-resource languages where bilingual workers may be scarce. For example, two workers, each understanding a different low-resource language, can communicate the meaning of the target word to each other and collaboratively create and evaluate its translation pair.

Fig. 3 User interface for creation tasks

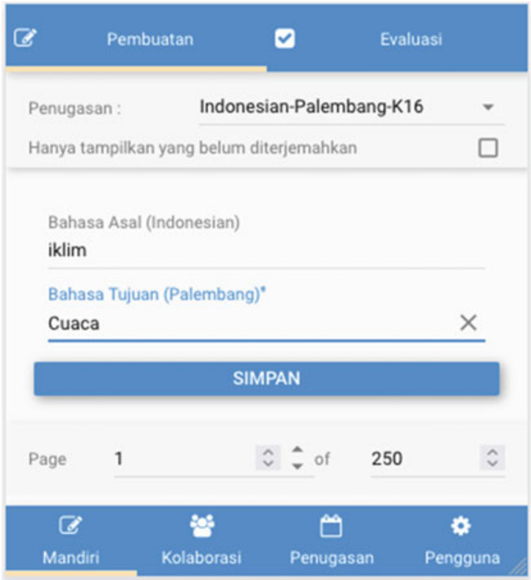
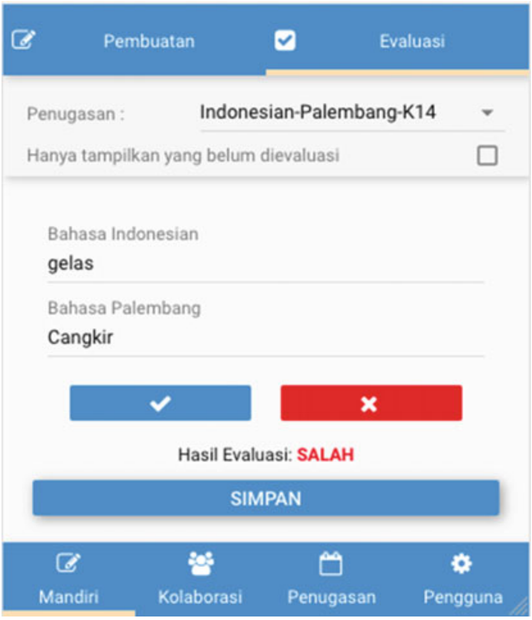


Fig. 4 User interface for evaluation tasks



4 Reliable Crowdsourced Services for Creating Language Resources

4.1 Introduction

Crowdsourcing is a service for requesting work from a large and open group of people via the Internet, and it can be used to order a large number of works that require human labor. Crowdsourced service is especially used to request relatively difficult tasks for computers but not so difficult for humans. However, in crowdsourcing, where the tasks are executed by an unspecified number of workers, the abilities of whom vary, it is difficult to guarantee the quality of the execution results. Especially, in the case of bilingual dictionary creation between low-resource languages [19], the number of people who can speak multiple low-resource languages is limited, and the average ability of workers is low. This results in the method of assigning the same task to multiple workers and using majority voting has a high possibility of obtaining wrong answers, and quality control cannot be performed well.

Therefore, we aim to improve quality in an environment with a small number of highly reliable workers by using an answer aggregation method on hyper-questions (multiple tasks considered together as one task). Since workers with high ability tend to agree on the answers to hyper-questions, the method increases the possibility that workers with high ability will be in the majority. To this end, we address the following two problems.

Selecting highly reliable evaluators In the answer aggregation method on hyper-questions, it is assumed that a small number of high-quality workers are involved. Therefore, it is necessary to select highly reliable evaluators from a crowd.

Reducing the number of tasks Even if a worker is able to correctly evaluate whether a translation pair is correct or not, in the case of wrong translation pairs, the worker may have to redo the translation, which increases the number of tasks.

For these problems, we dynamically evaluate the reliability of workers based on their work results, and selected workers who were estimated to be highly skilled. Specifically, we set a parameter "Reliability" for each worker and increased or decreased the reliability based on the task results. In addition, we adjust the probability of task assignment based on the reliability of each worker.

4.2 Issues in Crowdsourcing

4.2.1 Quality Control

One of the most important research topics in crowdsourcing is quality control. Since tasks are performed by humans, it is not always possible to obtain correct results. In addition, since tasks are requested from an unspecified number of people, there is a possibility that workers with low ability or workers who intentionally perform low-quality work (spammers) will perform tasks. Therefore, the quality of the results cannot be guaranteed only by the results of a single worker. In the research of quality control, there are two main approaches: an approach to aggregate work results for improving the overall quality and an approach to improve the quality of individual work results.

The former is mainly an approach that attempts to obtain high-quality results by removing errors from the work results. As an example, the method of assigning the same task to multiple workers, and then taking a majority vote is used. However, the majority voting can lead to the correct answer when the ability of the workers is high, while it is difficult to obtain the correct answer when the ability of the workers is low (less than 50% correct in the case of binary choice type tasks) [35]. For such cases where experts are in the minority, an answer aggregation method using hyper-questions has been proposed as an effective method [14]. A hyper-question is a set of single questions, in which multiple questions are considered together as one. Since experts are more likely to agree on the answers to multiple questions than non-experts, majority voting on hyper-questions is particularly effective when there are few workers with high ability.

The latter is an approach that attempts to improve the results of task execution itself by designing rewards and tasks or selecting workers before requesting workers to perform tasks. Especially, the method of extracting workers who are estimated to have high ability in advance and assigning tasks to the extracted workers is expected to improve the quality of the work results, because it can eliminate low-ability workers and spammers before executing the task, and only workers who are estimated to have a high ability can actually perform the task.

4.2.2 Task Assignment

In the task assignment, it is necessary to estimate the abilities of workers in advance in order to extract workers who can be expected to deliver high-quality work results. However, it is difficult to know the abilities of workers in advance because the abilities of workers in crowdsourcing vary widely.

Therefore, a method of detecting workers with high ability by using a task the correct answer of which is known in advance (gold task) has been used. For example, there are two methods: one is to assign gold tasks in advance and filter workers by evaluating their answers, and the other is to blend gold tasks into normal tasks to measure and select the ability of workers [12]. When a worker is judged to have a low ability by these methods, it is possible to take countermeasures such as not assigning tasks to the worker afterward, placing restrictions on some tasks, or not using the results of the worker's output. These methods are considered to be the most effective ways of estimating the abilities of workers when the average ability of workers is not high. However, if the gold tasks are mixed in with the actual tasks, the reward for answering the gold tasks, whose answers are already known, must be paid, which reduces the cost-effectiveness of the method. In the case of measuring workers' abilities in advance, it is necessary to assign gold tasks to all workers, which simply reduces the efficiency of the workload. Furthermore, it is known that it is very difficult and costly to generate gold tasks, so a method to automatically generate gold tasks based on data collected has been proposed [26]

In this paper, we assume the bilingual dictionaries creation using crowdsourcing in low-resource languages. Therefore, the number of workers who can speak these languages is small, and the average ability of workers is not high. Therefore, we aim to improve the quality of the created bilingual dictionary by combining an answer aggregation method that is effective even for such a crowd with low average ability and a task assignment method based on workers' reliability calculated from the results of each worker's work.

4.3 Crowdsourced Workflow

Considering a workflow consisting of a creation task and multiple evaluation tasks (Fig. 5), we ensure redundancy by performing multiple evaluation tasks for each bilingual creation task. In other words, the final evaluation of the translation pair produced by a creation task is determined by a majority vote on the results of evaluation tasks. If a "Correct" translation pair is produced and it is evaluated "Correctly," the "Correct" translation pair is obtained. If a "Wrong" translation pair is produced and it is evaluated "Wrongly," the "Wrong" translation is obtained. Otherwise, the translation pair is ignored. If no translation pair is obtained, the process is repeated from a creation task until translation pairs for all words are obtained.

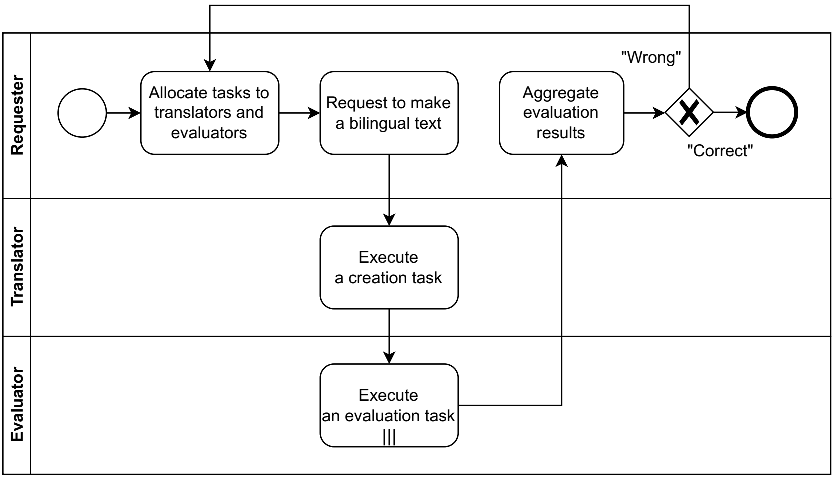


Fig. 5 Workflow for bilingual dictionary creation

We assume that there are two types of tasks assigned to workers: a creation task, which is a free-input task to create a translation from a given word or sentence, and an evaluation task, which is a binary-choice task to evaluate whether the translation created by the previous task is "Correct" or "Wrong."

4.4 Evaluation Aggregation with Hyper-Questions

The common aggregation methods, such as majority voting, often fail when the majority of workers do not know the correct answers. To emphasize the answers of a few high-quality workers, the aggregation method on hyper-questions was proposed [14]. A hyper-question consists of a subset of original single questions, and an answer to a hyper-question is a set of answers to the questions included in the hyper-question. A set of k original single questions is defined as a k-hyper-question. As the specific answer aggregation method on hyper-questions, we use majority voting on hyper-questions for evaluation tasks.

Given a set of some evaluation tasks Q, our evaluation method constructs k-hyper-questions by combining single evaluation tasks in Q. Then conducting a majority voting for each hyper-question results in an answer to the hyper-question. The aggregated results of the hyper-questions are decoded into answers to the single questions. Finally, another round of majority voting is carried out for each question. Consequently, the results of the first round of majority voting on hyper-questions are aggregated to obtain the final answer for every single question.

Figure 6 shows the procedure of majority voting on hyper-questions, which consists of five evaluators e_1, e_2, e_3, e_4, and e_5, and four evaluation tasks q_1, q_2, q_3, and q_4 in which the evaluators determine whether each translation pair (Indonesian–Minangkabau) is "P(correct)" or "N(wrong)." In this example, k is set to 3. "P" is the correct answer for all of the evaluation tasks. In the first step, four 3-hyper-questions are created from the four evaluation tasks. An answer to a hyper-question is the concatenation of the answers to the constituent single evaluation task. In the second step, majority voting for each hyper-question; in this case, the answer "PPP" is chosen for the first three hyper-questions, and the answer for the last one is not determined. In the third step, each of the majority answers to the hyper-questions votes for the single evaluation task included in it. Finally, in the fourth step, another round of majority voting aggregates the votes to the single evaluation task to obtain the final answers. Simple majority voting fails in the evaluation task q_2, but majority voting on hyper-questions succeeds. If there are no majority answers in the second step and some of the single evaluation tasks do not get the final answers, another round of majority voting is taken among the evaluators who voted majority answers for the rest of the evaluation tasks. By narrowing the evaluators and reusing the evaluation results from the narrowed evaluators, we can reduce the number of evaluation tasks.

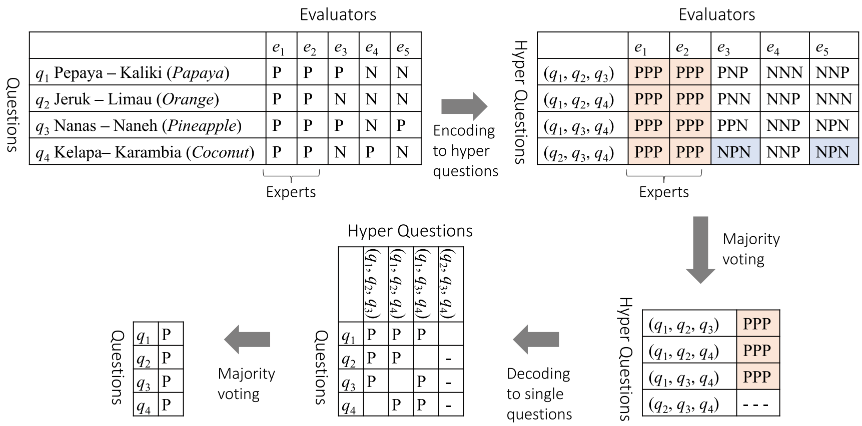


Fig. 6 Example of majority voting on hyper-questions procedure

4.5 Task Assignment Based on Workers' Reliability

In this research, we aim to improve the quality and reduce the cost of crowdsourcing by identifying workers who are estimated to be highly skilled based on their work results and proactively assigning tasks to them. For this purpose, we propose a method to dynamically evaluate the reliability of workers based on their work results.

4.5.1 Workers' Reliability

A "reliability" is set for each worker, and the initial value is 0. The reliability is calculated based on the results of creation tasks and evaluation tasks as follows.

- If the translation pair created by a creation task is evaluated as "correct" by evaluation tasks, the reliability of the translator is increased by $+1$.
- If the translation pair created by a creation task is evaluated as "wrong" by evaluation tasks, the reliability of the translator is increased by -1.
- If a worker's evaluation of all the created translation pairs in a given task set Q is a majority of the final evaluation obtained from the aggregation of the evaluation tasks, the reliability of the evaluator is increased by $+1$.
- If a worker's evaluation of all the created translation pairs in a given task set Q is a minority of the final evaluation obtained from the aggregation of the evaluation tasks, the reliability of the evaluator is increased by -1.

This calculation is performed each time the evaluation of all the created translations in one problem set Q is completed.

4.5.2 Task Assignment

By using the reliability of each worker, we proposed two types of task assignment methods:

- Assigning evaluation task using a threshold
- Task assignment using weighted probabilities.

For the first method, we placed restrictions on workers to allocate evaluation tasks. For the bilingual evaluation task, we consider a worker whose reliability is 1 or higher to be a trusted worker, and only trusted workers can perform evaluation tasks. This method is expected to reduce the number of errors in evaluation tasks.

For the second method, the probability of task assignment for both creation tasks and evaluation tasks is adjusted based on the weight of each worker using his/her reliability. When the total number of workers who can perform a task is n, the weight w_i of the ith worker is calculated as in Eq. (1).

$$w_i = 1 + r_i - r_{min} \tag{1}$$

The r_i shows the reliability of the ith worker, and the r_{min} is the lowest reliability among all workers who can perform the task. By calculating the weight as in Eq. (1), we can avoid that the weight of the worker with the lowest reliability becomes 0 (the probability of being assigned the task becomes 0). As the work progresses, the difference in the weights increases as the difference in the reliability among the workers becomes larger.

The probability that a task is assigned to a worker, p_i, can be calculated by using weights, as in Eq. (2).

$$p_i = \frac{w_i}{w_1 + w_2 + w_3 + \cdots + w_i + \cdots + w_n} \tag{2}$$

By performing these calculations each time a task is assigned, we can make it easier to assign a task to a worker with high reliability (a worker who is estimated to be highly capable) and harder to assign a task to a worker with low reliability (a worker who is estimated to be less capable), thereby automatically eliminating workers who are estimated to be less capable. This can be expected to improve accuracy and reduce costs.

4.6 Evaluation

4.6.1 Models

For the evaluation, we modeled crowdsourcing workers and tasks for creating a bilingual dictionary between low-resource languages.

Workers

The higher the ability of the worker, the quality of the task execution result is higher. In this paper, the ability of a worker is defined as the vocabulary in multiple languages and is represented by $x (0 \leq x \leq 1)$. When x is closer to 1, the worker recognizes more vocabulary, and the more likely he/she is to perform the task correctly. On the other hand, when x is closer to 0, the worker recognizes less vocabulary and the possibility that the task will be incorrect increases. For simplicity, we assume that the quality of the task execution result is probabilistically determined by the ability of a worker. In this paper, we follow previous studies and represent the ability of a worker using a beta distribution. The probability density function $f(x|a, v)$ is represented by Eq. 3 [7].

$$f(x|a, v) = \mathrm{Beta}(\frac{a}{\min(a, 1-a)v}, \frac{1-a}{\min(a, 1-a)v}) \quad (3)$$

$a \in (0, 1)$ is the normalized value of workers' ability and $v \in (0, 1)$ is the parameter that determines the variance of workers' ability. When v is closer to 0, the variance is closer to 0, and when v is closer to 1, the variance in the beta distribution with the average a is larger. The above model of workers was adopted by [7].

Tasks

We assume that the result of a creation task is "Correct" if the worker knows the translation of the given word and "Wrong" if the worker does not know the translation of the given word. Therefore, it is completely dependent on the ability of the worker whether a correct translation pair is produced or not (Fig. 7). However, since an evaluation task is a binary choice task, if the worker knows the correct translation for a given word, he/she will evaluate it as "Correct." However, if the worker does not know the translation of the word, he/she will randomly select one of the two values "Correct" or "Wrong" (Fig. 8). Therefore, in an evaluation task, no matter how low the ability of the worker is, it is guaranteed that the worker will make a "Correct" evaluation with a probability of more than 50%.

4.6.2 Evaluation Method

The methods, including the proposed method, are evaluated in terms of the accuracy of the produced translation pairs and the work quantity required to obtain all the translation pairs.

Proposed Method 1 (Reliable_hyper_reuse) A model that combines the answer aggregation on hyper-questions and the task assignment based on workers' reliability. In the case of failure of majority voting using hyper-questions, this model

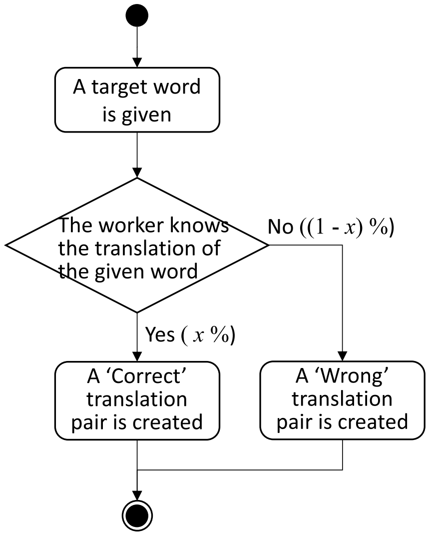


Fig. 7 A creation task model

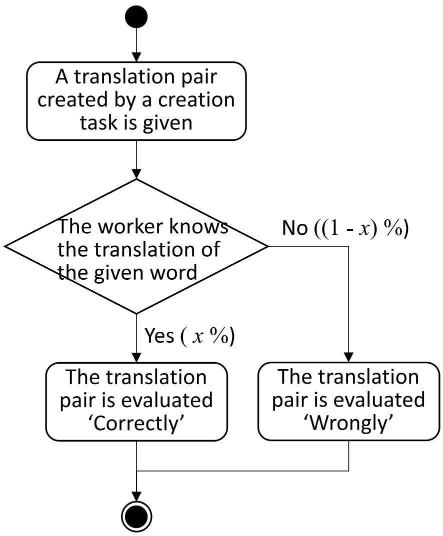


Fig. 8 An evaluation task model

takes another majority voting by reusing the evaluation results from the evaluators who voted majority answers for the successful evaluation tasks.

Proposed Method 2(Reliable_hyper) A model that combines the answer aggregation on hyper-questions and the task assignment based on workers' reliability.

Comparison Method 1 (Random_hyper) A model that combines the answer aggregation on hyper-questions and the random task assignment for the entire workers.

Comparison Method 2 (Reliable) A model that combines a simple majority voting in evaluation tasks and the task assignment based on workers' reliability.

Comparison Method 3 (Random) A model that combines a simple majority voting in evaluation tasks and the random task assignment for the entire workers.

In order to measure the performance of each method described above, we use the following indicators.

1. Accuracy of the produced translation pairs
   The accuracy of the produced translation pairs by each method is calculated as follows:

$$\text{Accuracy} = \frac{\text{Number of translation pairs produced correctly}}{\text{Total number of obtained translation pairs}} \tag{4}$$

   This indicator helps to compare the simple quality of the outputs from each method.
2. Work quantity required to obtain all the translation pairs.
   The work quantity is the total unit times of the creation tasks and the evaluation tasks, which are executed until all the translation pairs are obtained. A unit time is calculated from the estimated time taken for doing the task. Since creation tasks are more difficult than evaluation tasks, we defined that a creation task takes 3 units and an evaluation task takes 1 unit. The cost model was adopted by [25]. This indicator helps to compare the efficiency and cost of each method.

In order to evaluate the indicators described above, we conducted simulations using each method. We set the number of workers to 20 and assumed that there were 1,000 target words. The ability of each worker is determined based on the model in 4.6.1, and we varied the average of workers' abilities between 0.2 and 0.7 with a variance of 0.5. To eliminate bias due to random numbers, we used the average of the results of 100 simulations for each method.

4.6.3 Results

The accuracy of the proposed methods, Reliable_hyper_reuse and Reliable_hyper, were almost the same and the highest, followed by Reliable_hyper_reuse, Reliable, Random_hyper, and Random. The difference in accuracy between the proposed methods and Reliable, the second highest, was about 5–10%, as illustrated in Fig. 9.

The work quantity tended to be larger for Reliable_hyper_reuse, Reliable_hyper, and Random_hyper, which are the models using the answer aggregation on hyper-questions. However, for Reliable_hyper_reuse, the work quantity was the smallest when the average of the workers' ability was 0.5 or higher, as shown in Fig. 10, illustrating the cost reduction by reusing the evaluation results of reliable workers.

Both Reliable and Random_hyper were more accurate than Random, indicating that the task assignment based on workers' reliability and the answer aggregation

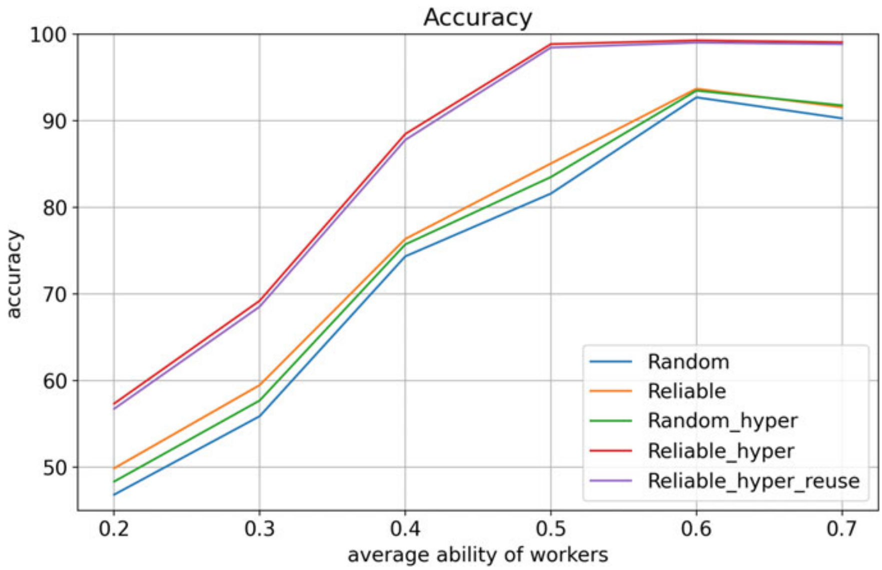


Fig. 9 Accuracy

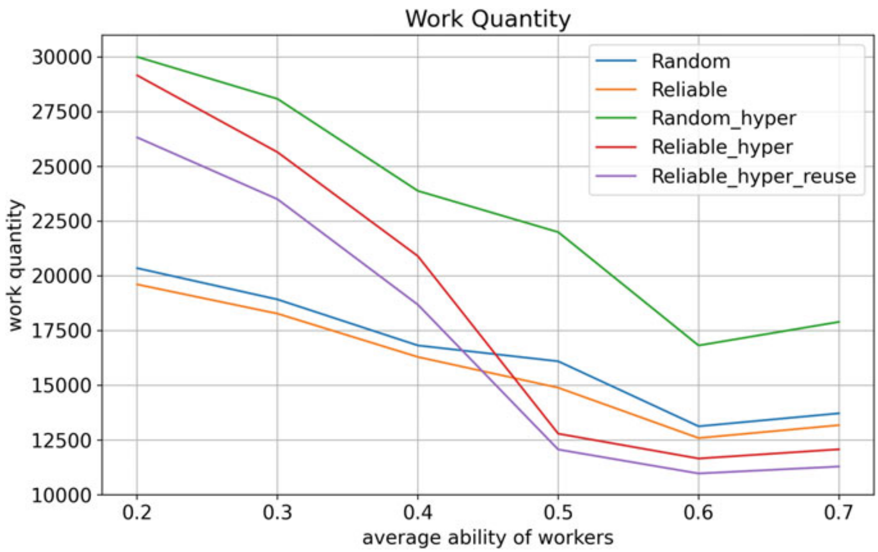


Fig. 10 Work quantity

on hyper-questions are effective. In addition, when we compared Reliable and Random_hyper, the accuracy of Reliable was higher than that of Random_hyper, indicating that it is more effective to assign tasks to workers with high reliability than to improve the quality of answer aggregation. Furthermore, the accuracy of Reliable_hyper_reuse and Reliable_hyper, which combine the task assignment based on workers' reliability and the answer aggregation on hyper-questions, were particularly high, indicating that these methods are more effective when combined than when used individually.

Since the work quantity for Reliable_hyper_reuse, Reliable_hyper, and Random_hyper, which use the answer aggregation on hyper-questions, tended to be larger, it is easy to assume that many redos occurred. This may be because the majority voting on hyper-questions makes it more difficult to reach an agreement than in simple majority voting. Therefore, the evaluation aggregations often fail. However, when the average worker's ability was 0.5 or higher, the work quantity for Reliable_hyper_reuse and Reliable_hyper got lower rapidly. This shows that if evaluation tasks can be assigned to high-quality workers from a crowd with more than a certain number of high-ability workers, the majority voting on hyper-questions is more likely to be successful, and redoing the task is less likely to occur. Furthermore, in Reliable_hyper_reuse and Reliable_hyper, creation tasks are also assigned preferentially to the worker with the highest reliability, resulting in few wrong translation pairs created in the first place. Regarding the number of reliable workers whose abilities are more than 0.7, there were two reliable workers when the average of workers' abilities was 0.4, and there were four reliable workers when the average of workers' abilities was 0.5. This shows that two reliable workers are too few to assign evaluation tasks as well as creation tasks to them, which results in the majority voting on hyper-questions not working well even if they perform creation tasks very well.

5 AI Services for Augmenting Language Resources

5.1 Introduction

Crowdsourced bilingual dictionary creation between low-resource languages is challenging, especially for languages with fewer speakers. This challenge is primarily due to high manual costs and the scarcity of bilingual workers. Numerous studies have explored the semi-automatic or automatic creation of bilingual lexicons, leveraging various language resources such as parallel corpora, comparable corpora, WordNet, and existing bilingual dictionaries. However, these methods often fail when applied to low-resource languages, which typically lack substantial parallel corpora. To address this issue, this section proposes two machine induction methods that utilize small existing bilingual dictionaries as seed data. The first method is a pivot-based approach. It generates a new bilingual dictionary by linking two existing dictionaries through a pivot language. However, this approach must address the inherent ambiguity caused by polysemous words in the pivot language when identifying correct translation pairs between the languages. The second method is a neural network approach that infers spelling transformation rules from the seed data based on the orthographic similarity of cognates.

5.2 Pivot-Based Approach

The pivot-based approach is commonly used in bilingual dictionary induction, especially when the only available language resources are dictionaries. This method constructs a graph, termed a "*transgraph*," by connecting two bilingual dictionaries via a shared pivot language. To model a transgraph, we utilize a tripartite graph. Figure 11 illustrates an example of a transgraph between language A and C via pivot language B. each vertex denotes a word, while each edge represents a translation relation between two vertices. In the basic form of a transgraph, every pivot vertex must be linked to at least one non-pivot vertex and be interconnected through non-pivot vertices. Transgraphs are merged when there exists at least one edge connecting a pivot vertex in one transgraph to a non-pivot vertex in the other. From this graph, reachable word pairs between two non-pivot languages are extracted as "*translation pair candidates*," such as pairs (w_1^A, w_1^C), (w_1^A, w_2^C), (w_2^A, w_1^C), (w_2^A, w_2^C), (w_3^A, w_1^C), and (w_3^A, w_2^C). Subsequently, correct translation pairs are identified from these candidates. Wushouer et al. formalized the pivot-based bilingual dictionary induction as an optimization problem [45]. They assumed that translation pairs between closely related languages were one-to-one mapping and cognates (words originating from the same word in a proto-language). Based on this assumption, they solved the constraint optimization problem to induce a Uyghur-Kazakh bilingual dictionary using Chinese as the pivot language. In this research, we aim to develop a generalized framework for the constraint-based bilingual dictionary by relaxing the existing one-to-one mapping assumption into the many-to-many assumption.

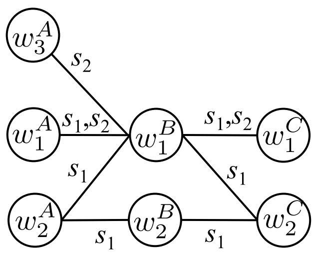


Fig. 11 Example of transgraph

5.2.1 Symmetry Assumption

Given that dictionaries incorporating sense information, as denoted by s_1 and s_2 in Fig. 11, correct translation pairs can be readily derived from a transgraph by identifying cognate pairs, each pair of which has a complete overlap in their senses. For instance, the cognate pair (w_1^A, w_1^C) shares two senses, namely s_1 and s_2, through the pivot word w_1^B. Also, the cognate pair (w_2^A, w_2^C) shares only the sense s_1 through the pivot words w_1^B and w_2^B. However, available machine-readable bilingual dictionaries with sense information are limited, especially for low-resource languages. Therefore, we assume that connected words share at least one sense. Furthermore, non-pivot words symmetrically connected through pivot word(s) are presumed to share all their senses and are thus identified as cognates. In Fig. 11, the pairs of (w_1^A, w_1^C), (w_3^A, w_1^C), and (w_2^A, w_2^C) are regarded as cognates. We employ this symmetry assumption for extracting cognates between closely related languages because most linguists argue that lexical comparison alone is insufficient for cognate identification [3].

5.2.2 N-Cycle Symmetry Assumption

Machine-readable bilingual dictionaries for low-resource languages are often limited in size and lack the desired quality. Such dictionaries may miss translation relations essential for constructing a symmetrical topology in a transgraph. Figure 12 illustrates an asymmetry transgraph, where the dashed edge, (w_2^B, w_1^C), is expected to be a missing translation relation. The pivot-based approach adds these missing edges to a transgraph with some costs to satisfy the symmetry assumption.

The existing one-to-one approach identifies missing edges only once to ensure the symmetry assumption of initial translation pair candidates linked by solid edges. In Fig. 13a, the five translation pair candidates are extracted, and the four missing dashed edges are identified to satisfy the symmetry assumption of all the candidates. Since this compensation for missing edges is limited to only initial translation pairs, we call this "one-cycle symmetry assumption." To apply this compensation to new translation pair candidates linked by the added edges, we iterate the one-cycle symmetry assumption n times, called the "n-cycle symmetry assumption." Figure 13b illustrates the second cycle after Fig. 13a. The three more candidates, 6, 7, and 8, are extracted from the previously added solid edges. Users can specify the maximum number of iterations for the experiment.

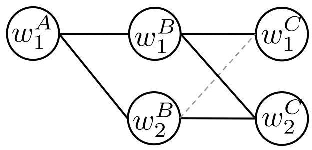
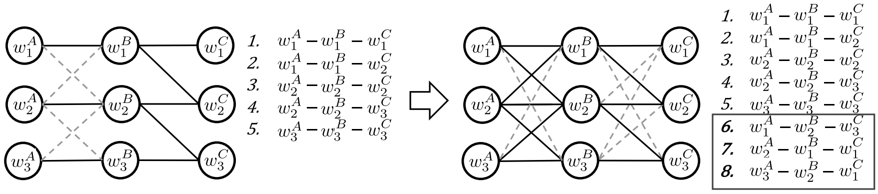


Fig. 12 Asymmetry transgraph



(a) Translation pair candidates from **existing solid edges** in 1-cycle symmetry assumption

(b) Translation pair candidates from **existing** and **previously added solid edges** in 2-cycle symmetry assumption

Fig. 13 N-cycle symmetry assumption extension

5.2.3 Formalization

Constraint optimization problems have been commonly introduced into many natural language processing and web service composition problems [9, 16]. Wushouer et al. [45] applied a Weighted Partial MaxSAT (WPMaxSAT) to a bilingual dictionary induction. Following them, we also adopted CNF encoding in our formalization [1]. A literal is defined as either Boolean variable x or its negation $\neg x$ and a clause C as a disjunction of literals $x_1 \vee \ldots \vee x_n$. In weighting a clause C, we represent it as a pair (C, ω), where ω, a weight, denotes the penalty for violating the clause C. In the case of a hard clause, infinity (∞) is assigned as a weight. A propositional formula φ_c^ω is a conjunction of one or more clauses $C_1 \wedge \ldots \wedge C_n$. A formula with soft clauses and one with hard clauses are represented as φ_c^+ and φ_c^∞, respectively. A WPMaxSAT problem comprises multiple formulae φ_c^ω. The solution of the WPMaxSAT problem provides an optimal assignment to the variables in C, resulting in the minimal cost of that assignment.

To apply the WPMaxSAT to bilingual dictionary induction, we introduced two types of variables for the literal: e and c. e indicates edge existence between a given word pair, while c represents cognates for a given word pair. For instance, the edge existence between word w_i^A in language$A$ and word w_j^B in language B is denoted by $e(w_i^A, w_j^B)$, and the cognate pair between words w_i^A and w_j^B by $c(w_i^A, w_j^B)$.

To represent various word pairs for e and c, we define five sets of word pairs: E_E, E_N, D_C, D_{Co}, and D_R. The first two sets focus on the existence of edges. E_E and E_N are a set of word pairs connected by existing edges and missing edges, respectively. In contrast, the rest three sets are related to translation pairs between non-pivot languages. Specifically, D_C denotes a set of translation pair candidates, D_{Co} signifies a set of cognate pairs, and D_R indicates a set of all the translation pairs identified by the WPMaxSAT solver.

5.2.4 Heuristics to Find Cognate

We have introduced two heuristics to the cognate identification modeled by the WPMaxSAT problem: cognate pair coexistence probability and cognate form similarity.

Cognate Pair Coexistence Probability

In assessing the likelihood that a translation pair candidate $t(w_i^A, w_k^C)$ is a cognate pair $c(w_i^A, w_k^C)$, we calculate the cognate coexistence probability, denoted as H_{coex}. This probability is derived by multiplying two chain rules, as given in Eqs. (5) and (6), which results in Eq. (7). A marginal probability $P(w_i^A)$ represents the likelihood that w_i^A connects to any word in language C. A conditional probability $P(w_i^A|w_k^C)$ indicates the likelihood that w_k^C connects to w_i^A when w_k^C connects to any word in language A. A joint probability $P(w_i^A, w_k^C)$ signifies the likelihood that w_i^A and w_k^C are interconnected. $P(w_i^A)$ and $P(w_k^C)$ are independent because they are from different bilingual dictionaries. Thus, $P(w_k^C, w_i^A) = P(w_i^A)P(w_k^C)$ and Eq. (7) can be converted to Eq. (8). To calculate $P(w_i^A|w_k^C)$ and $P(w_k^C|w_i^A)$, we employ a generative probabilistic process which is commonly used in previous works [5, 20, 33, 43] in Eq. (9).

$$P(w_i^A, w_k^C) = P(w_k^C|w_i^A)P(w_i^A) \tag{5}$$

$$P(w_k^C, w_i^A) = P(w_i^A|w_k^C)P(w_k^C) \tag{6}$$

$$P(w_i^A, w_k^C)P(w_k^C, w_i^A) = P(w_i^A|w_k^C)P(w_k^C|w_i^A)P(w_i^A)P(w_k^C) \tag{7}$$

$$P(w_i^A, w_k^C) = P(w_i^A|w_k^C)P(w_k^C|w_i^A) \tag{8}$$

$$P(w_i^A|w_k^C) = \sum_{j=0} P(w_i^A|w_j^B)P(w_j^B|w_k^C) \tag{9}$$

Cognate Form Similarity

The symmetry assumption may sometimes fail to identify the correct cognate from the translation pair candidates when a pivot word has multiple in-degrees/out-degrees. To correctly identify cognates, not only the word sense represented by edges but also the word form is useful. We, therefore, calculate cognate form similarity $H_{formSim}$ of the translation candidate $t(w_i^A, w_k^C)$ using the Longest Common Subsequent Ratio (LCSR). This ratio ranges from 0 (0% form-similarity) to 1 (100% form-similarity) [17]. In Eq. (10), $LCS(w_i^A, w_k^C)$ is the longest common subsequence of w_i^A and w_k^C; $|x|$ is the length of x; and $max(|w_i^A|, |w_k^C|)$ returns the longest length.

$$LCSR(w_i^A, w_k^C) = \frac{|LCS(w_i^A, w_k^C)|}{max(|w_i^A|, |w_k^C|)} \tag{10}$$

$$t(w_i^A, w_k^C).H_{formSim} = LCSR(w_i^A, w_k^C) \tag{11}$$

5.2.5 Constraints to Identify Cognates

All the constraints for the WPMaxSAT are summarized in Table 1.

Edge Existence. In the transgraph, there exists an edge between words that share similar meanings. Edges that currently exist in the transgraph are encoded as TRUE in the CNF formula. Specifically, edges such as $e(w_i^A, w_j^B)$ and $e(w_j^B, w_k^C)$ are represented as hard constraints φ_1^∞.

Edge Non-existence. In the transgraph, there does not exist an edge between words that do not share similar meanings. This non-existence of edge is encoded as the negation of the edge existence literal in the CNF formula. Specifically, $\neg e(w_i^A, w_j^B)$ and $\neg e(w_j^B, w_k^C)$ are represented as soft constraint φ_2^+.

Symmetry. Cognates share all of their senses, resulting in a symmetrical topology via the pivot language in the transgraph. We convert

Table 1 Constraints for cognates extraction

ID	CNF formula
φ_1^∞	*Edge Existence:* $\left(\bigwedge_{(w_i^A, w_j^B)\in E_E} \left(e(w_i^A, w_j^B), \infty\right)\right) \wedge \left(\bigwedge_{(w_j^B, w_k^C)\in E_E} \left(e(w_j^B, w_k^C), \infty\right)\right)$
φ_2^+	*Edge Non-existence:* $\left(\bigwedge_{(w_i^A, w_j^B)\in E_N} \left(\neg e(w_i^A, w_j^B), \omega(w_i^A, w_j^B)\right)\right) \wedge \left(\bigwedge_{(w_j^B, w_k^C)\in E_N} \left(\neg e(w_j^B, w_k^C), \omega(w_j^B, w_k^C)\right)\right)$
φ_3^∞	*Symmetry:* $\left(\bigwedge_{\substack{(w_i^A, w_j^B)\in E_E\cup E_N \\ (w_i^A, w_k^C)\in D_C}} \left((\neg c(w_i^A, w_k^C) \vee e(w_i^A, w_j^B)), \infty\right)\right) \wedge \left(\bigwedge_{\substack{(w_j^B, w_k^C)\in E_E\cup E_N \\ (w_i^A, w_k^C)\in D_C}} \left((\neg c(w_i^A, w_k^C) \vee e(w_j^B, w_k^C)), \infty\right)\right)$
φ_4^∞	*Uniqueness:* $\left(\bigwedge_{\substack{k\neq n \\ (w_i^A, w_k^C)\in D_C \\ (w_i^A, w_n^C)\in D_C}} \left((\neg c(w_i^A, w_k^C) \vee \neg c(w_i^A, w_n^C)), \infty\right)\right) \wedge \left(\bigwedge_{\substack{i\neq m \\ (w_i^A, w_k^C)\in D_C \\ (w_i^A, w_n^C)\in D_C}} \left((\neg c(w_i^A, w_k^C) \vee \neg c(w_m^A, w_k^C)), \infty\right)\right)$
φ_5^∞	*Extracting at Least One Cognate:* $\left(\left(\bigvee_{(w_i^A, w_k^C)\notin D_R} c(w_i^A, w_k^C)\right), \infty\right)$
φ_6^∞	*Encoding Cognate:* $\bigwedge_{(w_i^A, w_k^C)\in D_{Co}} \left(c(w_i^A, w_k^C), \infty\right)$

$$c(w_i^A, w_k^C) \rightarrow e(w_i^A, w_1^B) \wedge e(w_i^A, w_2^B) \wedge \ldots \wedge e(w_i^A, w_n^B)$$
$$\wedge\, e(w_1^B, w_k^C) \wedge e(w_2^B, w_k^C) \wedge \ldots \wedge e(w_n^B, w_k^C)$$

into

$$(\neg c(w_i^A, w_k^C) \vee e(w_i^A, w_1^B)) \wedge (\neg c(w_i^A, w_k^C) \vee e(w_i^A, w_2^B)) \wedge \ldots$$
$$\wedge(\neg c(w_i^A, w_k^C) \vee e(w_i^A, w_n^B)) \wedge (\neg c(w_i^A, w_k^C) \vee e(w_1^B, w_k^C))$$
$$\wedge(\neg c(w_i^A, w_k^C) \vee e(w_2^B, w_k^C)) \wedge \ldots \wedge (\neg c(w_i^A, w_k^C) \vee e(w_n^B, w_k^C)).$$

In the transgraph, the symmetry assumption is encoded as a hard constraint φ_3^∞. However, challenges arise with low-resource languages. Due to the small size of their dictionaries, these languages often lack senses, leading to many missing edges in the transgraph. To compensate for the missing edges, we introduce new edges, ensuring that cognate pairs share all senses. This is achieved by violating the soft constraint φ_2^+ for edge non-existence and incurring a cost based on user-selected heuristics, namely the cognate pair coexistence probability and cognate form similarity. Essentially, we operate under the assumption that these edges exist. A higher cognate pair coexistence probability and greater cognate form similarity increase the likelihood of a pair being cognate. Consequently, the cost of introducing a new edge to such a pair is lower. In the CNF formula, these new edges in the transgraph are encoded as FALSE, represented as $\neg e(w_i^A, w_j^B)$ or $\neg e(w_j^B, w_k^C)$, and visually depicted as dashed edges in the transgraph. The weights of the new edges, whether from a non-pivot word w_i^A to a pivot word w_j^B or vice versa, are defined as $\omega(w_i^A, w_j^B)$ and $\omega(w_j^B, w_k^C)$. Both of these weights are equivalent to $t(w_i^A, w_k^C).H_{coex} + t(w_i^A, w_k^C).H_{formSim}$.

Uniqueness. The uniqueness constraint ensures that only one-to-one cognates that share all of their pivot words are regarded as correct translation pairs. This limits a cognate of a word in language A to just one word in language C. This constraint is encoded as a hard constraint φ_4^∞.

Extracting at Least One Cognate. Due to the iterative interaction between the framework and the WPMaxSAT solver, the hard constraint φ_5^∞, which is a disjunction of all $c(w_i^A, w_k^C)$ variables, ensures that at least one of these variables is evaluated as TRUE. As a result, each iteration identifies the most possible cognate pair, storing it in both D_{Co} and D_R as a correct translation pair result.

Encoding Cognate. We filter out previously selected translation pairs in D_{Co} from the list of translation pair candidates. These pairs are encoded as TRUE, represented by $c(w_i^A, w_k^C)$, and are encoded as the hard constraint φ_6^∞. Additionally, they are excluded from "φ_5^∞".

5.2.6 Generalized Framework

We define two main CNF formulae: $CNF_{cognate}$ as shown in Eq. (12) and CNF_{M-M} as shown in Eq. (13) [21]. The former aims at identifying unique cognate pairs, and the latter at extracting many-to-many translation pairs by omitting uniqueness constraint φ_4^{∞}.

$$CNF_{cognate} = \varphi_1^{\infty} \wedge \varphi_2^{+} \wedge \varphi_3^{\infty} \wedge \varphi_4^{\infty} \wedge \varphi_5^{\infty} \wedge \varphi_6^{\infty} \tag{12}$$

$$CNF_{M-M} = \varphi_1^{\infty} \wedge \varphi_2^{+} \wedge \varphi_3^{\infty} \wedge \varphi_5^{\infty} \wedge \varphi_6^{\infty} \tag{13}$$

To construct various constraint-based bilingual dictionary induction methods suitable for available language resources and target languages, we generalize the constraint-based framework based on the above two CNF formulae. This allows users to choose the set of constraints such as $CNF_{cognate}$ and CNF_{M-M}, the number of iterations for the symmetry assumption, and individual or combined heuristics. The generalized framework is defined in Backus Normal Form as follows:
$\langle situatedMethod \rangle ::= \langle cycle \rangle$“:” $\langle method \rangle$“:” $\langle heuristic \rangle$
$\langle cycle \rangle ::=$ “1” | “2” | “3”| “4”| “5”| “6” | “7” | “8” | “9”
$\langle method \rangle ::=$ “C” | “M”
$\langle heuristic \rangle ::=$ “H1” | “H2” | “H12”

- *cycle*: the number of iteration for symmetry assumption (*cycle*$\geq$ 1).
- *method*: C indicating $CNF_{cognate}$ or M denoting CNF_{M-M}).
- *heuristic*: an individual or combined heuristics. The heuristics involves *H1* indicating cognate pair coexistence probability and *H2* denoting cognate form similarity.

Using this generalized framework, we can express the previous constraint-based methods. $CNF_{cognate}$ formula with 1-cycle symmetry assumption and heuristic 1 is represented by 1:C:$H1$, identical with one-to-one approach [44] and Ω_1 in our prior work [21]. CNF_{M-M} formula with 1-cycle symmetry assumption and heuristic 1 is represented by 1:M:$H1$, identical with Ω_2 in our prior work [21].

5.3 Experiment for Pivot-Based Approach

We conducted experiments using 6 methods derived from our generalized framework. Three of them extract unique cognate pairs (1–1) with the combined heuristics and 1-cycle symmetry assumption (1:C:$H12$), 2-cycle one (2:C:$H12$), and 3-cycle one (3:C:$H12$). The remaining three methods extract many-to-many translation pairs (M-M) with the combined heuristics and 1-cycle symmetry assumption (1:M:$H12$), 2-cycle one (2:M:$H12$), and 3-cycle one (3:M:$H12$). For comparison, we utilized two baseline methods employed in the previous constraint-based methods: 1:C:$H1$

and 1:M:$H$1. Furthermore, we also compared the 6 variations with the inverse consultation method (IC)[39] and translation pairs generated from the Cartesian product of each transgraph (CP).

5.3.1 Experimental Settings

We targeted three Indonesian ethnic languages for evaluating our methods: Minangkabau (min), Riau Mainland Malay (zlm), and Indonesian (ind) as the pivot language (min-ind-zlm). The language similarities between Minangkabau and Indonesian, Indonesian and Riau Mainland Malay, and Minangkabau and Riau Mainland Malay are 69.14%, 87.70%, and 61.66%, respectively, obtained from ASJP [10, 42]. This experiment aims to induce a Minangkabau-Malay bilingual dictionary from two bilingual dictionaries between Minangkabau and Indonesian and Malay and Indonesian. To create the gold standard for evaluating precision and recall, we generated all possible translation pairs using the Cartesian product (CP) of each transgraph, which were then verified by the Minangkabau-Malay bilingual crowd workers. Table 2 summarizes the details of the input dictionaries and the gold standard.

5.3.2 Experiment Result

In this experiment, all transgraphs achieve full symmetric connectivity by the third cycle, obtaining all possible translation pair candidates. To extract many-to-many translation pairs, the soft-constraint violation threshold is used to filter out all translation pairs whose costs surpass the threshold. Decreasing the threshold could yield high precision but low recall while increasing the threshold could yield high recall but low precision. To balance the precision and recall, we utilize the harmonic mean of precision and recall, F-measure. Table 3 presents the results targeting the optimal threshold for the highest F-score. For min-ind-zlm, our best M-M method (2:M:$H$12) achieves an F-score that is 3.4% higher than CP and 12.9 times higher than IC. Meanwhile, our best 1–1 method (3:C:$H$12) achieves precision that is 1.3% higher than our previous method (1:C:$H$1).

Table 2 Details of input dictionaries and gold standard

Language	min	ind	zlm
Headword	520	625	681
CP within transgraph	1,757		
CP across transgraph	354,120		
Gold standard	1,246		

Table 3 Comparison of thresholds producing the highest F-score

Method	Cognate threshold	Precision	Recall	F-score
3:M:H12 (M-M)	4.79	0.656	0.998	0.792
2:M:H12 (M-M)	4.79	0.735	0.923	0.818
1:M:H12 (M-M)	4.17	0.836	0.713	0.770
3:C:H12 (1-1)	4.79	0.884	0.331	0.481
2:C:H12 (1-1)	4.79	0.884	0.331	0.481
1:C:H12 (1-1)	4.17	0.878	0.328	0.478
Baseline: 1:M:H1 (M-M)		0.836	0.713	0.770
Baseline: 1:C:H1 (1-1)		0.873	0.327	0.475
Baseline: CP (M-M)		0.654	0.998	0.791
Baseline: IC (M-M)		0.950	0.031	0.059

5.4 Neural Network Approach

Given a set of translation pairs as a bilingual dictionary, we can utilize the translation pairs to train a model that transforms a source word into a target word, which augments the size of the dictionary. Therefore, we introduced a neural network approach to acquire the transformation rules or patterns between words in closely related languages. Seq2seq model consisting of an encoder and a decoder, one of the neural network approaches, is commonly used to learn a model to transform one language to another. We have employed it with Bi-LSTM as the encoder and LSTM as the decoder. The encoder receives a word in a hub language among closely related languages and produces a context vector, while the decoder takes the vector from the encoder and generates a word in another closely related language. The encoder for the hub language can be suitable for transfer learning applied to word translation tasks in other closely related languages because the hub language is most similar to the other closely related languages. In this research, we have validated two tokenization methods for applying the sequence-to-sequence (seq2seq) model to word translation tasks: character-based and subword-based tokenization.

5.4.1 Character-Based Sequence to Sequence

The first method employs character-based tokenization. Figure 14 shows the seq2seq model, where the encoder reads the input sequence character-by-character and the decoder also produces an output sequence character-by-character in which

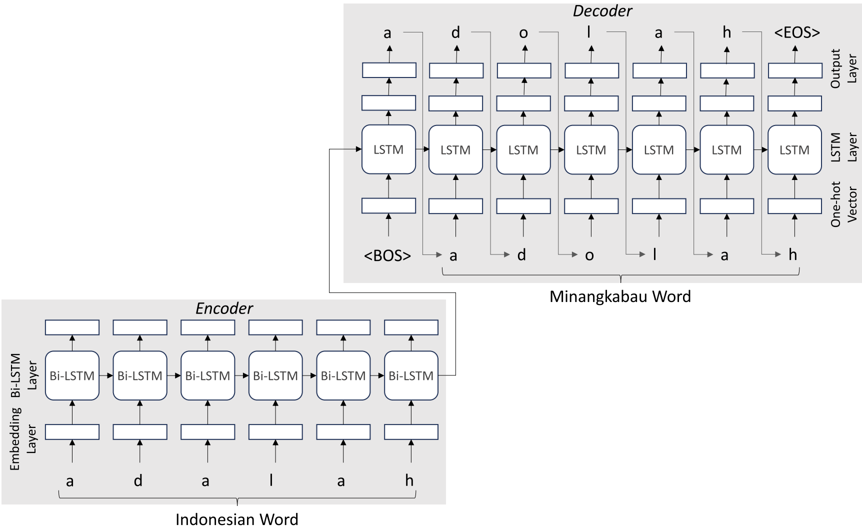


Fig. 14 Character-based sequence-to-sequence model

each character affects the subsequent character. For example, the encoder for Indonesian can accept 28 types of input tokens, and the decoder for Minangkabau generates 31 types of output tokens, including special tokens like ⟨*bos*⟩ and ⟨*eos*⟩. The token ⟨*bos*⟩ and the token ⟨*eos*⟩ denotes the beginning of a sentence triggering to produce a translated word and the end of a sentence determining when to stop predicting the subsequent character, respectively [38]. In Fig. 14, the encoder receives the word "*adalah* (is)" character-by-character. On the other hand, the decoder takes the token ⟨*bos*⟩ and the context vector from the encoder and outputs "*a*." Subsequently, this "*a*" is input into the decoder, which then outputs "*d*." This process continues until the token ⟨*eos*⟩ is outputted.

5.4.2 Byte-Pair Encoding-Based Sequence to Sequence

The second method employs SentencePiece as subword tokenization. SentencePiece builds subword vocabulary with the specified vocabulary size by using the byte-pair encoding (BPE) segmentation method, which divides words into chunks of characters [13]. The BPE starts with a vocabulary consisting of all symbols found in the set of words, then continues to combine two symbols most frequently co-occurring from the vocabulary to create a new symbol until the vocabulary size reaches the specified size [34]. Subword-based tokenization is expected to work because the phonemes of Indonesian ethnic languages are similar due to the closely related languages, and a similar chunk of the alphabet is assigned to them. To explore the appropriate vocabulary size, which means the number of the most frequent co-occurring characters,

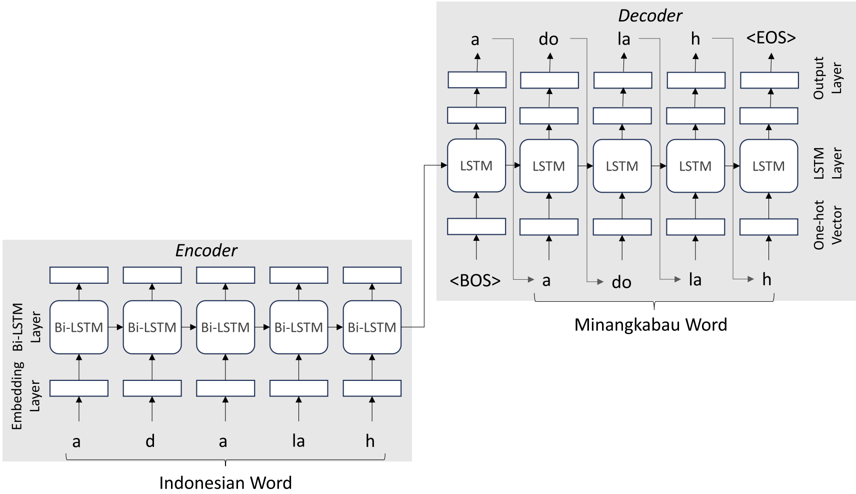


Fig. 15 Byte-pair encoding-based sequence-to-sequence model

we have applied the BPE-based seq2seq model with various vocabulary sizes. From this perspective, the character-based seq2seq model is regarded as a special case of the BPE-based seq2seq model with a vocabulary size of 28, the total number of alphabets. As shown in Fig. 15, an input word "*adalah* (is)" is tokenized by the BPE method in the preprocess and then each token, "*a*," "*d*," "*a*," "*la*," and "*h*" is input to the encoder. The decoder also chooses a token from the built vocabulary one by one.

The vocabularies, except for alphabets, obtained by BPE with the sizes of 40 and 100 are summarized in Table 4. Overall, the same number of vocabularies in Indonesia and Minangkabau (7 and 68, respectively) are acquired. The symbol "_" indicates the beginning of the word. For example, the difference between the "sa" and "_sa" in Minangkabau is that "sa" can occur in any place in a word. Table 5 shows the tokenization results of "*yang* and *nan* (which)," "*pada* and *pado* (on),""*adalah* and *adolah* (is),""*segera* and *sagiro* (quick)," and "*dasarnya* and *dasanyo* (basically)" with the learned vocabularies.

5.5 *Experiment for Neural-Based Approach*

5.5.1 Experimental Settings

We conducted an experiment to find the optimal tokenization method for applying the seq2seq model to a word translation task. The experiment targeted Indonesian as a source language and Minangkabau as a target language, the language similarity of which is 69.14% based on ASJP. The 10,278 translation pairs are split into 8,221 pairs for training data and 2,056 pairs for test data.

Table 4 Vocabularies obtained from BPE Indonesian-Minangkabau

Language	Vocab size = 40	Vocab size = 100
Indonesian	an, ng, nya, ta, kan, _di, _men,	an, ng, kan, ta, _di, la, nya, ra, da, si, _ke, _ber, ti, ba, li, ga, ri, ja, er, tu, bu, _se, at, in, _men, ma, sa, _per, ka, en, di, wa, ku, _meng, ya, na, _me, _pen, te, mp, ca, _p, _ter, ru, du, _mem, de, pa, or, un, ar, ju, is, _ka, bi, _ko, _ma, re, on, _ba, _pe, _pem, tan, pu, gu, al, ran, asi
Minangkabau	an, ang, _pa, _di, _ma, _ba, ng	an, ng, _di, _ba, ra, si, la,_pa, nyo, _ka, ta, da, ang, _ma, ik, kan, li, ri, ti, ak, tu, ka, **_sa**, _man, ja, ah, _ta, bu, ga, ek, in, ba, ku, **sa**, ma, su, di, ru, ya, _a, mp, _pan, to, wa, pa, ca, ran, du, ro, lu, tan, lo, mba, angan, ju, bi, pu, re, han, en, te, do, de, ko, gu, gi, _mam

Table 5 Example of tokenization BPE with different vocabulary size Indonesian-Minangkabau

Vocab size = 40		Vocab size = 100	
Indonesian	Minangkabau	Indonesian	Minangkabau
_,y,a,ng	_,n,an	_,ya,ng	_,n,an
_,p,a,d,a	_,pa,d,o	_pa,da	_,pa,do
_a,d,a,la,h	_a,d,o,l,a,h	_a,da,la,h	_a,do,la,h
_,s,e,g,e,ra	_,s,a,g,i,r,o	_,se,ge,ra	_,se,ge,ra
_,d,a,s,a,r,nya	_,d,a,s,a,n,y,o	_,da,sa,r,nya	_,da,sa,nyo

5.5.2 Experiment Result

As shown in Table 6, the results demonstrate that character-based tokenization outperforms BPE tokenization for a word translation task. The experiment was iterated seven times with different vocabulary sizes; the minimum and maximum sizes were 33 and 300, respectively. The smaller the vocabulary size of BPE is, the higher the performance is, and the performance with the minimal size of 33 is approximately the same as the character-based tokenization. This shows that a vector length for a token has an impact on the performance compared to the number of tokens, resulting in the fewer choices being more significant than fewer choice times. For example, in the case of "*adolah*," the vector length for a token and the number of tokens in character-based tokenization are 31 and 6, while the ones in BPE-based tokenization are 300 and 3.

Table 6 Comparison experiment results

Method	K-fold cross-validation Indonesian-Minangkabau					
	K = 1	K = 2	K = 3	K = 4	K = 5	Average precision
Character-based	**84.72**	83.70	83.31	83.60	84.30	**83.92**
SentecePiece(size = 33)	79.96	76.55	78.84	**81.71**	80.78	**79.56**
SentecePiece(size = 35)	76.11	76.89	79.42	74.31	**80.73**	77.49
SentecePiece(size = 40)	72.12	72.88	75.23	**75.99**	71.64	73.59
SentecePiece(size = 50)	67.12	62.15	66.97	**67.41**	64.29	65.58
SentecePiece(size = 80)	58.73	**59.32**	53.35	54.12	56.47	56.39
SentecePiece(size = 100)	49.36	48.24	49.46	**49.70**	48.78	49.10
SentecePiece(size = 300)	34.85	34.93	30.31	35.76	**36.19**	34.40

6 Markov-Based Composite Service for Human–Machine Collaboration

This chapter has proposed a crowdsourced method for language resource creation and machine induction methods for language resource augmentation. However, the accuracy of these methods heavily depends on the quality of the input data and the similarity between the target language pairs. When languages are closely related, securing reliable bilingual workers for crowdsourcing becomes more straightforward, thus reducing costs in bilingual dictionary creation. Conversely, inducing a bilingual dictionary from less similar languages can result in decreasing accuracy. This low accuracy can lead to mistranslations, leading to the need for corrections and increasing the overall costs. Therefore, strategic planning for service composition is necessary to determine the optimal combination of two interdependent services, crowdsourced human services and machine induction services, and to prioritize the language pairs to be targeted.

To this end, we have proposed a plan optimizer to produce a feasible optimal plan for creating multiple bilingual dictionaries. Considering uncertainties inherent in constraint-based induction and crowdsourced creation, this optimizer employs a Markov Decision Process (MDP) to decide the most cost-effective bilingual dictionary creation method for each state [25].

6.1 Formalizing Plan Optimization

A Markov Decision Process is commonly used in the services computing domain, especially for modeling workflow composition and optimization with uncertainty [6, 46]. To deal with the inherent uncertainty in the constraint-based bilingual dictionary induction, we model the plan optimization for creating bilingual dictionaries as a directed acyclic graph with the MDP. To apply the MDP to our plan optimization problem, we need to define a set of states ($s, s' \in S$), a set of actions ($a \in A$), a transition probability distribution $T(s, a, s')$ representing the likelihood that the process transitions from state s to state s' upon taking action a, and a cost function $C(s, a, s')$ that associates a cost with each state transition.

6.1.1 State

In the case of n target languages, the total number of all possible combinations of language pair is $h = \binom{n}{2}$. Each state contains h bilingual dictionaries, each of which between language x and y, denoted by $d_{(x,y)}$, can take four types of status:

- n: not existing
- eu: existing, but the dictionary size is below the user's requested minimum size
- $pu(z)$: induced by the pivot action with pivot language z, but the dictionary size is below the minimum size
- s: existing, and the dictionary size satisfies the minimum size.

A state is defined as a combination of the above statuses for each dictionary. In the initial state, all bilingual dictionaries must take either status n, eu, or s, while the final state consists of all the dictionaries whose statuses are s.

6.1.2 Action

We have two actions to create or augment a dictionary $d_{(x,y)}$: one is *pivot action* $a^p_{(x,z,y)}$ where z is the pivot language and the other is *crowdsourced creation* $a^i_{(x,y)}$. Both actions aim at changing the status of a bilingual dictionary from n, eu, or $pu(z)$ to s. The set of available actions for each state is determined by the following rules.

- If a dictionary in a state takes status n or eu, it can be augmented by both pivot action and crowdsourced action.
- If a dictionary in a state takes status $pu(z)$, it can be augmented by only crowdsourced action.
- If both input dictionaries to create a dictionary $d_{(x,y)}$ take status s, eu, and pu, pivot action $a^p_{(x,z,y)}$ is available.

6.1.3 State Transition Probability

An action to create or augment a dictionary transitions from one state to another by updating the status of the target dictionary. A crowdsourced action can deterministically decide the next state, as workers can be instructed to create translation pairs until the dictionary satisfies the user's specified minimum size. In contrast, a pivot action non-deterministically decides the next state because the size of the

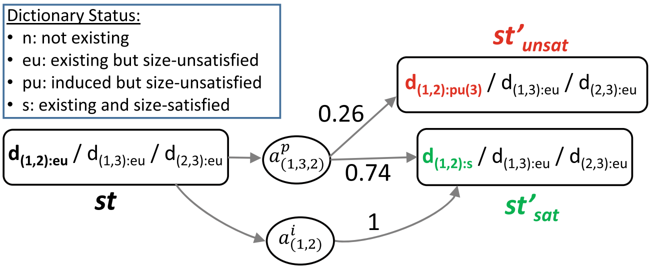


Fig. 16 Example of state transition

output dictionary depends on the input dictionaries, resulting in either status s or $pu(z)$. Figure 16 illustrates state transition triggered by both actions to augment a dictionary $d_{(1,2}$ between language 1 and 2. The crowdsourced action $a^i_{(1,2}$ ensures the subsequent state is st'_{sat}, where the status of $d_{(1,2)}$ is s, and the statuses of the other dictionaries remain unchanged from the previous state st. The pivot action $a^p_{(1,3,2)}$ whose pivot language is language 3, can lead to two potential subsequent states: st'_{sat} and st'_{unsat}. If the output dictionary size satisfies the minimum criteria, the next state becomes s'_{sat}. Otherwise, it transitions to s'_{unsat}, where the status of $d_{(1,2)}$ is updated to $pu(3)$, and the other dictionaries remain unchanged.

The state transition probability from one state to another after the pivot action is obtained by estimating the output dictionary size. This size is influenced by the size of the two input dictionaries used in the pivot action. In practice, we assume the number of translation pair candidates, $size(d^c_{(x,y)})$, to be double that of the smaller input dictionary, either of $size(d_{(x,z)})$ or $size(d_{(y,z)})$. By multiplying the number of translation pair candidates with the precision of the pivot action, we can calculate the number of induced translation pairs, $size(d^i_{(x,y)})$.

$$size(d^c_{(x,y)}) = 2 \times \min\left\{size(d_{(x,z)}), size(d_{(y,z)})\right\} \tag{14}$$

$$size(d^i_{(x,y)}) = precision(a^p_{(x,z,y)}) \times size(d^c_{(x,y)}) \tag{15}$$

To satisfy the minimum criteria, we can define the minimum precision k of the pivot action as the following expression.

$$k = \frac{minimumSize - size(d_{(x,y)})}{size(d^c_{(x,y)})} \tag{16}$$

We have introduced a beta distribution parameterized by language similarity as α and polysemy of topology as β to model the precision of the pivot action. Using this model, we can calculate the state transition probability that the pivot action changes from the current state s to the state s'_{unsat} where it fails to match the minimum size.

Given the cumulative distribution function $F(k; \alpha, \beta)$ for the beta distribution, the transition probability is defined as follows.

$$T(s, a^p_{(x,z,y)}, s'_{unsat}) = F(k; \alpha, \beta) = \int_0^k f(x; \alpha, \beta)dx \quad (17)$$

In contrast, the state transition probability from the current state s to the state s'_{sat} where the pivot action successfully satisfies the minimum size is defined as follows.

$$T(s, a^p_{(x,z,y)}, s'_{sat}) = 1 - F(k; \alpha, \beta) = 1 - \int_0^k f(x; \alpha, \beta)dx \quad (18)$$

6.1.4 Cost

In the MDP, a reward is received after transitioning from one state to another caused by an action. In the case of a bilingual dictionary creation, we need to pay some cost to manually create and evaluate translation pairs, resulting in that we alternatively regard the cost as a negative reward. The reward and cost are interchangeable in the previous MDP studies [41].

In the crowdsourced action, we instruct workers to manually create and evaluate translation pairs until they reach the minimum size. The cost of the crowdsourced action $a^i_{(x,y)}$ from state s to state s' is defined as the cost for one translation pair, the sum of $creationCost$ and $evaluationCost$, multiplied by the required number of translation pairs. Furthermore, by estimating the accuracy of the crowdsourced action at 0.8, the cost of the crowdsourced action is finally as follows.

$$C(s, a^i_{(x,y)}, s') = \frac{minimumSize - size(d_{(x,y)})}{0.8} \times (creationCost + evaluationCost) \quad (19)$$

On the other hand, when we already have the input dictionaries to induce a new dictionary with a pivot action, we can create translation pairs without cost, that is $creationCost = 0$, but still need to pay a cost for evaluating it. The cost of the pivot action $a^p_{(x,z,y)}$ from state s to state s' is defined as the evaluation cost for one translation pair, $evaluationCost$, multiplied by the number of translation pair candidates.

$$C(s, a^p_{(x,z,y)}, s') = size(d^c_{(x,y)}) \times evaluationCost \quad (20)$$

6.2 Experiment

To evaluate the MDP-based plan optimizer for bilingual dictionary creation, we conducted an experiment under Indonesia language sphere project [19]. Since our

Table 7 Similarity matrix of the target languages

Language	Indonesian	Malay	Minangkabau	Javanese	Sundanese
Indonesian	–				
Malay	87.70%	–			
Minangkabau	69.14%	61.66%	–		
Javanese	24.09%	21.36%	25.01%	–	
Sundanese	39.43%	41.12%	30.81%	21.82%	–

pivot-based bilingual dictionary induction method works better on closely related languages, we targeted Indonesian, Malay, and Minangkabau, whose language similarities are high, as shown in Table 7. Additionally, we also selected Javanese and Sundanese, considering the population of their speakers. Thus, we targeted 5 languages, Indonesian (ind), Malay (zlm), Minangkabau (min), Javanese (jav), and Sundanese (sun), and created or augmented 10 dictionaries for every combination of the target languages. The users' specified minimum size is 2,000 translation pairs, that is $minimumSize = 2,000$. We also decided on the cost of creating and evaluating translation pairs based on the availability of the native speakers.

6.2.1 Modeling Task for Native Speaker

We have two types of tasks by native speakers: a creation task and an evaluation task. Even though Indonesia is a multiethnic country where various ethnic people coexist, it is difficult to recruit a bilingual native speaker between the two ethnic languages because ethnic languages are not taught in school, and only Indonesian, the national language of Indonesia, is commonly used in education. To overcome this limitation, $s_{(ind,x)}$, a native bilingual speaker of Indonesian language and ethnic language x, and $s_{(ind,y)}$, a native bilingual speaker of Indonesian language and ethnic language y, collaboratively create and evaluate translation pairs by communicating the senses in Indonesian. By considering this collaboration, we classify the native speakers' tasks into four: an individual creation task $T1(ind, x)$ and an individual evaluation task $T2(ind, x)$ of a bilingual dictionary $d_{(ind,x)}$, and a collaborative creation task $T3(x, y)$ and a collaborative evaluation task $T4(x, y)$ of a bilingual dictionary $d_{(x,y)}$ between ethic language x and y.

Based on the preliminary experiments, we estimated the creation and evaluation cost for each translation pair with a unit time taken for doing the evaluation task $T2(ind, x)$, which is the simplest task. The cost of the creation task $T1(ind, x)$ is calculated as three times the cost of its evaluation task $T2(ind, x)$. On the other hand, the costs of the creation task $T3(x, y)$ and the evaluation task $T4(x, y)$ are calculated as eight times and four times the cost of $T2(ind, x)$ if they need the collaboration of two native speakers, respectively. Otherwise, they are six times and two times the cost of $T2(ind, x)$.

To ensure the quality of the manually created bilingual dictionary $d_{(ind,x)}$, the created translation pairs should be evaluated by the different native bilingual speaker $s_{(ind,x)}$. We only pay for correct translation pairs to motivate them to create translation pairs carefully. In this way, we couple a creation task and an evaluation task to make two composite tasks: $CT1(ind, x)$ consists of $T1(ind, x)$ and $T2(ind, x)$ between Indonesian and an ethnic language, and $CT2(x, ind, y)$ consists of $T3(x, ind, y)$ and $T4(x, ind, y)$ between ethnic languages via Indonesian as a pivot language.

6.2.2 Estimated Plans

To show the effectiveness of our method, we compare them with an all-crowdsourced action plan as a baseline. The estimated cost of the baseline is summarized in Table 8. This cost is estimated by the total number of translation pairs manually created and evaluated by workers. By considering the accuracy of the crowdsourced action as 0.8 and no payment for creating wrong translation pairs, the total cost is the cost of creating the required number of correct translation pairs and evaluating all created translation pairs, including wrong translation pairs. The number of all the created translation pairs is the required number divided by the accuracy, 0.8.

Table 8 Estimated cost of all-crowdsourced action plan

Plan	#Creation tasks[1]	#Evaluation tasks	#Paid tasks[2]	{Total cost[3]} (unit time)
CT1(ind, zlm)-711 exist	1,611	1,611	2,900	5,478
CT1(ind, jav)	2,500	2,500	4,500	8,500
CT1(ind, sun)	2,500	2,500	4,500	8,500
CT2(zlm, min)-1,246 exist	943	943	1,697	9,802
CT2(jav, sun)	2,500	2,500	4,500	26,000
CT2(zlm, jav)	2,500	2,500	4,500	26,000
CT2(min, sun)	2,500	2,500	4,500	26,000
CT2(zlm, sun)	2,500	2,500	4,500	26,000
CT2(min, jav)	2,500	2,500	4,500	26,000
Total				**162,280**

[1] Estimating 0.8 human accuracy
[2] #Paid Tasks = #Creation Tasks*0.8 + #Evaluation Tasks
[3] Total Cost for CT1= #Creation Tasks*0.8*3 + #Evaluation Tasks*1
Total Cost for CT2= #Creation Tasks*0.8*8 + #Evaluation Tasks*4

Table 9 Estimated cost of the MDP optimal plan

Plan	#Induced translation	Induction precision[2]	#Paid tasks[2]	Total cost[3] (unit time)
CT1(ind, zlm)-711 exist	–	–	2,900	5,478
CT1(ind, jav)	–	–	4,500	8,500
CT1(ind, sun)	–	–	4,500	8,500
P(zlm, ind, min)-1,246 exist	2,792	0.6981	0	0
T4(zlm, ind, min)	–	–	2,792	11,170
P(jav, ind, sun)	3,285	0.6108	0	0
T4(jav, ind, sun)	–	–	3,285	13,139
P(zlm, ind, jav)	3,283	0.6094	0	0
T4(zlm, ind, jav)	–	–	3,283	13,134
P(min, ind, sun)	2,727	0.6817	0	0
T4(min, ind, sun)	–	–	2,727	10,907
P(zlm, ind, sun)	3,644	0.6563	0	0
T4(zlm, ind, sun)	–	–	3,644	14,578
P(min, zlm, jav)	2,694	0.6735	0	0
T4(min, zlm, jav)	–	–	2,694	10,776
Total				**96,182**

[1] Estimated from beta distribution: α as language similarity and β as topology polysemy = 3
[2] #Paid Tasks for CT1 = #Creation Tasks*0.8 + #Evaluation Tasks
#Paid Tasks for T4 = #Evaluation Tasks for Induced Translation Pairs
[3] Total Cost for CT1 = #Creation Tasks*0.8*3 + #Evaluation Tasks*1
Total Cost for T4 = #Evaluation Tasks for Induced Translation Pairs*4

On the other hand, we generated the MDP optimal plan by modeling the pivot action precision with prior beta distributions. We employed the language similarities as the α parameter shown in Table 7 and a topology polysemy, 3 in practice, as the β parameter. The generated optimal plan and its estimated cost are summarized in Table 9. The plan column indicates the task order in the plan. The cost calculation for the crowdsourced actions in this plan, CT1, is the same as the all-crowdsourced plan. Meanwhile, the cost of the pivot actions is estimated by the only number of evaluating translation pairs induced by the pivot actions.

6.2.3 Experiment Result

To validate the MDP optimal plan in Table 9, we conducted a real experiment in Indonesia collaboratively with the Islamic University of Riau and Telkom University. In this experiment, 34 native speakers, consisting of 5 Minangkabau speakers, 8 Malay speakers, 9 Javanese speakers, and 12 Sundanese speakers, joined as crowd workers. The real costs are summarized in Table 10.

This result shows that the MDP optimal plan outperformed the all-crowdsourced plan with 42% cost reduction, and the real cost of the optimal plan is close to the

Table 10 Real cost of the MDP optimal plan

Plan	Topology polysemy[1]	#Induced translation	Induction precision[2]	Human accuracy[1]	#Paid tasks[3]	Total cost (unit time)
CT1(ind, zlm)-711 exist	–	–	–	0.868	3,338	6,440
CT1(ind, jav)	–	–	–	0.790	4,573	8,610
CT1(ind, sun)	–	–	–	0.830	4,517	8,615
P(zlm, ind, min)-1,246 exist	3.355	1,940	0.885	–	0	0
T4(zlm, ind, min)	–	–	–	1	1,940	7,760
P(jav, ind, sun)	2.498	2,071	0.824	–	0	0
T4(jav, ind, sun)	–	–	–	1	2,071	8,284
CT2(jav, sun)	–	–	–	0.838	715	4,164
P(zlm, ind, jav)	2.583	2,018	0.801	–	0	0
T4(zlm, ind, jav)	–	–	–	1	2,018	8,072
CT2(zlm, jav)	–	–	–	0.843	892	5,200
P(min, ind, sun)	3.300	2,239	0.802	–	0	0
T4(min, ind, sun)	–	–	–	1	2,239	8,956
CT2(min, sun)	–	–	–	0.732	435	2,557
P(zlm, ind, sun)	2.824	2,029	0.833	–	0	0
T4(zlm, ind, sun)	–	–	–	1	2,029	8,116
CT2(zlm, sun)	–	–	–	0.840	665	3,896
P(min, zlm, jav)	3.192	2,069	0.739	–	0	0
T4(min, zlm, jav)	–	–	–	1	2,069	8,276
CT2(min, jav)	–	–	–	0.957	678	4,760
Total						**93,707**[4]

[1] The average topology polysemy and human accuracy are close to our estimation in Table 9
[2] All pivot-based bilingual dictionary induction precisions are higher than our estimation in Table 9
[3] #Paid Taks = #Creation Tasks*Human Accuracy + #Evaluation Tasks
[4] There are 42% of cost reduction compared to the estimated all-crowdsourced plan in Table 8 *and 3% of cost reduction compared to the estimated MDP optimal plan in Table* 9

estimated cost in Table 9 within a 3% margin of error. Furthermore, our estimated human accuracy of 0.8 and the topology polysemy were also validated because the average human accuracy is 0.837 and the average topology polysemy is 2.958 in the real experiment.

6.3 Discussion

The current plan optimization algorithm is offline and generates the optimal policy based on approximate models beforehand. This results in the optimal plan can be sub-optimal after taking a few actions in the plan. For instance, although all the pivot actions in the MDP optimal plan shown in Table 9 successfully satisfy the required number of translation pairs, five out of six pivot actions in the real experiment failed to satisfy the required number, in spite of the higher pivot action precision compared to the beta distribution-based estimation. This is caused by the low accuracy of estimating the number of translation pair candidates. This could cause a difference between the estimated and real costs.

One possible way to solve this problem is to change the offline algorithm to an online one by recursively reformalizing the planning problem with newly acquired information on the environment, such as the number of translation pair candidates and created correct translation pairs, every time after executing an action. This allows the plan optimizer to adapt to the dynamic and uncertain environment.

7 Conclusion

To create a multilingual service platform for smart cities, it is necessary to collect language resources in low-resource languages as well as high-resource languages for language equality. However, existing multilingual service platforms are targeted mainly at official languages but not ethnic languages, spoken more in Asia than in Europe, because there exist fewer resources in ethnic languages. Multiethnic countries such as Indonesia require a multilingual service platform to support their ethnic languages. This chapter focused on creating language resources in ethnic languages by combining crowdsourced human services and automatic machine services.

Crowdsourcing is widely adopted to create language resources when there is less data on the Web. By introducing hyper-questions to aggregate answers from crowd workers into the crowdsourced workflow, this chapter aimed at improving the evaluation accuracy under a majority of less reliable workers and assigning creation tasks to highly reliable workers preferentially. The proposed workflow has been demonstrated to achieve higher accuracy than the existing methods regardless of the ratios of less reliable workers.

Additionally, induction methods are employed to acquire language resources from a large amount of data. To complement the lack of data for ethnic languages, this chapter utilized similarities between ethnic languages, such as cognates. Assuming that cognates maintain several common senses and have similar spelling, we filtered mistranslation pairs from candidates with constraints optimization techniques and obtained spelling transformation rules between cognates with neural networks. The proposed methods have been empirically shown to achieve higher recall than the existing methods.

Moreover, to optimally combine crowdsourced creation and machine induction of language resources, this chapter modeled resource creation planning as a Markov Decision Process (MDP). The MDP calculated the optimal policy that decided which actions, such as manual creation and machine induction, are best to minimize the total cost. This chapter proved that the proposed planning method significantly reduced the total cost with a close real cost estimate compared to entirely manual creations.

Acknowledgements This research was partially supported by a Grant-in-Aid for Scientific Research (B) (21H03561,2021–2024, 21H03556, 2021–2023) and a Grant-in-Aid Young Scientists (A) (17H04706, 2017–2020) from the Japan Society for the Promotion of Science (JSPS).
I would like to express my sincere gratitude to Prof. Toru Ishida at Hong Kong Baptist University for his valuable advice and helpful discussions. I am also particularly grateful to Associate Prof. Arbi Haza Nasution at the Islamic University of Riau for his collaboration on my research. I wish to thank Prof. Mirna Adriani, Prof. Totok Suhardijanto, and Dr. Kemas Muslim Lhaksmana for organizing the field experiments. Finally, I appreciate my students in the Social Intelligence Laboratory at Ritsumeikan University, especially Hiroki Chida and Kartika Findra Resiandi, for contributing to the Indonesia Language Sphere project.

References

1. Biere A, Heule M, van Maaren H (2009) Handbook of satisfiability, vol 185. IOS Press
2. Calzolari N, Gratta R, Francopoulo G, Mariani J, Rubino F, Russo I, Soria C (2012) The lre map harmonising community descriptions of resources. In: International conference on language resources and evaluation (LREC 2012). ELRA, pp 1084–1089
3. Campbell L (2013) Historical linguistics. Edinburgh University Press
4. Chida H, Murakami Y, Pituxcoosuvarn M (2022) Quality control for crowdsourced bilingual dictionary in low-resource languages. In: Proceedings of the thirteenth language resources and evaluation conference. European Language Resources Association, Marseille, France, pp 6590–6596. URL https://aclanthology.org/2022.lrec-1.709
5. Déjean H, Gaussier E, Sadat F (2002) An approach based on multilingual thesauri and model combination for bilingual lexicon extraction. In: Proceedings of the 19th international conference on computational linguistics—vol. 1, COLING ’02. Association for Computational Linguistics, Stroudsburg, PA, USA, pp 1–7. https://doi.org/10.3115/1072228.1072394
6. Doshi P, Goodwin R, Akkiraju R, Verma K (2004) Dynamic workflow composition using Markov decision processes. In: Proceedings of the IEEE international conference on web services, pp 576–582. https://doi.org/10.1109/ICWS.2004.1314784
7. Goto S, Ishida T, Lin D (2016) Understanding crowdsourcing workflow: modeling and optimizing iterative and parallel processes. In: Proceedings of the AAAI conference on human computation and crowdsourcing
8. Gratta R, Frontini F, Khan A, Mariani J, Soria C (2014) The lremap for under-resourced languages. In: International workshop on collaboration and computing for under-resourced languages in the linked open data era (CCURL 2014), pp 78–83
9. Hassine AB, Matsubara S, Ishida T (2006) A constraint-based approach to horizontal web service composition. In: International semantic web conference. Springer, pp 130–143
10. Holman EW, Brown CH, Wichmann S, Müller A, Velupillai V, Hammarström H, Sauppe S, Jung H, Bakker D, Brown P et al (2011) Automated dating of the world’s language families based on lexical similarity. Curr Anthropol 52(6):841–875
11. Ishida T, Murakami Y, Lin D, Nakaguchi T, Otani M (2018) Language service infrastructure on the web: The language grid. Computer 51(6):72–81. https://doi.org/10.1109/MC.2018.2701643
12. Kazai G, Kamps J, Koolen M, Milic-Frayling N (2011) Crowdsourcing for book search evaluation: impact of hit design on comparative system ranking. In: Proceedings of the 34th international ACM SIGIR conference on research and development in information retrieval, pp 205–214
13. Kudo T (2018) Subword regularization: improving neural network translation models with multiple subword candidates. In: Proceedings of the 56th annual meeting of the association for computational linguistics (vol. 1: Long Papers). Association for Computational Linguistics, Melbourne, Australia, pp 66–75. https://doi.org/10.18653/v1/P18-1007

14. Li J, Baba Y, Kashima H (2017) Hyper questions: unsupervised targeting of a few experts in crowdsourcing. In: Proceedings of the 2017 ACM on conference on information and knowledge management, pp 1069–1078
15. Lösch A, Mapelli V, Piperidis S, Vasiļjevs A, Smal L, Declerck T, Schnur E, Choukri K, van Genabith J (2018) European language resource coordination: collecting language resources for public sector multilingual information management. In: Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018). European Language Resources Association (ELRA), Miyazaki, Japan. https://aclanthology.org/L18-1213
16. Matsuno J, Ishida T (2011) Constraint optimization approach to context based word selection. In: IJCAI Proceedings-international joint conference on artificial intelligence, vol 22
17. Melamed ID (1995) Automatic evaluation and uniform filter cascades for inducing n-best translation lexicons. CoRR cmp-lg/9505044. http://arxiv.org/abs/cmp-lg/9505044
18. Miyabe M, Yoshino T, Shigeno A (2011) Sharing multilingual resources to support hospital receptions. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 119–132. https://doi.org/10.1007/978-3-642-21178-2_8
19. Murakami Y (2019) Indonesia language sphere: an ecosystem for dictionary development for low-resource languages. J Phys Conf Ser 1192:012001. IOP Publishing
20. Nakov P, Ng HT (2012) Improving statistical machine translation for a resource-poor language using related resourcerich languages. J Artif Intell Res:179–222
21. Nasution AH, Murakami Y, Ishida T (2016) Constraint-based bilingual lexicon induction for closely related languages. In: Proceedings of the tenth international conference on language resources and evaluation (LREC 2016). Paris, France, pp 3291–3298
22. Nasution AH, Murakami Y, Ishida T (2017a) A generalized constraint approach to bilingual dictionary induction for low-resource language families. ACM Trans Asian Low Resour Lang Inf Process 17(2):9:1–9:29. https://doi.org/10.1145/3138815
23. Nasution AH, Murakami Y, Ishida T (2017b) Plan optimization for creating bilingual dictionaries of low-resource languages. In: 2017 International conference on culture and computing, pp 35–41. https://doi.org/10.1109/Culture.and.Computing.2017.21
24. Nasution AH, Murakami Y, Ishida T (2018) Designing a collaborative process to create bilingual dictionaries of indonesian ethnic languages. In: Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018). European Language Resources Association (ELRA), Paris, France, pp 3397–3404
25. Nasution AH, Murakami Y, Ishida T (2021) Plan optimization to bilingual dictionary induction for low-resource language families. ACM Trans Asian Low Resour Lang Inf Process 20(2). https://doi.org/10.1145/3448215
26. Oleson D, Sorokin A, Laughlin G, Hester V, Le J, Biewald L (2011) Programmatic gold: targeted and scalable quality assurance in crowdsourcing. In: Workshops at the twenty-fifth AAAI conference on artificial intelligence, pp 43–48
27. Piperidis S (2012) The META-SHARE language resources sharing infrastructure: principles, challenges, solutions. In: Proceedings of the eighth international conference on language resources and evaluation (LREC'12). European Language Resources Association (ELRA), Istanbul, Turkey, pp 36–42. http://www.lrec-conf.org/proceedings/lrec2012/pdf/1086_Paper.pdf
28. Piperidis S, Labropoulou P, Deligiannis M, Giagkou M (2018) Managing public sector data for multilingual applications development. In: Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018). European Language Resources Association (ELRA), Miyazaki, Japan. https://aclanthology.org/L18-1205
29. Rehm G, Berger M, Elsholz E, Hegele S, Kintzel F, Marheinecke K, Piperidis S, Deligiannis M, Galanis D, Gkirtzou K, Labropoulou P, Bontcheva K, Jones D, Roberts I, Hajič J, Hamrlová J, Kačena L, Choukri K, Arranz V, Vasiļjevs A, Anvari O, Lagzdiņš A, Meļņika J, Backfried G, Dikici E, Janosik M, Prinz K, Prinz C, Stampler S, Thomas-Aniola D, Gómez-Pérez JM, Garcia Silva A, Berrío C, Germann U, Renals S, Klejch O (2020) European language grid: an overview. In: Proceedings of the twelfth language resources and evaluation conference. European Language Resources Association, Marseille, France, pp 3366–3380. https://aclanthology.org/2020.lrec-1.413

30. Rehm G, Piperidis S, Bontcheva K, Hajic J, Arranz V, Vasiļjevs A, Backfried G, Gomez-Perez JM, Germann U, Calizzano R, Feldhus N, Hegele S, Kintzel F, Marheinecke K, Moreno-Schneider J, Galanis D, Labropoulou P, Deligiannis M, Gkirtzou K, Kolovou A, Gkoumas D, Voukoutis L, Roberts I, Hamrlova J, Varis D, Kacena L, Choukri K, Mapelli V, Rigault M, Melnika J, Janosik M, Prinz K, Garcia-Silva A, Berrio C, Klejch O, Renals S (2021) European language grid: a joint platform for the European language technology community. In: Proceedings of the 16th conference of the European chapter of the association for computational linguistics: system demonstrations. Association for Computational Linguistics, pp 221–230. https://doi.org/10.18653/v1/2021.eacl-demos.26
31. Resiandi K, Murakami Y, Nasution AH (2022) A neural network approach to create Minangkabau-Indonesia bilingual dictionary. In: Proceedings of the 1st annual meeting of the ELRA/ISCA special interest group on under-resourced languages. European Language Resources Association, Marseille, France, pp 122–128. https://aclanthology.org/2022.sigul-1.16
32. Resiandi K, Murakami Y, Nasution AH (2023) Neural network-based bilingual lexicon induction for Indonesian ethnic languages. Appl Sci 13(15). https://doi.org/10.3390/app13158666
33. Richardson J, Nakazawa T, Kurohashi S (2015) Pivot-based topic models for low-resource lexicon extraction. In: PACLIC
34. Sennrich R, Haddow B, Birch A (2016) Neural machine translation of rare words with subword units. In: Proceedings of the 54th annual meeting of the association for computational linguistics (vol 1: Long Papers). Association for Computational Linguistics, Berlin, Germany, pp 1715–1725. https://doi.org/10.18653/v1/P16-1162
35. Sheng V, Provost F, Ipeirotis P (2008) Get another label? improving data quality and data mining using multiple, noisy labelers. In: Proceedings of the ACM SIGKDD international conference on knowledge discovery and data mining, pp 614–622. https://doi.org/10.1145/1401890.1401965
36. Simons GF, Fennig CD (eds) (2017) Ethnologue: languages of the world, 20th edn. http://www.ethnologue.com
37. Suo Y, Miyata N, Morikawa H, Ishida T, Shi Y (2009) Open smart classroom: extensible and scalable learning system in smart space using web service technology. IEEE Trans Knowl Data Eng 21(6):814–828. https://doi.org/10.1109/TKDE.2008.117
38. Sutskever I, Vinyals O, Le QV (2014) Sequence to sequence learning with neural networks. In: Proceedings of the 27th international conference on neural information processing systems—vol 2, NIPS'14. MIT Press, Cambridge, MA, USA, pp 3104–3112
39. Tanaka K, Umemura K (1994) Construction of a bilingual dictionary intermediated by a third language. In: Proceedings of the 15th conference on Computational linguistics-vol 1. Association for Computational Linguistics, pp 297–303
40. Weber F, Jarisch G (2023) Virtual personal assistant prototype YouTwinDi. Springer International Publishing, Cham, pp 355–360. https://doi.org/10.1007/978-3-031-17258-8_29
41. White DJ (1993) A survey of applications of Markov decision processes. J Oper Res Soc 44(11):1073–1096
42. Wichmann S, Holman EW, Brown CH (eds) (2022) The ASJP database (version 20)
43. Wu H, Wang H (2007) Pivot language approach for phrase-based statistical machine translation. Mach Transl 21(3):165–181. https://doi.org/10.1007/s10590-008-9041-6
44. Wushouer M, Lin D, Ishida T, Hirayama K (2014) Pivot-based bilingual dictionary extraction from multiple dictionary resources. Springer International Publishing, Cham, pp 221–234. https://doi.org/10.1007/978-3-319-13560-1_18
45. Wushouer M, Lin D, Ishida T, Hirayama K (2015) A constraint approach to pivot-based bilingual dictionary induction. ACM Trans Asian Low Resour Lang Inf Process 15(1):4:1–4:26. https://doi.org/10.1145/2723144
46. Yu J, Buyya R, Tham CK (2005) Cost-based scheduling of scientific workflow applications on utility grids. In: First international conference on e-science and grid computing (e-Science'05), pp 8–147. https://doi.org/10.1109/E-SCIENCE.2005.26

Data Analysis for Smart Cities

Automatic Big Data Analysis Using AI-Based Service Composition for Smart City

Incheon Paik

Abstract The new wave of the next industrial revolution is beginning. Many metaverse platforms have been launched successfully. One of these digital twins will use automatic big data analysis technologies to affect our real-world efficiently. Following the analysis of various large amounts of data on digital twins in the metaverse, smart cities will be constructed more efficiently. These analyses of big data from virtual worlds should be customizable for various goal tasks; therefore, the analysis workflow requires higher intelligence, but it is very difficult to overcome this high barrier. A possible solution is to use an automatic service composition technique. In this chapter, automatic service composition architecture, in addition to discovery and composition methods using a heuristic deep learning approach, will be introduced. In addition, an example framework using service composition to analyze big data will be explained. Finally, this chapter will show how automatic big data analysis is processed in a service composition sequence that is supported by AI.

1 Concept of Automatic Service Composition

1.1 Introduction

According to Singh and Huhns [1], service-oriented computing (SOC) can enhance the productivity of programming and administering applications in open distributed systems, and provide new flexible and scalable business applications. Web services, which offer useful APIs for open systems on the Internet, are evolving into an automatic development environment for agent-based applications thanks to the semantic web. The automatic service composition (ASC) is aimed at creating more capable and novel value-added services for users by composing existing services [2].

ASC typically involves four stages, as proposed by Agarwal et al. [3]: planning a workflow of individual service types, locating services from a service registry,

---

I. Paik (✉)
School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu, Japan
e-mail: paikic@u-aizu.ac.jp

Y. Murakami and K. Kimura (eds.), *Human-Centered Services Computing for Smart Cities*, https://doi.org/10.1007/978-981-97-0779-9_4

selecting the best candidate services based on nonfunctional properties (NFPs), and executing the selected services. In case of exceptions during execution, services may need to be retried or the planning and selection stages may have to be redone. However, there is no fully integrated framework for ASC, and previous studies have mainly focused on individual stages or integrating them for more realistic composition. There is also confusion about the roles of composition stages across different studies.

To overcome these issues, Paik et al. [4] present a more comprehensive framework for ASC, which includes making functional goals scalable and the composition seamless. This framework addresses the problems of automated composition and provides a starting point for future research.

1. **Scalable functional goals: Nested workflow management**. Service compositions in existing literature commonly assume that the composition can be completed at one time. However, this is not always feasible due to the distributed and dynamic nature of compositions, which can occur across enterprise boundaries. To address this, ASC can include nested dynamic compositions at sublevels to achieve comprehensive functional goals. Additionally, ASC must consider dynamically changing workflows to fulfill new goals introduced at higher levels of abstraction. In a nested architecture, the workflow manager can control replanning and reselection in case of exceptions, as well as orchestrate nested composition flows.
2. **Seamless composition: Identification of Composition Properties (NFPs)**. Services possess both functional and nonfunctional properties. Functional properties (FPs) generally refer to requirements within the domain of a service request, while nonfunctional properties (NFPs) encompass requirements on the services themselves. NFPs encompass quality-of-service (QoS) parameters by including preferences and similar "soft" constraints. FPs must be met, while NFPs do not necessarily need to be met. Although there is no clear distinction between FPs and NFPs, it is helpful to separate them when describing goals and services. A user or developer specifies a composition goal consisting of FPs or constraints that include their attributes (e.g., "Arrange a trip from Aizu to Los Angeles" and "Total Cost is less than 100,000") and NFPs (e.g., "Cost for the composed services to arrange the trip is less than 5,000" and "Time to execute the composed services must be less than 30 s"). These are specified at the business (abstract) level to a service composer. Typically, during the discovery stage, service instances that match the functional operation signature are located, but composition properties, such as NFPs and some of the FP's attributes in the user's request, which are abstract as in the examples above, are not available as explicit operations with detailed parameters that can be considered during the selection process. However, identifying and considering the composition properties are necessary for seamless ASC.
3. **Framework for ASC: Modified four-stage process**. Previous studies on service composition have primarily focused on one or two of the four stages, often in abstract or comparative terms. However, to gain a complete understanding of ASC

and its interactions, it is necessary to consider all functional blocks and the entire composition structure. This can be accomplished through hierarchical structural and behavioral object analysis, with top-level analysis focusing on more abstract concepts and lower level analysis delving into algorithm and method choices. By integrating the modified four-stage architecture with the two additional issues previously identified, our UML-based framework can shed light on the organic structure and behavior of ASC and provide direction for future development.

1.2 Preliminaries for Service Composition

1.2.1 The Four Stages of Composition

The four stages for automatic service composition are depicted in Fig. 1, which has been augmented with more information from that presented by Agarwal et al. [5]. Formally, we denote the following:

- R: Set of user's requests at the service level
- $W = \{t_1, t_2, t_3, \ldots t_l\}$: Set of l abstract tasks in an abstract workflow W
- Planning Π: $R \rightarrow W$
- $I_i = \{i_{i1}, i_{i2}, i_{i3}, \ldots, i_{im}\}$: Set of m service instances advertised in a service registry for an abstract task t_i. I is the set of I_i, for $1 \le i \le l$. If each task in a workflow has m instances, then the total number of service instances available for the workflow is $l \times m$
- Discovery Δ: $W \rightarrow I$

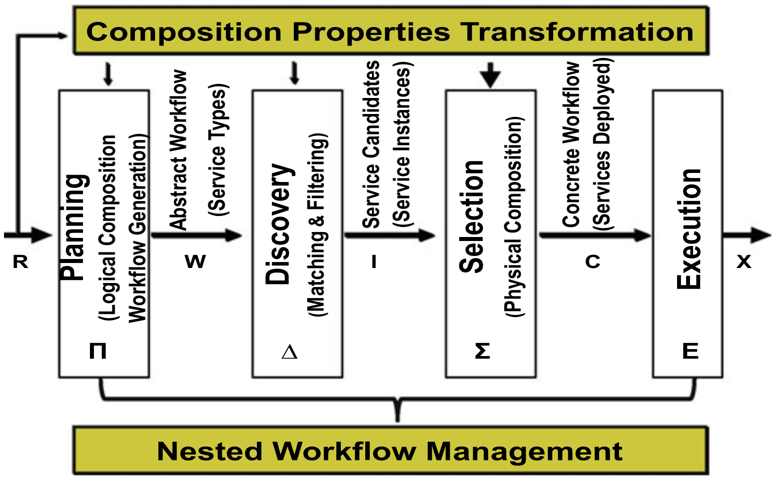


Fig. 1 Four stages of service composition

- $C_j = \{c_{j1}, c_{j2}, c_{j3}, \ldots, c_{jp}\}$: Set of p selected service instances to be executed from the service instance set I_j. C is the set of C_j where $1 \leq j \leq l$
- Selection Σ: I $\rightarrow$ C
- $X = \{x_1, x_2, x_3, \ldots, x_q\}$: Set of q executed service traces
- Execution E: C $\rightarrow$ X

There are differing viewpoints regarding the service composition process, but generally, it is divided into four stages: logical composition in the planning stage, and physical composition in the selection stage [5]. We chose this four-stage process as the basis of our approach because:

1. It is widely accepted and many existing approaches can be easily mapped or related to it.
2. Our approach builds upon it to enhance service composition's flexibility and improve its usability.
3. Our approach incorporates exception handling and backtracking to handle failures, avoiding the need to return to the planning stage when a failure occurs.

1.2.2 Motivating Scenario and Our Extended Framework for Seamless ASC

During the planning stage of automated service composition, a plan is generated to fulfill a functional goal. However, this plan may be a subprocedure of a higher level goal. For instance, consider a tour group traveling from Aizu to Los Angeles. To create the tour group package, a composition of three subprocesses is required: (1) scheduling the trip, (2) making group reservations, and (3) repeating the trip scheduling and reservations for each participant. In the first composite service for finding a trip schedule, the ASC tries to find the best workflow for the trip and candidate services for the workflow dynamically. After deciding on the trip schedule, the service must book all travel resources according to the schedule from the previous step. If the booking fails, the composition manager must return to the planning or selecting stages. The four-stage process characterizes composition from an abstract workflow to a concrete one but does not address seamless ASC. To address this, we have extended the process with the Nested Workflow Management and Composition Properties Transformation components, as shown in Fig. 1. The nested workflow management block orchestrates nested workflows for each stage of ASC to handle composite goals.

1.3 *Workflow Orchestration in a Nested Composition for Scalability*

1.3.1 Orchestration in Service Composition

Previous research on ASC has only focused on one-step composition, neglecting the need for additional processes to reach the final goal when one-step composition falls short. This type of procedure can be considered a multistep composition that involves the orchestration of nested workflows. In the given scenario, the trip scheduling service can be generated by an ASC, where the ASC planner develops an abstract workflow using staged composition and execution for traffic routes and hotels between Aizu and LA. Next, the ASC discovers and selects optimal service candidates using QoS and user constraints on the workflow. However, to achieve the final goal, the selected trip schedule must be passed to the reservation process, and the results of both processes must be orchestrated to create the group tour. As a result, an outer ASC is required to combine the results of the subprocesses to achieve the final goal.

1.3.2 Conceptual Model of Nested Composition

There are two approaches to managing workflow in service composition: centralized orchestration and distributed choreography [6]. In general, ASC employs the centralized management of services for a specific goal and adopts the orchestration paradigm.

At a higher level, workflow management mirrors orchestration and involves integrating three types of services: dynamic composite services, static composite services, and atomic services, as depicted in Fig. 2. A dynamic composite service is created on the fly by an ASC, while a static composite service is a predefined service that may have been produced manually or through extraction tools. An atomic service may be found in a service repository, and we consider such services to be static ones. These services are utilized by a nested ASC (NASC) to accomplish a goal at an outer level. Generally, a hierarchy of compositions can be established to attain the final goal, which can be located by matching their Input–Output–Precondition–Effects (IOPEs) during the composer's discovery function.

1.3.3 Workflow Orchestration in Nested Composition

Top-Down Approach

The conventional approach of ASC for services composition is the orchestration strategy, which follows a top-down approach. The top-down approach modifies the discovery stage and the execution stage to accommodate the characteristics of

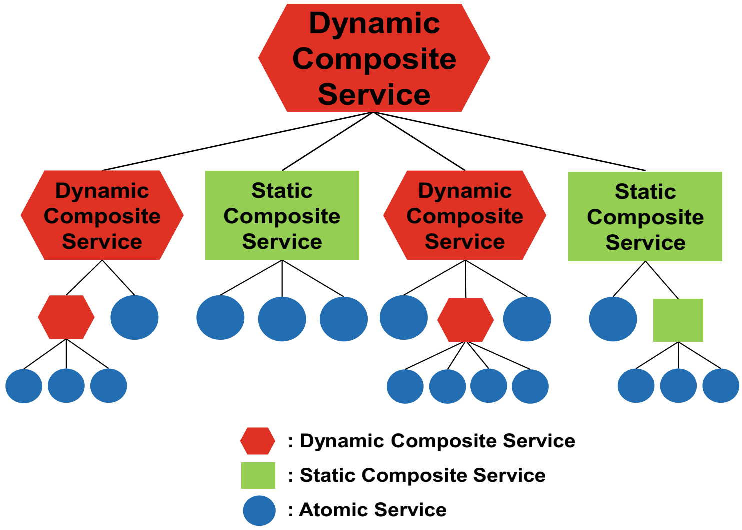


Fig. 2 General composition model with nested composition

dynamic services, which potentially conform to more than one static service. In the planning stage, the ASC generates several sequences of service types and selects optimal service instances for each type based on the operators required. The ASC then combines the selected operators to output composite services. To fulfill a desired composition, Procedure 1 outlines the essential methods and parameters for nested composition using the top-down approach. The flow of the procedure starts with the ASC function taking the *PlanningDomain* variables and a set of *UserConstraints*. *PlanningDomain* has a problem space and a planning space that are used to invoke planning for one domain, while *UserConstraints* contain information about user constraints and functional goals. *AbstractWorkflow* stores the state of the results derived from the planning and discovery stages, while *ConcreteWorkflow* stores the state of the results derived from the selection and execution stages. The *generateAbstractWorkflow* method in line 7 receives the *PlanningDomain* variable and generates an *AbstractWorkflow* set, whose size depends on the number of generated sequences from *ServiceTypes*.

The top-down composition approach's discovery stage is illustrated between lines 9 and 20 in Procedure 1. This stage discovers the service instances for each service type. If a *ServiceType* cannot be discovered from the service repositories or is deemed a dynamic service, the internal ASC is recursively invoked to generate a *ServiceInstance* set for the *ServiceType*. Once the inner ASC is called, the *generatePlanningDomain* method generates a *PlanningDomain* variable, which provides the necessary information (required planning and problem spaces) for a planner to derive the desired workflows for the domain. Using the generated *PlanningDomain*, the inner

ASC creates services that fulfill the functional goal in the domain. The *setServiceInstances* and *setServiceTypes* methods (lines 17 and 19) are then used to write the obtained parameters to the corresponding variables.

At line 22, the *doCPSelection* method is called, which generates a set of *ConcreteWorkflows*. This method selects the optimal *ServiceInstance* for each *ServiceType* from the discovered *ServiceInstances* by using the NFPs information and the user constraint satisfaction measures. Therefore, the generated *ConcreteWorkflow* set has optimal service instances for the service types. This process results in obtaining the results such as the selection stage in Fig. 2.

Procedure 1: Managing nested workflow by a top-down approach to ASC

```
Require: PlanningDomain pd, UserConstraint uc
Ensure: ExecutableWorkflow ←
        ASC(PlanningDomain pd, UserConstraint uc)
1: Let ServiceType be service types for an AbstractWorkflow
2: Let ServiceInstance be service instances for a ServiceType
3: Let AbstractWorkflow be workflows, consisting of a set of
        ServiceTypes
4: Let ConcreteWorkflow be workflows, consisting of a set of
        ServiceTypes
5: Let ExecutableWorkflow be executable workflow languages
6: // Planning Stage
7: AbstractWorkflow ← generateAbstractWorkflow(pd)
8: // Discovery Stage
9: for i=0 to AbstractWorkflow.length do
10: ServiceType ← AbstractWorkflow[i].getServiceTypes()
11: for j = 0 to ServiceType.length do
12: ServiceInstance ← discoverServices(ServiceType[j])
13:      if ServiceInstance.length is 0 or ServiceType[j]
        calls a dynamic service then
14: pd2 ← generatePlanningDomain(ServiceType[j])
15: ServiceInstance ← (ServiceInstance)ASC(pd2, uc)
16: end if
17: ServiceType[j].setServiceInstances(ServiceInstance)
18: end for
19: AbstractWorkflow[i].setServiceTypes(ServiceType)
20: end for
21: // Selection Stage
22: ConcreteWorkflow ← doCPSelection(pd, uc, AbstractWorkflow)
23: // Prepared for Execution Stage
24: ExecutableWorkflow ← generateExecutableServices(pd,
        ConcreteWorkflow)
25: for i = 0 to ExecutableWorkflow.length do
26: ExecutableWorkflow[i].temporalpublish()
27: end for
28: if Process is in Inner ASC then
29: return ExecutableWorkflow
30: else
31: return Invocation Results of ExecutableWorkflow
32: end if
```

In the top-down composition approach, the execution stage corresponds to lines 25 and 32 in Procedure 1. At line 24, the *generateExecutableServices* method receives the *PlanningDomain* and *ConcreteWorkflow* sets, generating a set of *Executable-Workflows*. Once the *ExecutableWorkflow* set is generated, it is temporarily published as services that can be accessed by a client application. If the process is in the inner ASC, the generated *ExecutableWorkflow* set denoted in line 29 is returned. In this process, the parent ASC can obtain the *ServiceInstance* set from the Inner ASC shown in line 15. Also, in line 15, the *ExecutableWorkflow* set is converted to a *ServiceInstance* set. This implies that *ExecutableWorkflow* is substantially similar to *ServiceInstance*. At line 31, the invocation results of the executable workflow are returned. This process is invoked when the highest services located on the nested structure are generated. Thus, by using Procedure 1, the inner ASC can be called recursively, and the nested dynamic composition structure can be generated.

Bottom-Up Approach

When it comes to managing nested dynamic service composition, a bottom-up approach can involve human inputs for service selection. In practice, clients' manual selections are essential as automated selections by ASC may not always meet their requirements. To implement the bottom-up approach, the selection stage of the top-down orchestration can be modified, replacing the selection based on composition properties such as QoS information and user constraints with human selections. Several dialogue forms can be used to obtain the human selections, such as asking for preferred service type sequences or preferred service instances. For further details on this approach, refer to Lécué [7].

1.3.4 Lower Level Architecture for Example Scenario

While the high-level architecture outlines the conceptual framework of the scalable ASC, the lower level architecture focuses on implementation details. Our research does not encompass a complete transformation of the design specification, such as in a model-driven architecture. Instead, we illustrate in detail how our upper level architecture can be designed and implemented by adopting a divide-and-conquer approach based on domain details. This approach enables us to focus on the two core composition parts, namely the planning stage and the selection stage, as well as the two new parts of transformation and orchestration.

User Request for the Group Tour Scenario

We use the scenario introduced in Sect. 2.3 to provide details of the request and composition. In this scenario, a user contacts an agent to plan a tour for a group to travel to Los Angeles. The user specifies their goal as a functional requirement, along with non-functional requirements such as the departure and arrival dates and locations. The user may also provide additional constraints and preferences.

User Request (Input)

Users can describe their requests in various forms, such as natural language, logic languages, a graphical user interface (GUI), or dedicated goal modeling languages. Inferencing is necessary to automatically derive real services from abstract goals or requests for composition. In our scenario, the functionality and non-functionality requirements are expressed in first-order logic. However, the terms used may not be terminal, requiring transformation.

1.3.5 Investigation of Functional Scalability

The proposed nested multilevel composition provides functional scalability as a composition parameter. Two approaches to scalability are possible: bottom-up and top-down. A bottom-up approach is suitable for user-driven composition, while a top-down approach is more appropriate for machine-planned composition without user intervention. In this paper, we only introduce the top-down approach, and for the sake of clarity, we limit scalability to a single independent domain to simplify the complex composition problem. An example that demonstrates functional scalable composition in the travel domain can be found on our demonstration site [8].

1.4 Architecture for Scalable ASC

In this section, we will present the complete architecture for workflow orchestration, nested composition, and NFP transformations in ASC using UML. Figure 3 displays the composition architecture as a whole [4].

1.4.1 Top-Level Architecture

While the existing four-stage architecture seeks to identify services that fulfill a specific goal, two additional parts deal with nested workflow orchestration for more general goals. This contributes to functional scalability and links NFPs from the abstract to the concrete level. As describing the entire structure and behavior of the ASC architecture is complex, we will use top-level class and sequence diagrams in this subsection to illustrate the abstract concepts of the architecture. A more detailed design using middle-level diagrams, with an instance of our motivating scenario, will be provided in the following subsection. Figure 3 contains top-level class diagrams for the ASC's structure. The following sections mainly explain the

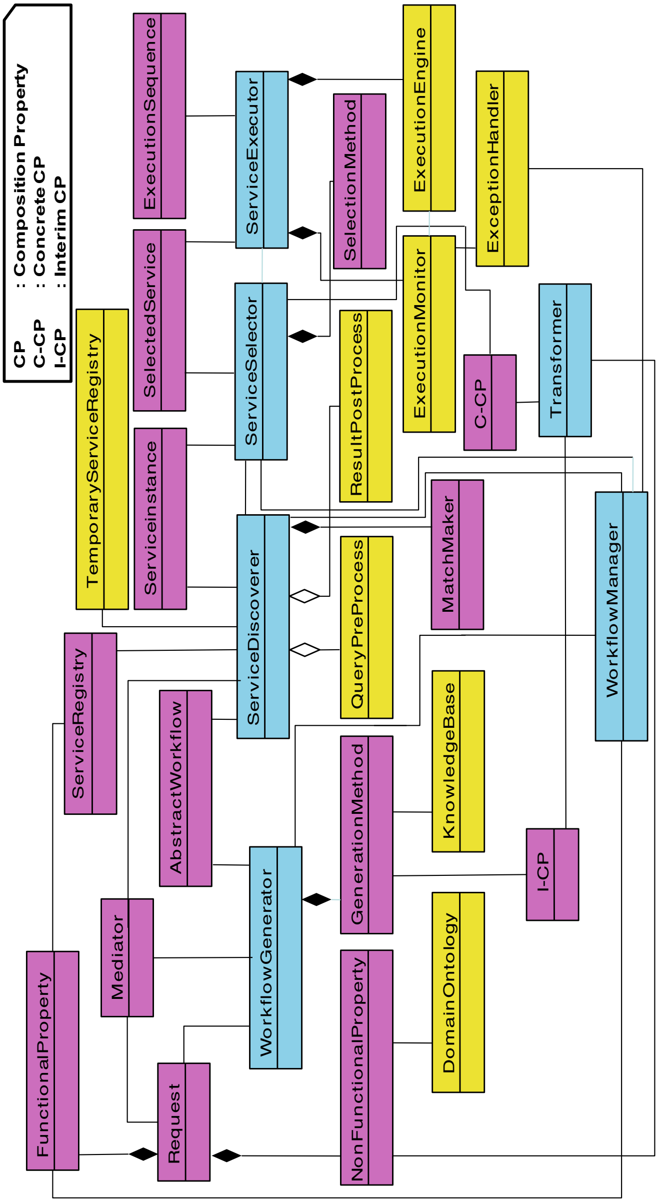


Fig. 3 Top-Level UML class diagram of scalable ASC

classes' functionalities. The first four classes cover the functions found in the existing four stages.

The *WorkflowGenerato*r creates an abstract workflow to meet the user's request's *FunctionalProperty*. The methods may include planning algorithms, FSM, workflow generation, or a dedicated application. The *ServiceDiscoverer* provides discovery services, with the primary objective of providing candidate services (service instances) to fulfill the tasks' functionalities from the *WorkflowGenerator*. The *ServiceSelector* selects service instances to meet the NFPs from the user or the *WorkflowGenerator*. All abstract NFPs must be transformed into concrete NFPs with binding information before selection. The *ServiceSelector* can use any selection method, such as planning, integer linear programming, or CSP-constraint optimizing. The *ServiceExecutor* executes the selected service instances from the *ServiceSelector*. It has the *ExecutionMonitor* class, which tracks the services' execution in the *ExecutionEngine* to maintain performance quality.

The next two classes enable scalable composition with seamless processing of ASC. The Transformer converts abstract NFPs into concrete NFPs with binding information, so the NFPs can be understood by the *ServiceSelector*. The Transformer captures the meaning of the terms in the abstract NFPs to link them to intermediate NFP terms composed of terminal terms. It then transforms the intermediate NFPs into concrete NFPs. As described in Sect. 4, ontology matching between terminal terms of the intermediate NFPs and the service domain ontology processes the transformation.

The *OrchestrationManager* orchestrates the entire service composition. It refines the user's generic goals into concrete goals found in the registry and analyzes the goals to identify services that meet the goals as required. The manager orchestrates all the composition steps in the ASC's nested structure to achieve the final goal. It allows the ASC to create new composite services for ones not found in the service registry, as described in the previous section. The nested workflow orchestration management gives our system multi-level functionally scalable composition with dynamically derived goal parameters.

Planning with HTN

In the planning stage, we can use the composition of predefined processes in OWL-S, BPEL, or WSMO to describe the workflow for reaching a goal statically. However, to deal with more general goals, we have adopted a Hierarchical Task Network (HTN) planner [9] to develop the workflow dynamically. It provides a strong foundation for planning an abstract workflow at this stage. We can encode an OWL-S process model or any formation of the user's requests in the HTN planner. The HTN planner then develops a plan that describes a workflow to reach the goal state. Definition 6 formally describes HTN, which has been drawn in UML to combine it with the upper level ASC architecture.

Definition 1 HTN planning problem and plan. *A planning problem is a tuple Plan* $=$ *< State, Task, Domain >, and Domain* $=$ *< Axiom, Operator, Method > where:* A plan, $\Pi = (\pi_1, \pi_2, \ldots, \pi_n)$, is a sequence of instantiated operators that will achieve Task from State in Domain, which will be the abstract workflow in our composition scheme.

In the composition system, a π_i is to be mapped to a task of the abstract workflow directly. For example, in the scenario, the planner generates an abstract workflow, (*moveByVehicle(Aizu,Koriyama), moveByVehicle(Koriyama, Tokyo), moveByVehicle(Tokyo, LA)…, stayAtHotel(LA)).*

Property Transformation

A user's original nonfunctional property may contain complex terms that need to be transformed, and the domain ontology for the transformation is expected to have clear meanings for real services. However, finding concrete terms from abstract constraints requires human intelligence. Our transformation design focuses on an algorithm that transforms intermediate constraints into concrete constraints, which are described in Sect. 4.2 along with their relations and context. The algorithm should also consider the attributes of real services, which have variable domains related to the domain ontology and references.

To include all service classes and variables in the transformation, the proposed algorithm uses ontologies that can be updated if new services and conditions are added. The ontology is divided into unchangeable and variable concepts, and we construct the former using existing knowledge and ontologies. The latter is added by searching for existing websites and services, and synonymous terms are merged when adding new classes to the ontology. This process continues until no new classes are found in the websites selected for our target.

Service Selection for Execution Using CSP

To find a concrete sequence of service instances for execution that matches those discovered for the abstract service sequence, service selection (also known as physical composition) must identify services that best satisfy nonfunctional properties. Our implementation employs a Constraint Satisfaction Problem (CSP) to identify such service instances.

Definition 2 NFP-Based Service Discovery and Service Selection Using CSP.

In our approach, we distinguish between nonfunctional properties (NFP) and other nonfunctional properties related to system characteristics. NFPs are considered in the discovery stage, while user preferences and constraints are taken into account during the selection stage after NFPs have been identified.

The function of extracting candidate services from abstract workflow by NFP is:

- *GenerateCandidates*: $\Pi \times \text{NFP} \rightarrow \text{X}$, where:

- Π is a series of abstract tasks generated by the planning stage, and NFP represents attributes for NFP.
- X is a series of candidate services generated by "*GenerateCandidates*" that were filtered by NFP attributes.

We describe the service selection based on CSP, CSP = $< X, D, C >$, where:

- X is the same as in Definition 7;
- D is a set of instances of the ontology of input or output for the real process, e.g., the ontology of <IOPE> in OWL-S; and
- C is a set of constraints, which can be user's constraints or preferences in the form of relations and terms.

Constraints can be dynamically altered due to system effects or user actions. In our scenario, a sequence of service instances that best satisfy nonfunctional properties can be selected and passed to the next stage for a combination of service operators. It's important to note that other service instance sequences can also be chosen.

Orchestration for Scalable Composition

The primary goal of our scenario, "*MakeATourGroup*," involves developing a plan for achieving the goal by creating a sequence of subtasks like "*TripSchedule*," "Reservation," and "*MakeTourGroup*." The inner ASC recursively generates another plan for the "*TripSchedule*" task, as explained in this section. In some situations, there may be a need for more compositions upwards or downwards. For instance, when a flight departs early in the morning in the "*TripSchedule*," the passenger may need to stay in a hotel near Narita, requiring another plan at a lower level of composition. The orchestration manager controls multiple compositions to provide functional scalability as a composition parameter.

2 Better Service Composition Using Service Graph

2.1 Introduction

This section presents a new method for improving the quality of service composition by utilizing a global social service network. Service composition is the process of creating new services from existing ones, which has been studied by both academia and enterprises. Various approaches to service composition have been proposed, such as template-based, Petri-net-based, AI-planning-based, graph-theory-based, and logic-based approaches. However, these approaches have not adequately addressed the issue of optimizing end-to-end quality requirements. Semantic-based and QoS-based approaches have been proposed to improve quality, but they still suffer from scalability issues. The ontology reasoning and QoS optimization among isolated service islands are the primary reasons for this. To address this, functional clustering-based approaches have been proposed to preprocess time-consuming operations among isolated services. However, these approaches have ignored local optimization of QoS attributes within a cluster, which is an NP-hard problem. Heuristic and genetic approaches have been proposed, but they have limitations in terms of scalability and practicality. Instead, a "good enough" composition satisfying global constraints is a more practical approach to large search spaces. However, current approaches do not consider service sociability, which is crucial for interdependent web services. To address this, the proposed methodology utilizes a global social service network to enhance the service's social activities and improve the quality of service composition. The approach preprocesses time-consuming operations and reduces the search space for QoS optimization by mapping the GSSN into a service cluster network with local optimization of aggregated QoS attributes.

2.2 Background and Related Work

In this section, we appraise the existing service composition approaches in terms of scalability and sociability; then we argue that services' sociability provides the missing ingredients that will evolve isolated services into a GSSN to improve the quality of service composition.

2.2.1 Scalability Issue

First, semantics have been proposed as a key to increasing automation in applying web services and managing web processes within and across enterprises. Currently, many semantic service composition approaches have been proposed and many semantic composition projects such as IRS [10], and SHOP2 have been developed. However, these existing methods and semantic tools are registry-based, such as UDDI, which has several drawbacks. For instance, services are treated as isolated service islands, knowing only about themselves, but not about the peers that they would like to work with in compositions or that they would compete against in service selection. As a consequence, they still either stay in a semiautomated state, which means that composition requires a high level of user interaction, or risk low efficiency of producing a composition plan because a direct reasoning style is required. In overall terms, performance issues resulting from extensive ontology reasoning and other intensive manual operations are still the main problems in current semantic service composition approaches. Second, the QoS-based approach selects the best composition solution and component services that satisfy the end-to-end quality requirements. Yu and Lin [11] define the problem as a multidimensional multichoice 0–1 knapsack problem, as well as a multiconstraint optimal path problem. Zeng et al. [8] presented a global planning approach to select an optimal execution plan by means of a linear programming model. Ardagna and Pernici [12] modeled the service composition as a mixed integer–linear problem in which both local and global constraints are taken into account. Linear programming methods are very effective when the problem is small. However, these methods suffer from poor scalability because of the exponential time complexity of the applied search algorithms. Lécué and Mehandjiev [13] proposed heuristic algorithms that can be used to find a near-optimal solution more efficiently than exact solutions. Lécué [7] presented a method for semantic web service composition based on genetic algorithms and using both semantic links between I/O parameters and QoS attributes. Despite the significant improvement of these algorithms compared with exact solutions, neither algorithm scales with respect to the number of candidate web services, and hence they are not suitable for real-time service composition.

2.2.2 Sociability Issue

Nowadays, services consider only their own functional and nonfunctional detail through the life cycle of the service and ignore services' social activities. As a consequence, service composition approaches do not record services' past social interactions and cannot promise the quality of service composition. To address these issues, the sociability issue is introduced to improve the quality of service discovery and composition. A service's sociability is the skill, tendency, or property of being sociable or social, of interacting well with related services, which is supported by network models we refer to here as service social networks. A service's sociability issue is to capture how web services interact via service social networks, to know with whom they have worked in the past and with whom they would potentially like to work in the future. A service social network is constructed to reflect services' social reality, describing the mutual consciousness of mutual agreement about a social situation and supporting future services' social activities [14]. Therefore, by connecting distributed services into one single service social network, we can capitalize on users' willingness to interact, share, collaborate, and make recommendations for improving the quality of service composition.

Some approaches have been proposed to use service social networks to enable GPS-like support service discovery and service composition. Tan et al. [15] proposed a service map to enable recommending relevant services for service consumers and to find an operation chain to connect two operations based on others' past usage. Zhang et al. [16] proposed a novel approach of proactively recommending services in a workflow composition to help domain scientists find relevant services and to reuse successful processes based on social networks. Maamar et al. [17] proposed an approach to use social networks for web services discovery. However, all these approaches struggle to construct a single service social network dynamically, as they do when constructing the WWW, which hampers the services' sociability. These approaches do not consider interlinking web services on the open web into one single service social network to enhance services' sociability, as two persons are interlinked when they are friends in a social network or two actors are linked in an actor collaboration graph when they have acted in the same movie. In this paper, we propose a methodology to interlink distributed services into a GSSN with social links using the quality of the social link to consider a service's social activities and its popularity to provide a network model on a global scale for supporting services' social activities.

2.3 Motivating Example

Conventional methods for service composition (such as semantic-based and QoS-based approaches) have typically viewed services as disconnected islands, as depicted in Fig. 1a. This perspective gives rise to several difficulties, including poor scalability, an exponentially increasing search time in large search spaces, and a lack of service sociability resulting from the segregation of services. Please note that the web services {Si} demonstrated in Fig. 1 can be found in Chen and Paik [18].

The goal of this study was to enhance the quality of service composition by establishing connections between isolated services and creating a Global Social Service Network (GSSN), as depicted in Fig. 1b. By doing so, a quality-driven service composition approach could be developed, as shown in Fig. 4c, where a group of red services (S1, S2, S5, S10, S11, S7, S8) were composed to form a workflow service for users. However, this presented several challenges, including the quantitative measurement of relationships between services, constructing the GSSN with generic aspects, and exploiting the GSSN to improve service composition. To address these issues, the study proposed the "quality of social links" approach to quantify the strength of relationships between services, considering not only the functional and non-functional details but also past social interactions and popularity. A novel platform was also developed to construct the GSSN, considering four generic aspects of the network. Finally, a quality-driven service composition approach was introduced, with key features including the GSSN as a network model for improved sociability, preprocessing ontology reasoning, and semantic-related computing to reduce search time and improve scalability, mapping the GSSN into service cluster networks to reduce search space, and the development of a novel quality-driven workflow search algorithm based on the GSSN and quality of social links.

2.4 Connecting Isolated Services into GSSN

In this section, we present our novel framework for constructing a GSSN to address the issue of isolated service islands and improve the sociability of services, resulting in better quality web service compositions. The GSSN is designed to connect distributed services across domains through social links, similar to how RDF links connect distributed data into a single global data space in the web of data. This interlinking of services into a GSSN enhances their sociability and collaboration, thereby facilitating service compositions. Additionally, the GSSN includes descriptions of

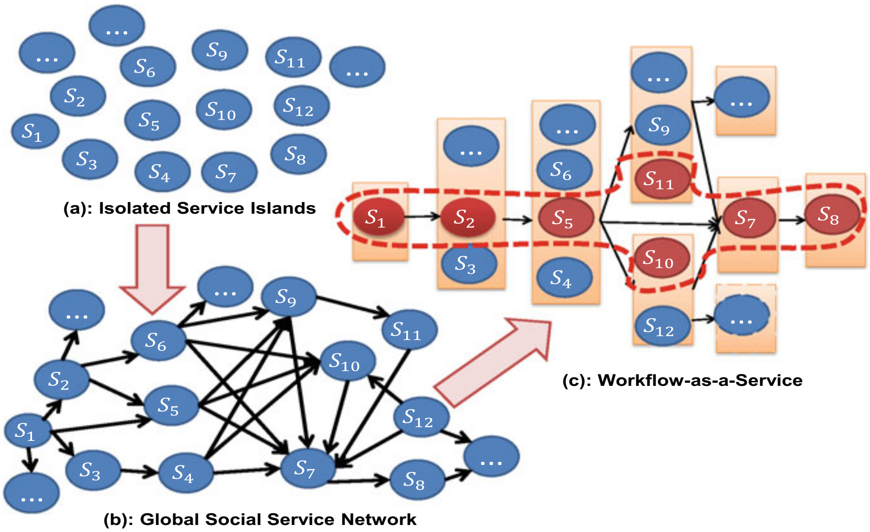


Fig. 4 Example of quality-driven service composition based on the GSSN

service society features, enabling higher level functions for composition components, such as inferring, planning, and coordinating social activities in the space.

Definition 3 (GSSN). A GSSN is a global space for a service social network to describe a service's social properties; its structure is a directed graph G = < V, E > on the web, where:

- V represents a set of nodes, with each node being a linked social service; and
- E represents a set of directed edges, with each edge corresponding to social link.

2.4.1 Pattern of Social Link

We interconnect isolated services on a GSSN by creating social links between them. These social links are patterned to make typed statements that can link any services. The patterns of social links indicate the functional relationships between the resource service and the target services based on service data correlations, which are data mappings between the input/output (I/O) attributes of services. The target service is an object in an RDF triple, which is published on the open web and usually linked to resource services. In contrast, the resource service is a subject that is linked to target services. We introduce the concept of Peer social link to make typed statements that connect peer services that can work together to provide a more complex service. *Peer social link* can be illustrated by the following rules, including sequential ($L(\leftarrow)$ and $L(\rightarrow)$), parallel ($L(<\oplus)$ and $L(\oplus>)$), and conditional ($L(<||)$ and $L(||>)$) routing. Further, to make typed statements that link services that perform a specific common function, *Cluster social link L*(=) is proposed to connect services offering similar functionalities. For more details about the definition of pattern of social link, please refer to our previous work.

2.4.2 Quality of Social Link

Four generic quality criteria for social links have been considered, which are denoted as L(R, T): (1) functionality homophily $Q_{FH}(R, T)$; (2) QoS preference $Q_{QoS}(R, T)$; (3) sociability preference $Q_{SP}(R, T)$; and (4) preferential service connectivity $Q_{PSC}(R, T)$. The four generic quality criteria can be combined to evaluate the quality of a social link using quality aggregation rules. For more details about these generic aspects, you can refer to our previous work [19]. The quality vector of social link is defined by the quality of social link Q(R, T).

Definition 4 (Quality of Social Link). Given resource service R and a set of target services T_n, the quality of social link $Q(R, T_n)$ provides a measure for the quality of the links between R and T_n for social link, and can be described as follows:

$$Q(R, Tn) = \langle QDSR(R,Tn), QQoS(R,Tn), QSP(R,Tn), QPSC(R,Tn)\rangle \quad (1)$$

Equation (1) means that the social link quality between R and T_n relies on the four quality criteria. Higher values indicate better quality of social link. Given a resource service R and a set of target services T_n, one can select the link with the best functional quality $Q_{DSR}(R, T_n)$, best nonfunctional quality values $Q_{QoS}(R, T_n)$ (such as the cheapest and fastest services), sociability preference $Q_{SP}(R, T_n)$, preferential service connectivity $Q_{PSC}(R, T_n)$, or the selection can be a compromise (according to the user preferences) among the four criteria. First, selecting links with the best functional quality $Q_{DSR}(R, T_n)$) will make sure easy end-to-end integration between services by minimizing semantic and syntactic mediators, and by providing seamless deployment and execution of compositions.

To reduce the cost of data mediation, the selection of appropriate criteria is crucial. Choosing links that include services with superior nonfunctional quality values, such as price and response time, will ensure quality of social links that are easily understood by most users. This approach is particularly effective when shared data among services is relatively homogeneous, such as when services align their data during the description phase and most of their exchanged data matches perfectly, or when all data mediators are known during the design phase.

Furthermore, selecting links that include services with a high sociability preference will enhance the quality of social links by considering past and future collaborative partners. This approach reflects the service social reality, where services that have interacted frequently in the past are likely to be linked with social links.

Lastly, selecting links that include services with the best preferential service connectivity will improve the quality of social links by linking with well-known and popular services that have high connectivity, increasing the probability that they will be recruited by other well-known services.

2.4.3 Construction of *GSSN*

When constructing a GSSN for improved service composition, there are four generic aspects that must be taken into consideration. Firstly, the growth aspect, which highlights that a GSSN is constantly evolving and expanding as new services are added. This means that the number of vertices, N, will continue to increase over time, much like the exponential growth of the www and the research literature. Secondly, the preferential service connectivity aspect, which emphasizes that connections between vertices in a GSSN are not random or uniform, but rather exhibit preferential service connectivity. This means that vertices with a larger number of connections are more likely to link to other vertices. Thirdly, the competitive aspect, which underscores that each node in a GSSN has an inherent ability to compete for edges at the expense of other nodes. For instance, a higher dependency satisfaction rate between a resource service and a target service can make the target service more competitive. Lastly, the adaptation aspect, which highlights that social links in a GSSN are regularly updated to reflect service social reality based on the quality of the social link. This means that "old" links may be replaced by "new" social links with higher quality of social link.

To account for the growing nature of the network, we begin with a small number (m_0) of vertices and add a new vertex at every time step with $m \le m_0$ edges, linking the new vertex to m different vertices already present in the network. To incorporate preferential attachment, we assume that the probability of a new vertex being connected to vertex i (represented as Π_i) is dependent on the connectivity (k_i) of that vertex, such that $\Pi_i = k_i / \Sigma_j k_j$. It should be noted that in this study, T_n is treated as a vertex i because it is a primary functional component of the service network. Therefore, the connectivity k_i of vertex i can be calculated as $k_i = \sum_{j=1}^{n} k_j/n$. To incorporate the competitive aspect, we assign a fitness parameter η_i to each vertex so that when a new service W is added to the GSSN at every time step, it has a fitness value η_i that depends on Q_{DSR}, $Q_{QoS,}$ and Q_{Sp}. In order to reflect the service's social reality, we rewrite "old" social links with low quality of social link and add new social links with high quality of social link throughout the lifetime of the network. Based on the previous analysis, we also quantify the quality of social link as

$$Q(R, Tn) = \frac{\eta i k i}{\sum_j \eta j k j} \tag{2}$$

where k_i is the degree of node i and η_i is a fitness parameter that represents the internal superiority of the ith node, as each node has the intrinsic ability to compete for edges at the expense of other nodes. η_i can be calculated as

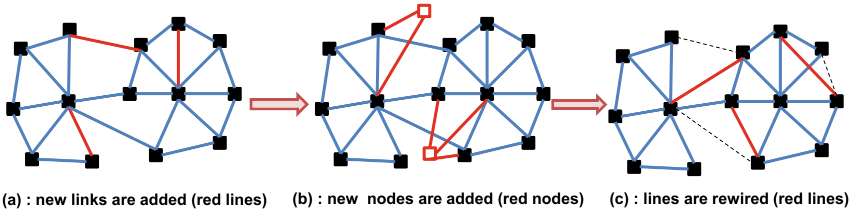


Fig. 5 GSSN construction process

$$\eta i = \left(wDSRQ^{l}\mathrm{DSR}(\mathrm{R,T}n) + wQosQ^{l}\mathrm{QoS}(\mathrm{R,T}n) + wSPQ^{l}\mathrm{SP}(\mathrm{R,T}n)\right). \quad (3)$$

where $W_{DSR} + W_{QoS} + W_{SP} = 1$ and l is a constant in positive integer number, which is used to set the weight of η_i. Higher value indicates that node i is better at competition.

Taking into account the aforementioned concerns, we adopt an algorithm for constructing our network, which relies on four parameters: m_0 (the initial number of nodes), m (the number of links added or rewired at each step of the algorithm), p (the probability of adding links), and q (the probability of rewiring edges).

The algorithm for building the network takes into account four parameters: m_0 (initial number of nodes), m (number of links added or rewired per step), p (probability of adding links), and q (probability of edge rewiring). The procedure starts with m0 nodes and performs one of three actions at each step.

1. With probability p and $m < = m_0$, new links are added by selecting the endpoint based on the quality of social link given by Eq. (2) (shown in Fig. 5a). This process is repeated m times.
2. With probability q, m edges are rewired by selecting a random node I and its link L_{ij} with the lowest quality of social link. The link L_{ij} is removed, and another node z is selected based on the quality of social link (shown in Fig. 5b), and a new link L_{iz} is added.
3. with probability *1–p–q,* a new node with m links is added, and the new links connect to m other nodes selected based on the quality of social link (shown in Fig. 5c).

The algorithm stops once the desired number of nodes (N) is reached. Actions 1 and 2 satisfy the adaptation aspect of the network, while action 3 achieves the growth aspect. The quality of social link with fitness parameter η_i satisfies the competitive and preferential attachment aspects.

Properties of GSSN

The probability Π_i that a new node in *GSSN* will connect to an already present node i depends on two factors, namely, the connectivity k_i and the fitness parameter η_i of that node, as given by Eq. (2):

$$\prod i = Q(\mathrm{R,Tn}) = \frac{n_i k_i}{\sum_j k_j n_j} \tag{4}$$

The simplest possible way to incorporate the joint influence of fitness and connectivity on the rate of adding new links to a node is through the generalized preferential attachment. Therefore, at each time step, a new node *i* is added to the service network with fitness parameter *i* selected from the distribution $\rho(\eta)$. To analyze the scaling properties of this model, we use a continuum theory to predict the connectivity distribution, which is the probability that a node has *k* social links. The connectivity of a node *i*, denoted as k_i, increases at a rate proportional to the probability (4) that a new node will attach to it, resulting in the following expression:

$$\frac{\partial k_i}{\partial t} = m \frac{n_i k_i}{\sum_j k_j n_j} \tag{5}$$

The variable m considers that each new node contributes *m social links* to the network. To solve Eq. (5), we make an assumption that, similar to the scale-free model, the time evolution of ki follows a power law with a fitness-dependent function represented by $\beta(n)$. However, the network exhibits multiscaling, meaning that the dynamic exponent depends on the fitness parameter of each node, denoted by η_i.

$$k_{n_i}(t, t_i) = m\left(\frac{t}{t_i}\right)^{\beta(n_i)} \tag{6}$$

where the dynamic exponent satisfies

$$\beta(n) = \frac{n}{C} \; with \; C = \int \rho(n) \frac{n}{1 - \beta(n)} dn \tag{7}$$

Hence, the function β is characterized by a range of values determined by the distribution of fitness. Consequently, the connectivity distribution *P(k)*, indicating the likelihood of a node having k social links, can be determined by summing up different power laws with varying weights. Specifically, we need to compute the cumulative probability that a node's connectivity $k_\eta(t)$ exceeds *k*, in order to find *P(k).*

$$P(k_n(t) > k) = P\left(t_i < t\left(\frac{m}{k}\right)^{C/n}\right) = t\left(\frac{m}{k}\right)^{\frac{C}{n}} \tag{8}$$

Thus, the connectivity distribution is given by the integral:

$$P(k) = \int_0^{n_{\max}} dn \frac{\partial P(k_n(t) > k)}{\partial t} \propto \int dn \rho(n) \frac{C}{n} \left(\frac{m}{k}\right)^{\frac{C}{n}+1} \tag{9}$$

2.5 Workflow as a Service

In the previous sections, a GSSN was created to connect distributed services and provide a network model for their social activities. In this section, we introduce a new approach for exploring a GSSN that offers workflow as a service. As business processes and scientific problems become increasingly complex, service composition plans can grow to involve hundreds of thousands of services. Unfortunately, traditional service composition approaches struggle with creating large service compositions or guaranteeing their quality. To address this, we preprocess the time-consuming ontology reasoning and other semantic-related computing during GSSN construction, and we reduce the search dimension by mapping the GSSN into a service cluster network based on social links. Our novel quality-driven workflow search algorithm improves the success rate of service composition by considering four generic quality criteria: $Q_{DSR}(R, T_n)$, $Q_{QoS}(R, T_n)$, $Q_{SP}(R, T_n)$, and $Q_{PSC}(R, T_n)$. This algorithm is based on the quality of social link and differs from traditional service network-based workflow-search algorithms.

Definition 5 (Workflow as a service). Given some existing services S_n (1 < n < N), including the original service S_O and the service for destination S_d, e.g., an uncompleted workflow, Workflow as a service aims to find a subnetwork that starts with S_O and ends with S_d based on the GSSN containing a finite set of services, S_1, S_2, … S_i, … S_j, … S_m, such that:

1) $L(S_i, S_j)$ is a peer social link;

2) $foreach Si, Si.I \in \bigcup_{j=1}^{m} Sj.O;$

3) $\sum_{i,j \in \{adjacent\,node\}}^{m} (1 - Q(\mathrm{S}i, Sj)) shall\,be\,\mathrm{min}\,imal.$

2.5.1 Method for Workflow as a Service

In this subsection, we present a new method for representing service social activities and finding service chains for service composition on the GSSN, analogous to exploring social networks for friends. Our approach involves translating the GSSN into a service cluster network by following social links, which reduces the search space. We then calculate the adjacency matrix and reachability matrix based on the service cluster network and propose an algorithm for providing workflow as a service with a cost assignment scheme based on the quality of social link.

Step 1: Constructing a Service Cluster Network Following Social Links.

As defined in Sect. 4.1, services that are grouped by cluster social links perform a common function and can be clustered into a Service Cluster (SC). An SC is a group of services that have similar functionality, and is denoted as follows:

$$\mathrm{SC} = \{S_1, S_2, \ldots, S_n\},$$

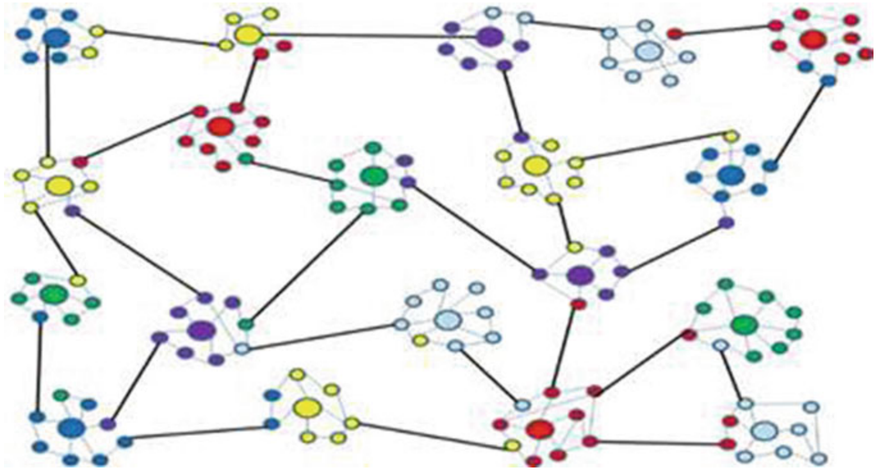


Fig. 6 Service cluster network

where S_1, …, S_n represent n services that all implement the same specific function of the service cluster. SCs linked by peer social links are interdependent in terms of functionality and QoS, which creates a network model called a service cluster network, as shown in Fig. 6.

Definition 6 (Service Cluster Network). A service cluster network is a directed graph $G' = \langle V', E' \rangle$, $V' = \{SC_1, SC_2, \ldots, SC_m\}$ and m is the number of service clusters, where:

- V' denotes a collection of nodes where each node represents a service cluster that has services connected by cluster social links;
- E' denotes a collection of directed edges, where each edge corresponds to a peer social link between service clusters.

To categorize the GSSN into a service cluster network by following social links, we propose Algorithm 1, which involves two sub-steps.

First, we explore the cluster social links to identify all related services that belong to the same service cluster. Starting from an unexplored node, we locate all social services linked by cluster social links (lines 3–10). Then, we add the social services to the service cluster and explore their social services, following their cluster social links. This process continues until all social services belonging to the service cluster are explored (lines 11–21). In the second sub-step, we establish social links between service clusters by following peer social links (lines 22–27), with social links between service clusters inheriting social links between services.

Step 2: Calculate the Adjacency Matrix and Reachability Matrix based on the Service Cluster Network.

To begin, we calculate an adjacency matrix A(G′) for the service cluster network with m vertices, which shows the relationships between service clusters according to peer social links. The adjacency matrix A(G′) for G′ = < V′,E′ > with m vertices SC_m can be denoted as

$$A(G') = (a_{ij})_{m\times m} \tag{10}$$

where a_{ij} represents the social link status between SC_i and SC_j; it is defined by

$$aij = \begin{cases} 1 \text{ with } L(SCi, SCj) \\ 0 \text{ with out } L(SCi, SCj) \end{cases} \tag{11}$$

After mapping service clusters into the adjacency matrix, we can extract the reachability matrix for pairs of service clusters based on the adjacency matrix. The reachability matrix R(G′) for G = < V′,E′ > with m vertices SC_m can be denoted as

$$R\left(G'\right) = (r_{ij})_{m\times m} \tag{12}$$

where r_{ij} describes the reachability relation between SC_i and SC_j. We first calculate the Lth power of A(G′) as

$$(A(G'))^L = A^L = (a_{ij}^{(L)})_{m\times m}(L \geq 2) \tag{13}$$

where $a_{ij}^{(L)}$ represents the number of L-step connections (or paths of length L) from SC_i to SC_j, and can be defined by

$$a_{ij}^{(L)} = \sum_{k=1}^{m} a_{ik}^{(L-1)} \cdot a_{kj}^{(1)}, A^1 = A = (a_{ij})_{m\times m} \tag{14}$$

Therefore, r_{ij} in Eq. (12) can be defined by

$$r_{ij} = \begin{cases} 1 if a_{ij}^{(1)} + a_{ij}^{(2)} + \cdots + a_{ij}^{m-1} > 0 \\ 0 \, otherwise \end{cases} \tag{15}$$

Step 3: Quality-Driven Algorithm for Workflow-Search and Service Composition Quality.

In order to enhance the quality of service composition in terms of scalability, we preprocess semantic-related computing during the network construction stage and construct the service cluster network based on the GSSN to reduce the search space. Additionally, to increase the success rate of service composition in terms of quality, we propose an algorithm in this section to find the path chaining services with the best quality, based on the quality of social links.

To compare the costs of potential solutions and identify the optimal quality of composition solution, we introduce a cost assignment scheme. We use weight functions to represent the cost of a solution graph of node R in a service cluster network, when node R is explored. The cost is denoted by Cost(R). We define the cost of node R recursively as follows:

Algorithm 1: Service cluster network construction

Input: *GSSN* G = <V, E>

Output: Service Cluster Network G′ = <V′, E′>

```
∀S ∈V, S.flag = false, Q = Φ .
foreach S_i ∈G.V do
  if(S_i.flag == true)
    continue;
  SC_i = new ServiceCluster(S_i);
  foreach S_k ∈G.V do
    if((<S_k, S_i> ∈E && L(S_k = S_i)) ||
       (<S_i, S_k> ∈E && L(S_i = S_k))) then
       Q.enqueue(S_k)
    end
  end
  while(Q != Φ ) do
    S_j := Q.dequeue;
       SC_i = SC_i ∪S_j;
       S_j.flag = true;
    foreach S_k ∈G.V do
      if((<S_k, S_i> ∈E && L(S_k = S_i) ||
        (<S_i, S_k> ∈E && L(S_i = S_k))) then
          Q.enqueue(S_k)
      end
    end
  end
  V' := V' ∪ SC_i;
  foreach <S_i, S_j> ∈E do
    if(S_i ∈CS_m && S_j ∈CS_n &&CS_m != SC_n) then
      E' = E' ∪<S_m, CS_n>;
    end
  end
end
```

$$\mathrm{Cost(R)} = <\ \mathrm{E(R); Cost(T_1), Cost(T_2), ...Cost(T_m)} >, \tag{16}$$

where $T_1, T_2, \ldots, T_m$ are super node R, $Cost(T_m)$ is the cost of the solution graph of node T_m, and E(R) is the cost or weight affected by node R.

On the other hand, to achieve the quality of service composition, we assign E(R) based on the quality of social link, which was defined in Eq. (2) as

$$\mathrm{E(R)} = 1 - \mathrm{Min\{Q(R, T_n)\}}. \tag{17}$$

Algorithm 2: Workflow-Search Algorithm

Input: Original Service S_o, Destiny Service S_d, G′ = <V′, E′>, R(G′)

Output: Workflow Solution with Minimal Cost

```
//WF(SCi): the workflow explored until SCi from Sd
SCo := getServiceCluster(So);
SCd:= getServiceCluster(Sd);
SQ.enqueue(SCd);
while SQ != Φ do
  SCi = dequeueMinCostNode(SQ);
  if(SCi==SCo)
    return WF(SCi);
  end
  {SC} = getNextNodewithIncomingSocialLink(SCi);
   For each SCj ∈{SC} do
     if(!reachability(SCj, So, R(G')))
        continue;
     if(L(SCi< ⊕ SCj) || L(SCj ⊕ >SCi) )
       {SCn} = getParallelNode(SCj);
       if( ∀SCn ∈ {SCn} : SCn.flag==1 )
           Cost(SCj) = Σ_{1}^{k} Cost(SCk) + E(SCi, SCk);
           WF(SCj) = WF(SCi) ∪{SCk};
          if(SCj.flag==0)
             SQ.enqueue(SCj);
             SCj.flag=1;
           end
        end
      end
      else if(L(SCi←SCj) || L(SCj⟶SCi)
          || L(SCi<||SCj) || L(SCj||>SCi) )
          if(Cost(SCj) > Cost(SCi)+E(SCi,SCj))
            Cost(SCj) = Cost(SCi)+E(SCi,SCj)
            WF(SCj) = WF(SCi) ∪SCj;
            if(SCj.flag==0)
              SQ.enqueue(SCj);
              SCj.flag=1;
            end
          end
       end else
    end for
  end while
```

Higher values of Q(R, T_n), show lower costs. Therefore, based on Eqs. (16) and (17), Cost(R) can be set to two different values in the process of search, depending on the social links pattern:

$$\text{Cost}(R) = \begin{cases} Min\{E(R) + Cost(Ti)\}, L(R \leftarrow Ti) \text{ or } \text{L(R<||Ti)} \\ \sum_1^n Cost(Ti) + E(R), L(\text{R}< \oplus \text{Ti}) \end{cases}. \quad (18)$$

In the first part, we have a recursive definition of the Cost(R) function used to evaluate the quality of a composition solution. If R is an $(L(SC_i \leftarrow SC_j)$ or $L(SC_i < ||SC_j)$ node, the cost is assigned the minimal value of the sum of the immediate parent of R plus the cost of R itself. If R is an $L(SC_i < \oplus SC_j)$ node, the cost is the sum of the costs of the solution graphs of all the immediate parents of R plus the cost of R itself. The lower the Cost(R) value, the better the quality of the composition solution.

In the third step, we propose a quality-driven workflow search algorithm based on the service cluster network to provide users with workflow as a service. The algorithm uses the Cost(R) function from the previous step to evaluate the quality of possible solutions. First, we find the service clusters SC_o and SC_d, and add SC_d to queue SQ (lines 1–3). Then, we find the node SCi in SQ with the smallest cost and check if it is the final solution (lines 5–8). Next, we find the nodes related to SC_i in G′ by following peer social links and check their reachability from related nodes to So (lines 9–12). We select the node with the next smallest cost to keep track of the minimal cost. If the selected node $L(SC_i < \oplus SC_j) \| L(SC_j \oplus >SC_i)$, we calculate the cost of its solution graph by taking the sum of the costs of its parents and the cost of the parallel node itself (lines 13–23). Otherwise, we calculate the cost of its solution graph by taking the cost of its parents and the cost of the node itself (lines 24–32). Finally, we select the workflow with the minimal cost and the highest quality of social link.

3 Better Service Discovery Using Attention of Service Invocation

Web services enable different software applications to communicate with each other over a network, relying on standard technologies such as XML, WSDL, SOAP, and UDDI. They are extensively used in e-business and have gained popularity among application developers. However, the growing number of services has made it challenging for consumers to find the most suitable services, impeding the development of web services. To tackle this issue, web service discovery plays a crucial role in matching customers' requests with appropriate services.

Service clustering, which involves grouping related services based on their domains or features, is an effective way to boost service discovery or composition. The process typically includes three main steps: Requirement Analysis, Feature Extraction, and Matcher, as illustrated in Fig. 7. Requirement Analysis helps understand consumers' needs, and Feature Extraction formats data in a way that computers can understand. Matcher then identifies the target services based on the request

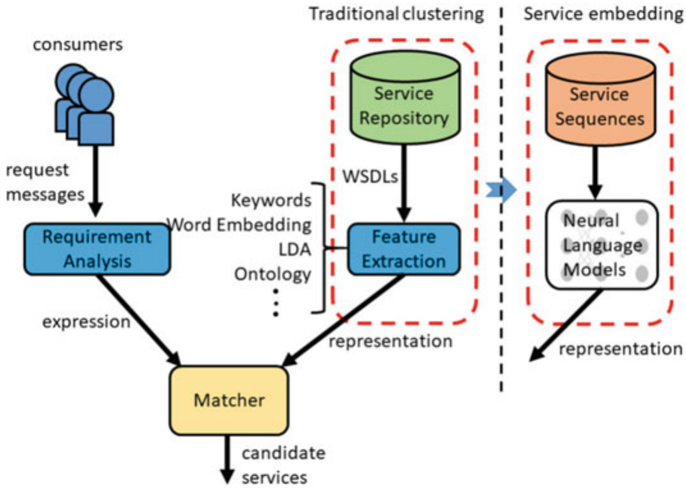


Fig. 7 Service clustering with service embedding in web service discovery

expression. If a single service cannot meet the consumer's requirements, service composition is introduced.

To extract features from WSDL documents, WSDL-based approaches such as keywords, word embedding, LDA, and ontology are commonly used in traditional service clustering. These approaches often involve service signatures, such as IOPEs, which include the names of operations, inputs, outputs, preconditions, and effects.

To achieve more realistic service clustering, considering the invocation association between services reflecting the actual invocation situation during service execution is another approach. This study proposes a novel service embedding method based on successful word embedding techniques in various contexts to facilitate this.

The increasing trend of microservices due to the rapid development of IoT, edge, and fog computing has led to better quality service compositions and more efficient mashup development. Service composition is frequently implemented in cloud and edge computing environments, which lack sufficient resources to support large-scale deep learning models. Therefore, more efficient lightweight approaches for service clustering are necessary.

This paper proposes a lightweight deep learning-based approach for service clustering that uses a BERT-based service embedding model with a novel transformer's encoder to perform semantic clustering of service composition. First, service embedding builds an informative cyclic framework in web service composition, with neural language networks learning service composition sequences and understanding the invocation relationship between services. Second, the pre-trained model generates representation vectors of all sequences, which are then clustered to obtain different semantic clusters.

The approach addresses the main issues of proposing service embedding for informative cyclic framework construction in service composition and suggesting the use of neural language models for service embedding. To deal with the complexity of existing models, a lightweight deep neural language model is developed, which has similar performance to the base model but is faster. Comprehensive experiments with a real-world dataset show that the approach effectively performs clustering.

3.1 Related Work

This study presents the first fully deep learning-based approach to service clustering. Related works are categorized based on different aspects.

3.1.1 Web Service Discover and Clustering

Traditional web service clustering methods use features extracted from WSDL documents to compute similarities between services. For example, Elgazzar et al. [20] used WSDL documents to capture features and compute similarity, while [21] used both WSDL documents and tags. Kumara et al. [22] utilized ontology learning to calculate similarity, while [23] presented a word embedding augmented LDA model. Zou et al. integrated service composability into deep semantic features for clustering. In contrast, we propose to use neural language models to represent services as representation vectors and perform clustering based on these vectors.

WSDL documents are difficult for machine algorithms to understand from a semantic perspective, so semantic web service discovery has been proposed. Ontology is a promising approach to enriching web services with machine-processable semantics. Martin et al. [24] used the Web Ontology Language for web services, while [25] explored ontology for service discovery. Instead of extracting semantic knowledge from WSDL documents or constructing ontology based on them, we attempt to reveal semantic information from service composition sequences, as the invocation relationship between services contains semantic information.

3.1.2 Social Relationship for Web Service Discovery

Social relationship-based service discovery is a promising approach that connects related services based on functionality, quality of service, or sociability. Maamar et al. [17] developed social networks for service discovery, while [18] presented the Global Social Service Network. Cor-bellini et al. mined social web service repositories for social relationships to aid discovery. In contrast, we adopt neural networks to learn service composition sequences and extract the invocation relationship for clustering services.

3.1.3 Deep Learning for Application Programming Interface (API) Learning

In order to ease the workload of developers, deep learning techniques have been applied to API learning. Gu et al. [26] utilized a neural LM to project natural language queries into API usage sequences, while [27] proposed a novel neural synthesis algorithm for learning programs with APIs. Wu et al. [28] suggested an approach for automatically binding answers for natural language questions related to APIs from tutorials and stack overflow. These studies demonstrate the capability of neural language networks to understand both natural language and API usage sequences.

Inspired by these works, we aim to leverage neural LMs to learn service composition sequences and extract important information for service clustering. Firstly, this section reviews various traditional service clustering approaches that typically rely on WSDL documents. Then, existing studies that perform service discovery based on social relationships between services are reviewed, although they do not consider the invocation relationship. Finally, Fig. 8 illustrates some cases of deep learning models used for API learning, which indicate their ability to comprehend API invocation sequences. These studies motivate us to propose a new approach for service clustering using service embedding with invocation sequences.

3.2 Service Embedding

In this section, we introduce the concept of service embedding in web service composition. Web service discovery aims to provide suitable services for consumers, but when a single service cannot meet the complex requirements of consumers, the discovery task changes to service composition by combining several services to provide value-added services. As shown in Fig. 8, the web service composition framework consists of three main components: Service Matcher, Composition Generator, and Evaluation Engine. When Composition Generator receives service requests from customers, it processes the requests and obtains relevant services from Service Matcher to create candidate service compositions. These compositions are then sent to Evaluation Engine for testing, and the final tested service composition provides value-added services that can satisfy the complex functionality required by consumers.

Composition Generator generates service compositions based on rules or knowledge, and these service sequences contain the invocation relationship. Determining precise information or knowledge can be helpful in service clustering, which can be performed based on the invocation relationship. To this end, we propose service embedding in the framework to learn service composition sequences using appropriate models. The sequences can then be projected into representative vectors by the pretrained models, and related services can be determined by computing these representative vectors. The significance of service embedding can be summarized as follows: the representative vectors generated by the pretrained model can be used

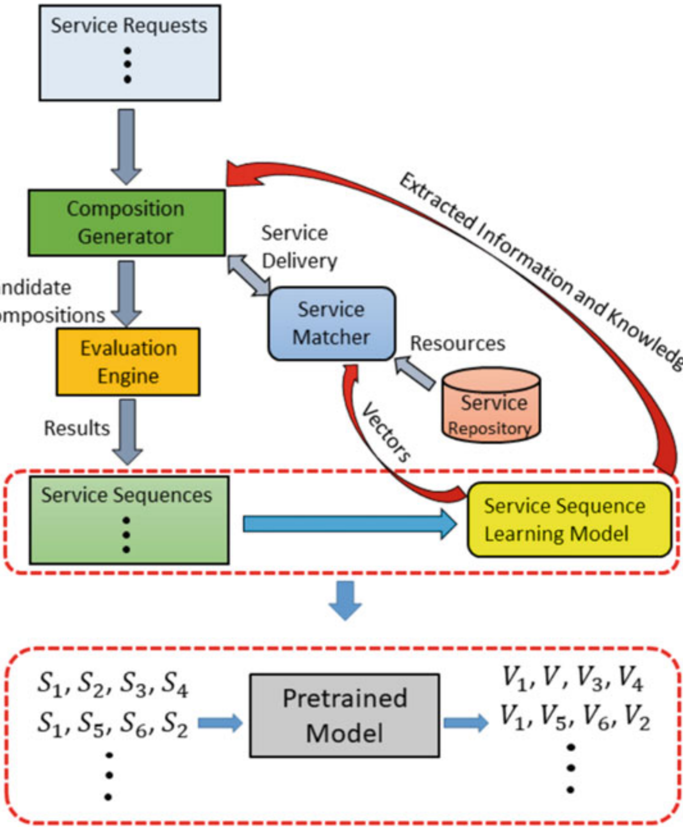


Fig. 8 Service embedding in web service composition

to find relevant services, the extracted information and knowledge can contribute to the service composition procedure, and the model is independent and open in the cyclic framework because the input and output are service composition sequences and representative vectors, respectively, making it easy to share and exploit such data.

3.3 Service Embedding with Deep Neural Language Networks

The use of Transformer as a state-of-the-art model in neural machine translation has been well established. BERT, which is composed of stacked layers of Transformer's encoder, has been used for service embedding in this paper. However, the base model is heavy and still under development. Therefore, a lightweight BERT-based model has also been developed for service embedding. This section provides a detailed description of both models.

3.3.1 Transformer and BERT

In natural language processing (NLP), language models (LMs) are essential for tasks such as machine translation, question answering, and sentiment analysis. LMs are responsible for representing word sequences in a form understandable by machines and estimating the probability distribution of words, phrases, and sentences.

Recently, neural networks have been used to learn the probability of LMs, resulting in significant improvements. Transformer and its stacked layers, known as BERT, have demonstrated exceptional language sequence learning capabilities [29, 30]. As shown in Fig. 3, Transformer relies solely on a self-attention mechanism and is composed of the Encoder and Decoder. The main components of Transformer are Multi-head Attention, Feed Forward, and Add & Norm. Feed Forward consists of two linear transformations with a Rectified Linear Unit activation function in between. Add & Norm is a residual connection [31] and layer normalization. Multi-head Attention is a crucial part that implements a self-attention mechanism and is shown in Fig. 4. It comprises several attention layers running in parallel, with h representing the number of heads or parallel layers. The input vectors, query (Q), keys (K), and values (V), are transformed to Scaled Dot-Product Attention through linear projections. In a self-attention layer, all queries, keys, and values come from the same place. The Scaled Dot-Product Attention can be formulated as follows:

$$Attention(Q, K, V) = soft\max\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

In the Decoder component, the mask operation restricts the current position from observing anything beyond its prior positions. All attention weights are concatenated and then transformed via linear projection. BERT is composed of stacked transformer encoder layers, and its proposal splits the NLP process into two stages: upstream representation and downstream tasks, with BERT being employed in the former. To pretrain BERT, two unsupervised tasks are used: masked language modeling and next sentence prediction (NSP). The input is a concatenation of two masked sentences, with the first position being [CLS]. NSP necessitates the model to predict whether the second sentence is the following sentence of the first sentence, and the outcome position is probability. [SEP] is a unique separator token for separating two sentences, like questions and answers. Masked language models predict the masked token in the input sentences. Pretrained BERT can be utilized in a variety of downstream tasks, such as machine translation, Q&A systems, and others. The self-attention mechanism can learn an exceptional representation of input sequences through unsupervised learning. In our study, we propose using BERT to learn service composition sequences and capture invocation relationships via its self-attention mechanism. In our situation, some adjustments were made to the model. The NSP task and segment embedding are eliminated, as depicted in Fig. 9. The input is a single masked service sequence, and the embedding layer is comprised of two procedures: token embeddings and position embeddings. The model performs masked language modeling. In masked language modeling, 15% of the masked token positions are randomly chosen for prediction. Suppose the last API invocation sequence is "getText toLowerCase replace split" and the selected position is the last one. In that case, the input and label are as follows:

$$\text{Input}: \text{ getText toLowerCase replace [MASK]}$$
$$\text{Label}: \text{ [MASK]} = \text{split}$$

The model's prediction is the label "split." The masked token positions can be predicted by performing mask operation on all input service sequences, and the model can be trained on the samples to learn these service sequences. The model predicts the label "split" for the given input sequence, where the masked position is replaced with the [MASK] token 80% of the time, a random API method 10% of the time, or remains unchanged 10% of the time. The mask operation is performed on all input service sequences, and the labels of the masked positions are obtained. The model is

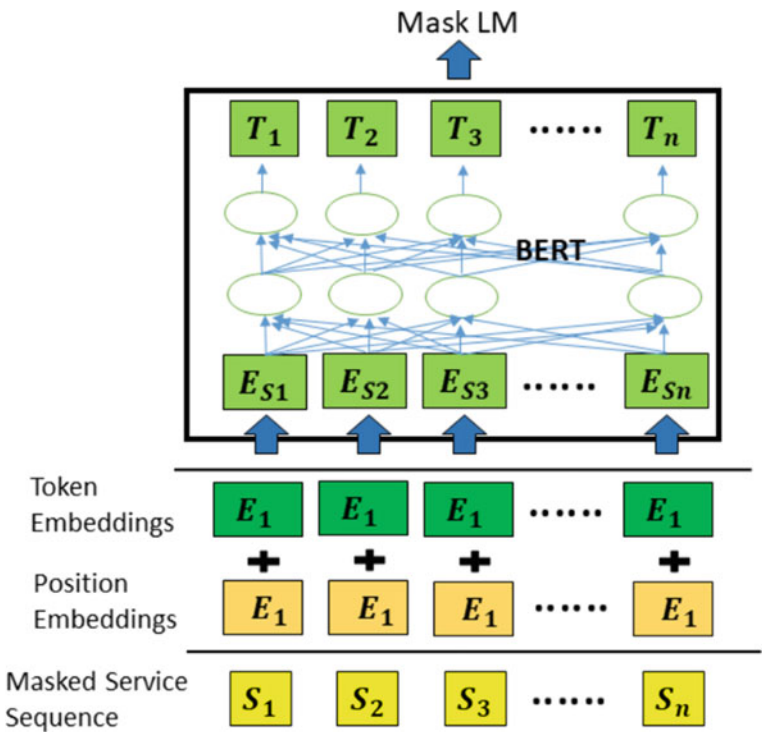


Fig. 9 BERT-based service embedding model

trained with these samples to learn the service sequences by predicting the masked positions.

3.4 Semantic Service Clustering Based on Service Embedding

Contextual knowledge plays a vital role in semantic segmentation in NLP. With the help of BERT, the semantics of words can be comprehended by considering the context, thereby resolving lexical ambiguity. Additionally, a single service can result in different service compositions, each having unique functions. By pretraining a BERT-based service embedding model with these composition sequences, the model can effectively capture the semantics of services and generate corresponding representation vectors. As a result, we can discover similar semantic services and retrieve matching semantic compositions. The entire process is depicted in Fig. 10. The semantic clustering of service composition can be divided into two stages:

The first stage involves service embedding, which entails pretraining a neural LM with service sequences to generate representation vectors through the model. In

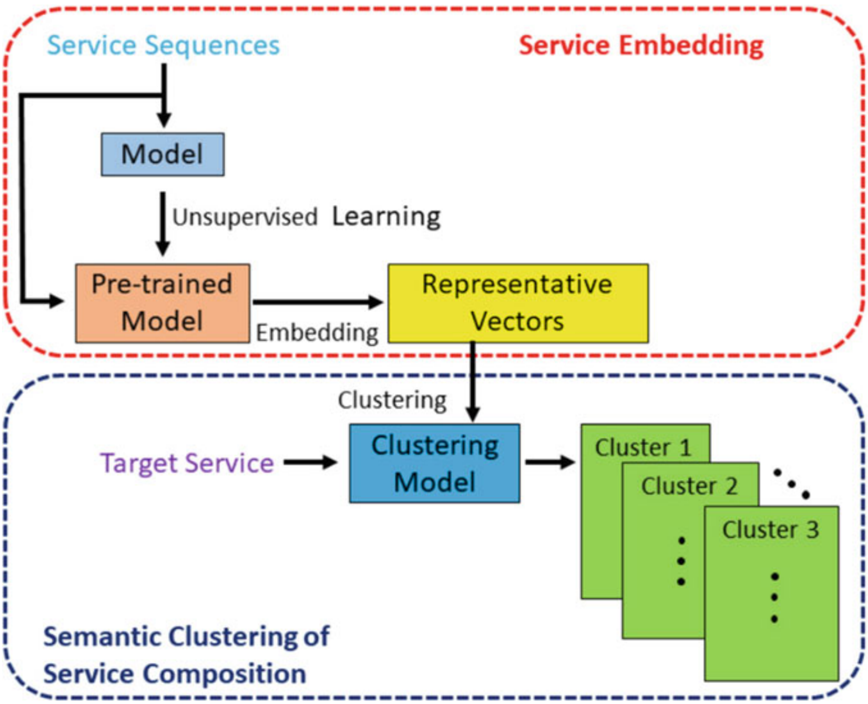


Fig. 10 Semantic clustering of service composition with service embedding

this study, we opted for BERT-based service embedding models as they leverage the self-attention mechanism to capture the invocation relationship between services. This knowledge encapsulates semantic information about the services and is already conveyed via the embedding process. The second stage revolves around clustering. Specifically, we utilized the unsupervised K-means clustering technique to cluster the representation vectors. Consequently, a semantic clustering model was developed, which can return various semantic clusters when a target service is entered into the model.

3.5 Data Preparation

For our experimental dataset, we opted for the invocation sequences of web APIs. We acquired Java source codes from GitHub, which were designed for implementing the Twitter APIs. Figure 11 illustrates the data preparation process. Initially, we parsed the source code into abstract syntax trees to identify the methods in each calling method or class. As our research focus was on Twitter APIs, we had to distinguish the relevant Twitter API methods and filter out irrelevant methods. Ultimately, we obtained Twitter API invocation sequences within a specific definition scope. During the experiments, we utilized approximately 3000 API invocation sequences as training data, with the number of methods totaling around 800. Compared to other NLP datasets, our dataset is relatively small. This is primarily due to two reasons: First, the model's complexity is low, with our models containing only 1:6 M and 2:5 M parameters, in contrast to base BERT in NLP, which has 110 M parameters. As a result, a large dataset is not required. Additionally, our model is not a full BERT model, as it doesn't learn sentence pairs, but rather just predicts the masked position to embed the sequences, simplifying the task. Second, our dataset type is different, as it consists of API invocation sequences that aren't as complex or creative as natural language. Moreover, nearly all Twitter APIs are already included.

3.6 Experiment and Discussion

This section addresses two key issues: service embedding using the lightweight BERT architecture and semantic service clustering. Regarding the former, we evaluate the computational complexity and reduction in model size. Furthermore, we present the experimental outcomes of service embedding. With respect to the latter, we delve into the discussion of semantic service clustering using lightweight BERT-based service embedding through invocation sequences, analyzing the clustering performance.

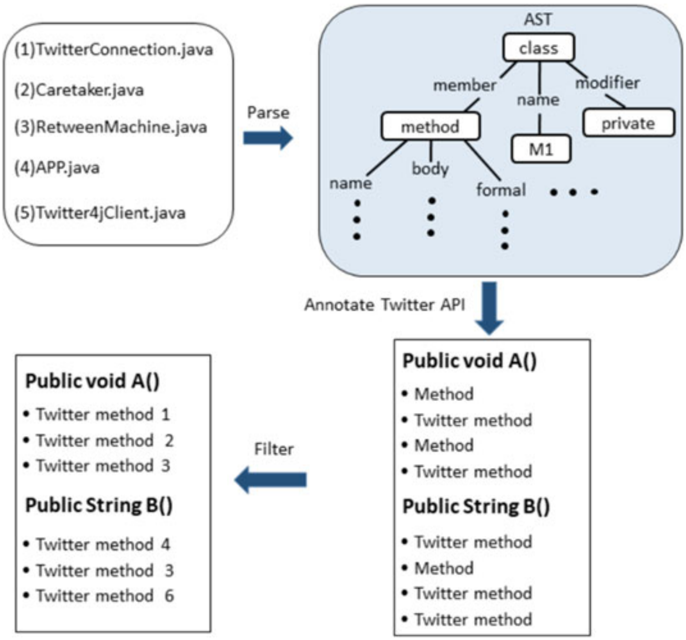


Fig. 11 Processing of the dataset

Table 1 Hyperparameters of models

Model	N	d_{model}	d_{ff}	h	Filter size
Base	3	384	768	6	–
Lightweight	3	384	768	6	3*9

3.6.1 Service Embedding with Lightweight BERT-Based Models

Calculation of Computational Complexity

The aim of this experiment is to compare the performance of the base BERT model with the proposed lightweight architecture. The hyperparameters for both models are set in Table 1, with a batch size of 12, maximum sequence length of 128, vocabulary size of 800, and other configurations following the original literature [17]. As discussed in Sect. 5, the computational complexity of both models can be calculated with the increase of the embedding dimension dm leading to a dramatic increase in time complexity and number of parameters for both models. However, the lightweight model shows a reduction in both time complexity and number of parameters when compared to the base model.

When dm is set to 384, the time complexity of the base model is approximately 322 M, with 2.5 M parameters. For the lightweight model, the time complexity is about 221 M, with 1.6 M parameters. This represents a reduction of 19–56% in time complexity and 22–46% in the number of parameters for the lightweight model when compared to the base model, making it theoretically faster and more lightweight. This is especially important in deep learning-based applications where response time is crucial, with inference time being the dominant factor. When performing the same inference task on edge computing, the inference time of the lightweight model can be reduced by 19–56% compared to the base model.

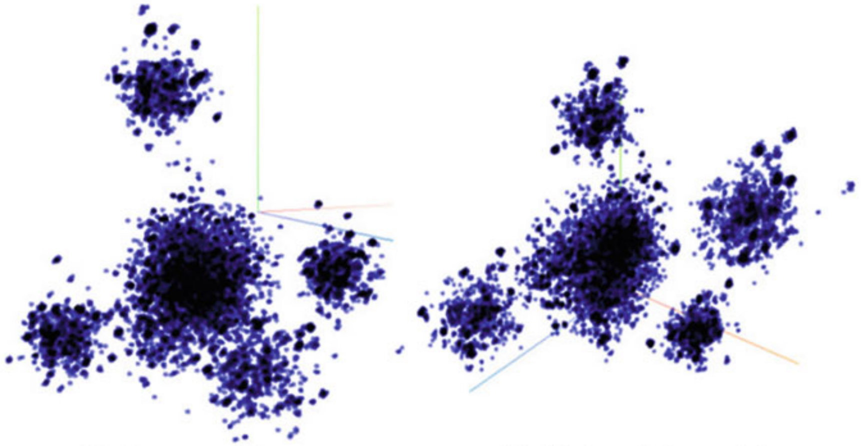
Both models were trained on a GTX 1080 Ti, with the base model taking about 10 h and the lightweight model taking about 6 h. The results show that the loss of the base model becomes stable at around 300 K steps, while the lightweight model completes training at about 150 k steps. This further confirms that the lightweight model can be trained faster than the base model, consistent with the previous comparison of computational complexity.

Visualizing Service Embedding

After pretraining, we obtain representation vectors of all sequences using the pretrained models. To visualize these vectors, we use principal component analysis for dimension reduction, and the results are presented in Fig. 12. The distribution of points is quite similar, and the points are divided into several large groups, but this does not have clear significance. However, there are several small clusters, indicating promising capability of the model in service embedding. To compare the results, we compute the nearest points of some target method. For instance, if we choose "(185)setMedia" as a target, "185" represents the number of sequences in the dataset, and "setMedia" is the name of the API method. By computing the cosine distance, we can determine the nearest points in space, as shown in Fig. 13. Although there are a few differences in the order, the points are the same. We use several target API methods to compare the difference between the nearest points, and the results are consistent. Thus, by measuring the visualization result, the performances of the two models are comparable.

3.6.2 Semantic Service Clustering

Our approach aims to achieve semantic service clustering, where consumers input a target service, and the clustering model returns different semantic clusters that contain the target service, as shown in Fig. 14. In this experiment, we use the K-means clustering algorithm to construct a clustering model. K-means clustering is an unsupervised learning algorithm widely used in clustering tasks. The number of clusters K needs to be determined in advance, so we use several values to compare the performance of clustering models in different k values. To evaluate cluster quality, we use purity and entropy, and we adjust entropy accordingly in our case. Detailed experimental results can be found in the literature [32].



(a) Base model (b) Lightweight model

Fig. 12 Visualization of the representation vectors

Nearest points in the original space:

(1032)setMedia	0.002
(2106)setMedia	0.002
(2657)setMedia	0.003
(2079)setMedia	0.003
(2697)setMedia	0.003
(1875)setMedia	0.003
(2937)setMedia	0.004
(2596)setMedia	0.004
(2060)setMedia	0.004

(a) Base model

Nearest points in the original space:

(2657)setMedia	0.005
(2106)setMedia	0.006
(1032)setMedia	0.006
(2697)setMedia	0.006
(2079)setMedia	0.008
(1875)setMedia	0.009
(2596)setMedia	0.010
(2937)setMedia	0.010
(2060)setMedia	0.010

(b) Lightweight model

Fig. 13 Nearest points of (185) setMedia

4 Intelligent Big Data Analysis with ASC for Virtual World

This section presents an example of an intelligent big data analysis architecture based on Automatic Service Composition (ASC) for virtual world applications. Big Data is generated by both human and machine sources, resulting in massive amounts of data pouring in from every direction. According to the [33], the amount of information created and replicated is nearly as many as the number of stars in the physical universe, indicating the exponential growth of digital data due to various factors such as volume, velocity, variety, value, veracity, etc. The importance and challenge of

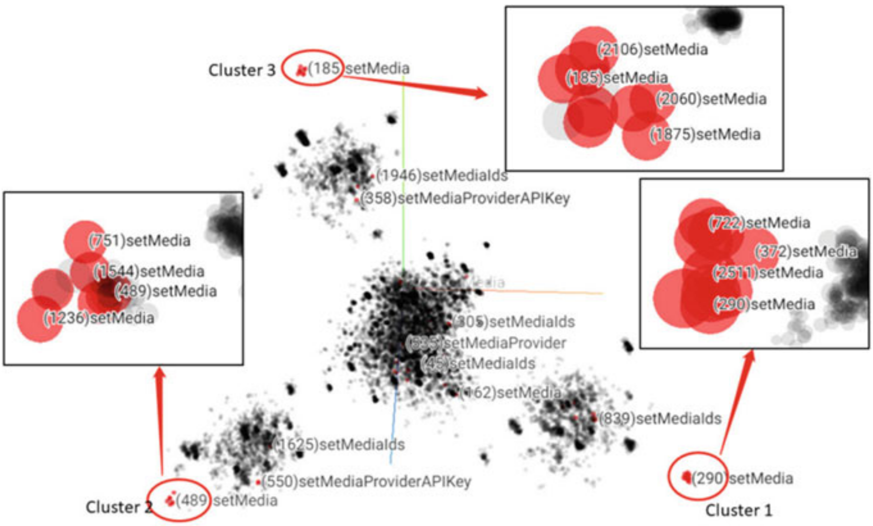


Fig. 14 Example of Semantic Clustering for "setMedia"

manipulating Big Data are increasing exponentially, making it difficult to synchronize all factors to achieve a final solid solution.

Currently, Big Data Analytics (BDA) is performed by manually accumulated tasks, which hinders faster decision-making on real-time applications for efficient data analytics. Moreover, the diversified analytical requirements and multidisciplinary datasets in BDA make the data mining process complex, requiring a comprehensive data mining methodology to efficiently fulfill the requirements. While CRoss Industry Standard Platform for Data Mining (CRISP-DM) is a useful standard for BDA, its manual process and rigorous steps make it time-consuming and not suitable for real-time applications.

To address these issues, we propose a novel architecture that automates the BDA process with CRISP-DM using Nested Automatic Service Computing (NASC) as a key technology to automate the multi-step process while maintaining scalability. The proposed architecture integrates intelligent and innovative technologies to create a scalable, intelligent, and real-time BDA solution.

4.1 Related Work

The existing literature on scalable intelligent architectures for BDA is limited. Geerdink [34] proposed a reference architecture for BDA and presented indicative evidence of its effectiveness. [35] provided an intelligent multi-agent solution for a specific domain. Zhong et al. [36] introduced a memory-centric real-time BDA solution, while [37] discussed a real-time BDA solution for monitoring health. Ayhan

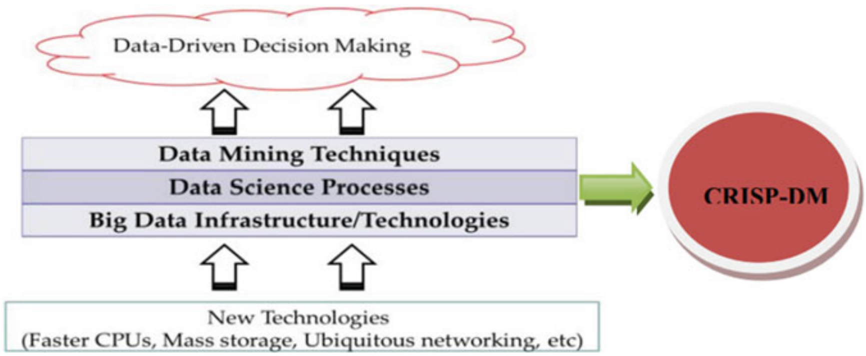


Fig. 15 Standard process of BDA

et al. [38] presented a predictive BDA solution for the aviation industry by considering various factors. Oracle [39] introduced a method for making informed predictions and gaining business insights from the constant flow of information within different business domains. Additionally, wu et al. [40] proposed the HACE theorem characterizing the features of the Big Data revolution and presented a Big Data processing model. However, most of these solutions are domain-specific and only some of them provide real-time support to the analytical process. In contrast, our proposed solution offers a domain-independent scalable approach to the BDA process.

4.2 Preliminaries for Big Data Analytics

4.2.1 Big Data Analytics Process

Big Data Analytics (BDA) involves gathering, structuring, and examining large data sets to uncover patterns and insights that can aid organizations in comprehending the data and identifying essential information for business decisions. The data will be processed through data science technology and mined using data mining techniques in a data warehouse. A data science process will be utilized for data manipulation, with the CRoss Industry Standard Platform for Data Mining (CRISP-DM) shown in Fig. 15 as our methodology of choice.

4.2.2 CRISP-DM Process

The process of BDA involves the collection, organization, and analysis of large datasets to uncover patterns and useful information. This helps organizations better understand the data and identify the most important information for future business decisions. The CRISP-DM model has six stages that effectively address data

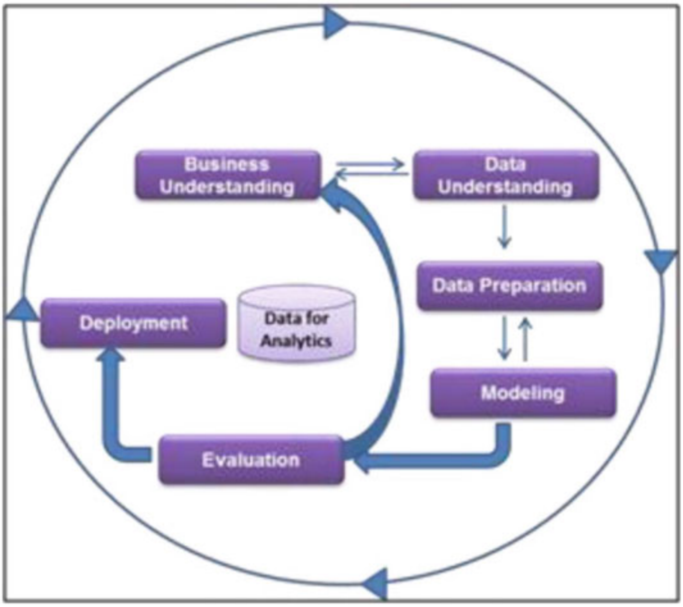


Fig. 16 CRISP-DM process

science requirements in the Big Data domain. Figure 16 provides a graphical view of the model. The Business Understanding stage focuses on understanding objectives and requirements from a domain perspective, and a preliminary plan is designed to achieve those objectives. The Data Understanding stage begins with the given dataset and continues with tasks until the first insights into the data are discovered. In the Preparation stage, the final purified and rectified dataset is prepared for the next stage. The Modeling stage applies various modeling techniques, usually data mining techniques, based on the requirements. In the Evaluation stage, a thorough insight into the model is gained using matured data, and a decision is made whether to use the mining process results. In the Deployment stage, the result is organized for customer readability and the project is deployed. Business understanding and data understanding have already been confirmed manually by this project, and we will now automate the remaining four stages using NASC technology.

4.2.3 Nested Automatic Service Composition

NASC, which is based on the service-oriented architectural design pattern, is used in this study to automate BDA. To achieve intelligent automation of the BDA process, we must first define service concepts for each step of the CRISP-DM process, and then logically match each step to a composition step. The development of an intelligent BDA process involves the following steps:

1. Development of service types and instances for BDA;
2. Definition of a workflow for BDA;
3. Development of a service discovery algorithm for BDA;
4. Development of a service selection algorithm for BDA; and
5. Development of a service algorithm for BDA results.

4.3 Architecture for Intelligent BDA

Our focus is to develop a comprehensive architectural solution that translates real-world problems into technical language. Due to the size and complexity of Big Data solutions and the need for quick time-to-market, new software engineering approaches are required to design software architectures [3]. One such approach is a software Reference Architecture (RA) that allows for the systematic reuse of knowledge and components when developing a concrete System Architecture (SA). As a result, we were able to easily generate an implementation-level UML class diagram.

4.3.1 Reference Architecture

RA is an architectural solution that provides a template solution for a complex problem domain. The RA for the BDA process is shown in Fig. 17 and provides a solid foundation for extracting the SA from it. SA is a conceptual model that defines the structure, behavior, and other views of a system. The RA is a layered solution that gives a high-level view of how each component and technology of the product behaves and how it maintains interactions between each of them. This layered pattern is closely connected to an architectural principle of "loose coupling." From the RA perspective, we have identified three main building block layers: the top-level layer is called the Analytical Layer, the middle layer is called the Technology Layer, and the bottom layer is called the Infrastructure Layer. Let us now summarize the identification of each layer.

- **Infrastructure layer**: It mainly includes the data warehouse and data mart layer, consisting of the Hadoop ecosystem for managing Big Data infrastructure, web service pools, and two relational database management systems (RDBMS) for data manipulation and maintaining analytical clusters. These components can exist on both Intranet and Internet platforms. For instance, a Hadoop cluster can be distributed geographically across data centers, requiring dealing with Hadoop beyond the intranet level. Web services can also be distributed over the internet and local networks. One of the RDBMSs is used to import data from Hadoop and

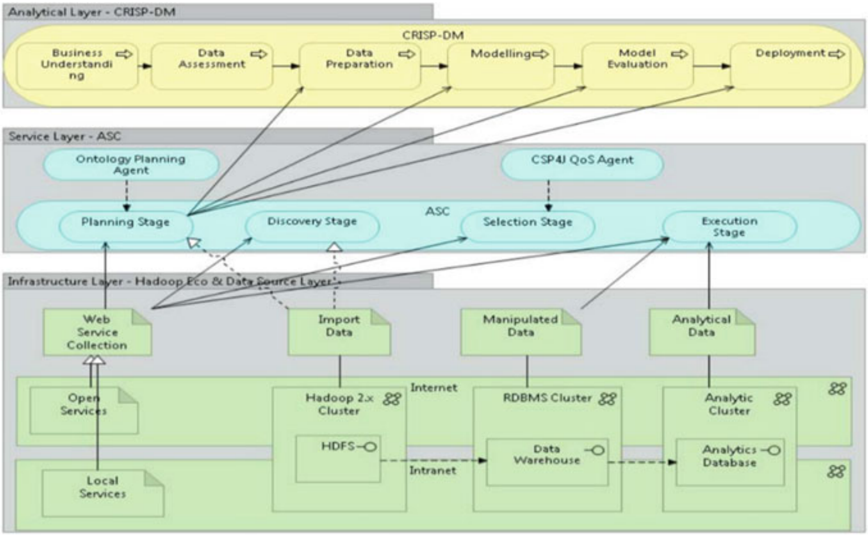


Fig. 17 Reference architecture of the BDS solution

facilitate data processing, while the other RDBMS is responsible for managing the analytical cluster and related activities of the analytical process.

- **Technology layer**: It is mainly dominated by NASC, which supports technologies such as the quality of service agent and intelligent planning agent to provide intelligent workflow automation. It identifies the requirements and utilizes respective resources distributed along the system to fulfill both functional and non-functional requirements of the project.
- **Analytical layer**: This layer is dominated by CRISP-DM, providing the data mining process of the project. The first two out of six stages of CRISP-DM have already been decided manually, so the NASC will only deal with the remaining four stages.

4.3.2 System Architecture

In scenario 1, the ABC Air Port Company needs to analyze flight delay data to identify factors causing the delay and make necessary decisions to reduce/avoid it. We used the RA to derive the SA and applied it to our scenario. The behavior of each layer during execution time and the resulting output from the NASC execution stage are clearly shown in Fig. 18. The SA shows the existing technologies and their responsibilities, as well as the communication between layers across the entire solution.

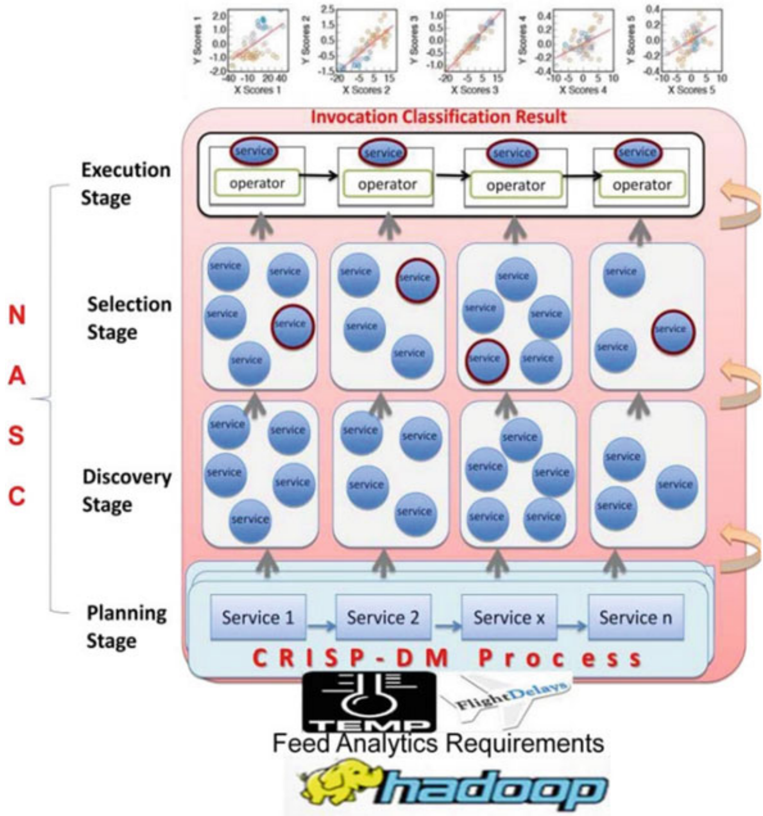


Fig. 18 NASC versus CRISP-DM and classification results

4.3.3 Top-Level UML Class Diagram

Using the RA, we successfully integrated the main technologies to achieve intelligent real-time analytics for BDA and derived the SA for our scenario. Finally, we designed a detailed top-level UML class diagram of a scalable BDA based on the RA and the SA. We identified two packages, one for ASC and the other for CRISP-DM-related services. Additionally, there are two utility packages providing services to the system: Planning Agent and Quality of Service (QoS) Agent. Figure 19 displays a high-level view of the UML class diagram.

The NASC Package is the base package of the solution, allowing for the identification of functional and non-functional requirements for analytics (see Fig. 20). The CRISP-DM Package is responsible for dealing with web services related to the complex, dynamic, and diversified tasks of the BDA process that are requested by the NASC. Note that we manually accomplished the first two stages of CRISP-DM according to the scenario, and the NASC will automate the remaining four stages.

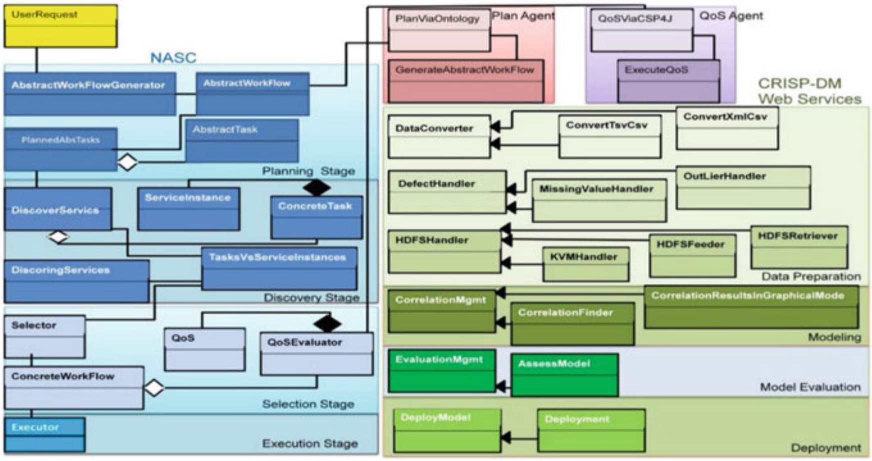


Fig. 19 Top-level UML class diagram of Intelligent BDA process

Fig. 20 NASC stage definitions

R: Set of user's requests at the service level.
$W = \{t_1, t_2, t_3, \ldots t_l\}$: Set of l abstract tasks in an abstract workflow W.

Planning: $\prod$: $R \to W$.
$I_i = \{i_{i1}, i_{i2}, i_{i3}, \ldots, i_{im}\}$: Set of m service instances advertised in a service registry for an abstract task t_i. I is the set of I_i, for $1 \le i \le l$. If each task in a workflow has m instances, then the total number of service instances available for the workflow is $l \times m$.

Discovery: $\Delta : W \to I$
$C_j = \{c_{j1}, c_{j2}, c_{j3}, \ldots, c_{jp}\}$: Set of P selected service instances to be executed from the service instance set I_j. C is the set of C_j where $1 \le j \le l$

Selection: $\Sigma : I \to C$.
$X = \{x_1, x_2, x_3, \ldots, x_q\}$: Set of q executed service traces.

Execution: $E : C \to X$

The two utility packages are Planning Agent and QoS Agent. The Planning Agent can be selected by the developer to fulfill the planning requirement, such as HTN. We chose the Planning Agent by Ontology reasoning for the planning process. The QoS Agent uses Constraint Satisfaction Problem-Solving Agent.

4.4 Evaluation

The following are the main advantages observed and studied: NASC technology can create a scalable solution for real-time analytics. The ASC approach is useful for automating the CRISP-DM process, as it separates workflow management from functional modules, which is expected to be a more technically effective solution than the conventional manual path. The technologies used in RA can be customized based on user preferences, and the solution is a layered architecture, making it loosely coupled and interoperable from an architectural perspective.

We have successfully designed an intelligent BDA using RA, derived the SA based on that RA, and simulated the SA with our scenario. We have also designed a UML class diagram for the software development process of BDA. As we believe that this scalable architectural solution will work effectively for our scenario, we are confident in the success of this approach [41].

5 A Scenario of Smart City Simulation on Virtual Space and Conclusion

5.1 Motivation Scenario

5.1.1 For Batch Processing

An institute in Japan, specializing in advanced industries, has launched several projects in the renewable energy sector. One of these projects involves a weather-data analysis program in Fukushima Prefecture, aimed at identifying effective renewable energy sources. The institute's objective is to encourage the use of renewable energy, find the most efficient energy sources, and reduce dependency on existing nuclear power plants in northern Japan. The project includes collaborations with Fukushima University and the University of Aizu, with one of the contributions being in the areas of smart grid and energy IoT as shown in Fig. 20 [42].

5.1.2 For Real-Time Processing

The Japan Meteorological Department is currently researching the development of a real-time earthquake detection model to send life-saving alerts to relevant authorities for earthquakes that exceed a magnitude of 6. The proposed solution aims to improve accuracy and speed with the incorporation of machine learning technology for near real-time computation results. It should be capable of handling multiple sources of data and designed for easy use, enabling bulk transmission of alerts within 60 s of the foreshock. Additionally, the system should be able to collect real-time data, perform analytics, and send alerts to earthquake-prone areas within a limited time frame as shown in Fig. 21 [42].

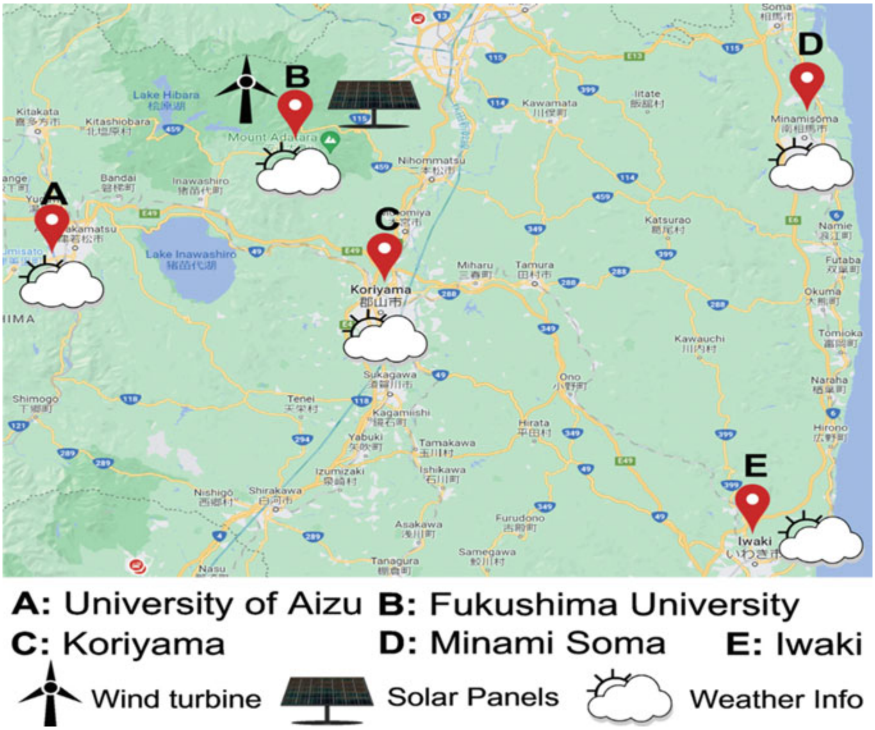


Fig. 21 Batch processing scenario for the sustainable power solution

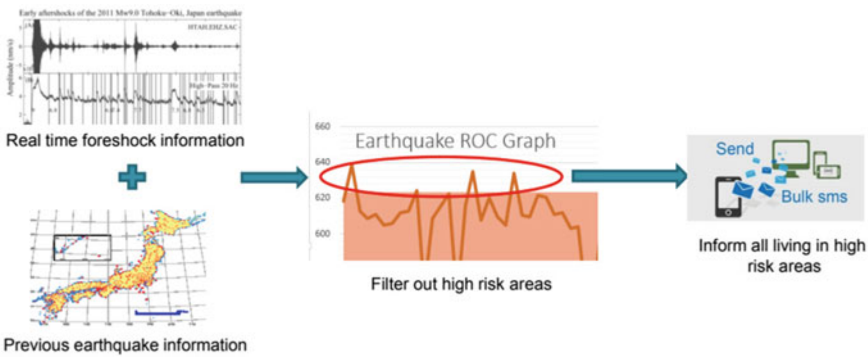


Fig. 22 Real-time processing scenario for the earthquake detection

5.1.3 CRISP-DM Process

This section introduces the CRISP-DM process and a motivating scenario for BDA, which aims to use large sets of data to discover patterns and other useful information to support decision making The CRISP-DM process involves various data mining techniques applied to data stored on specific infrastructure displaying the six phases of operating CRISP-DM over big data. The first phase is business understanding, which focuses on understanding project objectives and requirements from the business domain. In the data understanding phase, data scientists familiarize themselves with the data and identify quality problems. The data preparation step involves all activities that prepare raw data to yield the final dataset for the modeling tool. In the modeling step, various modeling techniques are applied to analyze the dataset, followed by an evaluation phase to ensure it meets business requirements. The section then describes how the CRISP-DM process is used in a batch processing scenario for BDA in the renewable energy field. The main objective is to find optimal renewable energy sources that can reduce or halt the use of nuclear power plants in northern Japan, specifically Fukushima Prefecture. Researchers created two profiles for renewable energy sources across Fukushima Prefecture: the first profile involves weather data collected from five locations, and the second profile involves energy data collected from one location. The weather profile contains six types of data, including irradiance, temperature, wind direction, wind speed, humidity, and pressure, while the power profile contains three types of data, including voltage, current, and temperature of the panel's photovoltaic surface, and wind turbine data. Researchers identified the core influential factors from the weather and power data generated for the two profiles and treated them as variables upon which cluster analysis was performed. The preprocessed dataset was fed into a big-data file system such as the HDFS, and sophisticated models were built using clustering and classification algorithms. Various analytic techniques were used to verify the reality of the resulting cluster or classification results. The model was then deployed by respective authorities to generate required reports to the Japanese government. Based on the results, the advance industry institution proposed the most sustainably optimal power solution to the power crisis in Fukushima Prefecture.

5.1.4 Application Examples to ASC for Smart City Simulation on Virtual Space

All the processes of CRISP-DM can be efficiently implemented by ASC to automate a diverse range of applications. The ASC handles the highly dynamic and constraint-oriented BDA problem domain in a sophisticated manner. The analysis workflow will be automated by 5 ASC stages: planning, discovery, selection, verification, and execution stages. This automation enables efficient analysis of any complex task. Scenarios related to sustainable power solutions or earthquake detection can be implemented as digital twins in a virtual space. Several traffic congestion simulations for smart city can be supported by the ASC concept too. Considering the technological stages of a digital twin, ranging from physical mirroring level to global data analysis or interoperable environment as autonomous agents [43], several scenarios can be considered. ASC is capable of efficiently handling various dynamic situations on multiple digital twins. If you are looking for detailed and up-to-date information on ASC and its automatic analysis capabilities, you can refer to the following literature: Siriweera [42] and Siriweera et al. [44].

5.2 Conclusion

Automatic service composition closely simulates human intelligence, and the current deep learning systems have gone beyond data learning with respect to high-level inferences. The currently available AI systems, such as ChatGPT, simply learn the probability of relationships between words. However, ASC can enable the construction of intelligent systems to create value-added complex services based on human intelligence.

In this chapter, some core techniques for ASC, such as general ASC architecture, heuristic method, and service discovery, were introduced and applied in ASC to analyze big data using AI. ASC can be applied to automatic data analytics and deep learning generation systems, which can be used in virtual environment systems, such as digital twins, or in creating value-added services to supplement existing AI services (e.g., ChatGPT services). Finally, I expect this ASC technique can be used effectively to compose digital twins or AI services in Society 5.0.

Acknowledgements I am grateful to Prof. Wuhui Chen, Dr. Akila Siriweera, and Dr. Kungan Zeng for initial research and collaboration.

References

1. Singh MP, Huhns MN (2005) Service-oriented computing: semantics, processes, agents. Wiley, West Sussex
2. Milanovic N, Malek M (2004) Current solutions for web service composition. IEEE Internet Comput 8(6):51–59
3. Agarwal V, Chafle G, Mittal S, Srivastava B (2008) Understanding approaches for web service composition and execution. In: Proceedings of the 1st Bangalore annual COMPUTE conference, Bangalore, India, pp 18–20
4. Paik I, Chen W, Huhns MN (2012) A scalable architecture for automatic service composition. IEEE Trans Serv Comput 7(1):82–95
5. Agarwal V et al (2005) A service creation environment based on end to end composition of web services. In: Proceedings of the 14th international conference on World Wide Web, Chiba, Japan, pp 128–137
6. Maximilien EM, Singh MP (2004) A framework and ontology for dynamic web services selection. IEEE Internet Comput 8(5):84–93

7. Lécué F (2009) Optimizing QoS-aware semantic web service composition. In: the semantic web–ISWC 2009: 8th international semantic web conference, vol 9, pp 375–391
8. Zeng L, Benatallah B, Ngu AH, Dumas M, Kalagnanam J, Chang H (2004) QoS-aware middleware for web services composition. IEEE Trans Software Eng 30(5):311–327
9. Sirin E, Parsia B, Wu D, Hendler J, Nau D (2004) HTN planning for web service composition using SHOP2. J Web Semant 1(4):377–396
10. Cabral L, Domingue J, Galizia S, Gugliotta A, Tanasescu V, Pedrinaci C, Norton B (2006) IRS-III: a broker for semantic web services based applications. The semantic Web-ISWC 2006: 5th international semantic web conference, ISWC 2006, Athens, GA, USA, 5–9 November 2006, proceedings 5. Springer, Heidelberg, pp 201–214
11. Yu T, Lin KJ (2005) Service selection algorithms for composing complex services with multiple QoS constraints. Service-oriented computing-ICSOC 2005: third international conference, Amsterdam, The Netherlands, 12–15 December 2005, Proceedings 3. Springer, Heidelberg, pp 130–143
12. Ardagna D, Pernici B (2007) Adaptive service composition in flexible processes. IEEE Trans Software Eng 33:369–384
13. Lécué F, Mehandjiev N (2010) Seeking quality of web service composition in a semantic dimension. IEEE Trans Knowl Data Eng 23(6):942–959
14. Chen W, Paik I (2013) Improving efficiency of service discovery using linked data-based service publication. Inf Syst Front 15:613–25
15. Tan W, Zhang J, Madduri R, Foster I, Roure DD, Goble C (2011) ServiceMap: providing map and GPS assistance to service composition in bioinformatics. In: 2011 IEEE international conference on services computing, pp 632–639
16. Zhang J, Tan W, Alexander J, Foster I, Madduri R (2011) Recommend-as-you-go: a novel approach supporting services-oriented scientific workflow reuse. In: 2011 IEEE international conference on services computing, pp 48–55
17. Maamar Z, Hacid H, Huhns MN (2011) Why web services need social networks. IEEE Internet Comput 15(2):90–94
18. Chen W, Paik I (2014) Toward better quality of service composition based on a global social service network. IEEE Trans Parallel Distrib Syst 26(5):1466–1476
19. Xia B, Fan Y, Tan W, Huang K, Zhang J, Wu C (2015) Category-aware API clustering and distributed recommendation for automatic mashup creation. IEEE Trans Services Comput 8(5):674–687
20. Elgazzar K, Hassan AE, Martin P (2010) Clustering WSDL documents to bootstrap the discovery of web services. In: 2010 IEEE international conference on web services, pp 147–154
21. Wu J, Chen L, Zheng Z, Lyu MR, Wu Z (2014) Clustering web services to facilitate service discovery. Knowl Inf Syst 38:207–229
22. Kumara BT, Paik I, Chen W, Ryu KH (2014) Web service clustering using a hybrid term-similarity measure with ontology learning. Int J Web Services Res 11(2):24–45
23. Shi M, Liu J, Zhou D, Tang M, Cao B WE-LDA: a word embeddings augmented LDA model for web services clustering. In: 2017 IEEE international conference on web services (ICWS), pp 9–16
24. Martin D et al (2004) Bringing semantics to Web services: The OWL-S approach. Semantic web services and web process composition: first international workshop, SWSWPC 2004, San Diego, CA, USA, 6 July 2004, Revised Selected Papers 1. Springer, Heidelberg, pp 26–42
25. Fariss M, El Allali N, Asaidi H, Bellouki M (2019) Review of ontology based approaches for web service discovery. In: Smart data and computational intelligence: proceedings of the international conference on advanced information technology, services and systems (AIT2S-18), 17–18 October 2018, Mohammedia 3, Springer International Publishing, pp 78–87
26. Gu X, Zhang H, Zhang D, Kim S (2016) Deep API learning. In: Proceedings of the 2016 24th ACM SIGSOFT international symposium on foundations of software engineering, pp 631–642
27. Bhupatiraju S, Singh R, Mohamed AR, Kohli P (2017) Deep API programmer: learning to program with APIs. 1704.04327

28. Wu D et al (2018) Poster: automatically answering API-related questions. In: Proceedings of the 40th international conference on software engineering (Companion Proceedings), pp 270–271
29. Devlin J, Chang M-W, Lee K, Toutanova K (2018) BERT: pre-training of deep bidirectional transformers for language understanding. 1810.04805
30. Vaswani A et al (2017) Attention is all you need. In: Proceedings of advances in neural information processing systems, pp 5998–6008
31. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 770–778
32. Zeng K, Paik I (2021) Semantic service clustering with lightweight BERT-based service embedding using invocation sequences. IEEE Access 9:54298–54309
33. IDC Digital Universe Study (2011) Sponsored by EMC, June 2011. http://chucksblog.emc.com/chucks_blog/2011/06/2011-idc-digital-universe-study-big-data-is-here-now-what.html
34. Geerdink B (2015) A reference architecture for big data solutions-introducing a model to perform predictive analytics using big data technology. Int J Big Data Intell 2(4):236–249
35. Markic I, Stula M, Maras J (2014) Intelligent multi agent systems for decision support in insurance industry. In: 2014 37th international convention on information and communication technology, electronics and microelectronics (MIPRO), pp 1118–1123
36. Zhong T, Doshi KA, Tang X, Lou T, Lu Z, Li H (2013) On mixing high-speed updates and in-memory queries: a big-data architecture for real-time analytics. In: 2013 IEEE international conference on big data, pp 102–109
37. Moore P, Xhafa F, Barolli L, Thomas A (2013) Monitoring and detection of agitation in dementia: towards real-time and big-data solutions. In: 2013 eighth international conference on P2P, parallel, grid, cloud and internet computing, pp 128–135
38. Ayhan S et al (2013) Predictive analytics with aviation big data. In: 2013 integrated communications, navigation and surveillance conference (ICNS), pp 1–13
39. Oracle (2015) http://www.oracle.com/technetwork/database/options/advanced-analytics/bigdataanalyticswpoaa1930891.pdf. Accessed 13 Mar 2015
40. Wu X, Zhu X, Wu GQ, Ding W (2013) Data mining with big data. IEEE Trans Knowl Data Eng 26(1):97–107
41. Akila TH, Siriweera TA, Paik I, Kumara BT, Koswatta CK (2015) Architecture for intelligent big data analysis based on automatic service composition. Int J Big Data 2:1–14
42. Siriweera A (2019) Architecture for intelligent big data analysis based on automatic service composition. In Ph.D Dissertation in Graduate Department of Computer and Information Systems, The University of Aizu
43. Saracco R, Henz P (2022) Digital twins-part2. In IEEE internet computing, pp 5–6
44. Siriweera A, Paik I, and Huang H (2023) Constraint-driven complexity-aware data science workflow for AutoBDA. In IEEE transactions on big data, accepted, pp 1–20
45. Bizer C, Heath T, Berners-Lee T (2009) Linked data: the story so far. J Semantic Web Inf 5(3):1–22
46. Brogi A, Corfini S, Popescu R (2008) Semantics-based composition-oriented discovery of web services. ACM Trans Internet Technol 8(4):1–39
47. Han J, Park S, Kim J (2020) Dynamic OverCloud: realizing microservices-based IoT-cloud service composition over multiple clouds. Electronics 9(6):969
48. Hayyolalam V, Kazem AA (2018) A systematic literature review on QoS-aware service composition and selection in cloud environment. J Netw Comput Appl 110:52–74
49. Oh S, Lee D, Kumara SRT (2008) Effective web service composition in diverse and large-scale service networks. IEEE Trans Serv Comput 1(1):15–32
50. Paik I, Maruyama D (2007) Automatic web services composition using combining HTN and CSP. In: 7th IEEE international conference on computer and information technology (CIT 2007), pp 206–211
51. Sirin E, Parsia B, Hendler JA (2005) Template-based composition of semantic web services. In: AAAI Fall symposium: agents and the semantic web, pp 85–92

52. Tripathy AK, Patra MR, Khan MA, Fatima H, Swain P (2014) Dynamic web service composition with QoS clustering. In: 2014 IEEE international conference on web services, pp 678–679
53. Wang X, Han Y, Leung VCM, Niyato D, Yan X, Chen X (2020) Convergence of edge computing and deep learning: a comprehensive survey. IEEE Commun Surveys Tuts 22(2):869–904

Privacy-Preserving Data Collection and Analysis for Smart Cities

Yuichi Sei

Abstract Smart cities leverage real-world data to digitally replicate city-related aspects such as disaster prevention, transportation, and pandemics, creating an encompassing virtual environment. The construction of this realistic virtual world necessitates the collection of individual behavioral data through Internet of Things (IoT) environments. However, the challenge lies in ensuring the privacy of individuals during this data collection process. While numerous studies exist on privacy-preserving data mining, most target clean, complete, and independent personal data. This overlooks the reality of real-world personal data, which often contains noise, missing values, and evidence of interpersonal interactions. To build a human-centric smart city, it is crucial to consider these imperfect and interactive data while preserving privacy. In this paper, we propose a novel framework for privacy-preserving data collection and analysis in smart cities. This framework acknowledges the inherent sensing errors and interpersonal interactions, ensuring a more accurate representation of real-world conditions while maintaining stringent privacy safeguards.

1 Introduction

Comprehensive efforts have been made to establish an intelligent urban environment by developing sophisticated digital twins. Digital twins encapsulate the diverse functionalities of urban landscapes and accurately represent the behavioral patterns of their inhabitants. By creating a virtual counterpart to a city from data gathered in the real world, a myriad of urban-related events and processes, including disaster mitigation, transportation management, and pandemic response, are simulated within the digital realm. Subsequently, the insights gleaned from these simulations can be fed back into the physical world to facilitate well-informed decision-making and foster more sustainable urban development.

Y. Sei (✉)
The University of Electro-Communications, 1-5-1, Chofugaoka, Chofu 1828585, Japan
e-mail: seiuny@uec.ac.jp

Y. Murakami and K. Kimura (eds.), *Human-Centered Services Computing for Smart Cities*, https://doi.org/10.1007/978-981-97-0779-9_5

However, the construction of intricate digital twins and the realization of a truly smart city require the collection of detailed information about the behaviors and characteristics of individuals within the physical world. For instance, to model the specific attributes that people possess and the actions they undertake, a variety of personal attributes, such as age, gender, occupation, and income levels, must be gathered and analyzed [47]. Consequently, the protection of privacy has emerged as a critical concern that must be addressed in order to implement digital twin technology and achieve the vision of a smart city [28].

During the collection of information pertaining to an individual's attributes and behaviors in relation to their environment, it is crucial to ensure that the privacy of each person is adequately protected. Additionally, as humans are inherently social beings who interact with one another, the development of a human-centric digital twin requires careful consideration of the interactions between individuals and the associated information about each person [61]. Moreover, accounting for the potential measurement noise and missing values that may arise from sensing errors is essential when addressing individual privacy [55, 60].

Regrettably, existing privacy-preserving data mining solutions have neglected to consider the impact of measurement noise and missing values, which has led to low accuracy in data analysis. Furthermore, the lack of consideration for human interactions has resulted in increased privacy leakage beyond anticipated levels. This chapter aims to address three primary concerns: the loss of accuracy due to missing data, the loss of accuracy caused by observation noise, and the heightened privacy leakage that results from human interaction. These challenges are particularly pronounced in the context of a smart city environment. The content of this chapter is grounded in the author's previous publications [52, 54, 55, 60]. There are several other issues concerning LDP for smart cities. My previous articles have addressed these issues [56, 57, 59].

In this chapter, local differential privacy (LDP) [12] serves as the principal metric for evaluating privacy. LDP is a highly significant privacy-preserving technique that has been widely adopted to protect user data while enabling meaningful analysis. As a variant of differential privacy [11], LDP offers robust privacy guarantees for individual data points by introducing randomness directly at the data source, prior to any data being shared with an aggregator or analyst. Several prominent examples of LDP in action can be found in industry applications. For instance, Apple leverages LDP in its data collection processes to ensure user information remains private and secure [3]. Similarly, Google employs LDP in its RAPPOR (Randomized Aggregatable Privacy-Preserving Ordinal Response) project, which collects anonymized statistics from user browsers while preserving privacy [13]. The definition of LDP will be detailed in Sect. 2.3.

1.1 Purpose of This Research

The goal is to safely obtain people's attributes and behaviors under the LDP and analyze them with high statistical accuracy to realize smart cities. Detailed personal information is not needed for advanced smart cities; global information is sufficient. Laws that protect privacy have been enacted for countries, such as Japan's Personal Information Protection Law and Europe's General Data Protection Regulation (GDPR). Therefore, it is necessary not to violate people's privacy. On the other hand, it is impossible to achieve privacy protection that is 100% safe. The LDP can control the amount of privacy leakage by adjusting the value of ϵ, which represents privacy loss. This ϵ value can be specified by system administrators or by individuals. Within this range, personal data is collected from people and statistically analyzed. Research on LDP has been actively conducted in the past decade, but as mentioned in Sect. 1, there have been several challenges. The main goal of this chapter is to solve these challenges as follows.

- Treating measurement noise under LDP
- Treating missing values under LDP
- Treating human-to-human interactions under LDP.

Whereas the first two challenges have an effect on the accuracy of statistical analysis, the third is related to privacy leakage.

1.2 Structure of This Chapter

Section 2 commences with a presentation of motivational examples that emphasize the necessity of personal information for realizing advanced smart cities while concurrently underscoring the importance of privacy protection. This section also acknowledges that personal information is frequently gathered from sensors integrated into IoT systems and smartphones, and that this may result in inaccurate or missing data. Lastly, this section introduces the privacy protection metric utilized throughout the paper, which is LDP.

Section 3 explores the treatment of observational error in an LDP context. Although privacy-preserving data mining has been investigated extensively over the past decade, limited attention has been devoted to error in data values. LDP can be achieved by adding privacy noise to a target value that should be protected. However, if the target value already contains measurement error, the amount of privacy noise to add can be reduced. This section proposes a novel privacy model called true-value-based differential privacy (TDP). This model applies traditional differential privacy to the "true value", which is not known by the data owner or anonymizer, but not to the "measured value" that contains error. By leveraging TDP, our solution reduces the amount of noise to be added by LDP techniques by approximately 20%.

Consequently, the error of generated histograms is reduced by 40.4 and 29.6% on average.

Section 4 discusses the processing of missing values in an LDP setting. Privacy-preserving data mining techniques are valuable for analyzing diverse types of information, such as COVID-19-related patient data. Nonetheless, collecting substantial amounts of sensitive personal information poses a challenge. Moreover, this information may contain missing values, and this fact is not considered in existing methods that ensure data privacy while collecting personal information. Neglecting missing values diminishes the accuracy of data analysis. In this paper, we propose a method for privacy-preserving data collection that accounts for various types of missing values. Patient data are anonymized and transmitted to a data collection server. The data collection server generates a generative model and a contingency table suitable for multi-attribute analysis based on expectation-maximization and Gaussian copula methods. We conduct experiments on synthetic and real data, including COVID-19-related data. The results are 50–80% more accurate than those of existing methods that do not consider missing values.

Section 5 examines the management of human interactions in an LDP environment. Under LDP, a privacy budget is allocated to each user. Each time a user's data are collected, some of the user's privacy budget is consumed, and their privacy is protected by ensuring that the remaining privacy budget is greater than or equal to zero. Organizations and previous studies assume that an individual's data are entirely unrelated to another individuals' data. However, this assumption is invalid in situations where data for an interaction between two or more users are collected from those users. In such cases, each user's privacy is inadequately protected because their privacy budget is, in fact, overspent. In this study, we clarify the problem of LDP for person-to-person interactions. We propose a mechanism that satisfies LDP in a person-to-person interaction scenario. Mathematical analysis and experimental results demonstrate that the proposed mechanism maintains higher data utility while ensuring LDP than do existing methods.

2 Background

2.1 Motivating Examples

At present, IoT devices are capable of collecting and estimating various kinds of attribute information about individuals, including location, heart rate, health status, age, and movement patterns [70]. By leveraging this attribute data, individuals can access a wide range of services, such as recommender systems for smart cities. Additionally, the data collector can function as a data anonymizer, anonymizing the acquired data and transmitting it to the data receiver (refer to Fig. 1).

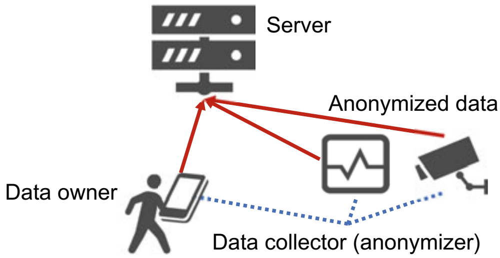


Fig. 1 Data receiver collects user data from people and/or sensing platforms under LDP

Two types of attribute data are considered in this context: the first comprises numerical attributes, such as heart rate measured in beats per minute, whereas the second encompasses categorical attributes, such as disease names (e.g., COVID-19).

The gathered attribute data often contain sensing errors, as accurately sensing and estimating the attributes of individuals can be challenging. In the most unfavorable circumstances, attribute data may not be collectable at all. Missing data can be approximated using techniques like multiple imputation or predictions based on regression models [81]. However, these estimated values tend to exhibit a significant degree of error.

In recent years, generative AI technologies such as ChatGPT and Stable Diffusion have undergone rapid advancements. While the majority of training data for these models are publicly sourced from the web, it is anticipated that future generative AI models will increasingly engage in the direct collection and training of data from individuals. The methods proposed in this paper are particularly well-suited for these emerging scenarios.

2.2 Attack Model

We assume an honest-but-curious adversary. That is, the adversary follows the protocol and rules of the system but attempts to learn as much as possible about individual users from the available data. They do not actively manipulate or tamper with the data but try to exploit the information they can access within the system's constraints.

Furthermore, each anonymized datum may contain original sensing error or intentionally added noise; therefore, the attacker cannot accurately estimate people's true data but can estimate the probability distribution of the data.

2.3 Local Differential Privacy (LDP)

In technical terms, LDP is defined as ϵ-LDP, where parameter ϵ represents a privacy budget. There are several relaxation concepts related to ϵ-LDP, such as (ϵ, δ)-LDP and Renyi differential privacy [38]. Although the concepts discussed in this chapter can be applied to the relaxations of LDP, we focus on ϵ-LDP to simplify the discussion. ϵ-LDP is defined as follows.

Definition 1 (ϵ-LDP) Let X represent the domain of a user's data, and let Y be an arbitrary set. A randomized mechanism M provides ϵ-LDP if and only if for any $x, x' \in X$ and any $y \in Y$,

$$P(M(x) = y) \leq e^{\epsilon} P(M(x') = y). \tag{1}$$

Several techniques have been proposed for achieving LDP. One of the most commonly used techniques is the Laplace mechanism [11]. To introduce the Laplace mechanism, we first define the concept of global sensitivity.

Definition 2 (Global sensitivity) For a function $f : X \to Y$, the global sensitivity of f is defined as follows.

$$\Delta f = \max_{x,x' \in X} |f(x) - f(x')|. \tag{2}$$

Theorem 1 (Laplace mechanism[11]) *Let Δf be the global sensitivity of a function $f : X \to Y$ and let $\mathcal{L}(v)$ represent the Laplace distribution, with a mean of zero and the scale parameter as v. The following mechanism M ensures ϵ-LDP.*

$$M(x) = f(x) + \mathcal{L}(\frac{\Delta f}{\epsilon}). \tag{3}$$

In the context of LDP, the magnitude of privacy safeguarding is modulated by the parameter ϵ. Deliberation on the appropriate selection of this value is beyond the purview of the present discourse; however, strategies such as the automatic determination predicated on the uniqueness of each attribute value [39] may be employed.

3 Measurement Noise Under LDP

3.1 Introduction

To realize smart cities, the collection and analysis of personal data through devices such as IoT is indispensable. However, it is crucial to consider the noise present in data measured from IoT devices. Additionally, the protection of privacy is imperative. In this section, we propose methodologies suitable for this scenario. In this section, an original value without error is referred to as a "true" value; the data owner or anonymizer may not have knowledge of these values. Conversely, sensed values that may contain errors are denoted as "measured" values. Existing studies on differential privacy do not consider true values, only measured values. Our study aims to investigate whether additional noise should be introduced to protect privacy if the

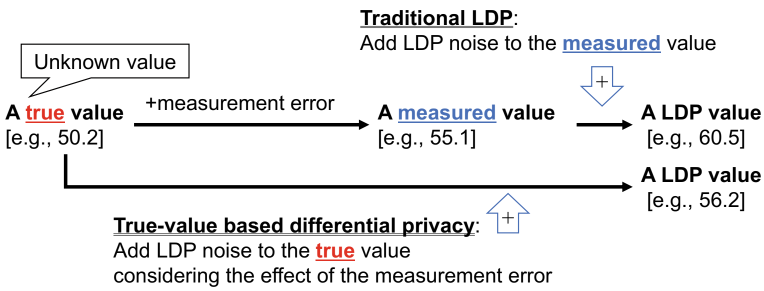


Fig. 2 Concept of true-value-based differential privacy (TDP). Traditional differential privacy adds LDP noise to the measured value. In contrast, TDP adds LDP noise to the true value after considering the measurement error

Table 1 Relationship between the error distribution knowledge and the TDP

The correctness of the knowledge	Whether we can achieve TDP
Correct	✓
Not correct (Underestimation)	✓
Not correct (Overestimation)	×

target value already contains error. This research proposes a new privacy model that safeguards the true value rather than the measured value. Because the data owner may not be aware of the true value, it is assumed that the true data exhibit a specific probability distribution, such as the normal distribution. This probability distribution is based on the data owner's or anonymizer's knowledge or the theory of errors [67]. The distinction between the traditional approach to differential privacy and the proposed true-value-based differential privacy (TDP) is illustrated in Fig. 2. According to the concept of TDP, the amount of noise to add to the measured value can be reduced.

We assume that the anonymizer can estimate the distribution of measurement error to some degree. Therefore, TDP can be achieved even if the anonymizer's estimation is inaccurate, as long as they do not overestimate the magnitude of the sensing error. The relationship between the anonymizer's error distribution and the TDP is presented in Table 1. Consequently, if the anonymizer is uncertain about the error distribution, they can guarantee TDP by conservatively estimating the amount of error. If the amount of error is predicted to be zero, the outcome aligns with traditional differential privacy. Thus, TDP can decrease the amount of error introduced relative to traditional differential privacy while still achieving the desired privacy protection level specified by ϵ.

If we possess no information about the error distribution, the proposed method in this chapter cannot be employed. However, we believe that there are numerous situations where it is feasible to make estimates under the condition that we can underestimate the amount of error.

The motivation, research gap, and contribution of this study are summarized below.

Motivation: This study aims to estimate the distribution of personal data sensed in IoT environments while protecting user data using differential privacy. We assume that the sensed data contains sensing noise.

Research gap: Existing methods do not take sensing noise into account. As a result, they introduce excessive privacy noise into the sensed data.

Contribution: First, we propose true-value-based differential privacy (TDP), a novel differential privacy concept that considers sensing noise. Second, we propose anonymization algorithms for numerical and categorical data that satisfy TDP. Third, we demonstrate that the proposed algorithms ensure TDP. Fourth, we show that the proposed algorithms can reduce the amount of differential privacy noise using synthetic and real datasets. Fifth, we illustrate that the proposed algorithms can decrease error in the estimated distribution for personal data using the same datasets.

3.2 Models

3.2.1 Assumptions

Anonymizers may not know the true values of an attribute, but they can estimate them. However, these estimated values may contain error. Anonymizers can also estimate the error distribution of numerical attribute values. The normal distribution is considered the error model for numerical attributes, as measurement errors follow the normal distributions in many cases [37]. The normal distribution is characterized by the parameter σ, which represents its standard deviation. However, please note that the concept of TDP can be applied to other error models.

The probability of wrong classification $p_{i \to j}$ is considered with reference to categorical attributes. This probability signifies that the ID of the true category is i. However, the anonymizer is unaware of the true category ID and assumes that the category ID is j.

In this section, parameters σ and $p_{i \to j}$ for all i, j are referred to as "error parameters."

Three scenarios are assumed.

Scenario I: The anonymizer knows the exact error parameters.

Scenario II: The anonymizer does not know the exact error parameters. The estimated parameters may differ from the actual parameters; however, the anonymizer is not pessimistic about the degree of error. The mathematical definitions of the numerical attributes are described in Sect. 3.3.1, and those of the categorical attributes are described in Sect. 3.3.2.

Scenario III: The anonymizer does not know the exact error parameters and has no estimate for them.

In this chapter, we do not focus on Scenario III. As Scenario I is somewhat unrealistic, we generally focus on Scenario II.

3.2.2 Privacy Metric

Suppose that a person has an attribute value, and the person or anonymizer who collects the attribute value anonymizes the value. Let ϵ be a positive real number. The, the differential privacy is defined as follows.

In this section, it is considered that the value of x may contain sensing error. Therefore, the focus must be placed on the true value of x, which is an unknown value, even for the data owner and the anonymizer. TDP is proposed to handle the privacy of unknown values.

Definition 3 (TDP) Let x and x' be true values and let ϵ be a positive real number. A measurement function $\mathcal{M}$ acquires an input x and outputs a measured value. A randomized mechanism $\mathcal{A}$ satisfies TDP if and only if for any output y, the following equation holds:

$$P(\mathcal{A}(\mathcal{M}(x)) = y) \leq e^{\epsilon} P(\mathcal{A}(\mathcal{M}(x')) = y) \quad \text{for all} \quad x, x'. \tag{4}$$

Theorem 2 *In an anonymized data collection scenario, Definition 1 is the same as Definition 3 when the measured values contain no error.*

Proof When the measured values contain no error, the equations $x = \mathcal{M}(x)$ and $x' = \mathcal{M}(x')$ hold. Therefore, in this case, Eqs. 1 and 4 are equivalent. □

3.3 True-Value-Based Differential Privacy (TDP)

Existing studies define x and x' in Definition 1 as measured values. In this section, they are defined as true values. The anonymization mechanisms for both numerical and categorical attributes are described next (Table 2).

Table 2 Notation

ϵ	Privacy budget for differential privacy
Δ	Range of possible values for a numerical attribute
M	Number of categories for a categorical attribute
τ	Probability of correct estimation of category ID by IoT devices
σ	Standard deviation of the normal distribution
b	Scale parameter of the Laplace distribution (equal to Δ/ϵ)

3.3.1 Numerical Value Anonymization

The Laplace mechanism (Theorem 1), which adds noise based on the Laplace distribution, can be used for numerical attributes. However, the Laplace mechanism does not take sensing error into consideration. As a result, the noise of the normal distribution is added to true values as sensing error, and additional noise based on the Laplace mechanism is added to the noisy value. This is the traditional approach, which is referred to as the **baseline approach** for numerical attributes, and it always adds the Laplace noise. The resulting probability density function, which represents the probability of the distance between the final noisy and the true values, can be calculated by convoluting the normal and Laplace distributions.

Let $\mathcal{N}(x;\sigma^2)$, $\mathcal{L}(x;b)$ represent the probability density functions of the normal distribution, with the standard deviation being σ and the scale parameter of the Laplace distribution being b. Centered distributions that peak at zero are only considered without loss of generality.

A convolution of the normal distribution with a standard deviation of σ and of the Laplace distribution with a scale parameter of b is represented by

$$\mathcal{U}(x;\sigma^2,b) = \mathcal{N} \star \mathcal{L} = \int_{t=-\infty}^{\infty} \mathcal{N}(t;\sigma^2)\mathcal{L}(x-t;b)dt$$
$$= \frac{e^{\frac{\sigma^2-2bx}{2b^2}} \left(\mathrm{erfc}\left(\frac{\sigma^2-bx}{\sqrt{2}b\sigma}\right) + e^{\frac{2x}{b}} \mathrm{erfc}\left(\frac{\sigma^2+bx}{\sqrt{2}b\sigma}\right)\right)}{4b} \quad (5)$$

where erfc is the complementary error function, which is represented by

$$\mathrm{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt. \quad (6)$$

It is noted that for Scenario II, the value of σ can be wrong, as long as it is not pessimistic. Let σ_t and σ represent the true standard deviation and the standard deviation assumed by the anonymizer, respectively. Here, pessimistic means that

$$\sigma > \sigma_t. \quad (7)$$

$\exp(\epsilon)$, $1/\exp(\epsilon)$, and the ratio of the probability density function values whose distance is Δ with respect to $\mathcal{N}(x;\sigma^2)$, $\mathcal{L}(x;\Delta/\epsilon)$, and $\mathcal{U}(x;\sigma^2,\Delta/\epsilon)$, where ϵ and σ are set to one, are presented in Fig. 3. The ratio of the probability density function values whose distance is Δ with respect to the normal distribution is calculated by

$$R_{\mathcal{N}(x;\sigma^2)} = \frac{\mathcal{N}(x+\Delta/2;\sigma^2)}{\mathcal{N}(x-\Delta/2;\sigma^2)} = e^{-\frac{\Delta x}{\sigma^2}}. \quad (8)$$

Equation 8 shows that $R_{\mathcal{N}(x;\sigma^2)}$ approaches ∞ when x is close to $-\infty$. Therefore, even if σ is very large, extra noise needs to be added to achieve ϵ-differential privacy.

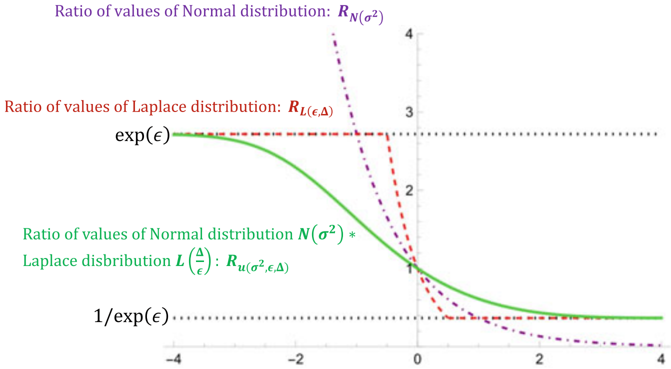


Fig. 3 Ratio of probability density function values of the normal distribution and Laplace distribution, and the convolution of the two distributions ($\sigma = \epsilon = \Delta = 1$)

Similarly, in Fig. 3, $R_{\mathcal{L}(x;\epsilon,\Delta)}$ and $R_{\mathcal{U}(x;\sigma^2,\epsilon,\Delta)}$ are defined as the ratio of the probability density function values whose distance is Δ with respect to $\mathcal{L}(x; \Delta/\epsilon)$ and $\mathcal{U}(x; \sigma^2, \Delta/\epsilon)$, respectively.

The ratio of the probability density function values whose distance is Δ should appear between the lines of $\exp(\epsilon)$ and $1/\exp(\epsilon)$, according to the definition of ϵ-differential privacy. Figure 3 shows that $R_{\mathcal{L}(x;\epsilon,\Delta)}$ and $R_{\mathcal{U}(x;\sigma^2,\epsilon,\Delta)}$ satisfy this condition; therefore, the $\mathcal{L}(x; \Delta/\epsilon)$ and $\mathcal{U}(x; \sigma^2, \Delta/\epsilon)$ mechanisms achieve ϵ-differential privacy (here $\sigma = \Delta = \epsilon = 1$). Although $R_{\mathcal{U}(x;\sigma^2,\Delta/\epsilon)}$ approaches $\exp(\epsilon)$ (or $1/\exp(\epsilon)$) when $|x|$ is large, its convergence to $\exp(\epsilon)$ (or $1/\exp(\epsilon)$) is slower than that of $R_{\mathcal{L}(x;\Delta/\epsilon)}$. Consequently, the mechanism adds much more noise than is required.

The algorithm proposed in this section is simple but effective; Laplace noise is not added when the calculated Laplace noise is smaller than the predefined threshold w. Thus, the total loss is expected to become smaller (i.e., the ratio of the probability density function values whose distance is Δ is expected to approach $\exp(\epsilon)$ and $1/\exp(\epsilon)$ faster).

However, the definition of an appropriate value for w is complex. If the threshold w is very large, the resulting value cannot achieve either traditional ϵ-differential privacy or TDP. Conversely, the resulting value contains unnecessary noise if the threshold w is very small.

The probability density function, which adds the Laplace noise only when the noise x satisfies $\text{abs}(x) \geq w$[1], is represented by

[1] More formally, this is a combination of a probability density function and a probability mass function.

$$\widehat{\mathcal{L}}(x;b,w) = \begin{cases} \int_{-w}^{w} \mathcal{L}(t;b)dt & x = 0 \\ \frac{e^{-x/b}}{2b} & x \geq w \\ \frac{e^{x/b}}{2b} & x \leq -w \\ 0 & otherwise. \end{cases} \tag{9}$$

Therefore, the probability density function obtained from the original sensing error and the Laplace noise defined in Eq. 9 can be represented by

$$\begin{aligned} \mathcal{V}(x;\sigma^2,b,w) &= \int_{-\infty}^{\infty} \mathcal{N}(t;\sigma^2)\widehat{\mathcal{L}}(x-t;b,w)dt \\ &+ \mathcal{N}(x;\sigma^2)\int_{-w}^{w} \mathcal{L}(t;b)dt \\ &= \frac{e^{-\frac{w+x}{b}-\frac{x^2}{2\sigma^2}}}{4b\sigma} \times \left\{\sigma e^{\frac{1}{2}\left(\frac{2bw+\sigma^2}{b^2}+\frac{x^2}{\sigma^2}\right)}\left[\text{erfc}\left(\frac{b(w-x)+\sigma^2}{\sqrt{2}b\sigma}\right)\right.\right. \\ &+ e^{\frac{2x}{b}}\text{erfc}\left(\frac{b(w+x)+\sigma^2}{\sqrt{2}b\sigma}\right)\left.\right] + 2\sqrt{\frac{2}{\pi}}b\left(e^{\frac{w}{b}}-1\right)e^{\frac{x}{b}}\left.\right\} \end{aligned} \tag{10}$$

For the proposed algorithm, the ratio of the probability density function values whose distance is Δ is represented by the following:

$$R_{\mathcal{V}(x;\sigma^2,\epsilon,\Delta,w)} = \frac{\mathcal{V}(x+\Delta/2;\sigma^2,\Delta/\epsilon,w)}{\mathcal{V}(x-\Delta/2;\sigma^2,\Delta/\epsilon,w)} \tag{11}$$

The objective is to find an appropriate value of w such that $R_{\mathcal{V}}$ approximates $\exp(\epsilon)$ but $R_{\mathcal{V}}$ does not overestimate $\exp(\epsilon)$ or $1/\exp(\epsilon)$.

The following theorem is considered (see Fig. 4):

Theorem 3 *If w is near ∞, the value of $R_{\mathcal{V}}$ approaches the value of $R_{\mathcal{N}}$. If w is near zero, the value of $R_{\mathcal{V}}$ approaches the value of $R_{\mathcal{U}}$.*

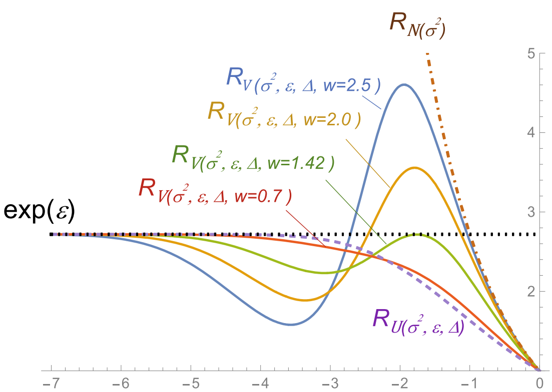


Fig. 4 $R_{\mathcal{V}}$ for various values of w ($\sigma = \epsilon = \Delta = 1$). It can be seen that if the value of w is too large, the requirement for differential privacy is not met. Alternatively, it can be seen that if the value of w is too small, more noise is added than is necessary

Proof $\mathcal{U}(x; \sigma^2, \Delta/\epsilon)$ (Eq. 5) and $\mathcal{N}(x; \sigma^2)$ (Eq. 8) can be obtained by calculating the limit of $\mathcal{V}(x; \sigma^2, \Delta/\epsilon, w)$ (Eq. 10) of w as w approaches zero and ∞, respectively. □

The ratio between $x + \Delta/2$ and $x - \Delta/2$ is defined in this study; therefore, the range $-w - \Delta/2 < x < 0$ can be considered to check whether or not the maximum ratio is greater than $\exp(\epsilon)$. It is noted that only the range $x < 0$ needs to be checked because $\mathcal{V}$ is symmetrical with respect to the point $(x, y) = (0, 1)$, where y represents the ratio of the probability density function values whose distance is Δ.

Algorithm 1 describes the method that yields the anonymized value. In Algorithm 1, the value of w is calculated at Lines 1–16. $\mathrm{erfc}(x)$ can be computed using approximate equations, such as

$$\begin{aligned}\mathrm{erfc}(x) &= 1 - \mathrm{erf}(x) \approx 1 - \sqrt{1 - e^{-x^2 \frac{4/\pi + 0.147x^2}{1+0.147x^2}}} \\ &\text{(maximum relative error: } 1.3 \cdot 10^{-4}\text{)}\end{aligned} \tag{12}$$

when $x \geq 0$ from [76]. Note that we can obtain an approximate value of $\mathrm{erfc}(x)$ with $x < 0$ from the property of

$$\mathrm{erfc}(x) = 2 - \mathrm{erfc}(-x). \tag{13}$$

After checking the approximate values, precise values must be calculated. Mathematical tools such as Maxima[2], which is a popular free software program, can be employed.

3.3.2 Categorical Values Anonymization

The randomized response mechanism [75] can be used for categorical attributes. First, a sensed value is categorized into one of the predefined categories. Another category replaces that category with a certain probability, and then the resulting category ID is sent to the data receiver. The randomized response is referred to as the **baseline approach** for categorical attributes.

The retention probability of an unchanging category ID is p_α, and the probabilities of other IDs are $(1 - p_\alpha)/(M - 1)$, where M is the number of categories. The equation

$$\max\left(\frac{p_\alpha}{(1 - p_\alpha)/(M - 1)}, \frac{(1 - p_\alpha)/(M - 1)}{p_\alpha}\right) \leq e^\epsilon \tag{14}$$

should hold to satisfy ϵ-differential privacy. Therefore, the following is set:

$$p_\alpha = e^\epsilon/(M - 1 + e^\epsilon). \tag{15}$$

Because $M \geq 2$, $p_\alpha > 0.5$ is obtained.

[2] https://maxima.sourceforge.net/.

Algorithm 1 Proposed randomization mechanism for numerical attributes

Input: Privacy budget ϵ, Standard deviation of the normal distribution for sensing error σ, Range of possible values Δ, Measured value v_s
Output: TDP value

$w_{max} \leftarrow$ sufficiently large value
$w_{min} \leftarrow 0$
while True **do**
  $w' \leftarrow (w_{max} + w_{min})/2$
  $r \leftarrow \max_{-w-\Delta/2 \le x \le 0} (R_{\mathcal{V}(x;\sigma^2,\epsilon,\Delta,w')} - \exp(\epsilon))$.
  if $r > 0$ **then**
    $w_{max} \leftarrow w'$
  else
    if $w' - w_{min}$ is sufficiently small **then**
      $w \leftarrow w'$
      Break.
    else
      $w_{min} \leftarrow w'$
    end if
  end if
end while
Generate Laplace noise l based on $\mathcal{L}(0, \Delta/\epsilon)$.
if $l < w$ **then**
  Return v_s.
else
  Return $v_s + l$.
end if

Let $p_{i\to j}$ represent the probability that the true category ID C_i is (mis-)classified to C_j due to sensing error. It is assumed that the retention probability is greater than any other probability; that is, the following equation is assumed:

$$p_{i\to i} > \max_{j\neq i} p_{i\to j}. \tag{16}$$

It is assumed that the values of $p_{i\to j}$ for all i, j can be estimated. Let

$$\boldsymbol{p_i} = \{p_{i\to 1}, p_{i\to 2}, \ldots, p_{i\to M}\}. \tag{17}$$

For Scenario II, these values can be wrong, as long as they are not pessimistic.

Let $p_{i\to j,t}$ and $p_{i\to j}$ represent the true probability and the probability that the anonymizer assumes, respectively. Here, pessimistic estimation means that

$$\begin{cases} p_{i\to i} < p_{i\to i,t} \text{ for any } i, \\ p_{i\to j} > p_{i\to j,t} \text{ for any } i, j\,(i \neq j). \end{cases} \tag{18}$$

First, the following expression is satisfied:

$$\frac{p_{i \to j}}{p_{i' \to j}} \leq e^{\epsilon} \text{ for all } i, i', j. \tag{19}$$

This case clearly holds TDP. In this case, the random mechanism $\mathcal{A}$ in Definition 3 does not need to do anything. In other words, the TDP can be satisfied by outputting the measured input values as they are.

If Eq. 19 is not satisfied, the following simultaneous equations with respect to $x_{i \to j}$ for all i and j are solved:

$$\begin{aligned} \boldsymbol{p_i} \cdot \boldsymbol{x_i} &= p_{\alpha} \quad \text{for} \quad i = 1, \ldots, M, \\ \boldsymbol{p_i} \cdot \boldsymbol{x_j} &= \frac{1 - p_{\alpha}}{M - 1} \quad \text{for} \quad i, j = 1, \ldots, M \quad \text{s.t.} \quad i \neq j, \end{aligned} \tag{20}$$

where

$$\boldsymbol{x_i} = \{x_{1 \to i}, x_{2 \to i}, \ldots, x_{M \to i}\} \tag{21}$$

and $\cdot$ represents the scalar product of two vectors.

The value of $x_{i \to i}$ may be greater than one, and the value of $x_{i \to j}$ may be less than zero. Therefore, the obtained values are normalized by

$$\begin{aligned} x_{i \to i} &\leftarrow \min(1, x_{i \to i}) \quad \text{for} \quad i = 1, \ldots, M, \\ x_{i \to j} &\leftarrow \max(0, x_{i \to j}) \quad \text{for} \quad i, j = 1, \ldots, M \quad \text{s.t.} \quad i \neq j. \end{aligned} \tag{22}$$

Finally, when the measured category ID is C_i, the anonymizer generates the anonymized version C_j with probability $x_{i \to j}$.

Algorithm 2 shows the method that yields the anonymized category ID.

Algorithm 2 Proposed randomization mechanism for categorical attributes

Input: Privacy budget ϵ, Probabilities $p_{i \to j}$ for all i, j, measured category ID s, IDs of categories K

Output: TDP value Scenarios I and II

1: Calculate p_{α} from Equation 15.
2: **if** Equation 19 holds **then**
3: Return s.
4: **else**
5: Solve simultaneous equations 20 and obtain $\boldsymbol{x_i}$ for all i.
6: Normalize $\boldsymbol{x_i}$ using Equation 22.
7: Randomly select j each with a probability $x_{i \to j}$, and return j.
8: **end if**

3.3.3 Proof of Achieving True Value-Based Differential Privacy

Next, it is proved that the proposed algorithms (for Scenarios I and II) realize TDP.

Numerical Attributes

First, Scenario I is considered. Because Algorithm 1 ensures that $1/\exp(\epsilon) \leq R_{\mathcal{V}(x;\sigma^2,\epsilon,\Delta,w)} \leq \exp(\epsilon)$ for the true value if σ is correct, it achieves TDP based on Definition 3.

Next, Scenario II is considered. It is assumed that the anonymizer's knowledge about the sensing error is not correct, but that their assumption about the measurement error is not pessimistic. The concept "pessimistic" is defined in Eq. 7 in relation to numerical attributes.

Let the ratio of the probability density function values whose distance is Δ with respect to $\mathcal{N}(x;\sigma^2)$ be $R_{\mathcal{N}(x;\sigma^2)}$. By differentiating $R_{\mathcal{N}(x;\sigma^2)}$ with respect to σ, we obtain

$$\frac{\partial R_{\mathcal{N}(x;\sigma^2)}}{\partial \sigma} = \frac{2\Delta e^{-\frac{\Delta x}{\sigma^2}} x}{\sigma^3}. \tag{23}$$

When x is less than zero, the value of differentiating $R_{\mathcal{N}(x;\sigma^2)}$ with respect to σ is always less than zero. Therefore, if σ becomes larger, the value of $R_{\mathcal{N}(x;\sigma^2)}$ becomes smaller. It can be concluded that $R_{\mathcal{V}(x;\sigma^2,\epsilon,\Delta,w)}$ becomes smaller when σ becomes larger, because the proposed probability density function $\mathcal{V}(x;\sigma^2,\Delta\epsilon,w)$ is a convolutional function of $\mathcal{N}(x;\sigma^2)$ and Eq. 9, which does not depend on σ. Therefore, if the anonymizer's assumption about the measurement error is not pessimistic, then $R_{\mathcal{V}(x;\sigma^2,\epsilon,\Delta,w)} \leq R_{\mathcal{V}(x;\sigma_t^2,\epsilon,\Delta,w)}$ for $x \leq 0$. If the anonymizer sets the value of error parameters as pessimistic (i.e., set σ to a small value), the amount of noise added by the proposed mechanism is larger than the amount needed. Although the usefulness of the proposed algorithm is less in this case, the ratio of the anonymization probabilities generated by the proposed mechanism from two neighboring databases is between $\exp(\epsilon)$ and $1/\exp(\epsilon)$, with some extra space available. However, the total loss of the proposed mechanism is less than that of the baseline approach, even in this case. When $x > 0$ is considered, the discussion is similar, and then $R_{\mathcal{V}(x;\sigma^2,\epsilon,\Delta,w)} > R_{\mathcal{V}(x;\sigma_t^2,\epsilon,\Delta,w)}$ for $x > 0$.

Because $1/\exp(\epsilon) \leq R_{\mathcal{V}(x;\sigma_t^2,\epsilon,\Delta,w)} \leq \exp(\epsilon)$ for σ_t^2, then $1/\exp(\epsilon) \leq R_{\mathcal{V}(x;\sigma^2,\epsilon,\Delta,w)} \leq \exp(\epsilon)$ for σ^2. Therefore, Definition 3 holds.

Categorical Attributes

First, Scenario I is considered. It is assumed that the attacker obtains a category ID γ as the anonymized version of a categorical attribute. Let $P(v_a = \gamma | v_t = i)$ represent the anonymized version of the category ID γ when the probability that the true category ID is i. The proposed mechanism ensures that

$$P(v_a = \gamma | v_t = i) = \begin{cases} \frac{e^\epsilon}{M-1+e^\epsilon} & (i = \gamma) \\ \frac{1-\frac{e^\epsilon}{M-1+e^\epsilon}}{M-1} & (\text{otherwise}) \end{cases} \tag{24}$$

when we ignore the process in Eq. 22. The ratio of the two equations in Eq. 24 is e^ϵ or $1/e^\epsilon$. Therefore, Definition 3 holds. Based on the post-processing property of differential privacy, the values resulting from the process of Eq. 22 also satisfy TDP.

Next, Scenario II is considered. It is assumed that the anonymizer's knowledge about the sensing error is not correct but their assumption about the measurement error is not pessimistic. Let $x_{i\to j,t}$ and $x_{i\to j}$ represent the disguising probabilities based on the true error parameters and the assumed error parameters, respectively. If the error parameters are not pessimistic, then

$$\begin{cases} x_{i\to j,t} \geq x_{i\to j} & (i = j) \\ x_{i\to j,t} \leq x_{i\to j} & (\text{otherwise.}) \end{cases} \tag{25}$$

Therefore,

$$\begin{cases} P(v_a = \gamma | v_t = i) \leq \frac{e^\epsilon}{M-1+e^\epsilon} & (i = \gamma) \\ P(v_a = \gamma | v_t = i) \geq \frac{1-\frac{e^\epsilon}{M-1+e^\epsilon}}{M-1} & (\text{otherwise}) \end{cases} \tag{26}$$

From Eqs. 16 and 26, it is concluded that Definition 3 holds.

3.4 Analysis

3.4.1 Numerical Attributes

The proposed mechanism skips the addition of Laplace noise if the generated Laplace noise l is less than the threshold w. Then, the avoidance (or skipping) ratio can be calculated by

$$\int_{-w}^{w} \mathcal{L}(x; \Delta/\epsilon)dx = 1 - e^{-\epsilon w/\Delta}. \tag{27}$$

Let $\eta_\mathcal{U}$ and $\eta_\mathcal{V}$ represent the expected values of the amount of additional Laplace noise with respect to the baseline approach and the proposed mechanism, respectively. The value of $\eta_\mathcal{U}$ can be calculated by

$$\eta_\mathcal{U} = \int_{-\infty}^{\infty} |x| \cdot \mathcal{L}(x; \Delta/\epsilon)dx = \frac{\Delta}{\epsilon}, \tag{28}$$

and the value of $\eta_\mathcal{V}$ can be calculated by

$$
\begin{aligned}
\eta_{\mathcal{V}} &= \int_{-\infty}^{-w} -x\mathcal{L}(x;\,\Delta/\epsilon)dx + \int_{w}^{\infty} x\mathcal{L}(x;\,\Delta/\epsilon)dx \\
&= e^{-\frac{w\epsilon}{\Delta}}(\frac{\Delta}{\epsilon} + w)
\end{aligned}
\tag{29}
$$

3.4.2 Categorical Attributes

Let $\zeta_{\mathcal{U}}$ and $\zeta_{\mathcal{V}}$ represent the probabilities that the true category ID is equivalent to the anonymized category ID that corresponds to the baseline approach and the proposed mechanism, respectively. The baseline approach represents a method that always adds the Laplace noise with respect to numerical attributes, whereas the randomized response method adds the Laplace noise with respect to categorical attributes, as described in Sects. 3.3.1 and 3.3.2. Assuming that the true category ID is i,

$$\zeta_{\mathcal{U}} = p_{i\to i} \cdot p_{\alpha} + \sum_{j} p_{i\to j} \cdot \frac{1-p_{\alpha}}{M-1}, \tag{30}$$

and

$$\zeta_{\mathcal{V}} = p_{i\to i} \cdot x_{i\to i} + \sum_{j} p_{i\to j} \cdot x_{j\to i}. \tag{31}$$

3.5 *Evaluation*

3.5.1 Utility Metric

The data receiver intends to use the anonymized value for several services. Therefore, the estimated value should be close to the true value. Let N represent the number of people whose attribute values are collected. Let v_i and $\widetilde{v_i}$ represent the true value and the anonymized value, respectively, of an attribute of person i.

The utility is defined as follows with respect to numerical attributes:

$$U_n = \frac{1}{N}\sum_{i=1}^{N}\left(1 - \frac{|v_i - \widetilde{v_i}|}{\Delta}\right), \tag{32}$$

whereas the utility is defined as follows with respect to categorical attributes:

$$U_c = \frac{1}{N}\sum_{i=1}^{N}\delta_{v_i,\widetilde{v_i}}, \tag{33}$$

where $\delta_{i,j}$ is the Kronecker delta

$$\delta_{i,j} = \begin{cases} 1 & (i = j) \\ 0 & (i \neq j). \end{cases} \tag{34}$$

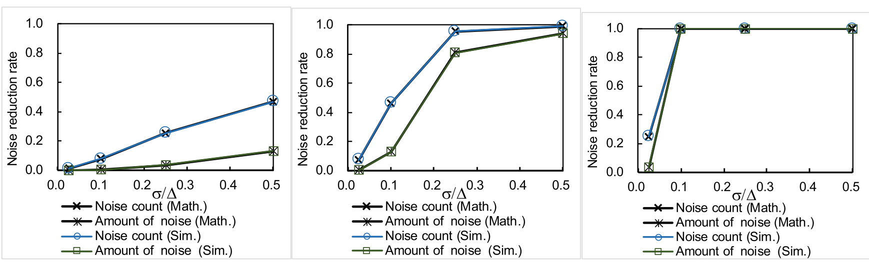
Both metrics are considered superior if their values are significant.

Some methods can estimate statistical values (e.g., averages) or generate cross-tabulations of the collected data. If the goal is to generate cross-tabulations, then a total loss, which compares the true cross-tabulation with the generated cross-tabulation, should be used. However, in this section, the focus is mainly on the data of a single individual; that is, the aim is not to do a statistical analysis but to use the attribute value for *each person*, because IoT-related services such as the health monitoring, context-aware recommender systems, and navigation described in Sect. 3.1 need to analyze an individual's attribute value.

3.5.2 Numerical Value Results

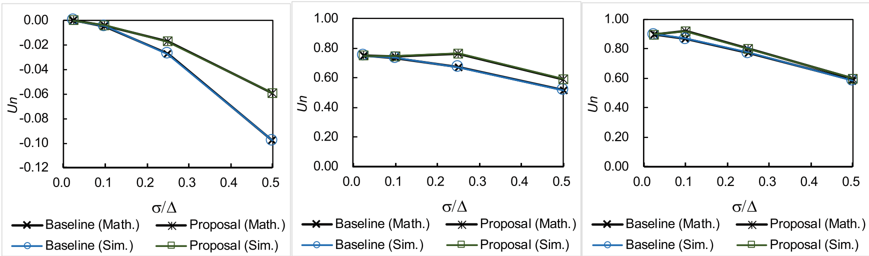
Δ is set within the range of 10–1,000, ϵ within the range of 1–10, and σ within the range from 1/40 of the value of Δ to 1/2 of the value of Δ. We evaluated the number of times the proposed mechanism skipped the addition of Laplace noise to a measured value, as well as the mechanism's ability to reduce the average amount of Laplace noise added. The results for $\Delta = 10$ are shown in Fig. 5, along with the computed results of Eqs. 27, 28, and 29. The results for $\Delta = 100$ and $\Delta = 1000$ are nearly identical to those in Fig. 5 and are therefore not shown.

The computed results based on Eqs. 27, 28, and 29 align closely with the simulation results for all parameter settings. The proposed mechanism reduced the frequency of Laplace noise addition and reduced the corresponding average Laplace noise. Large values of σ or ϵ result in a significant reduction rate. A high value of σ indicates a substantial sensing error noise has already been added to a true value, whereas a high value of ϵ signifies a low privacy protection level, meaning a large amount of noise is not necessary. Consequently, the proposed mechanism reduces additional



(a) $\epsilon = 1$. (b) $\epsilon = 4$. (c) $\epsilon = 10$.

Fig. 5 Reduction rate of the proposed mechanism with respect to noise addition counts and amount of Laplace noise. (The results are for $\Delta = 10$. Results for $\Delta = 100$ and $\Delta = 1000$ are almost the same)



(a) $\epsilon = 1$. (b) $\epsilon = 4$. (c) $\epsilon = 10$.

Fig. 6 U_n results (The results are for $\Delta = 10$. Results for $\Delta = 100$ and $\Delta = 1000$ are almost the same)

Laplace noise, particularly when the values of σ and ϵ are large. According to Eq. 8, the need to add noise cannot be entirely avoided. However, Fig. 5 shows that the noise skipping ratio approaches one.

We evaluated U_n using Eq. 32 with the same values for Δ and ϵ as above (Fig. 6). A large σ value results in a low U_n (i.e., a high total loss), even if none of the privacy protection mechanisms are used, so the difference between the proposed mechanism and the baseline approach is small. This is true not only when the value of σ is small, but also when it is large. However, if σ is set to a medium value, the proposed mechanism can reduce the total loss U_n by 25%–40% compared with the baseline approach. When ϵ is set to one, the difference between the proposed mechanism and the baseline approach is small. However, when the value of ϵ equals one, the average absolute value of the Laplace noise to be added is about 50 when $\Delta = 100$. This amount of noise appears to be quite large. Therefore, in typical cases, the value of ϵ should be larger.

We determined the actual ratio of probability density function values whose distance is Δ by conducting simulations. The true values to be protected were set to $-\Delta/2$ and $\Delta/2$. Noise from the normal distribution was randomly added to the true values independently. The noise-added values were anonymized using the proposed mechanism and the baseline approach, respectively. Histograms with 200 bins were created for the range -3Δ to 3Δ. This simulation was repeated 2^{31} times. In Fig. 7, we present an example of the average result with $\epsilon = 2$, $\Delta = 100$, and $\sigma = 25$. The ratio of the probability density function values of the normal distribution and the Laplace distribution, along with $\exp(\epsilon)$ and $1/\exp(\epsilon)$ functions, are also shown as a reference. The results for both the proposed and the baseline approaches lie within the range from $\exp(\epsilon)$ to $1/\exp(\epsilon)$. Consequently, we conclude that both mechanisms (for Scenarios I and II) achieve TDP. The ratio of the probability density function values of the Laplace distribution is the same as $\exp(\epsilon)$ and $1/\exp(\epsilon)$ in the range where $x < -\Delta/2$ and $\Delta/2 < x$; thus, the Laplace mechanism is optimal if the measured values have no error. As for the proposed mechanism, the ratio of the probability density function values approaches $\exp(\epsilon)$ and $1/\exp(\epsilon)$ at approximately $x = -\Delta/2$ and $x = \Delta/2$. However, this ratio deviates slightly from $\exp(\epsilon)$ and $1/\exp(\epsilon)$ at

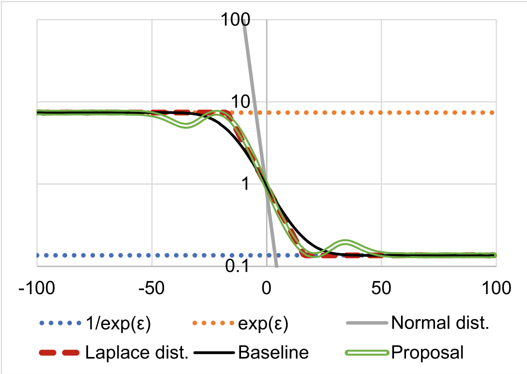


Fig. 7 Example of simulation results: the ratio of probability distributions for numerical attributes ($\epsilon = 2, \Delta = 100, \sigma = 25$)

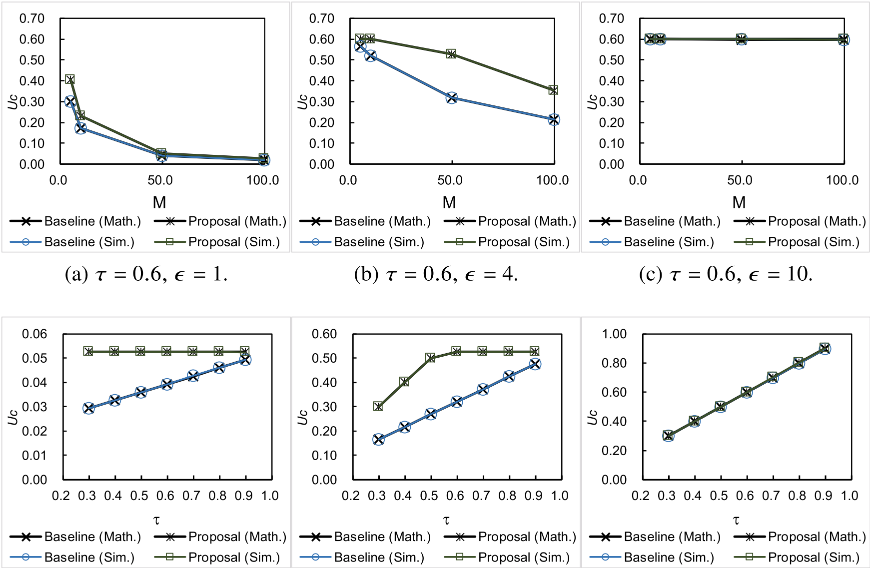
$x = -40$ and $x = 40$. In contrast, the baseline approach's ratio of the probability density function values reaches $\exp(\epsilon)$ and $1/\exp(\epsilon)$ at about $x = -30$ and $x = 30$. It is important to note that the probability density function values are large when x is near zero; therefore, high utility can be achieved if the ratio is close to $\exp(\epsilon)$ and $1/\exp(\epsilon)$ when x is near zero. Hence, the proposed mechanism attains high utility (i.e., low total loss) relative to the baseline approach.

We conducted additional simulations with different parameter settings. As a result, we confirmed that the ratio of the probability density function values of the proposed mechanism lies within the range from $\exp(\epsilon)$ to $1/\exp(\epsilon)$, except for those results that show considerable variation due to an insufficient number of samples in each bin.

3.5.3 Categorical Value Results

The value of ϵ was set to the range 1–10, the value of M was set to the range 5–100, and the value of τ was set to the range 0.3–0.9. The true category ID was set to a random integer, and the category ID with probability $1 - \tau$ was changed. Then, the category ID was randomized using the baseline mechanism and using the proposed mechanism. This simulation was repeated 2^{31} times. The results for ϵ equal to one are shown in Fig. 8. The simulation results along with the computed results calculated using Eqs. 30 and 31 are also presented. A close agreement can be observed between the simulated and computed results.

The values of U_c obtained using the proposed method are larger than or equal to those obtained using the baseline approach for all parameter settings. When M is large or ϵ is small, the values of U_c are small for both mechanisms since it is difficult to maintain high accuracy for both mechanisms in such cases. However, in other cases, the proposed mechanism reduces the total loss more than does the baseline approach, especially when ϵ is small, i.e., the privacy protection level is high. When ϵ is large, the experimental results of the proposed method are similar to those of other



(a) $\tau = 0.6, \epsilon = 1$. (b) $\tau = 0.6, \epsilon = 4$. (c) $\tau = 0.6, \epsilon = 10$.

(d) $M = 50, \epsilon = 1$. (e) $M = 50, \epsilon = 4$. (f) $M = 50, \epsilon = 10$.

Fig. 8 U_c results

methods; the value of ϵ is large enough that the noise added to achieve differential privacy is very small. This is why there was no difference in accuracy between the methods in such cases. Therefore, it is more important to experiment when the value of ϵ is small.

3.5.4 Real Dataset Results

Simulations were conducted using a real dataset called the Adult dataset [10], which is a widely used benchmark in research for privacy-preserving data mining. This dataset consists of six numerical attributes and nine categorical attributes, and it has 30,162 records when unknown values are excluded.

We assumed that each value in the Adult dataset was true. We also assumed that IoT devices estimated age, sex, race, and native country using estimation methods [21]. For numerical attributes, σ was set to 0.1 of the value of Δ, and ϵ was set to 8. For categorical attributes, τ was set to 0.6, and ϵ was set to 2.

Table 3 Adult dataset results [10]

(a) U_n results: numerical attributes

Attribute name	Age	fnlwgt	Education-num	Capital-gain	Capital-loss	Hours-per-week			
Δ	73	1470936	15	99999	4356	98			
Baseline	0.85	0.85	0.85	0.85	0.85	0.85			
Proposal	**0.92**	**0.92**	**0.92**	**0.92**	**0.92**	**0.92**			

(b) U_c results: categorical attributes

Attribute name	Workclass	Education	Marital-status	Occupation	Relationship	Race	Sex	Native-country	Salary
M	7	16	7	14	6	5	2	41	2
Baseline	0.36	0.22	0.36	0.24	0.39	0.42	0.58	0.10	0.58
Proposal	**0.55**	**0.33**	**0.55**	**0.36**	**0.60**	**0.60**	**0.60**	**0.16**	**0.60**

The simulation results are presented in Table 3. The names of the attributes, along with the values of Δ and M, are also shown. The proposed mechanism was able to increase U_n to approximately 92% from approximately 85% for all numerical attributes and to increase U_c by a maximum of 20% for the categorical attributes relative to the baseline approach. These results demonstrate that the proposed mechanism enhances utility (i.e., reduces total loss) for real datasets.

Lastly, simulations were conducted using other real datasets with the same parameter settings as above.

A dataset of activities based on multisensor data fusion (AReM dataset) [45] was used for numerical attributes. This dataset consists of 42,239 instances of six numerical attributes.

Datasets containing daily living activities as recognized by binary sensors (ADL dataset) [43], the activities of healthy older people using non-battery wearable sensors (RFID dataset) [68], and the localization of people's activity (Localization dataset) [26] were used for the categorical attributes. The numbers of instances in these datasets are 741, 75,128, and 164,860, respectively.

The simulation results are displayed in Table 4. These results show that the proposed mechanism outperforms the baseline approach on all datasets used in this study.

3.6 Related Research Work

A considerable amount of research has been conducted on anonymized data collection. Wang et al. [72] introduced a method for identifying the top-k most frequently used new terms by gathering term usage data from individuals under the constraint of differential privacy. Kim et al. [30] derived population statistics by collecting differentially private indoor positioning data. Encryption-based approaches for anonymized data collection have also been explored [36]. These methods primarily focus on obtaining aggregate values and are not intended to acquire individual

Table 4 Results for four real datasets

(a) U_n results: numerical attributes of AReM dataset [45]						
Attribute name	avg_rss12	var_rss12	avg_rss13	var_rss13	avg_rss23	var_rss23
Δ	56.25	17.24	35	11.42	40.33	13.61
Baseline	0.85	0.85	0.85	0.85	0.85	0.85
Proposal	**0.92**	**0.92**	**0.92**	**0.92**	**0.92**	**0.92**
(b) Results of U_c: categorical attributes of three datasets						
Dataset name	ADL dataset [43]	RFID dataset [68]	Localization dataset [26]			
M	10	4	11			
Baseline	0.30	0.47	0.28			
Proposal	**0.45**	**0.60**	**0.42**			

values. Furthermore, they do not account for error in the collected values. In contrast, the proposed scenario seeks to obtain each person's value as accurately as possible, as services like recommender systems require individual attribute values.

Abul et al. [2] and Sei et al. [53] put forth location anonymization methods that consider location error and achieve k-anonymity [41, 42, 58, 65], which is a fundamental privacy metric. However, these methods are not applicable to ϵ-differential privacy.

Ge et al. [15] and Krishnan et al. [32] proposed techniques for privately cleaning "dirty data". By employing differential privacy as a privacy metric, they focused on data cleaning to resolve inconsistencies in large databases containing the true data for multiple individuals. They assumed that each database value was accurate, and they utilized the Laplace mechanism without considering the potential error in the values.

Several studies have suggested the use of machine learning methods, such as deep neural networks (deep learning), to process IoT sensing values with differential privacy. Shi et al. [62] proposed a reinforcement technique for transportation network companies that use passenger data. Xu et al. [79] concentrated on mobile data analysis in edge computing, and Guan et al. [19] applied machine learning to the Internet of Medical Things. Although these studies employed differential privacy as a privacy metric; they did not consider the proposed true-value-based differential privacy (TDP). It is posited that the application of TDP could enhance the accuracy of these methods while preserving the desired levels of privacy protection.

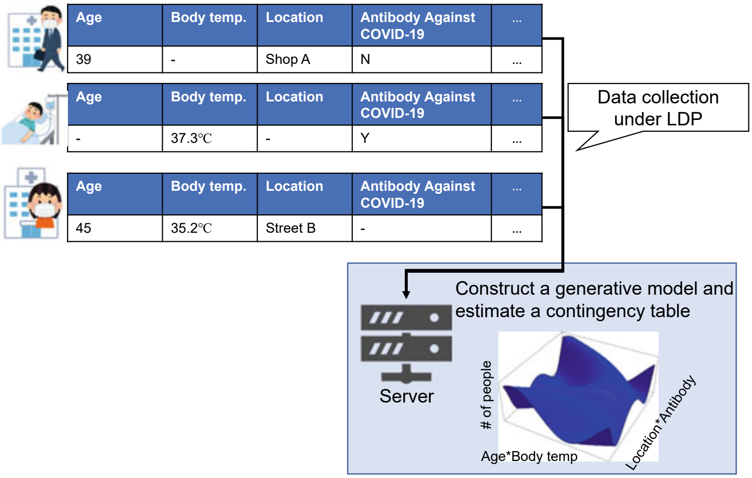


Fig. 9 Example application of missing values

4 Missing Values Under LDP

4.1 Introduction

To achieve smart cities, as previously mentioned, it is essential to collect and analyze vast amounts of personal data while ensuring the protection of privacy. Even when it is anonymized, a large amount of sensitive personal information is difficult to acquire. Moreover, this information may have missing values, as individuals are more likely to provide incomplete confidential information than to provide all their confidential information (Fig. 9).

In this section, we propose a method for privacy-preserving data collection that considers a large number of missing values. The personal data to be collected are anonymized on each person's device and/or computer in authorized entities, and are then sent to a data collection server. Each person can select which data to share or not to share. The data collection server creates a generative model and contingency table suitable for multi-attribute analysis based on the expectation–maximization and Gaussian copula methods.

We considered that if the value distribution of one or two attributes could be restored, the error in each attribute could be limited even when there are several missing values. Copula enables data generation when certain information (such as the correlation and mutual information) is available for each pair of attributes. We thus combined the features of copula with those of data recovery using differential privacy. To our knowledge, this idea is novel to privacy-preserving data collection.

Table 5 Notation

ϵ	Privacy budget for differential privacy
n	Number of participants
g	Number of attributes for data collection
A_j	jth attribute
V_j	Domain of A_j
f_j	Size of V_j
V_{jk}	kth value of V_j
s_{ij}	True attribute value of A_j of Person i
R_{ij}	Disguised attribute value of A_j of Person i
c	Number of targeted attributes for analysis (used in experiments only)
m	Missing value rate (used in experiments only)

4.2 Proposed Method

We leverage differential privacy to anonymize patient personal data on the client side. The server collects the anonymized data and reconstructs the distributions of each attribute, as well as all the combinations that include two attributes. From the two-attribute distributions, the mutual information of all attribute pairs is computed. Subsequently, a Gaussian copula [16, 51] is employed to calculate the generative model of patient personal data from the mutual information. As our proposed method needs only information about the combination of every attribute pair, it is robust to missing values. To visualize the generative model, we construct a contingency table using the generative model and the distribution of each attribute. The notation employed in this study is listed in Table 5.

In the proposed approach, the server constructs a copula model to analyze the collected differentially private data while mitigating the noise introduced by the differentially private technique. As detailed in Sect. 4.2.2, the construction of a copula model requires the value distribution of each attribute and the mutual information of all attributes. Therefore, the proposed method initially estimates the single-attribute distributions (Sect. 4.2.2) before estimating the attribute-pair distributions (Sect. 4.2.2). The generation of the copula model is described in Sect. 4.2.2. The copula model can generate an arbitrary number of data samples that do not have missing values. From these data samples, a contingency table is constructed (Sect. 4.2.2).

4.2.1 Anonymization on the Client Side

Let s_{ij} represent the value of attribute A_j of patient i. The number of attributes is g; that is, patient i has attribute values $s_{i1}, \ldots, s_{ig}$. Some values of s_{ij} may be missing. Let f_j be the number of categories of A_j.

We anonymize each non-missing value s_{ij}. Let V_j represent the domain of A_j and let V_{jk} represent the kth value of V_j. For example, assume that A_1 represents the attribute of a disease {COVID-19, flu, cancer}. In this case, $f_1 = 3$ and V_{11}, V_{12}, and V_{13} are COVID-19, flu, and cancer, respectively.

Based on a previous method [52], we create a value set R_{ij} for each attribute A_j as follows:

$$R_{ij} = \begin{cases} \{s_{ij}\} \cup Ran(V_j \backslash \{s_{ij}\}, h_j - 1) & \text{with prob.} p_j \\ Ran(V_j \backslash \{s_{ij}\}, h_j) & \text{otherwise,} \end{cases} \tag{35}$$

where $Ran(S, h)$ represents a function that randomly selects h elements without duplication from set S. For example, assume that $S = \{A, B, C\}$ and h=2. In this case, $Ran(S, h)$ outputs $\{A, B\}$, $\{B, C\}$, or $\{A, C\}$. To satisfy ϵ-differential privacy, the parameters h_j and p_j are respectively determined as

$$\begin{aligned} h_j &= \max\left(\left\lfloor \frac{f_j}{1 + e^{\epsilon}} \right\rfloor, 1\right) \quad \text{and} \\ p_j &= \frac{e^{\epsilon} h_j}{f_j - h_j + e^{\epsilon} h_j} \end{aligned} \tag{36}$$

following [52]. As there are g attributes in our scenario, each R_{ij} should satisfy ϵ/g-differential privacy [27].

Algorithm 3 is the anonymization algorithm on the client side.

The privacy budget allocated to each attribute is ϵ/g. Even if all the attributes are the same, i.e., the correlations between the attributes are all 1, we satisfy ϵ-differential privacy due to the composition property of differential privacy [27].

Algorithm 3 Anonymization algorithm for patient i

Input: Privacy parameter ϵ, original data $\{s_{i1}, \ldots, s_{ig}\}$, each domain V_j
Output: Anonymized version of $\{s_{i1}, \ldots, s_{ig}\}$

```
for j = 1, ..., g do
    f_j ← |V_j|
    Based on (36), determine p_j and h_j by substituting ε/g into ε
    Based on (35), obtain R_ij from s_ij and V_j
end for
return R_i = {R_i1, ..., R_ig}
```

4.2.2 Estimation on the Server Side

The data collection server first estimates the value distribution of each attribute as described in Sect. 4.2.2. It then estimates the value distribution of each attribute pair as described in Sect. 4.2.2. Using these estimated value distributions, the server creates a generative model (a Gaussian copula; see Sect. 4.2.2). Finally, it generates

n complete data records and creates a contingency table of target attributes, which is specified by a data analyzer (Sect. 4.2.2).

Separated estimation: estimation of a value distribution for each attribute

Each client sends its true value and $(h_j - 1)$ randomly selected values other than the true value with probability p_j, and each sends h_j randomly selected values other than the true value with probability $(1 - p_j)$ for attribute j, as represented in Algorithm 1. As a result, the probability that the true value is sent is p_j, and the probability that another value is sent is

$$q_j = \frac{p_j(h_j - 1)}{f_j - 1} + \frac{(1 - p_j)h_j}{f_j - 1} = \frac{h_j - p_j}{f_j - 1}, \tag{37}$$

as for attribute j. Here, because a total of h_j values are sent, $p_j + (f_j - 1)q_j = h_j$.

Let w_{jk} represent the number of occurrences of V_{jk} in $\{\boldsymbol{R_1}, \ldots, \boldsymbol{R_n}\}$, and let u_{jk} represent the true number of occurrences of V_{jk}. Thus, we have the following equation:

$$\begin{pmatrix} w_{j1} \\ w_{j2} \\ \vdots \\ w_{jf_j} \end{pmatrix} = M \begin{pmatrix} u_{j1} \\ u_{j2} \\ \vdots \\ u_{jf_j} \end{pmatrix}, \tag{38}$$

where M is the matrix in which the diagonal elements are p_j, and the other elements are q_j. The symbol z_{jk} represents the estimated number of occurrences of V_{jk}. We can easily estimate these values by calculating the following equation:

$$\begin{pmatrix} z_{j1} \\ z_{j2} \\ \vdots \\ z_{jf_j} \end{pmatrix} = M^{-1} \begin{pmatrix} w_{j1} \\ w_{j2} \\ \vdots \\ w_{jf_j} \end{pmatrix}, \tag{39}$$

where M^{-1} represents the inverse matrix of M. However, the estimation accuracy is very low [25]. Moreover, calculating the inverse function requires significant computational time, particularly for a large matrix. To overcome these limitations, we use the expectation-maximization (EM)-based algorithm. If we know the values of u_{jk}, we can calculate each expected value of w_{jk}. In our problem setting, we know the actual values of w_{jk}; however, we do not know u_{jk}. Therefore, with u_{jk} as an unobserved latent variable, the EM-based algorithm can provide maximum a posteriori estimation. It can find the unobserved latent variables that best explain the observed values. Moreover, the EM-based algorithm can ensure the increase in likelihood with each iteration [33, 78].

The symbol $\widetilde{n_j}$ represents the number of records that contain a value for attribute A_j:

$$\widetilde{n_j} = \sum_{k=1}^{f_j} w_{jk}. \tag{40}$$

Let z_{jk} represent the estimated number of occurrences of V_{jk} in A_j. From the expectation–maximization-based algorithm [52], we obtain z_{jk} by repeating the following substitution:

$$z_{jk} \leftarrow z_{jk}(p_j \mathcal{D}_k + q_j(\mathcal{E} - \mathcal{D}_k)), \tag{41}$$

where

$$q_j = \frac{h_j - p_j}{f_j - 1}, \tag{42}$$

$$\mathcal{D}_k = \frac{w_{jk}}{p_j z_{jk} + q_j(h_j \widetilde{n_j} - z_{jk})}, \tag{43}$$

and

$$\mathcal{E} = \sum_{k=1}^{f_j} \mathcal{D}_k. \tag{44}$$

Separated estimation: estimation of a value distribution for every two-attribute combination

Let $V_{jj'}$ be a combination of the elements of attributes A_j and $A_{j'}$:

$$V_{jj'} = V_j \times V_{j'}. \tag{45}$$

Let $w_{jj'kk'}$ represent the number of simultaneous occurrences of V_{jk} and $V_{j'k'}$ in each record in $\{\boldsymbol{R_1}, \ldots, \boldsymbol{R_n}\}$. The symbol $\widetilde{n_{jj'}}$ represents the number of records in which a value exists for both attributes A_j and $A_{j'}$:

$$\widetilde{n_{jj'}} = \sum_{k=1}^{f_j} \sum_{k'=1}^{f_{j'}} w_{jj'kk'}. \tag{46}$$

As an example, assume that Table 6 was created by the privacy-preserving data collection. The values of $\widetilde{n_1}$, $\widetilde{n_2}$, and $\widetilde{n_3}$ are 4, 2, and 3, respectively, because attribute A_1 has four values, attribute A_2 has two values, and attribute A_3 has three values. The value of $\widetilde{n_{1,2}}$ is 2 because two records (the first and fourth records) contain values for both A_1 and A_2 (the values are [39, 40, 58, 35.2, 35.5] and [33, 34, 88, 37.5, 37.6]). Similarly, the values of $\widetilde{n_{1,3}}$ and $\widetilde{n_{2,3}}$ are 3 and 1, respectively.

Table 6 Example table created by privacy-preserving data collection

Record ID	Age (A_1) (years)	Body temp. (A_2) (°C)	Location (A_3)
1	{39, 40, 58}	{35.2, 35.5}	–
2	{12, 22, 30}	–	{Shop A, Hospital D}
3	{25, 40, 61}	–	{Street B, Hospital D}
4	{33, 34, 88}	{37.5, 37.6}	{School C, Shop E}

As in Sect. 4.2.2, we estimate the occurrence of each combination V_{jk} and $V_{j'k'}$ of attributes A_j and $A_{j'}$ for n patients. By calculating these values for all combinations A_j and $A_{j'}$, we can estimate all value distributions of all attribute pairs.

After estimating the attribute-pair distribution, the mutual information of attributes j and j' is calculated as follows:

$$\sum_{k \in V_j} \sum_{k' \in V_{j'}} p(k, k') \log \frac{p(k, k')}{p(k)p(k')}, \tag{47}$$

where $p(k, k')$ represents the joint probability that V_{jk} and $V_{j'k'}$ occur, and $p(k)$ represents the probability that V_{jk} occurs.

Generative model construction: constructing a generative model as the Gaussian copula

Let $X_1, \ldots, X_g$ be random variables, and let $F(x_1, \ldots, x_g)$ represent the joint probability distribution function of $X_1, \ldots, X_g$. The marginal distribution functions $F_1, \ldots, F_g$ and the joint probability distribution function have the following relationship.

Theorem 4 (Sklar's Theorem[63]) *A function C uniquely satisfies the following expression:*

$$\begin{aligned} F(x_1, \ldots, x_g) &= Pr(X_1 \leq x_1, \ldots, X_g \leq x_g) \\ &= C(F_1(x_1), \ldots, F_g(x_g)). \end{aligned} \tag{48}$$

From Sklar's theorem, we have:

$$C(u_1, \ldots, u_g) = F(F_1^{-1}(u_1), \ldots, F_g^{-1}(u_g)), \tag{49}$$

for arbitrary $\boldsymbol{u} = (u_1, \ldots, u_g)$ where $(u_i \in [0, 1])$. Based on Sklar's Theorem, we have:

$$\begin{aligned} \Phi_g(x_1, \ldots, x_g; \Sigma) &= Pr(X_1 \leq x_1, \ldots, X_g \leq x_g) \\ &= C(\Phi(x_1), \ldots, \Phi(x_g)), \end{aligned} \tag{50}$$

where $\Phi(\cdot)$ represents the cumulative distribution function of a standard Gaussian distribution, and $\Phi_g(x_1, \ldots, x_g; \Sigma)$ represents the cumulative distribution function of a g-dimensional Gaussian distribution with random variables $X_1, \ldots, X_g$ and a covariance matrix Σ.

From (50), the cumulative distribution of the Gaussian copula can be expressed as

$$C(u_1, \ldots, u_g) = \Phi_g(\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_g); \Sigma). \tag{51}$$

The Gaussian copula C represents the cumulative distribution function of each marginal distribution, which is a uniform distribution in the range [0, 1]. The probability density function of the Gaussian copula $c(u_1, \ldots, u_g; \Sigma)$ satisfies the following relationship:

$$\phi(x_1, \ldots, x_g) = c(\Phi(x_1), \ldots, \Phi(x_g)) \prod_{j=1}^{g} \phi(x_j), \tag{52}$$

where $\phi(\cdot)$ represents the probability density function of a standard Gaussian distribution, i.e.,

$$\phi(x_1, \ldots, x_g) = \frac{1}{\sqrt{(2\pi)^g |\Sigma|}} \exp(-\frac{1}{2} x^T \Sigma^{-1} x). \tag{53}$$

Therefore, we have

$$c(u_1, \ldots, u_g) = \frac{1}{\sqrt{|\Sigma|}} \exp\,(-\frac{1}{2} \boldsymbol{\omega}^T (\Sigma^{-1} - \boldsymbol{I}) \boldsymbol{\omega}), \tag{54}$$

where $\boldsymbol{\omega} = \Phi^{-1}(\boldsymbol{u})$.

Σ must be estimated from the collected data. Let $\boldsymbol{u}^i$ and $\boldsymbol{\omega}^i$ represent the ith $\boldsymbol{u}$ and ith $\boldsymbol{\omega}$, respectively. Then, from (54), the log-likelihood function of the Gaussian copula is given by

$$l(\Sigma) = -\frac{n}{2} \ln |\Sigma| - \frac{1}{2} \sum_{i=1}^{n} \boldsymbol{\omega}^{i^T} (\Sigma^{-1} - \boldsymbol{I}) \boldsymbol{\omega}^i, \tag{55}$$

where $\boldsymbol{\omega}^j = \Phi^{-1}(\boldsymbol{u}^j)$. Differentiating (55) with respect to Σ^{-1}, we obtain [69]

$$\frac{\partial l(\Sigma)}{\partial \Sigma^{-1}} = \frac{n}{2} \Sigma - \frac{1}{2} \sum_{i=1}^{n} \boldsymbol{\omega}^i \boldsymbol{\omega}^{i^T}. \tag{56}$$

Therefore, the maximum likelihood estimator $\widehat{\Sigma}$ is

$$\widehat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\omega}^i \boldsymbol{\omega}^{i^T}. \tag{57}$$

To alleviate the high computational cost of (57), we estimate Σ using a suboptimal approach [51]. First, we calculate the mutual information of every pair of attributes using the reconstructed data in Sect. 4.2.2. We then determine each suboptimal element of Σ that minimizes the distance between the mutual information of the estimated joint distribution and that calculated from the reconstructed data (see Sect. 4.2.2).

Contingency table construction: generation of records based on the generative model

We generated n complete data from the Gaussian copula C and the reconstructed data in Sect. 4.2.2. The n values of each attribute A_j were determined based on the estimated attribute distribution in Sect. 4.2.2. We also generated random values $\bar{x_1}, \ldots, \bar{x_g}$ based on a g-dimensional Gaussian distribution with covariance matrix $\widehat{\Sigma}$. We then obtained $u_i = \Phi(x_j)$ for all $i = 1, \ldots, g$. From the reconstructed data in Sect. 4.2.2, we finally obtained $F_j^{-1}(u_j)$ for each attribute value, where F_j represents the marginal distribution of attribute A_j.

Contingency table construction: counting each combination of target attributes

After the above process, we obtained n complete data records with g attributes. If a contingency table is used for many attributes, it loses its primary value [18, 77]. Therefore, data analyzers generally select several attributes. The target contingency table is then constructed by simply counting the occurrences of each combination of attribute values from the n generated complete data records.

4.3 Evaluation

4.3.1 Evaluation Setting

We compared the performances of the proposed method and four state-of-the-art methods: O-RAPPOR [25], S2Mb [52], MDN [17], and PDE/ETE (the baseline approach).

The experimental results for the simple combination of the differentially private technique on the client side and the copula technique on the server side are also shown. This method is referred to as DF+Copula.

If the estimated contingency table generated by one of the methods was similar to that generated from the valid data, which was unknown to the data collection server, then the estimated contingency table was considered to be well-generated by the model.

In this study, a contingency table is considered to be a probability distribution of attribute values. To measure the difference between the probability distributions, we applied Jensen–Shannon (JS) divergence rather than the usual Kullback–Leibler (KL) divergence, because the KL divergence assumes all probabilities are non-zero. If any probabilities are zero, the KL divergence fails due to a division-by-zero error. The JS divergence is based on the KL divergence but does not impose the non-zero constraint.

In the Apple implementation, ϵ equals 1 or 2 per datum [66]. In the evaluations by the Apple differential privacy team, ϵ was set to 2, 4, and 8 [3]. Microsoft described their differentially private framework, and according to their paper, they set ϵ between 0.1 and 10 [9]. In the paper that proposed RAPPOR [13], which was developed by Google, $\epsilon = \log(3)$ was used as the main setting. Hsu showed that, in the literature, ϵ ranges from 0.01 to 10 [22]. Based on the settings reported in the literature, we set the value of ϵ between 0.01 and 10.

We varied the missing value rate m from 0.3 to 0.8, and we varied the number of attributes c in the analysis from 1 to 5. The reported results are the averages of 100 experiments for each parameter setting. For the default parameters, we set $m = 0.5$, $c = 3$, and $\epsilon = 5$.

Note that the missing value rate m is used only for the experiments, and the proposed algorithm does not require this information. The number of targeted attributes for analysis c can be freely determined by the data analyst according to the purpose of the analysis.

4.3.2 Experiments on Real Data

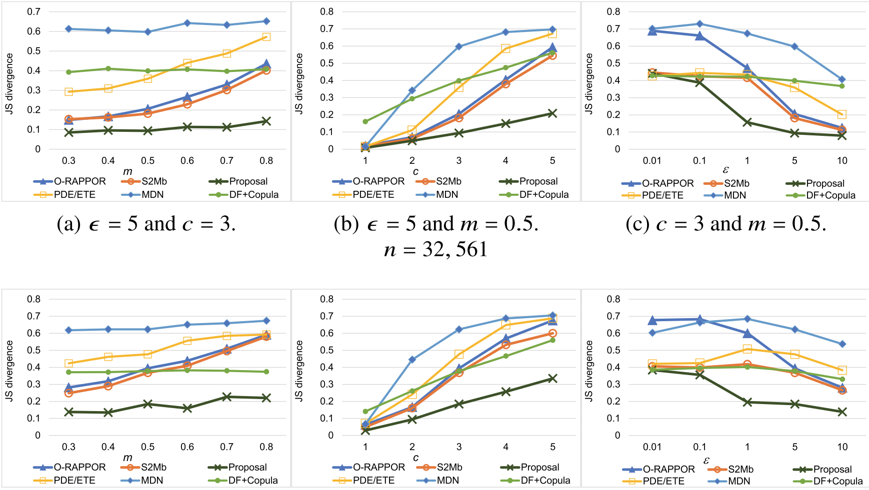
In the real-data experiments, we first investigated the Adult dataset [10], which is widely used in evaluations of privacy-preserving data mining techniques (for example, see [14, 24, 64]). The Adult dataset consists of 15 attributes (e.g., age, income) in 32,561 records. The number of categories in our experiments was set between 2 and 9 per attribute.

Figure 10a–c present the experimental results.

When the missing value rate was small or ϵ was large, the JS divergence of the proposed method was similar to the JS divergences of S2Mb, PDE/ETE, and O-RAPPOR. Similarly, when ϵ was small, the JS divergence of the proposed method was similar to the JS divergences of S2Mb, PDE/ETE, and DF+Copula.

However, at high rates of missing values, the proposed method outperformed the other methods, achieving a high level of privacy protection.

To determine whether the proposed method is applicable to small datasets, we randomly sampled 10% of the 32,561 records in the Adult dataset and measured their JS divergence. Figure 10d–f present the results. Owing to the data sparsity, this estimation task was more difficult than in the other experiments, and the JS divergence for all methods was higher for the 3,256 records than for the 32,561 records. However, the proposed method was robust to the small dataset. On a larger dataset with an insignificant missing value rate, the JS divergence was higher for



(a) $\epsilon = 5$ and $c = 3$. (b) $\epsilon = 5$ and $m = 0.5$. (c) $c = 3$ and $m = 0.5$.

$n = 32,561$

(d) $\epsilon = 5$ and $c = 3$. (e) $\epsilon = 5$ and $m = 0.5$. (f) $c = 3$ and $m = 0.5$.

$n = 3,256$

Fig. 10 Results for the adult dataset

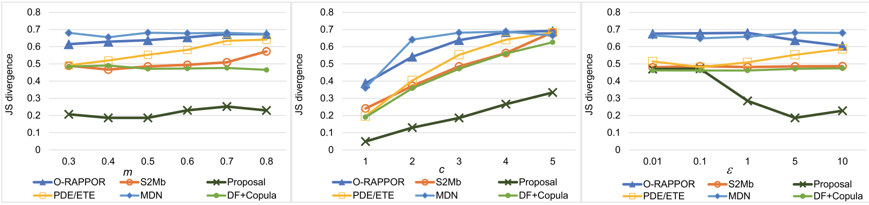
the proposed method than that for the existing methods. Therefore, regardless of the missing value rate, the proposed method outperformed the other methods on smaller datasets.

We then used the Communities and Crime Unnormalized dataset [1] (hereafter referred to as the Community dataset). This dataset contains 124 predictive attributes, such as the percentage of individuals aged 25 and over with a bachelor's or higher degree, which could be considered private information in some communities.

After removing 22 attributes that had more than 80% missing values, we retained 102 attributes for analysis.

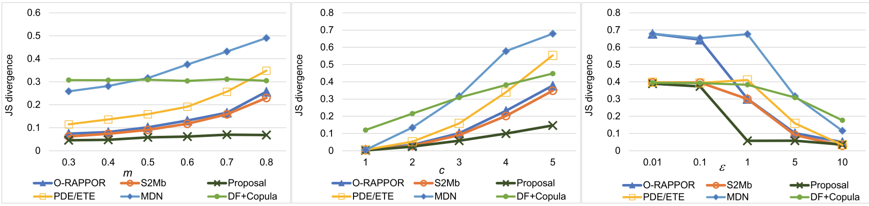
Figure 11 presents the experimental results for the Community dataset. The results are similar to those of the Adult dataset. For almost all parameter settings, the proposed method outperformed the other methods. As the number of participants n was smaller than in the previous experiments, increasing the missing value rate increased the JS divergence of the proposed method. However, the increase in JS divergence was not considerable.

We next used a default dataset containing 21,985 records with the following attributes: sex, job, income, number of loans from other companies, number of delayed payments, and a default flag (0 or 1). Here, the word default means that a debtor failed to pay off a loan. The results of this dataset, which was generated from authentic default data, are plotted in Fig. 12. As shown in Fig. 12a, the proposed method accurately reconstructed the contingency tables even when the missing value ratio (m) increased to 0.8. On the contrary, the accuracies of the existing methods greatly decreased as the missing value ratio increased. Increasing the number of attributes used for generating contingency tables (c) also increased the reconstructed



(a) $\epsilon = 5$ and $c = 3$. (b) $\epsilon = 5$ and $m = 0.5$. (c) $c = 3$ and $m = 0.5$.

Fig. 11 Results for the community and crime unnormalized datasets



(a) $\epsilon = 5$ and $c = 3$. (b) $\epsilon = 5$ and $m = 0.5$. (c) $c = 3$ and $m = 0.5$.

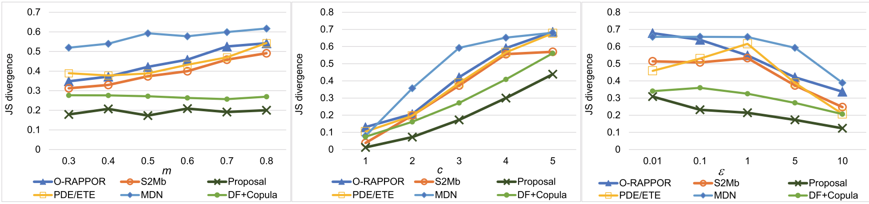
Fig. 12 Results for the default dataset

error (Fig. 12b). However, the proposed method was more resistant to an increasing c than were the other methods. Figure 12c shows the effect of ϵ on the reconstruction error in the five methods. When ϵ was sufficiently large, the accuracies of all methods were very similar, but when ϵ was small, the reconstructed error of the proposed method was clearly the lowest.

Finally, we applied a dataset related to the 2019 coronavirus disease (COVID-19) called Patient Medical Data for Novel Coronavirus COVID-19[3]. Hereafter, we refer to this dataset as the COVID-19 dataset. This dataset contains 427,036 records with 23 attributes. More than 90% of the values are missing for 12 of the attributes, and approximately 27% are missing even for basic attributes like age and sex. From the COVID-19 dataset, we extracted the Japanese medical data and analyzed the attributes that had few missing values (namely age, sex, administrative division, date of confirmation, and chronic disease status). The date of confirmation was categorized by month, and the number of categories in each attribute ranged from 2 to 29.

Figure 13 presents the results for the COVID-19 dataset. Under all parameter settings, the JS divergence was lower for the proposed method than for the other methods. As the rate of missing values in the original COVID-19 dataset was 68.7%, we concluded that the proposed method effectively processes real datasets with missing values.

[3] https://datarepository.wolframcloud.com/resources/Patient-Medical-Data-for-Novel-Coronavirus-COVID-19/ (accessed June 20, 2020).



(a) $\epsilon = 5$ and $c = 3$. (b) $\epsilon = 5$ and $m = 0.5$. (c) $c = 3$ and $m = 0.5$.

Fig. 13 Results of the patient medical data for novel coronavirus COVID-19 dataset

5 Human-to-Human Interactions Under LDP

5.1 Introduction

Smart cities aim to create efficient, sustainable, and livable urban environments by leveraging technology and data. In this context, data related to human-human interactions is pivotal for several reasons. For example, understanding the nature and frequency of human interactions can offer insights into community dynamics. Such data can inform city planners and local authorities about where community hubs or gathering spots might be needed, or where interventions to boost community interaction might be beneficial. Moreover, data on how and where people meet and interact can provide valuable insights into the design and placement of public spaces, transportation nodes, and amenities. For example, if a certain public square sees frequent human interaction, it might be worth investing in better seating, shading, or even establishing transit connections to that area.

Although LDP is considered to be the best technology for privacy protection [6, 60], these organizations apply additional explicit privacy policies for data collection. For example, Apple collects data from users regarding the users' emoji usage through LDP; however, it does not collect the users' identities.

In LDP, each user is assigned a privacy budget, which is a non-negative real value. When the user data are sent to the data collector, a portion(or the entirety) of the privacy budget for the user is consumed. The total privacy budget and the consumed amount of the privacy budget can be controlled through an agreement between the data collector and user. For example, suppose that the privacy budget for user A is 10.0, and the value of the privacy budget consumed by transmitting the data of this user is 1.0; the data collector can retrieve user A's data 10 times. To ensure continuous data collection, the total privacy budget for each user is regularly restored.

If the data collected by the data collector refer to a user's information regardless of other users, there are no issues because the user has already agreed to the privacy policy. However, what happens if the data collected concerns a person-to-person interaction? Suppose that user A sends an email to user B, and the data collector gathers information about word usage through LDP under a privacy policy agreed

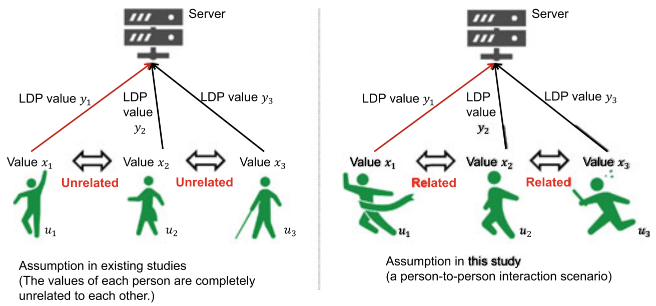


Fig. 14 Assumptions of previous studies and this study

upon with user A. The data collected are about the words used by user A, but for user B, the data are about the words they received. In other words, it is equivalent to collecting user B's data. Therefore, the data collector must also consider user B's privacy. However, whether user B has agreed to the privacy policy is not checked at present. Even if user B has agreed to the policy, no one has control over user B's privacy budget.

Figure 14 shows the difference between the assumptions used in previous studies and this study. According to previous studies, when user u_1 sends the LDP value y_1 of their true value x_1 to the data collection server, only u_1's private information is provided to the data collection server. This is because the values of each user are completely unrelated to each other. Moreover, suppose that each value depends on another value. In this case, when user u_1 sends LDP value y_1, information about users u_1, u_2, and u_3 is also provided to the data collection server. In other words, although u_2 does not send any information to the data collection server, through the behavior of u_1, some of u_2's information is provided to the data collection server.

In this study, this problem was formalized as a person-to-person interaction in LDP. To focus the discussion in this section on the new concept of person-to-person interaction under ϵ-LDP, we targeted the relatively simple task of obtaining average values from users. The recommendations in this section are expected to have a considerable impact on organizations that collect person-to-person interaction data using ϵ-LDP.

5.2 *Related Work and Real Applications*

5.2.1 Related Work on LDP

Many methods have been proposed for estimating a histogram distribution of users' values under ϵ-LDP, such as the Randomized Aggregatable Privacy-Preserving Ordinal Response, *Sarve*, and so on [13, 71]. Although such methods achieve high accuracy, their techniques cannot be applied to a person-to-person interaction scenario.

This is because they assume that each user's value is not dependent on any other user's value.

There have also been several methods proposed for estimating the average value of users. Xue et al. proposed (τ, ϵ)-personalized LDP (PLDP) as a privacy metric, Duchi's solution with PLDP (DCP), and piecewise mechanism with PLDP (PWP) [80]. The (τ, ϵ)-PLDP is a privacy metric that weakens ϵ-LDP, but DCP and PWP can be used for ϵ-LDP. We can assume that the range of a value is $[-1, 1]$ without loss of generality. In DCP, each user sends a randomized value v with a probability

$$Pr(\epsilon, x) = \frac{(e^{\epsilon} - 1) \cdot x}{(e^{\epsilon} + 1) \cdot 2} + \frac{1}{2}. \tag{58}$$

In PWP, each user randomly selects a value from a range around the true value with probability p, where the value of p is determined from ϵ. A value from a wider range is randomly selected with probability $1 - p$, and the selected value is sent to the server. Because the ratio of $p/(1 - p)$ is e^{ϵ}, PWP ensures ϵ-LDP.

Li et al. proposed the square wave mechanism (SW) [35]. This mechanism is similar to PWP, but the range of LDP values to be selected is different.

Many other LDP methods have been proposed. Navidan et al. proposed a framework that estimates the number of people in each area while protecting each user's location privacy using LDP [40]. In this framework, users measure the Received Signal Strength Indicator (RSSI) and determine their locations based on the RSSI. The users then perturb their location information and send it to the data aggregator, who estimates how many users are in each location. The experimental results showed that the proposed framework could estimate location frequency while ensuring differential privacy.

Kim and Jang [29] proposed a data collection method for workload-aware differentially private positioning. They assumed that location is hierarchical and aimed to estimate the density at each location for each level of the hierarchy by utilizing LDP. Their method provides an optimal perturbation scheme to minimize the estimation error for a given workload.

Although many studies target one-shot data-sharing scenarios, several studies have considered cases of data streaming. Please note that our proposed method can be used for data streaming cases by dividing the privacy budget by the number of data acquisitions. By using methods for specified data stream cases, the accuracy of the data analysis can be enhanced. For example, Ren et al. [48] proposed an LDP mechanism for an infinite data stream that targets w-event privacy, which ensures LDP for arbitrary time windows consisting of w consecutive time steps. In the future, we will propose a specialized method for measuring time series data.

Ren et al. [49] proposed an anonymous data aggregation scheme that allows the server to estimate the number of users located within each value area without knowing the location of individual users. In particular, the authors focus on high-dimensional values. The domain sizes of the datasets used in the experiments in [49] were 2^{16}, 2^{52}, and 2^{77}. Experiments with such high-dimensional datasets should be conducted in the future to test our proposed method.

These studies are excellent, but they do not take user interaction into account.

In recent years, studies on federated learning with LDP have gained attention [7, 23, 82]. In a typical federated learning scenario using LDP, the server sends to the clients the machine-learning model parameters that are to be trained. Each client independently trains the machine-learning model using private local data samples. The updated gradient information is sent to the server under the protection of LDP. If each private local data sample is completely unrelated to the private local data samples of other users, ϵ-LDP can be ensured in these studies. However, for the person-to-person interaction data envisioned in the current study, when one user sends information to a server through LDP, loss of privacy of other users must be considered as well.

The extant studies on LDP [6, 13, 73] assume that one user's value is independent of that of any other user. In many cases, this assumption is correct. However, in some scenarios, this assumption does not hold, as discussed in Sect. 5.2.2.

Example 1
Alice transferred \$50 to Bob on a single day. Alice has agreed to a 10-LDP (i.e., the amount of the privacy budget is 10), which allows a data collector to gather the amount per day transferred by Alice. Based on this policy, Alice sends the LDP value (e.g., \$53) to the data collector, which consumes a 10-privacy budget. Because Alice's identity is not sent to the data collector, the data collector only knows that someone transferred \$53 on that day.

In the above example, the information that is sent is related to Alice's money transfer. However, for Bob, the information sent is related to Bob's receipt of money. In this case, Alice's 10-privacy budget and Bob's 10-privacy budget are consumed. Therefore, if Bob's transmission information is also collected, the total amount of privacy budget consumed will be 20, which surpasses the upper limit of 10. Such problems occur in person-to-person interactions in LDP.

5.2.2 Application of LDP Under Person-to-Person Interactions

Recently, LDP has been widely applied to many real services. Apple collects pictogram usage information from users under LDP to analyze the use frequency of each pictogram [8]. However, Apple does not seem to care about the receiver's privacy.

Several email datasets contain anonymized text information and pseudo personal, sender, and receiver IDs [34]. Such data can be collected under LDP from each user. Emails are generally considered personal data that must be handled with care, regardless of the data that are sent or received. Therefore, if the email information of a sender is collected under LDP, this collection should consume the privacy budget of not only the sender but also the receiver.

Human relationship information, such as that from online social networks, is another form of privacy information. There are several anonymized datasets on human relationships, such as Epinions social network [50]. If the data collector gathers information about who a user is connected to and trusts, the privacy budget of not only the user but also the other person is consumed.

5.3 Problem Definition

We have defined the problem of LDP for person-to-person interactions. This scenario was not assumed in previous studies, but it occurs in real-world situations. One of the most important contributions of this work is to clarify this problem. Numerous forms of person-to-person interactions are possible, but to simplify the discussion in this section, we limit our analysis to the following interactions.

Definition 4 (ϵ-LDP in a person-to-person interaction scenario) Let X_i represent the domain of user u_i's data, and let $X_{i,j}$ represent the domain of the interaction data between two users u_i and u_j $(i, j = 1, \ldots, n\ (i \neq j))$. The value of $x_i \in X_i$ is obtained from $x_{i,j} \in X_{i,j}$ for all j except for $i = j$; i.e., $x_i = f(x_{i,1}, \ldots, x_{i,i-1}, x_{i,i+1}, \ldots, x_{i,n})$ for a function $f : X_{i,j}^{n-1} \to X_i$.

User u_i sends information x_i under ϵ-LDP using mechanism M, which is defined in Definition 1.

Theorem 5 (Consumed privacy budget of ϵ-LDP for person-to-person interactions) *In a scenario of ϵ-LDP for person-to-person interactions, the consumed privacy budget of user u_i is ϵ. The privacy budget of user u_j is also consumed, and this amount is represented by*

$$\begin{aligned} &\min \epsilon_j, s.t.\ \ P(M(f(x_{i,1}, \ldots, x_{i,n})) = y) \leq \\ &e^{\epsilon_j} P(M(f(\ldots, x_{i,j-1}, x'_{i,j}, x_{i,j+1}, \ldots)) = y), \end{aligned} \tag{59}$$

for any $x_{i,j}, x'_{i,j} \in X_{i,j}$.

Proof For user u_i, the consumed privacy budget is ϵ because x_i is collected under ϵ-LDP.

For user u_j $(j \neq i)$, the following expression should be satisfied for any $x_{i,j}, x'_{i,j} \in X_{i,j}$ to ensure ϵ_j-LDP because of Definition 1.

$$\begin{aligned} &P(M(f(x_{i,1}, \ldots, x_{i,n})) = y) \leq \\ &e^{\epsilon_j} P(M(f(x_{i,1}, \ldots, x_{i,j-1}, x'_{i,j}, x_{i,j+1}, \ldots, x_{i,n})) = y). \end{aligned} \tag{60}$$

The smaller the value of ϵ_j, the smaller the amount of privacy budget consumed and the more robustly the privacy is protected. Therefore, the consumed privacy budget is the minimum value that satisfies (60). □

The problem definition in this section is as follows.

Problem 1 (Obtaining the average value under ϵ-LDP in a person-to-person interaction scenario) Assume there are n users $(u_1, \ldots, u_n)$, and each privacy budget is set to ϵ_i. In a person-to-person interaction scenario, the average value of $x_1, \ldots, x_n$ is obtained with high accuracy while ensuring ϵ_i-LDP for each user u_i.

Note that we do not propose a new privacy metric, but we strictly follow ϵ-LDP. The difference between the objective in this section and that of previous studies is whether or not each user's data contain information about other users, which should be protected. To simplify the discussion, the goal of this analysis is to obtain the average value of all users' data. However, the concept of ϵ-LDP in a person-to-person interaction scenario can be applied to any other analysis, such as histogram estimation or machine learning. Such analysis remains to be undertaken in future work.

5.4 *Proposed Method*

The main notation used in this study is listed in Table 7. We mainly used a Laplace mechanism. The global sensitivity of each user should be clarified when this mechanism is used.

Definition 5 (Global sensitivity for a person-to-person interaction)) For user u_i, the global sensitivity is the same as that given in Definition 2. For user u_j $(j \neq i)$, the global sensitivity of f is defined as

$$\Delta f_{i,j} = \max_{x_{i,j}, x'_{i,j} \in X_{i,j}} |f(\ldots, x_{i,j}, \ldots) - f(\ldots, x'_{i,j}, \ldots)|. \tag{61}$$

Theorem 6 (Consumed privacy budget of the Laplace mechanism in a person-to-person interaction) *Suppose user u_i sends the value of x_i to the data collector*

Table 7 Notation

$X_{i,j}$	Domain of interaction data between u_i and u_j
X_i	Domain of u_i's data
$x_{i,j}$	True value of the interaction between u_i and u_j
x_i	True value of u_i, obtained from $x_{i,j} \in X_{i,j}$ for all j (except for $i = j$)
$\Delta f_{i,j}$	Global sensitivity of $x_{i,j}$
Δf_i	Global sensitivity of x_i
$\epsilon_{i,j}$	Privacy budget for $x_{i,j}$
ϵ_i	Privacy budget for x_i
$\mathcal{L}(v)$	Function of Laplace distribution, with a mean of zero and scale parameter v

under ϵ_i-LDP using a Laplace mechanism. Let Δf_i represent the global sensitivity of x_i and let $\Delta f_{i,j}$ represent the global sensitivity of $x_{i,j}$. In this case, ϵ_i of the privacy budget of user u_i and $\epsilon_i \Delta f_{i,j}/\Delta f_i$ of the privacy budget of everyone else, $p_{i,j}$, are consumed.

Proof For x_i, this mechanism ensures ϵ_i-LDP according to Equation (1).

For $x_{i,j}$, the global sensitivity is $\Delta_{i,j}$. The value sent to the server is represented by

$$f(x_i) + \mathcal{L}\left(\frac{\Delta f_i}{\epsilon_i}\right) = f(x_i) + \mathcal{L}\left(\frac{\Delta f_{i,j}}{\epsilon_i \Delta f_{i,j}/\Delta f_i}\right). \tag{62}$$

Therefore, this mechanism ensures $(\epsilon_i \Delta f_{i,j}/\Delta f_i)$-LDP for $x_{i,j}$. □

Example 2

Consider that users u_1, u_2, and u_3 are giving money to each other. The maximum amount of money given is limited to \$100. Therefore, $\Delta f_1 = \Delta f_2 = \Delta f_3 = 100$. User u_1 gives \$10 and \$20 to u_2 and u_3, respectively. User u_2 gives \$30 to u_1. User u_3 gives \$40 and \$50 to u_1 and u_2, respectively. User u_1 sends information about how many dollars u_1 gave, on average, to the server. In this case, $x_1 = f(x_{1,2}, x_{1,3}) = 15$, where function f is a function that calculates the average. In this example, $\Delta f_{i,j}$=50, because a change in $x_{i,j}$ can affect the value of x_i by up to 50.

When user u_1 sends a value less than 1-LDP to the server, i.e., the result of 15+$\mathcal{L}$(100/1) is sent to the server, this behavior consumes 1, 0.5, and 0.5 of the privacy budget of users $u_1, u_2, and u_3$, respectively.

Thus far, we have assumed that only one user, (u_i), sends their LDP value to the data collection server. When several users send their LDP values, the composition of the interaction should be considered.

Theorem 7 (Interaction-composition property of LDP in person-to-person interactions) *Suppose that the private information of $u_i (i = 1, \ldots, n)$ is collected under ϵ_i-LDP by the data collection server. Let $\epsilon_{i,j}$ represent the amount of the privacy budget of user u_j that is consumed by the collection of u_i's private information. In this case, the total privacy budget consumed for u_i is represented by*

$$\widehat{\epsilon_i} = \epsilon_i + \sum_{j \neq i} \epsilon_{j,i} \Delta f_{j,i}/\Delta f_j. \tag{63}$$

Proof The information related to u_i is represented by

$$\begin{cases} x_{i,1}, \ldots, x_{i,i-1}, x_{i,i+1}, \ldots, x_{i,n} \\ x_{1,i}, \ldots, x_{i-1,i}, x_{i+1,i}, \ldots, x_{n,i}. \end{cases} \tag{64}$$

The value of x_i is calculated based on the top line of (64); that is, $x_i = f(x_{i,1}, \ldots, x_{i,i-1}, x_{i,i+1}, \ldots, x_{i,n})$. This value is sent to the server under the privacy budget ϵ_i.

Each value $x_{j,i}$ in the lower part of (64) is sent by user u_j under the privacy budget $\epsilon_{j,i}$ for $x_{j,i}$. Because of the sequential composition property of differential privacy [12], the total privacy loss is calculated using (63). □

Example 3
Consider the same case described in Example 2. The values of $x_1, x_2, and x_3$ are 15, 30, and 45, respectively. Consider people u_1, u_2, and u_3 sending their values $x_1, x_2, and x_3$ under 1-LDP, 2-LDP, and 3-LDP, respectively. In this case, in each report by user u_i, the total privacy losses of u_1, u_2, and u_3 are (1+2/2+3/2 = 3.5), (2+1/2+3/2 = 4), and (3+1/2+2/2 = 4.5), respectively.

Thus far, we have discussed generalized scenarios where the global sensitivity and privacy budget are different for each user. Usually, however, these values are common for all users. In this case, the following theorem holds.

Theorem 8 *Consider that there are n users, and each user u_i sends x_i under ϵ-LDP. In this case, each transfer of data by u_i consumes $\epsilon/(n-1)$ of the privacy budget of another user. The total privacy loss of each user u_i is represented by $\epsilon + \sum_{j \neq i} \epsilon/(n-1) = 2\epsilon$.*

In the following text, the expected amount of error of the estimated mean under ϵ-LDP in a person-to-person interaction is discussed. We assume that the privacy budget of each user is ϵ, and that the global sensitivity for each is Δf. Let $\mathcal{L}(x; s)$ represent the probability density function (PDF) of the Laplace distribution with mean 0 and scale parameter s. The probability distribution of the sum of n Laplace random variables is represented by the following equation.

$$\begin{aligned} \mathcal{L}_n(x; s) &= \int_{x_1=-\infty}^{\infty} \cdots \int_{x_{n-1}=-\infty}^{\infty} \\ &\mathcal{L}(x_1; s) \cdots \mathcal{L}(x_{n-1}; s) \mathcal{L}(x - \sum_{i=1}^{n-1} x_i; s) dx_1 \cdots dx_{n-1} \\ &= \frac{e^{-\frac{|x|}{s}} \sum_{i=0}^{n-1} a_{n,i} s^i |x|^{n-i-1}}{2\, s^n \prod_{i=1}^{n-1} 2i}, \end{aligned} \tag{65}$$

where

$$a_{n,i} = \begin{cases} 0 & (n = i \text{ or } i = -1) \\ 1 & (n = 1 \text{ and } i = 0) \\ a_{n-1,i-1}(n + i - 2) + a_{n-1,i} & (\text{otherwise.}) \end{cases}$$

The resulting value represents the PDF of the summed noise. The expected absolute value of (65) is calculated as

$$E[|x|\mathcal{L}_k(x;s)] = 2\int_{x=0}^{\infty} x\mathcal{L}_k(x;s)dx = s\prod_{i=1}^{n-1}\frac{2i+1}{2i}. \quad (66)$$

The value of (66) represents the expected magnitude of error compared with the true value. The expected magnitude of error is then adjusted based on the desired value. For example, if the server wants to calculate the final average value, the expected magnitude of error is the value of (66) divided by n. When the target mean absolute error (MAE) of the expected average value is θ, the value of s should be

$$s = n \cdot \frac{\theta\sqrt{\pi}\Gamma(n)}{2\Gamma(1/2+n)}. \quad (67)$$

The expected squared error is calculated using

$$E[x^2\mathcal{L}_n(x;s)] = 2ns^2. \quad (68)$$

If the server wants to calculate the final average value, the value of (68) is divided by n^2. When the target mean squared error (MSE) of the expected average value is θ', the value of s should be

$$s = n\sqrt{\frac{\theta'}{2n}} = \sqrt{\frac{n\theta'}{2}}. \quad (69)$$

Algorithm 4 describes our proposed method.

Algorithm 4 Collection and analysis of LDP data in a person-to-person interaction

Input: Δf, target MAE θ or target privacy budget ϵ
Output: Expected average value

1: /*Process of the data collection server*/
2: **if** target MAE is set **then**
3: $s \leftarrow \frac{\theta\sqrt{\pi}\Gamma(n)n}{2\Gamma(1/2+n)}$
4: $\epsilon' \leftarrow \frac{2\Delta f}{s}$
5: **else if** target privacy budget is set **then**
6: $\epsilon' \leftarrow \frac{\epsilon}{2}$
7: **end if**
8: Send ϵ' to the users.
9: /*Process of each user u_i*/
10: $x_i' \leftarrow x_i + \mathcal{L}(\frac{\Delta f}{\epsilon'})$
11: Send x_i' to the data collection server.
12: /*Process of the data collection server*/
13: $v \leftarrow \frac{1}{n}\sum_{i=1}^{n} x_i'$
14: Return v.

If a target MSE is desired, Line 3 in Algorithm 4 is replaced by $s \leftarrow \sqrt{n\theta'/2}$.

Table 8 Real datasets

Database name	Num. of users	Num. of interactions	Min value	Max Value
E-mail dataset [5]	19,753	517,401	0	852
Who-trusts-whom network dataset [50]	75,879	811,480	1	3,044
Village dataset [44]	86	102,293	0	5,398
SFHH dataset [20]	405	70,261	0	2,120

5.5 *Evaluation*

5.5.1 Datasets

Initially, we created synthetic datasets that followed normal, uniform, and delta distributions with values ranging from 0 to 100.

In addition, we assessed four real datasets. The first dataset was an email dataset [34]. The first dataset was an email dataset [5]. We examined the sender, recipient, and content of each email in the dataset, and we identified 19,753 unique email addresses. Furthermore, we tallied the number of swear words used by each user. We sourced the list of swear words from https://www.noswearing.com/, which has been utilized in numerous studies (e.g., [4, 31, 46]). Data on the number of swear words sent by each user was gathered under ϵ-LDP.

The second dataset was a who-trusts-whom network dataset [50]. Under ϵ-LDP, we collected information on the number of users trusted by each user. The dataset contained 36,692 users, with trust values ranging from 1 to 3,044.

The third dataset consisted of observational contact data from 86 rural Malawian residents [44]. Participants wore sensors in pouches on the front of their clothing to detect close proximity. A "touch event" between two individuals was identified when their devices exchanged about one radio packet across 20 time intervals. After contact was established, it was deemed continuous if no more than one radio packet was exchanged every second during the following 20-second interval. Each device had an ID number that linked to the contact information of the person carrying the device.

The fourth dataset documented face-to-face interactions among 405 participants at the SFHH conference in Nice, France, held in 2009 [20]. Each participant had a device that sent wireless packets at regular intervals, using temporary addresses assigned to the device. The devices could detect face-to-face encounters at an approximate distance of 1 m.

Table 8 provides an overview of the characteristics of these four datasets.

5.5.2 Evaluation Results

We evaluated the effectiveness of our proposed method using synthetic and real datasets. We compared the proposed method with the DCP, PWP, and SW methods proposed in previous studies [35, 80] (see Sect. 5.2). Because these methods do not assume a person-to-person interaction scenario, it is necessary to derive a method for setting the value of the privacy budget.

For DCP, the maximum value of ratio $Pr(\epsilon, x)/Pr(\epsilon, x')$ based on Equation (58) is e^ϵ when $x, x' = 1, -1$. In our scenarios, the range of x depending on $x_{i,j}$ is not 2 but $2/(n-1)$. In this case, the maximum ratio is represented by

$$\gamma(\epsilon, n) = \frac{Pr(\epsilon, -1 + 2/(n-1))}{Pr(\epsilon, -1)} = \frac{e^\epsilon + n - 2}{n - 1}. \tag{70}$$

Therefore, other than for u_i, the privacy budget $\log \gamma(\epsilon, n)$ is consumed. If the total privacy loss should be ϵ, the privacy budget for x_i should be set to the value obtained using the following equation for ϵ':

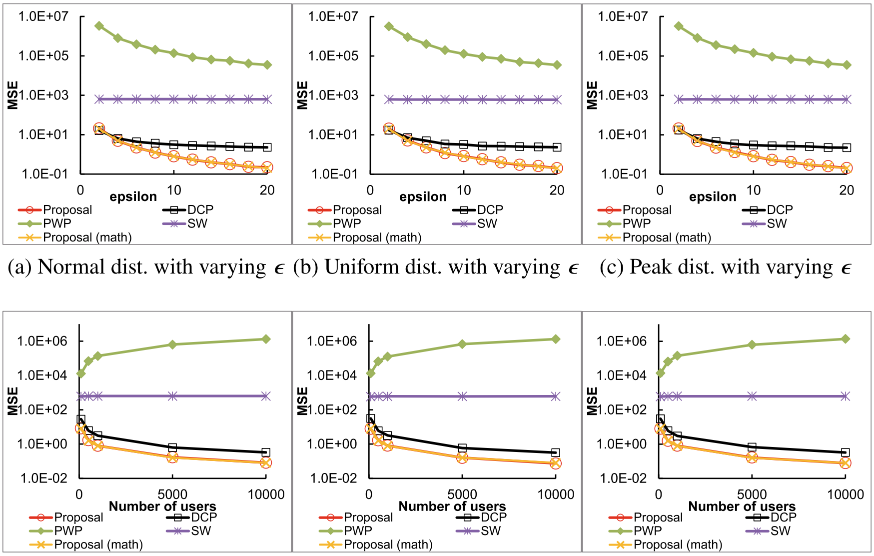
$$\epsilon' + (n-1)\log \gamma(\epsilon', n) = \epsilon. \tag{71}$$

It is difficult to solve (71) algebraically, but it can easily be solved numerically.

For PWP [80] and SW [35], the consumed privacy budget of u_j is also ϵ when user u_i sends the ϵ-LDP value of x_i to the server. Therefore, when n users send their LDP values to the server, the value of the privacy budget should be ϵ/n to ensure ϵ-LDP.

We experimentally evaluated the MSE and MAE. However, due to the space constraint, only the MSE results are shown in this section. The trends in the MSE results and MAE results were very similar. We repeated each experiment 1,000 times and obtained the average value. The range of ϵ was set to $[1, 20]$ based on [7, 74]. In several existing studies, ϵ was set to smaller values. In practice, the range $[1, 20]$ is sufficient for ϵ. In the setting we used for the synthetic datasets, each true value existed in the range $[0, 100]$. When ϵ was 1, the average amount of the Laplace noise per user was 200. The noise was large enough to ensure that the true value was not recognizable at all. When ϵ was 20, the average amount of Laplace noise per user was 10. Although the privacy protection level was relatively low, this value would be sufficient in some cases. The range of n (number of users) was set to $[100, 10000]$. The default values of ϵ and n were set to 10 and 1000, respectively.

The MSE results for the synthetic datasets are shown in Fig. 15. The results obtained with a varying value of ϵ are shown in Fig. 15a–c. The results of *Proposal (math)* represent the results of the mathematical analysis in (66) and (68). The results of PWP and SW are worse than those of the other methods. This is because when a user u_i sends an x_i value under ϵ'-LDP, this behavior consumes ϵ' of u_i's privacy budget and ϵ' of every u_j's privacy budget. DCP performed well when ϵ was small. However, for larger ϵ, the proposed method proved to be more effective than DCP. Originally, the DCP did not perform well when ϵ was large [7]. Figure 15d–f



(a) Normal dist. with varying ϵ (b) Uniform dist. with varying ϵ (c) Peak dist. with varying ϵ

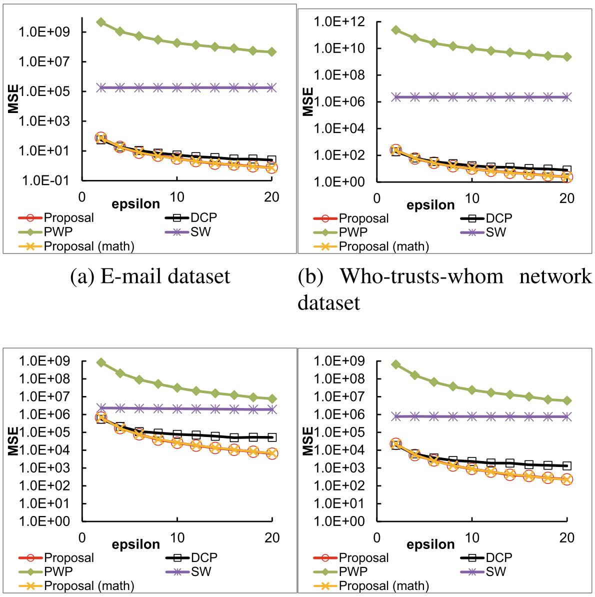
(d) Normal dist. with varying n (e) Uniform dist. with varying n (f) Peak dist. with varying n

Fig. 15 Mean squared error (MSE) results for synthetic datasets

show the results for different numbers of users. As the number of users increases, the amount of noise accumulated increases. However, if the noise added to each value is not too large, they cancel each other out, and the effect of each noise addition is mitigated. Owing to this tradeoff, the MSE increases or decreases depending on the method. For the proposed method and DCP, the MSE decreased as the number of users increased, because each noise addition was relatively small. In contrast, as PWP had larger noise values, the predicted average MSE increased with the number of users. For all datasets, the results were very similar to each other. As can be seen from Equations (66) and (68), the values of MSE and MAE do not depend on the content of the dataset but on the number of users and the value of ϵ.

The experimental results for the MSE are shown in Fig. 16. The performances of DCP and the proposed method were better than those of the other methods for all datasets. It is difficult to read the differences between DCP and the proposed method in Fig. 16, but there are significant differences in the MSE values. When ϵ was 10, the proposed method reduced the MSE by 47%, 40%, 66%, and 62%, respectively, compared with DCP.

Even if the amount of noise added to each value is large, the accuracy of the estimation can be increased by collecting a large amount of user data. Therefore, regarding the two large datasets (the e-mail and who-trusts-whom network datasets), the difference between the proposed method and other methods was relatively small. However, regarding the two small datasets (the Village and SFHH datasets), it was difficult for all the methods to estimate the average value with high accuracy. The



(a) E-mail dataset (b) Who-trusts-whom network dataset

(c) Village dataset (d) SFHH dataset

Fig. 16 MSE results for real datasets. Although the difference between the proposed method and DCP appears small, the proposed method reduces the MSE by 47%, 40%, 66%, and 62%, respectively, when $\epsilon = 10$

proposed method is particularly effective in this difficult task with a small number of users.

If the server collects data streams from each person, the budget will be small. Therefore, the performances of the proposed method and the DCP are similar in such a case. The performance of the proposed method for small values of ϵ will be improved in the future.

6 Conclusion

To create a human-centric digital twin for smart cities, it is crucial to collect extensive information about individuals' attributes and behaviors while ensuring privacy protection. However, existing privacy-preserving data mining solutions have not adequately addressed measurement noise, missing values, or human interactions, which has led to the loss of data analysis accuracy and privacy leakage. This chapter focused on addressing these challenges, which are particularly pronounced in smart city environments, and utilized local differential privacy (LDP) as the principal metric for evaluating privacy. LDP is a widely adopted privacy-preserving technique that introduces randomness at the data source to provide robust privacy guarantees. By refining the data collection and analysis methods, this chapter was intended to enhance the development of digital twins and the realization of truly smart cities. The proposed system has been demonstrated to achieve higher accuracy and enhanced privacy protection than existing methods through experiments using both synthetic and real-world data, as well as through theoretical analysis. It is believed that this system could serve as a foundation for the realization of more advanced smart cities. Moving forward, plans are in place to conduct pilot studies for the practical implementation of smart city development.

Acknowledgements This work was supported by JST, PRESTO under Grant JPMJPR1934. I am grateful to Prof. Akihiko Ohsuga, Assitant Prof. J. Andrew Onesimu, and Dr. Hiroshi Okumura for their collaboration on my research. I would like to thank Associate Prof. Yasuyuki Tahara, Associate Prof. Hiroyuki Nakagawa, Prof. Ryohei Orihara, and Dr. Shusaku Egami for useful discussions. I am also particularly grateful to Prof. Shinichi Honiden for his continuous support.

References

1. Communities and crime unnormalized data set (1995). https://archive.ics.uci.edu/ml/datasets/Communities+and+Crime+Unnormalized
2. Abul O, Bonchi F, Nanni M (2008) Never walk alone: uncertainty for anonymity in moving objects databases, pp 376–385
3. Apple DPT (2017) Learning with privacy at scale. Apple Mach Learn J 1:1–25
4. v Bosque LP, Garza SE (2014). Aggressive text detection for cyberbullying. Springer, pp 221–232. https://doi.org/10.1007/978-3-319-13647-9_21/COVER, https://link.springer.com/chapter/10.1007/978-3-319-13647-9_21
5. Cohen WW (2015) Enron email dataset. https://www.cs.cmu.edu/~./enron/
6. Cormode G, Maddock S, Maple C (2021) Frequency estimation under local differential privacy, pp 2046–2058
7. Cui L, Ma J, Zhou Y, Yu S (2023) Boosting accuracy of differentially private federated learning in industrial IoT with sparse responses. IEEE Trans Indus Inform 19:910–920. https://doi.org/10.1109/TII.2022.3161517
8. Differential-Privacy-Team: Learning with privacy at scale (2016). https://machinelearning.apple.com/research/learning-with-privacy-at-scale
9. Ding Z, Wang Y, Wang G, Zhang D, Kifer D (2018) Detecting violations of differential privacy. ACM, pp 475-489. https://doi.org/10.1145/3243734.3243818
10. Dua D, Graff C (2019) UCI machine learning repository. http://archive.ics.uci.edu/ml
11. Dwork C, McSherry F, Nissim K, Smith A (2006) Calibrating noise to sensitivity in private data analysis, pp 265–284
12. Dwork C, Roth A (2014) The algorithmic foundations of differential privacy. Found Trends® in Theoretical Comput Sci 9:211–407. https://doi.org/10.1561/0400000042
13. Úlfar E, Pihur V, Korolova A (2014) Rappor: randomized aggregatable privacy-preserving ordinal response, pp 1054–1067
14. Eyupoglu C, Aydin M, Zaim A, Sertbas A (2018) An efficient big data anonymization algorithm based on chaos and perturbation techniques. Entropy 20:373:1–373:18. https://doi.org/10.3390/e20050373, http://www.mdpi.com/1099-4300/20/5/373
15. Ge C, Ilyas IF, He X, Machanavajjhala A (2017) Private exploration primitives for data cleaning, pp 1–17. arXiv arXiv:1712
16. Genest C, MacKay J (1986) The joy of copulas: bivariate distributions with uniform marginals. The Am Statistician 40:280–283. https://doi.org/10.2307/2684602

17. Groat MM, Edwards B, Horey J, He W, Forrest S (2013) Application and analysis of multidimensional negative surveys in participatory sensing applications. Pervasive Mobile Comput 9:372–391
18. Grover R, Vriens M (2006) The handbook of marketing research: uses, misuses, and future advances. SAGE Publications, Inc
19. Guan Z, Lv Z, Du X, Wu L, Guizani M (2019) Achieving data utility-privacy tradeoff in internet of medical things: a machine learning approach. Fut Gener Comput Syst 98:60–68
20. Génois M, Barrat A (2018) Can co-location be used as a proxy for face-to-face contacts? EPJ Data Sci 7:1–18. https://doi.org/10.1140/EPJDS/S13688-018-0140-1 https://epjdatascience.springeropen.com/articles/10.1140/epjds/s13688-018-0140-1
21. Han H, Otto C, Liu X, Jain AK (2015) Demographic estimation from face images: human versus machine performance. IEEE Trans Pattern Anal Mach Intell 37:1148–1161
22. Hsu J, Gaboardi M, Haeberlen A, Khanna S, Narayan A, Pierce BC, Roth A (2014) Differential privacy: an economic method for choosing epsilon, pp 398–410
23. Hu R, Guo Y, Gong Y (2021) Concentrated differentially private federated learning with performance analysis. IEEE Open J Comput Soc 2:276–289. https://doi.org/10.1109/OJCS.2021.3099108
24. Jia J, Qiu W (2020) Research on an ensemble classification algorithm based on differential privacy. IEEE Access 8:93499–93513. https://doi.org/10.1109/ACCESS.2020.2995058
25. Kairouz P, Bonawitz K, Ramage D (2016) Discrete distribution estimation under local privacy, pp 2436–2444
26. Kaluža B, Mirchevska V, Dovgan E, Luštrek M, Gams M (2010) An agent-based approach to care in independent living, pp 177–186
27. Kasiviswanathan SP, Lee HK, Nissim K, Raskhodnikova S, Smith A (2013) What can we learn privately? SIAM J Comput 40:793–826
28. Khan LU, Saad W, Niyato D, Han Z, Hong CS (2022) Digital-twin-enabled 6g: vision, architectural trends, and future directions. IEEE Commun Mag 60:74–80. https://doi.org/10.1109/MCOM.001.21143
29. Kim JW, Jang B (2019) Workload-aware indoor positioning data collection via local differential privacy. IEEE Commun Lett 23:1352–1356. https://doi.org/10.1109/LCOMM.2019.2922963
30. Kim JW, Kim DH, Jang B (2018) Application of local differential privacy to collection of indoor positioning data. IEEE Access 6:4276–4286
31. Kontostathis A, Reynolds K, Garron A, Edwards L (2013) Detecting cyberbullying: query terms and techniques. In: Proceedings of the annual ACM web science conference (WebSci), pp 195–204. https://doi.org/10.1145/2464464.2464499, www.noswearing.com
32. Krishnan S, Wang J, Franklin MJ, Goldberg K, Kraska T (2016) Privateclean: data cleaning and differential privacy, pp 937–951. https://doi.org/10.1145/2882903.2915248
33. Leemput KV, Maes F, Vandermeulen D, Suetens P (1999) Automated model-based bias field correction of mr images of the brain. IEEE Trans Med Imaging 18:885–896. https://doi.org/10.1109/42.811268
34. Leskovec J, Lang KJ, Dasgupta A, Mahoney MW (2009) Community structure in large networks: natural cluster sizes and the absence of large well-defined clusters. Internet Math 6:29–123. https://doi.org/10.1080/15427951.2009.10129177
35. Li Z, Wang T, Lopuhaä-Zwakenberg M, Li N, Škoric B (2020) Estimating numerical distributions under local differential privacy, pp 621–635. https://doi.org/10.1145/3318464.3389700
36. Liu Y, Guo W, Fan CI, Chang L, Cheng C (2019) A practical privacy-preserving data aggregation (3pda) scheme for smart grid. IEEE Trans Indus Inform 15:1767–1774
37. Lyon A (2014) Why are normal distributions normal? Br J Philos Sci 65:621–649
38. Mironov I (2017) Rényi differential privacy, pp 263–275
39. Murakami T, Sei Y (2023) Automatic tuning of privacy budgets in input-discriminative local differential privacy. IEEE Internet of Things J
40. Navidan H, Moghtadaiee V, Nazaran N, Alishahi M (2022) Hide me behind the noise: local differential privacy for indoor location privacy. Institute of Electrical and Electronics Engineers Inc., pp 514–523. https://doi.org/10.1109/EUROSPW55150.2022.00061

41. Oishi K, Sei Y, Tahara Y, Ohsuga A (2020) Semantic diversity: privacy considering distance between values of sensitive attribute. Comput Secur 94:1–18. https://doi.org/10.1016/j.cose.2020.101823
42. Onesimu JA, Karthikeyan J, Sei Y (2021) An efficient clustering-based anonymization scheme for privacy-preserving data collection in IoT based healthcare services. Peer-to-Peer Netw Appl 14:1629–1649. https://doi.org/10.1007/S12083-021-01077-7/TABLES/7
43. Ordó?ez FJ, de Toledo P, Sanchis A, Ordóñez FJ, Toledo PD, Sanchis A (2013) Activity recognition using hybrid generative/discriminative models on home environments using binary sensors. Sensors 13:5460–5477
44. Ozella L, Paolotti D, Lichand G, Rodríguez JP, Haenni S, Phuka J, Leal-Neto OB, Cattuto C (2021) Using wearable proximity sensors to characterize social contact patterns in a village of rural Malawi. EPJ Data Sci 10:1–17. https://doi.org/10.1140/EPJDS/S13688-021-00302-W https://epjdatascience.springeropen.com/articles/10.1140/epjds/s13688-021-00302-w
45. Palumbo F, Gallicchio C, Pucci R, Micheli A (2016) Human activity recognition using multisensor data fusion based on reservoir computing. J Ambient Intell Smart Environ 8:87–107
46. Pamungkas EW, Basile V, Patti V (2020) Do you really want to hurt me? predicting abusive swearing in social media. European Language Resources Association, pp 6237–6246
47. Pang J, Huang Y, Xie Z, Li J, Cai Z (2021) Collaborative city digital twin for the covid-19 pandemic: a federated learning solution. Tsinghua Sci Technol 26:759–771. https://doi.org/10.26599/TST.2021.9010026
48. Ren X, Shi L, Yu W, Yang S, Zhao C, Xu Z (2022) Ldp-ids: local differential privacy for infinite data streams. Association for computing machinery, pp 1064–1077. https://doi.org/10.1145/3514221.3526190
49. Ren X, Yu CM, Yu W, Yang S, Yang X, McCann JA, Yu PS (2018) Lopub: high-dimensional crowdsourced data publication with local differential privacy. IEEE Trans Inf Forensics Secur 13:2151–2166. https://doi.org/10.1109/TIFS.2018.2812146
50. Richardson M, Agrawal R, Domingos P (2003) Trust management for the semantic web, pp 351–368
51. Rocher L, Hendrickx JM, de Montjoye YA (2019) Estimating the success of re-identifications in incomplete datasets using generative models. Nat Commun 10:1–9
52. Sei Y, Ohsuga A (2017) Differential private data collection and analysis based on randomized multiple dummies for untrusted mobile crowdsensing. IEEE Trans Inf Forensics Secur 12:926–939
53. Sei Y, Ohsuga A (2017) Location anonymization with considering errors and existence probability. IEEE Trans Syst, Man, Cybern: Syst 47:3207–3218. https://doi.org/10.1109/TSMC.2016.2564928
54. Sei Y, Ohsuga A (2022) Local differential privacy for person-to-person interactions. IEEE Open J Comput Soc
55. Sei Y, Ohsuga A (2022) Private true data mining: differential privacy featuring errors to manage internet-of-things data. IEEE Access 10:8738–8757
56. Sei Y, Ohsuga A (2023) Data collection of biomedical data and sensing information in smart rooms. Data in Brief 47:108922. https://doi.org/10.1016/j.dib.2023.108922
57. Sei Y, Okumura H, Ohsuga A (2022) Re-identification in differentially private incomplete datasets. IEEE Open J Comput Soc 3:62–72. https://doi.org/10.1109/OJCS.2022.3175999
58. Sei Y, Okumura H, Takenouchi T, Ohsuga A (2019) Anonymization of sensitive quasi-identifiers for l-diversity and t-closeness. IEEE Trans Dependable Secure Comput 16:580–593. https://doi.org/10.1109/TDSC.2017.2698472
59. Sei Y, Onesimu JA, Ohsuga A (2022) Machine learning model generation with copula-based synthetic dataset for local differentially private numerical data. IEEE Access 10:101656–101671. https://doi.org/10.1109/ACCESS.2022.3208715
60. Sei Y, Onesimu JA, Okumura H, Ohsuga A (2023) Privacy-preserving collaborative data collection and analysis with many missing values. IEEE Trans Dependable Secure Comput 2158–2173. https://doi.org/10.1109/TDSC.2022.3174887

61. Shengli W (2021) Is human digital twin possible? Comput Methods Programs Biomedicine Update 1:100014
62. Shi D, Ding J, Errapotu SM, Yue H, Xu W, Zhou X, Pan M (2019) Deep q-network based route scheduling for TNC vehicles with passengers' location differential privacy. IEEE Internet of Things J 6:7681–7692
63. Sklar A (1959) Fonctions de répartition àn dimensions et leurs marges. Publications de l'Institut Statistique de l'Universitè de Paris, vol 8, pp 229–231
64. Song F, Ma T, Tian Y, Al-Rodhaan M (2019) A new method of privacy protection: random k-anonymous. IEEE Access 7:75434–75445. https://doi.org/10.1109/ACCESS.2019.2919165
65. Sweeney L (2002) k-anonymity: a model for protecting privacy. Int J Uncertainty, Fuzziness Knowl-Based Syst 10:557–570
66. Tang J, Korolova A, Bai X, Wang X, Wang X (2017) Privacy loss in apple's implementation of differential privacy on macos 10.12. arXiv 1709.02753, pp 1–12
67. Taylor JR (1997) Introduction to error analysis: the study of uncertainties in physical measurements, 2nd edn. University Science Books
68. Torres RLS, Ranasinghe DC, Shi Q, Sample AP (2013) Sensor enabled wearable RFID technology for mitigating the risk of falls near beds, pp 191–198
69. Tozaka H, Yoshiba T (2005) Specific applications of copulas in financial practice. Financial Res 24:115–162
70. Tsuda A, Matsuzaki K, Sei Y (2023) Developing rem sleep prediction models using smart home sensor data. In: Proceedings of the IEEE world conference on applied intelligence and computing (AIC)
71. Varma G, Chauhan R, Singh D (2022) Sarve: synthetic data and local differential privacy for private frequency estimation. Cybersecurity 5:1–20
72. Wang N, Xiao X, Yang Y, Hoang TD, Shin H, Shin J, Yu G (2018) Privtrie: effective frequent term discovery under local differential privacy, pp 821–832
73. Wang S, Chang JM (2021) Privacy-preserving boosting in the local setting. IEEE Trans Inf Forensics Secur 1–15. https://doi.org/10.1109/TIFS.2021.3097822
74. Wang Y, Tong Y, Shi D (2020) Federated latent Dirichlet allocation: a local differential privacy based framework, pp 6283–6290. https://doi.org/10.1609/aaai.v34i04.6096
75. Warner SL (1965) Randomized response: a survey technique for eliminating evasive answer bias. Am Stat Assoc 60:63–69
76. Winitzki S (2008) A handy approximation for the error function and its inverse
77. Wood MJ, Ross-Kerr J (2010) Basic steps in planning nursing research: from question to proposal. Jones & Bartlett Publishers
78. Wu CFJ (1983) On the convergence properties of the EM algorithm on JSTOR. On the convergence properties of the EM algorithm 11:95–103. https://www.jstor.org/stable/2240463?seq=1#metadata_info_tab_contents
79. Xu C, Ren J, She L, Zhang Y, Qin Z, Ren K (2019) Edgesanitizer: locally differentially private deep inference at the edge for mobile data analytics. IEEE Internet of Things J 6:5140–5151
80. Xue Q, Zhu Y, Wang J (2022) Mean estimation over numeric data with personalized local differential privacy. Front Comput Sci 16. https://doi.org/10.1007/s11704-020-0103-0
81. Zahid FM, Heumann C (2019) Multiple imputation with sequential penalized regression. Stat Meth Med Res 28:1311–1327
82. Zhao Y, Zhao J, Yang M, Wang T, Wang N, Lyu L, Niyato D, Lam KY (2021) Local differential privacy-based federated learning for Internet of Things. IEEE Internet of Things J 8:8836–8853. https://doi.org/10.1109/JIOT.2020.3037194

Human-Centered Service Integration for Smart Cities

Automated Negotiations Protocols for Complex Utility Function as Social System

Katsuhide Fujita

Abstract This chapter focuses on automated negotiations based on multi-agent systems. It targets researchers and students in various communities of autonomous agents and multi-agent systems, such as agreement technology, cyber-physical system (CPS), electronic commerce, and so on. It helps readers understand automated negotiations, negotiation protocols, negotiating agents' strategies, and their applications. Negotiation is an essential aspect of daily life and represents an important topic in multi-agent system research. This chapter focuses on multiple interdependent issues negotiation, which is a more realistic situation than simple negotiations involving only multiple independent issues. The key impact of such issue dependencies is that their results in agent utility functions are complex. Existing negotiation protocols which are well-suited for linear utility functions are, however, often not able to find well agreements when applied to complex negotiations with issue dependencies. This chapter shows some negotiation protocols focusing on the multiple interdependent issues negotiations to find high-quality solutions for the complex agents' utility function.

1 Introduction

Multi-agent systems are one of the most promising technologies to emerge in recent decades, considering the applications of several fields such as distributed systems, economics, and social science. Many researchers have drawn a vision in which many tasks humans perform now are delegated to intelligent, autonomous, and proactive programs, generally called software agents [1]. Multi-agent system (MAS) is a system composed of multiple interacting intelligent agents. MAS can be used to solve difficult or impossible problems for an individual agent or a monolithic system. Intelligence may include some methodic, functional, procedural, or algorithmic Research, find, and process approach. In MAS, intelligent agents need to interact

K. Fujita (✉)
Tokyo University of Agriculture and Technology, 2-24-16 Koganei, Tokyo, Japan
e-mail: katfuji@cc.tuat.ac.jp

Y. Murakami and K. Kimura (eds.), *Human-Centered Services Computing for Smart Cities*, https://doi.org/10.1007/978-981-97-0779-9_6

with one another to achieve their individual objectives or manage the dependencies that follow from being situated in a common environment [2]. These interactions can vary from simple information interchanges to requests for particular actions to be performed and on to cooperation (working together to achieve a common objective) and coordination (arranging for related activities to be performed coherently).

AAMAS (Autonomous Agents and Multi-Agent Systems) is one of the prominent top conferences for research related to MAS. In addition, many research achievements related to MAS are presented in AAAI (Annual AAAI Conference on Artificial Intelligence) and IJCAI (International Joint Conference on Artificial Intelligence), which are the top conferences in Artificial Intelligence. Journal of Artificial Intelligence Research (JAIR) and the Journal of Artificial Intelligence (AIJ) are open-access journals covering a wide range of AI topics, including multi-agent systems. Autonomous Agents and Multi-Agent Systems (JAAMAS) is a journal associated with the IFAAMAS that publishes research on autonomous agents and MAS.

One of the relevant interactions in MAS is *negotiation* (the process by which a group of agents comes to a mutually acceptable agreement on some matter). Negotiation examines whether the agents (both artificial and human agents) should cooperate and is required when the agents are self-interested and cooperative. In other words, negotiation is a significant method of competitive (or partially cooperative) allocation of goods, resources, or tasks among agents. Negotiation is also an essential aspect of daily life and an important topic. They can be simple and ordinary, as in haggling over a price in the market or deciding on a meeting time, or international disputes and nuclear disarmament [3] issues that affect the well-being of millions. While the ability to negotiate successfully is critical for much social interaction, negotiation is an essential and challenging task. Something that might be perceived as a "simple" case of single-issue bilateral bargaining over a price in the marketplace can demonstrate the difficulties that arise during the negotiation process.

It is a subject that has been extensively discussed in game-theoretic, economic, and management research fields for decades (e.g. [4–11]). Although we already have more recent activities in this field [12–15], the key contributions done were in the field of automated negotiation systems that consist of intelligent software agents [16–18]. There has been extensive work in the area of automated negotiation, that is, where agents negotiate with other agents in such contexts as e-commerce [19–22], large-scale argumentation [23, 24], collaborative design [25, 26], and service-oriented computing [27, 28]. The model of the multi-agent system is necessary for cooperative work between agents, and automated negotiations between agents are required when they have conflicts. In addition, most researchers in multi-agent systems regard automated negotiation as the most critical topic for theoretical analysis or practical applications of agent-based systems. Thus, success in developing automated negotiation capabilities has excellent advantages and implications.

1.1 Main Flow of Automated Negotiations

The main flow of accomplishing the automated negotiations are Negotiation Environment, Preference Elicitation and Negotiation Strategy, and Negotiation Protocol.

Negotiation Environment: The negotiation environment defines the specific settings of the negotiation. Based on these settings, the researcher should take different considerations. The environment determines several parameters that dictate the number of negotiators taking part in the negotiation, the time frame, and the issues on which the negotiation is being conducted. The number of parties participating in the negotiation can be two (bilateral negotiations) or more (multilateral negotiations). The negotiation environment also consists of objectives and issues to be resolved. Various issues can be involved, including discrete enumerated value sets, integer-value sets, and real-value sets. Negotiations involving multi-attribute issues allow making complex decisions while considering multiple factors [29].

Preference Elicitation and Negotiation Strategy: Preference elicitation techniques attempt to collect as much information on users' preferences as possible to find efficient solutions [30, 31]. However, because users' preferences are always incomplete initially and tend to change in different contexts, in addition to user's cognitive and emotional limitations of information processing, preference elicitation methods must also be able to avoid preference reversals, discover hidden preferences, and assist users in making tradeoffs when confronted with competing objectives. In addition, negotiation agents should have an effective negotiation strategy to achieve significant agreements.

Negotiation Protocol: Automated negotiation protocol defines the formal interaction between the decision makers (Agents) in the negotiation environments -whether the negotiation is done only once (one-shot) or repeatedly- and how the exchange of offers between the agents is conducted. In addition, according to Jennings et al. [32], a negotiation protocol is a set of rules that govern the interaction and cover the permissible types of participants (e.g., the negotiators and any relevant third parties), the negotiation states (e.g., accepting bids, negotiation closed), the events that cause negotiation states to change (e.g., no more bidders, bid accepted) and the valid actions of the participants in particular conditions (e.g., which messages can be sent by whom, to whom, at what stage).

The agents in the negotiations can be non-cooperative or cooperative. Generally, cooperative agents try to maximize their social welfare (see Zhang [33]), while non-cooperative agents try to maximize their utilities regardless of the other side's utilities. These kinds of issues are focued, which have been widely studied in different research areas, such as game theory [8, 10], distributed artificial intelligence [34–36] and economics [9].

Figure 1 shows the main flow in accomplishing automated negotiations. This figure shows the example of designing a simple car among car designers:

- The negotiation environment, including negotiation issues, agents' actions, and objectives, is based on real-life negotiation.

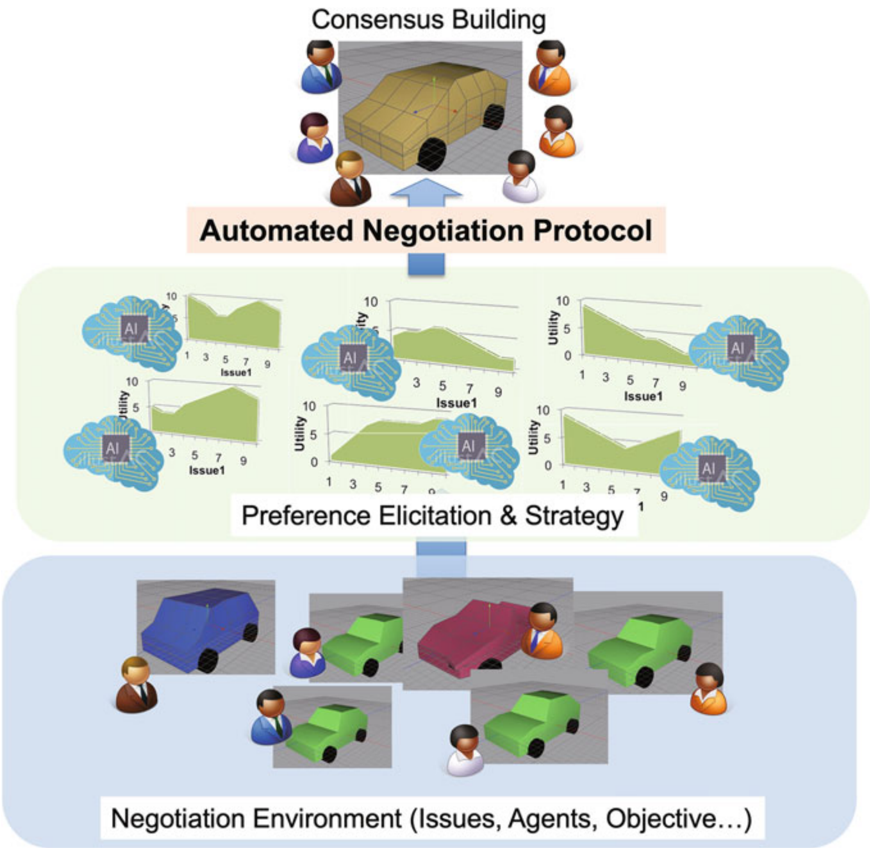


Fig. 1 Main flow of accomplishing the automated negotiations. The automated negotiation is composed as negotiation environment, preference elicitation and negotiation strategy, and negotiation protocol

- The preference of the users should be collected using some preference elicitation techniques. In addition, the negotiation agent has a strategy.
- Agents negotiate the car designs automatically based on the negotiation protocol.

One of the most critical parts of automated negotiation is the negotiation protocol, which has been extensively discussed in game-theoretic, economic, and management science literature for decades. In addition, many problems remain unsolved in the negotiation protocol, and these problems constitute the leading research theme in the multi-agent system field. The automated negotiation protocol to accomplish automated negotiations is focused. Finally, agents build a consensus for designing the car.

1.2 Complex Multi-issue Negotiation with Highly Nonlinear Utility Functions

In this chapter, the automated negotiation protocols between cooperative agents are focused on. While there has been a lot of previous work in this area [37–39], these efforts have, to date, dealt almost exclusively with simple negotiations involving multiple independent issues and, therefore, linear (single optimum) utility functions. An example of such representations widely used in the negotiation literature is linear-additive utility functions [35], which allow the modeling of independent issues.

Many real-world negotiation problems, however, involve multiple interdependent issues that are highly nonlinear. Adding such interdependencies complicates the agent's utility functions, making them nonlinear, with multiple optima. For example, interdependence between attributes in agent preferences can be described by using different categories of functions, like K-additive utility functions [40, 41], bidding languages [42] or constraints [43–45].

In the context of a multi-attribute negotiation, the complexity depends on the number of issues, the number of agents, and the level of interdependency between the preferences on the issues and the domain of the issues. The method to describe the agent's utility spaces also represents a fundamental measure of the complexity of the negotiation scenario.

Some studies have focused on negotiation with nonlinear utility functions. Klein et al. [36] present the first negotiation protocols specifically for complex preference spaces. They focus on the nonlinear utility function and describe a simulated annealing-based approach appropriate for negotiating complex contracts that achieves near-optimal social welfare for negotiations with binary issue dependencies. The important points in this work are the positive results regarding the use of simulated annealing to regulate agent decision-making and the use of agent expressiveness to allow the mediator to improve its proposals. In addition, most existing negotiation protocols, like a method based on Hill-climbing, which is well-suited for linear utility functions, work poor when applied to nonlinear problems. However, it was not applied to multilateral negotiations with higher-order dependencies. Higher-order dependencies and continuous-valued issues, common in many real-world contexts, generate more challenging utility landscapes that are not considered in their work.

One of the most relevant approaches focusing on the complex utility space is Ito et al. [43, 46]. They proposed the original constraint-based utility functions, which assume highly nonlinear and bumpy utility functions. Therefore, scalable and efficient negotiation protocols are required if the complexity of the negotiation environment is high. Also, they proposed a bidding-based protocol. In this protocol, agents generate bids by sampling their own utility functions to find local optima and then use constraint-based bids to compactly describe regions that have large utility values for that agent. A mediator considers then a combination of bids that maximizes social welfare. This protocol also had an impact on the automated negotiation field because many existing works didn't consider the highly nonlinear utility of agents.

In this chapter, the constraint-based nonlinear utility function is focused on. There are many multi-issue negotiation models except the use of constraints; however, there are several reasons in favor of using constraints in negotiation models. First, they implement efficient methods of preference elicitation. Moreover, constraints allow the expression of dependencies between the possible values of the different attributes. Finally, the use of constraints for offer expression allows for limiting the region of the solution space that must be explored in a given negotiation step. Reducing the area of the utility space under exploration according to the constraints exchanged by agents is a widely used technique in automated negotiation [47, 48], since it searches for agreements a more efficient process than when using positional bargaining, especially in complex negotiation scenarios.

1.3 Main Contributions of This Chapter

In the complex multi-issue automated negotiation protocol, the existing studies have some unsolved issues. This chapter deals with the followings aims.

Aim 1: Scalable and Efficient Negotiation Protocols

A significant problem is scalability for the number of agents and issues. In the negotiation setting, the utility space becomes highly nonlinear, making finding the optimal agreement point very difficult. For example, the bidding-based negotiation protocol does not have high scalability for the number of agents, and the mediator needs to find the optimum combination of submitted bids from the agents. However, the computational complexity for finding solutions is too large.

A Issue-grouping based negotiation protocol is proposed by decomposing the contract space based on issue interdependencies. A new protocol in which a mediator tries to reorganize a highly complex utility space into several tractable utility subspaces is proposed in order to reduce the computational cost. Issue groupings are generated by a mediator based on an examination of the issue interdependencies. First, a measure for the degree of interdependency between issues is defined. Next, a weighted non-directed interdependency graph is generated based on this information. By analyzing the interdependency graph, a mediator can identify issue subgroups. Note that while others have discussed issue interdependencies in utility theory [49–51], this previous work doesn't identify optimal issue groups. Finally, the experimental results demonstrate that the protocol has higher scalability than previous efforts and the impact on the optimality of the negotiation outcomes based on issue groups.

Aim 2: Negotiation Protocols Concerning Agents' Private Information

A negotiation protocol should concern about agents' private information (privacy). Such private information should be protected as much as possible in a negotiation because users generally want to keep their privacy in real life. For example, suppose several companies collaboratively design and develop a new car model. If one company reveals more private information than the other companies, the other companies will know more of that company's important information, such as utility information. As a result, the company will be at a disadvantage in subsequent negotiations, and the mediator might leak the agent's utility information. Therefore, this chapter aims to accomplish the negotiation protocols without revealing the agents' private information to others.

A threshold-adjusting mechanism was proposed. First, agents make bids that produce more utility than the common threshold value based on the bidding-based protocol proposed in [43]. Then, the mediator asks each agent to reduce its threshold based on how much each agent opens its private information to the others. Each agent makes bids again above the threshold. This process continues iteratively until an agreement is reached or there is no solution. The experimental results show that the method substantially outperforms the existing negotiation methods on the point of how much agents have to open their own utility space.

In addition, secure protocols are proposed to conceal all private information: the Distributed Mediator Protocol (DMP) and the Take it or Leave it (TOL) Protocol. They make agreements and conceal agent utility values. When searching in their search space, they employ Secure Gathering, with which they can simultaneously calculate the sum of the per agent utility value and conceal it. Furthermore, Distributed Mediator Protocol (DMP) improves the scalability for the complexity of the utility space by dividing the search space toward the mediators. In the Take it or Leave it (TOL) Protocol, the mediator searches using the hill-climbing search algorithm. The evaluation value is decided by responses that agents either take or leave, moving from the current state to the neighboring state. The Hybrid Secure Protocol (HSP) that combines DMP with TOL is proposed. In HSP, TOL is performed first to improve the initial state in the DMP step. Next, DMP is performed to find the local optima in the neighborhood. HSP can also reach an agreement and conceal per-agent utility information. Additionally, HSP can reduce the required memory for making an agreement, which is a major issue in DMP. Moreover, HSP can improve communication costs (memory usage) more than DMP by the experiments.

Aim 3: Addressing Weaknesses of the Nash Bargaining Solution in Nonlinear Negotiation

The Nash bargaining solution, which maximizes the product of the agent utilities, is a well-known metric that provably identifies the optimal (fair and social-welfare-maximizing) agreement for negotiations in linear domains [8, 52, 53]. In *nonlinear* domains, however, the Pareto frontier will often not satisfy the convexity assumption required to make the Nash solution optimal and unique [8, 52, 54]. There can, in other words, be multiple agreements in nonlinear domains that satisfy the Nash Bargaining Solution, and many or all of these will have sub-optimal fairness and/or social welfare.

A secure mediated protocol (SFMP) is proposed that addresses this challenge. The protocol consists of two main steps. In the first step, SFMP uses a nonlinear optimizer, integrated with a secure information-sharing technique called Secure Gathering [55], to find the Pareto front without causing agents to reveal private utility information. In the second step, an agreement is selected from the set of Pareto-optimal contracts using approximate fairness, which measures how equally the total utility is divided across the negotiating agents ([56] etc.). It demonstrates that SFMP produces better scalability and social welfare values than previous nonlinear negotiation protocols.

2 Multi-issue Negotiation with Highly Nonlinear Utility Functions

A model of non-linear multi-issue negotiation and a bidding-based negotiation protocol (basic bidding) designed as a multi-issue negotiation protocol suitable for agents with highly non-linear utility functions is described. The constraint-based utility functions are realistic because they allow us to produce bumpy and highly non-linear utility functions. In the basic bidding algorithm, agents generate bids by sampling their own utility functions to find local optima and then use constraint-based bids to compactly describe regions that have large utility values for that agent. These techniques make bid generation computationally tractable even in large utility spaces. A mediator then finds a combination of bids that maximize social welfare.

2.1 Basic Model of Multi-issue Negotiation

Definition 1: Agents and Mediator. N agents $(a_1, \ldots, a_N)$ want to reach an agreement with a mediator who manages the negotiation from a man-in-the-middle position.

Definition 2: Issues under negotiation. There are M issues $(i_1, \ldots, i_M)$ to be negotiated[1].

Definition 3: Contract Space. The negotiation solution space is defined by the values that the different values may take. To simplify, we assume that the issue takes a value drawn from the domain of integers $[0, X]$:

$$D = [0, X]^M$$

Definition 4: Contract or potential solution.

$$\mathbf{s} = (s_1, ..., s_M)$$

A contract is represented by a vector of issue values. Each issue s_j has a value drawn from the domain of integers $[0, X]$ $(1 \leq j \leq M)$.(*i.e.* $s_j \in \{0, 1, , \ldots, X\}$)[2].

[1] The number of issues represents the number of dimensions in the utility space. The issues are shared: all agents are potentially interested in the values for all M issues.

[2] A discrete domain can come arbitrarily close to a real domain by increasing its size. As a practical matter, many real-world issues that are theoretically real numbers (delivery date, cost) are discretized during negotiations.

Constraint-based Complex Utility Model

Some of the protocols and experiments in this chapter rely on the constraint-based utility model. In other words, an agent's utility function is described in terms of constraints. This produces a bumpy non-linear utility function and is a crucial departure from previous efforts on multi-issue negotiation, where the contract utility is calculated as the weighted sum of the utilities.

Definition 5: Constraint.

$$c_k \in C \ (1 \leq k \leq l).$$

There are l constraints in an agent's utility space. Each constraint represents a region in the contract space with one or more dimensions and an associated utility value.

Definition 5-1: Constraint Value. Constraint c_k has value $w_a(c_k, \mathbf{s})$ if and only if it is satisfied by contract $\mathbf{s}$ for the agent a.

Definition 5-2: Constraint Region. Function $\delta_a(c_k, i_j)$ is a region of i_j in c_k. $\delta_a(c_k, i_j)$ is $\emptyset$ if c_k has no region regarded as i_j.

Definition 5-3: The Number of Terms in the Constraint. Function $\epsilon_a(c_k)$ is the number of terms in c_k.

Definition 6: Utility function.

$$u_a(\mathbf{s}) = \sum_{c_k \in C, \mathbf{s} \in x(c_k)} w_a(c_k, \mathbf{s}),$$

where $x(c_k)$ is a set of possible contracts (solutions) of c_k.
An agent's utility for contract $\mathbf{s}$ is defined as the sum of the utility for all the constraints it satisfies.

Definition 7: The relationship between agents and constraints Every agent has *its own, typically unique*, set of constraints.

These definitions produce a "bumpy" nonlinear utility function with high points where many constraints are satisfied and lower regions where few or no constraints are satisfied. It represents a crucial departure from previous efforts on multi-issue negotiation. The utility is calculated as the weighted sum of the utilities for individual issues, producing utility functions shaped like flat hyperplanes with a single optimum.

Figure 2 shows an example of a utility space generated via a collection of binary constraints involving Issues 1 and 2; the number of issues is two. For example, one of the constraints has a value of 55, which holds if the value for Issue 1 is [3, 7] and the value for Issue 2 is [4, 6]. The utility function is highly nonlinear, with many hills and valleys. It assumes that many real-world utility functions are more complex than this, involving more than two issues and higher-order (e.g., trinary and quaternary) constraints. This constraint-based utility function representation allows us to capture the issue interdependencies common in real-world negotiations. However, this representation can also capture linear utility functions as a particular case (they

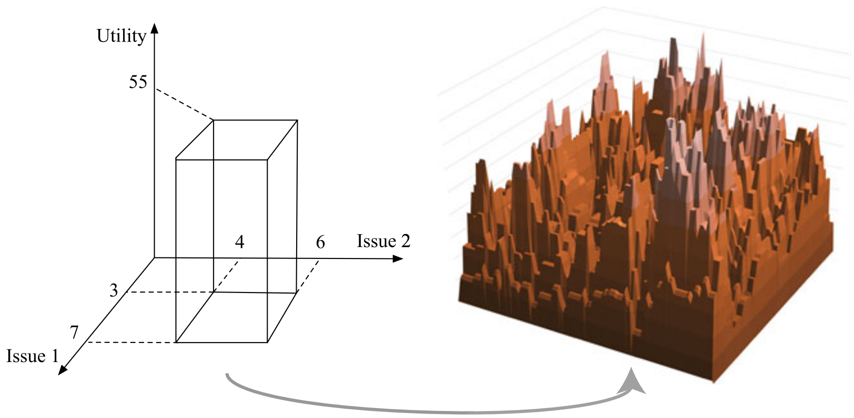


Fig. 2 An example of a utility space generated via a collection of binary constraints involving issues 1 and 2; the number of issues is two. One of the constraints has a value of 55, which holds if the value for issue 1 is [3, 7] and the value for issue 2 is [4, 6]. The utility space is highly nonlinear, with many hills and valleys

can be captured as a series of unary constraints). A negotiation protocol for complex contracts can, therefore, handle linear contract negotiations.

As is common in negotiation contexts, agents do not share their utility functions to preserve a competitive edge. It will generally be the case that agents do not fully know their desirable contracts in advance because each individual utility function is simply huge. For example, if we have 10 issues with 10 possible values per issue, this produces a space of 10^{10} (10 billion) possible contracts, too many to evaluate exhaustively. Agents must thus operate in a highly uncertain environment.

Objective Function

The objective function for the negotiation protocol can mainly be described as follows:

$$\arg\max_{\mathbf{s}} \sum_{a \in N} u_a(\mathbf{s}).$$

The negotiation protocol tries to find contracts that maximize social welfare, *i.e.*, the total utilities for all agents. Such contracts, by definition, will also be Pareto-optimal. Theoretically possible to gather all the individual agents' utility functions into one central place and then find all optimal contracts using well-known nonlinear optimization techniques such as simulated annealing (SA) or evolutionary algorithms (GA). However, centralized methods can't be applied for negotiation purposes because agents prefer not to share their utility functions to preserve a competitive edge as is common in negotiation contexts.

2.2 Basic Bidding Protocol

Agents reach an agreement based on the following steps. It is called a basic bidding protocol. This protocol is a remarkable result focusing on the complex automated negotiation with high nonlinearity. The proposed automated negotiation protocols are compared with this basic bidding protocol as the baseline for evaluation.

The basic bidding protocol consists of the following four steps:

Step 1: Sampling. Each agent samples its utility space to find high-utility contract regions. A fixed number of samples are taken from a range of random points drawn from a uniform distribution. Note that if the number of samples is low, the agent may miss some high-utility regions in its contract space and potentially end up with a sub-optimal contract.

Step 2: Adjusting. There is no guarantee that a given sample will lie on a locally optimal contract. Each agent, therefore, uses a nonlinear optimizer based on SA to try to find the local optimum in its neighborhood.

Step 3: Bidding. For each contract $\mathbf{s}$ found by adjusted sampling, an agent evaluates its utility by summing the values of the satisfied constraints. If that utility is larger than the reservation value δ, then the agent defines a bid that covers all the contracts in the region with that utility value. Steps 1, 2, and 3 can be shown as Algorithm 1.

Step 4: Deal identification. The mediator identifies the final contract by finding all the combinations of bids, one from each agent, that are mutually consistent, i.e., that specify overlapping contract regions. For example, if a bid has a region, such as $[0, 2]$ for issue 1, $[3, 5]$ for issue 2, the bid is accepted by a contract point $[1, 4]$, which means issue 1 takes 1, issue 2 takes 4. If a combination of bids, i.e., a solution, is consistent, there are definitely overlapping regions. For instance, a bid with regions (Issue 1, Issue 2) = ([0, 2], [3, 5]), and another bid with ([0, 1], [2, 4]) is consistent. If there is more than one such overlap appears, the mediator selects the one with the highest summed bid value (and thus, assuming truthful bidding, the highest social welfare). Each bidder pays the value of its winning bid to the mediator. The mediator employs a breadth-first search with branch cutting to find the social-welfare-maximizing overlaps. Step 4 can be shown as Algorithm 2.

It is easy to show in theory that this approach can be guaranteed to find optimal contracts. If each agent exhaustively samples every contract in its utility space and has a reservation value of zero, it will generate bids representing its complete utility function. With all agents' utility functions in hand, the mediator can use an exhaustive search over all bid combinations to find the social-welfare-maximizing negotiation outcome. However, this approach is practical only for tiny contract spaces. The computational cost of generating bids and finding winning combinations grows rapidly as the size of the contract space increases. As a practical matter, the threshold is applied to limit the number of bids the agents can generate. Thus, deal identification can terminate in a reasonable amount of time.

Algorithm 1 Bid-generation with SA(Th, SN, V, T, B)

SN: The number of samples
T: Temperature for Simulated Annealing
V: A set of values for each issue, V_m is for an issue m

```
P_smpl := ∅
while |P_smpl| < SN do
    P_smpl := P_smpl ∪ {p_i} (randomly selected from P)
    P := Π_{m=0}^{|I|} V_m, P_sa := ∅
end while
for p ∈ P_smpl do
    p' := simulatedAnnealing(p, T)
    P_sa := P_sa ∪ {p'}
end for
for p ∈ P_sa do
    u := 0, B := ∅, BC := ∅
    for c ∈ C do
        if c contains p as a contract and p satisfies c then
            BC := BC ∪ c
            u := u + v_c
        end if
        if u >= Th then
            B := B ∪ (u, BC)
        end if
    end for
end for
```

Algorithm 2 Search_solution(B)

Ag: A set of agents B: A set of Bid-set of each agent ($B = \{B_0, B_1, ..., B_n\}$, A set of bids from agent i is $B_i = \{b_{i,0}, b_{i,1}, ..., b_{i,m}\}$)

```
SC := ⋃_{j∈B_0} {b_{0,j}}, i := 1
while i < |Ag| do
    SC' := ∅
    for s ∈ SC do
        for b_{i,j} ∈ B_i do
            s' := s ∪ b_{i,j}
            if s' is consistent then
                SC' := SC' ∪ s'
            end if
            SC := SC', i := i + 1
        end for
    end for
end while
maxSolution = getMaxSolution(SC)
return maxSolution
```

3 Threshold Adjustment Mechanism for Keeping Agents' Privacy

Existing works on automated negotiation protocols with non-linear utility functions have not considered the agents' private information. Such private information should be kept secret as much as possible in their negotiations. A threshold adjustment mechanism is proposed. First, agents make bids that produce more utility than the common threshold value according to the basic bidding protocol [43]. Then the mediator asks each agent to reduce its threshold depending on how much private information it shares with others. Finally, each agent again makes bids above the threshold. This process continues iteratively until an agreement is reached or no solution is found. The experimental results show that the proposed method substantially outperforms existing negotiation methods regarding how much agents have to open their own utility space.

3.1 Threshold Adjustment Mechanism

The main idea of the threshold adjustment mechanism is that if an agent reveals a larger area of its utility space, it is given the opportunity to persuade other agents. On the other hand, when an agent shows a small area of its utility space, it should adjust its threshold to reveal a larger area if no agreement is reached. The revealed area can be defined by how the agent reveals its utility space according to its threshold value. The threshold values are initially set to the same value. Then each agent changes its threshold value based on the subsequent size of the revealed area.

Figure 3 shows an example of the threshold adjustment process among three agents. The upper and lower panels show the thresholds and the revealed areas before and after threshold adjustments, respectively. Specifically, Agent 3 revealed a small amount of its utility space in this case. Consequently, the increase in Agent 3's revealed utility space in this threshold adjustment is the largest among these three agents. In the protocol, this process is repeated until an agreement is achieved or until they cannot find any agreement. The mediator or the mechanism designer defines the exact rate of the change in the size of the revealed utility space and the amount of threshold decrease. The threshold adjustment protocol was the first to propose an external loop for an effective consent mechanism. The details of the threshold adjustment mechanism are shown in Algorithm 3.

The threshold adjustment process could reduce the computational cost of deal identification in Step 4 of the basic bidding protocol. The original Step 4 incurs an exponential computational cost because the computation consists of combinatorial optimization. In the proposed threshold adjustment process, agents incrementally reveal their utility spaces as bids. Thus, for each round, the mediator computes only the new combinations of bids submitted in that round. This process reduces the computational cost.

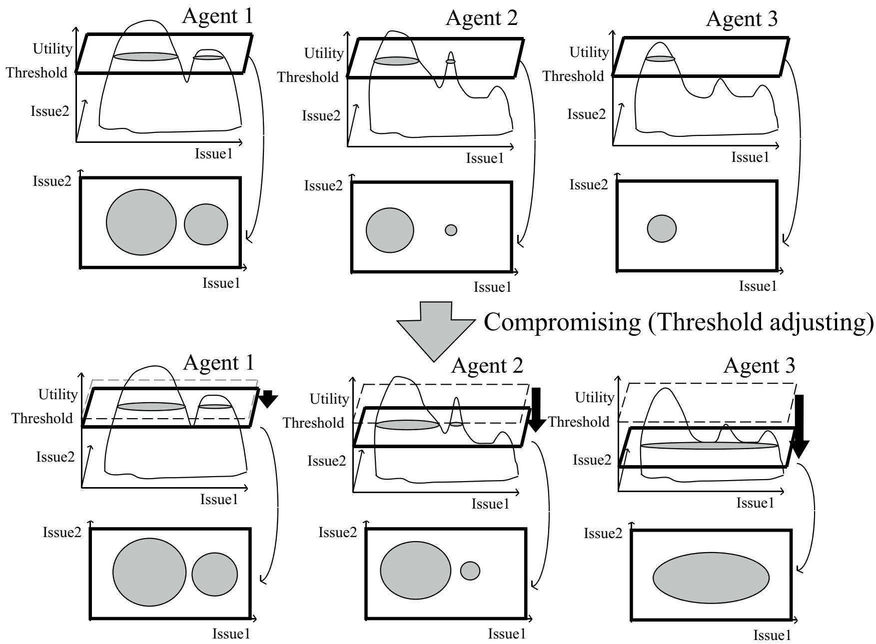


Fig. 3 Threshold adjustment process among three agents. The upper and lower panels show the thresholds and the revealed areas before and after threshold adjustments, respectively. Specifically, agent 3 revealed a small amount of its utility space in this case. Consequently, the increase in agent 3's revealed utility space in this threshold adjustment is the largest among these three agents

3.2 Experiments

Experimental Setting

The several experiments are conducted to evaluate the effectiveness of the proposed approach. 100 negotiations between agents in each experiment with randomly generated utility functions were ran. The threshold adjustment protocol was compared with the existing protocol without threshold adjustment in terms of optimality and privacy.

In the experiments on optimality, an optimizer to the sum of all the agent's utility functions was applied to find the contract with the highest possible social welfare. This value was used to assess the efficiency of the negotiation protocols (i.e., how closely they approached the optimal social welfare). Simulated annealing (SA) is used to find the optimum contract because an exhaustive search became intractable as the number of issues grew very large. The SA initial temperature was 50.0 and decreased linearly to 0 over the course of 2500 iterations. The initial contract for each SA run was randomly selected.

Algorithm 3 Threshold_adjustment( )

- Ar: Area Range of each agent ($Ar = \{Ar_0, Ar_1, ..., Ar_n\}$)
- *Bid_generation_with_SA(Th_i, V, SN, T, B_i)*: An Agent samples, adjusts and bids based on the basic bidding protocol.
- *Search_solution(B)*: The mediator employs breadth-first search with branch cutting to find social-welfare-maximizing overlaps. This step is based on the winner determination step of basic bidding protocol.

```
loop
  i := 1, B := Ø
  while i < |Ag| do
    bid_generation_with_SA(Th_i, V, SN, T, B_i)
  end while
  SC := Ø
  maxSolution := search_solution(B)
  if find maxSolution then
    maxSolution := getMaxSolution(SC)
    break loop
  else if all agent can lower the threshold then
    i := 1
    SumAr := Σ_{i∈|Ag|} Ar_i
    while i < |Ag| do
      Th_i := Th_i − C ∗ (SumAr − Ar_i)/SumAr
      i := i + 1
    end while
  else
    break loop
  end if
end loop
return maxSolution
```

Regarding privacy, the measure is the range of the revealed area. Namely, if an agent reveals one point of the utility space grid, it loses one privacy unit. If it reveals 1000 points, it loses 1000 privacy units. The revealed rate is defined as (Revealed rate) = (Revealed area)/(Entire area of utility space).

The parameters for the experiments were as follows: The number of agents is $N = 3$. The number of issues ranges from 2 to 10, and the domain for issue values is [0, 9]. The utility function per agent has 10 unary constraints, 5 binary constraints, 5 ternary constraints, and so forth. (a unary constraint is related to one issue, a binary constraint is related to two issues, and so on). The maximum value for a constraint is 100 $\times$ (Number of Issues). Constraints that satisfy many issues thus have larger weights on average. It seems reasonable for many domains. In meeting scheduling, for example, higher-order constraints affect more people than lower-order constraints, and hence they are more important. The maximum width for a constraint is 7. The following constraints, therefore, would all be valid: issue 1 = [2, 6], issue 3 = [2, 9], and issue 7 = [1, 3].

Three types of protocols were compared.

(A) w/o Threshold Adjustment: The basic bidding protocol is applied [43]. This protocol exhaustively explores the entire utility space.
(B) w/o Threshold Adjustment, w/ Bid Limitation: The basic bidding protocol with bids limitations is applied [43]. This protocol exhaustively explores the entire utility space. However, the number of an agent's bids is limited to $\sqrt[N]{6400000}$.
(C) w/ Threshold Adjustment: The proposed adjustment protocol is applied. This protocol does not have an explicit limitation on the number of bids. Each mechanism determines the amount of threshold decrease as $50 \times (\text{SumAr} - \text{Ar}_i)/\text{SumAr}$. SumAr is the sum of all agent's revealed areas and Ar_i indicates agent_i's revealed area.

The number of samples taken during random sampling is (Number of issues) × 200. The annealing schedule for sample adjustment is initial temperature 30, 30 iterations. Note that it is crucial that the annealer does not run very long or become very hot because then each sample will tend to find the global optimum instead of the peak of the optimum nearest the sampling point. The threshold used to select the bids to be made begins at 900 and decreases to 200 in the threshold adjustment mechanism. The protocol without the threshold adjustment process defines the threshold as 200. The threshold is used to eliminate contract points that have low utility. The limitation on the number of bids per agent is $\sqrt[N]{6400000}$ for N agents. Therefore, it was practical to run the deal identification algorithm only if it explored no more than about 6400000 bid combinations, which implies a limit of $\sqrt[N]{6400000}$ bids per agent, for N agents. In the experiments, 100 negotiations were ran in every condition. The code was implemented in Java 2 (1.5) and ran on a Core^{TM} 2 Duo processor iMac with 1.0 GB of memory under Mac OS × 10.4.

Experimental Result

Table 1 shows revealed rate (%), optimality rate, and number of bids of the comparable mechanisms. The mechanism without either threshold adjustment or bid limitation (A) increases the revealed rate. This means that if threshold adjustment and bid limitation are used, agents need to reveal much more of their utility space than in other mechanisms. The bid limitation is effective for keeping the increase in the revealed rate small. The revealed rate of the mechanism with bid limitation but without threshold adjustment starts decreasing when the number of issues is five; the reason is that bid limitation becomes active. Compared with the above two mechanisms, the mechanism with threshold adjustment drastically decreases the revealed rate.

The proposed threshold adjustment mechanism can effectively reduce the revealed rates. It shows that the optimality yielded by the proposed mechanism is very competitive with other mechanisms. Regarding optimality, the difference among (A), (B), and (C) is small, at a maximum of around 0.1 for about three to seven issues. When the amount of threshold decrease is not large, say 50, agents could miss agreement

Table 1 Revealed rate (%), optimality rate, and number of bids in the experiment. (A) w/o threshold adjustment, (B) w/o threshold adjustment, w/ bid limitation, (C) w/ threshold adjustment are compared in each metric

♯ of issues	Revealed rate (%)			Optimality rate			Number of bids		
	(A)	(B)	(C)	(A)	(B)	(C)	(A)	(B)	(C)
2	25.54	25.43	24.90	0.927	0.923	0.926	729	2028	1872
3	27.28	17.11	26.57	0.960	0.903	0.934	21952	146068	151686
4	34.31	10.96	34.29	0.965	0.893	0.931	194880	3373956	3326832
5	42.93	5.49	39.79	0.965	0.871	0.917	1329558	31649280	32256969
6	48.67	4.47	17.19	0.947	0.858	0.897	4424472	62097946	146866468
7	53.39	3.20	6.61	0.910	0.852	0.886	12037088	63202797	451196900
8	56.24	2.63	3.71	0.860	0.841	0.840	22945923	63521199	842949250
9	58.92	2.49	2.67	0.837	0.814	0.817	29855434	63521199	1348980237
10	69.58	2.09	2.02	0.813	0.804	0.800	42114800	63521199	2072179584

points with larger total utilities. It occurs when some agents have higher utility on an agreement point, but others have much lower utility on that point. (A) forces agents to submit to all agreement points with a larger utility than the minimum threshold. Thus, it can find such cases. However, (B) and (C) fail to capture such cases when the amount of decrease is small.

The number of bids indicates the utility space that must be explored and the time needed to find a possible deal. The number of bids for (A) increases exponentially. Actually, this program fails to compute the combinations completely at more than six issues when using (A). Threshold Adjustment drastically reduces the number of bids. (C) manually limits the number of bids. The increase in the number of bids stops at the limit defined above. On the other hand, the proposed mechanism successfully reduces the number of bids drastically.

4 Secure and Efficient Negotiation Protocols

Distributed Mediator Protocol (DMP) and the Take it or Leave it (TOL) Protocol are proposed, which makes agreements and conceals agent utility values. In the DMP, it is assumed that there is many mediators who search the utility space to find agreements. When searching in their search space, they employ the multi-party protocol to simultaneously calculate and conceal the sum of the per-agent utility value. Furthermore, the DMP scales better with the complexity of the utility space by dividing the search space between the mediators. In the TOL Protocol, the mediator searches using the hill-climbing (HC) search algorithm. The evaluation value is determined by the agents' responses, who either take or leave an offer to move from the current state to a neighboring state.

The Hybrid Secure Protocol (HSP) is also proposed, combining the DMP and TOL. In the HSP, TOL is performed first to improve the initial state in the DMP step. Next, the DMP is performed to find the local optima in the neighborhood. The HSP can also reach an agreement and conceal the per-agent utility information. Additionally, it can reduce the amount of memory required to make an agreement, which is a major issue in the DMP. Moreover, the HSP can reduce the communication cost (memory usage) more than the DMP can.

Although the DMP and HSP describe interactions among agents and mediators, they do not define the agreement search method, which is how the mediator searches for and finds agreement points. Thus, three agreement search methods are compared: HC, simulated annealing (SA), and a genetic algorithm (GA). HC and SA have been employed in previous works [43]. However, GAs also perform well in finding highly optimal contracts. Therefore, a GA-based method is compared with the other methods.

4.1 Secure Negotiation Protocol

Distributed Mediator Protocol (DMP)

It is assumed that there are more than two mediators (i.e., a distributed mediator) so that the DMP achieves distributed search and protection of the agents' private information by employing a multi-party protocol [55, 57]. The DMP is described as follows.

There are m mediators ($M_0, \ldots, M_m$) who can calculate the sum of all the agent utility values if k mediators get together, and n agents ($Ag_0, \ldots, Ag_n$). All mediators share q, which initially a prime number.

Step 1: The mediators divide the utility space (search space) and choose which mediator will manage it. The method of dividing the search space and assigning tasks is beyond the scope of this discussion. Parallel computation is possible if the search space is divided. This means that the computational complexity during searching can decrease.

Step 2: Each mediator searches its search space with a local search algorithm [58]. HC and SA are examples of local search algorithms. The objective function using a local search algorithm is used to maximize the social welfare. During the search, the mediator declares a multi-party protocol if it is searching in the state for the first time. Next, the mediator selects k mediators from all the mediators and asks for all agents to generate v (share).

Step 3: Agent i (A_i) randomly selects a k-dimensional formula that fulfills $f_i(0) = x_i$, and calculates $v_{i,j} = f_i(j)$. (x_i: agent's i's utility value). Next, A_i sends $v_{i,j}$ to M_j.

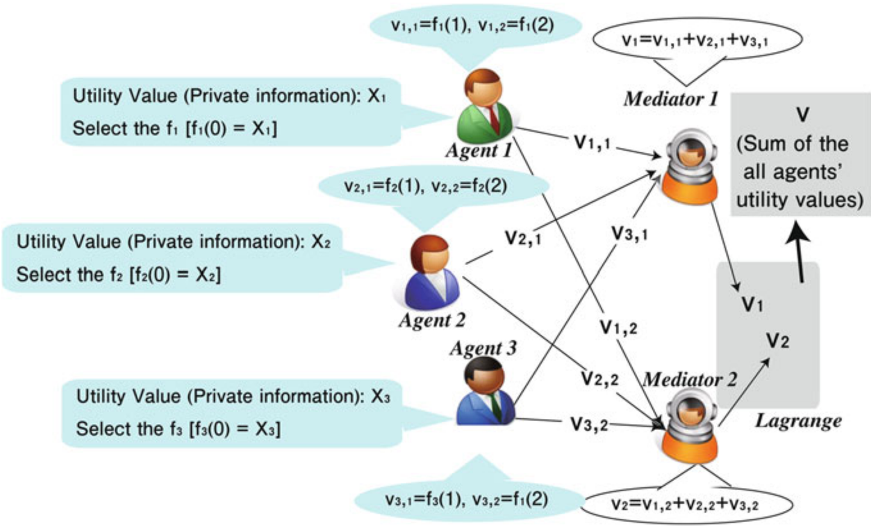


Fig. 4 Flow in distributed mediator protocol (DMP). There are three agents and two mediators. If two mediators get together, they can calculate the sum of the per-agent utility values. The skyblue area shows the steps that the agents perform without revealing them. As the figure indicates, the sum of all agent utility values can be calculated, and the values can be concealed by selecting the multi normal (f_i), generating the share (v), adding the share, and applying Lagrange s interpolating polynomial

Step 4: Mediator j (M_j) receives $v_{1,j}, \ldots, v_{n,j}$ from all the agents. M_j calculates $v_j = v_{1,j} + \cdots + v_{n,j} \ mod \ q$ and reveals v_j to other mediators.

Step 5: The mediators calculate the $f(j)$ that fulfills $f(j) = v_j$ by Lagrange's interpolating polynomial. Finally, s, which fulfills $f(0) = s$, is the sum of all the agents' utility values.

Steps 2–5 are repeated until they fulfill the at-end condition in the local search algorithm.

Step 6: Each mediator announces the maximum value (alternative) in his space to all mediators. Next, the mediators select the maximum value from all the alternatives.

Figure 4 shows the flow in the DMP. There are three agents and two mediators. If two mediators get together, they can calculate the sum of the per-agent utility values. The gray area shows the steps that the agents perform without revealing them. As the figure indicates, the sum of all agent utility values can be calculated, and the values can be concealed by selecting the multi normal (f_i), generating the share (v), adding the share, and applying Lagrange s interpolating polynomial.

The DMP has the advantages of keeping an agent's utility information private and scaling well with the size of the utility space. The details are given as follows.

Privacy: The DMP can calculate and conceal the sum of all the agents' utility values. The proof is identical to that for the multi-party protocol [57]. In the DMP, other agents and mediators cannot know the utility values without illegal collusion.

Additionally, k, which is the number of mediators performing the multi-party protocol, represents the trade-off between privacy issues and computational complexity. If k mediators exchange their shares (v) illegally, they can expose the agent utility values. Therefore, to protect an agent's private information, k should be so large that mediators are discouraged from colluding illegally because it requires considerable effort. However, a large k requires more computation time because more mediators have to stop searching.

Scalability: The computational cost can be greatly reduced because the mediators divide the search space. In existing protocols, they cannot find better agreements when the search space becomes huge. However, by dividing the search space, this protocol can locate better agreements in large search spaces.

The DMP has a limitation: Too many shares (v) are generated. This is because the shares are generated that correspond to the search space. Generating shares incurs a much greater communication cost as the number of agents increases than searching without generating shares. Thus, it is necessary to generate fewer shares with high optimality.

Take it or Leave it (TOL) Protocol for Negotiation

Take it or Leave it (TOL) Protocol is proposed, which can also reach agreements and conceal all the agents' utility information. The mediator searches using the HC search algorithm [58], which is a simple loop that continuously moves in the direction of increasing evaluated value. The values of each contract are evaluated by the decisions that agents make to take or leave offers to move from the current state to the neighboring state. The agents can conceal their utility value using this evaluation value. This protocol consists of the following steps.

Step 1: The mediator randomly selects the initial state.

Step 2: The mediator asks the agents to move from the current state to a neighboring state.

Step 3: Each agent compares its current state with the neighboring state and determines whether to take the offer or leave it. The agent takes the offer if the neighboring state provides a higher utility value than the current state. If the current state provides a higher or identical utility value than the neighboring state, the agent rejects (leaves) the offer.

Step 4: The mediator selects the next state that is declared by most agents as "take it." However, the mediator selects the next state randomly if more than two states are tied for being declared as "take it." The mediator can prevent local maxima from being reached by random selection.

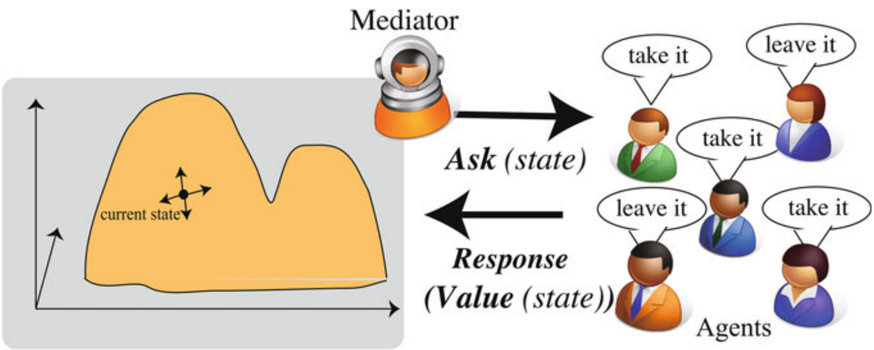


Fig. 5 Take it or leave it (TOL) Protocol. First, the mediator informs agents about the state whose evaluation value he wants to know. Second, agents search for their utility space and declare "take it" or "leave it." It determines the number of agents who declare "take it" (VALUE (state)). These steps are repeated until they satisfy the at-end condition

Steps 2, 3, and 4 are repeated until all agents declare "leave it," or the mediator determines that a plateau has been reached. A plateau is an area of the state space landscape where the evaluation function is flat.

Figure 5 shows the concept of the "Take it or Leave it (TOL) Protocol." First, the mediator informs agents about the state whose evaluation value he wants to know. Second, agents search for their utility space and declare "take it" or "leave it." It determines the number of agents who declare "take it" (VALUE (state)). These steps are repeated until they satisfy the at-end condition. The TOL Protocol has the advantage of lower time complexity because it easily rates the evaluated value. However, it cannot find optimal solutions when a plateau is reached.

4.2 Hybrid Secure Negotiation Protocol (HSP)

A proposed protocol that combines the DMP with TOL is proposed to address the DMP's limitation. This new protocol is called the HSP, which generates fewer shares than the DMP. It is described as follows.

Step 1: The mediators divide the utility space (search space) and choose a mediator to manage it.

Step 2: Each mediator searches its search space using TOL. The initial state is randomly selected. By performing TOL initially, the mediators can find somewhat more optimal solutions without generating shares (v).

Step 3: Each mediator searches its search space using steps 2–5 in the DMP as proposed in Sect. 4.1. The initial state is the solution found in the previous step. By performing the DMP after TOL, mediators can find the local optima in the neighborhood and conceal each agent's private information.

Steps 2 and 3 are repeated many times by changing the initial state.

Step 4: Each mediator communicates the maximum value (alternative) in their space to all the mediators. Next, the mediators select the maximum value from all the alternatives. Finally, they propose this alternative as the agreement point.

The HSP can find solutions with fewer shares than the DMP because the initial state in Step 3 is higher than that when only the DMP is performed. In addition, TOL does not generate shares, and the DMP searches in states in which TOL has not searched. Thus, the HSP can reduce the number of shares. Furthermore, TOL and the DMP can protect the agent's utility value (private value). Therefore, HSP can also preserve the agent's utility value.

Moreover, the HSP yields higher optimality. This is because TOL usually stops searching after reaching a plateau. Additionally, the main reason for lowering the optimality in the DMP is to reach the local optima, although the initial value in Step 3 is usually different because it is determined by TOL. Therefore, the HSP can produce agreements with higher optimality.

4.3 Experiments

Experimental Setting

100 negotiations between agents were ran in each experiment with randomly generated utility functions. In these experiments, the number of agents was six, and the number of mediators was four.

The following methods were compared:

- "(A) DMP (SA)" is the Distributed Mediator Protocol, and the search algorithm is simulated-annealing [58].
- "(B) DMP (HC)" is the Distributed Mediator Protocol, and the search algorithm is hill-climbing [58].
- "(C) DMP (GA)" is the Distributed Mediator Protocol, and the search algorithm is the genetic algorithm [58].
- "(D) HSP (SA)" is the hybrid secure protocol, and the search algorithm in the distributed mediator step is simulated annealing.
- "(E) HSP (HC)" is the hybrid secure protocol, and the search algorithm in the distributed mediator step is the hill-climbing algorithm.

In the optimality experiments, an optimizer is applied to the sum of all the agents' utility functions for each run to find the contract with the highest possible social welfare. This value was used to assess the efficiency of the negotiation protocols (i.e., how closely they approached the optimal social welfare). To find the optimum contract, SA is used because intractable as the number of issues grew very large. The SA initial temperature was 50.0 and decreased linearly to 0 throughout 2500 iterations. The initial contract for each SA run was randomly selected. The optimality rate is defined as (Maximum utility value calculated by each method)/(Optimum contract value using SA).

The number of agents was six, and the number of mediators was $2^{(\text{the number of issues})}$. In the DMP, the mediators can calculate the sum of the per-agent utility values if four mediators get together and the search space is divided equally.

Utility function: The domain for the issue values is [0, 9]. The constraints include 10 unary constraints, 5 binary constraints, 5 ternary constraints, and so forth (a unary constraint is related to one issue, a binary constraint is related to two issues, and so on). The value for a constraint is 100 × (Number of Issues). Constraints that satisfy many issues have, on average, which seems reasonable for many domains. To schedule meetings, for example, higher-order constraints affect more people than lower-order constraints; hence, they are more important. The maximum width for a constraint is 7.

The following parameters are set for HC, SA, and GA.

Hill climbing (HC): The number of iterations is 20 + (Number of issues) × 5. The final result is the maximum value achieved.

Simulated annealing (SA): The annealing schedule for the DMP includes an initial temperature of 50. For each iteration, the temperature is decreased by 0.1. Thus, it decreases to 0 after 500 iterations. 20 × (Number of issues) × 5 were conducted to search while varying the initial start point. The annealing schedule for the HSP in the DMP step includes an initial temperature of 10 with 100 iterations. Note that the annealer must not run very long or become very hot because then each initial state obtained by TOL will tend to find the global optimum instead of the peak of the optimum nearest the initial state in DMP.

Genetic algorithm (GA): The population size in one generation is 20+(Number of Issues) × 5. A basic crossover method combining two parent individuals to produce two children (one-point crossover) is used. The fitness function is the sum of all the agents' (declared) utility. 500 iterations were conducted. Mutations occurred with a very small probability. In a mutation, one of the issues in a contract vector was randomly chosen and changed. In the GA-based method, an individual is defined as a contract vector.

The code was implemented in Java 2 (1.5) and ran on a Core^{TM} 2 Duo processor iMac with 1.0 GB of memory under Mac OS X10.5.

Experimental Results

Table 2 shows the optimality rates and the average of shares (v) of five protocols. For (B) DMP (HC), the rate decreases rapidly as the number of issues increases because HC reaches local optima by increasing the search space. For (C) DMP (GA), it does not decrease rapidly even if the number of issues increases. Additionally, (A) DMP (SA) is the same as the optimal solution. Therefore, the optimality depends on the search algorithm in the DMP. (D) HSP (HC) achieves high optimality because the HSP performs the DMP after performing TOL. In addition, (D) HSP (HC) achieves

Table 2 Optimality rate and the number of shares per agent in the experiment. "(A) DMP (SA)," "(B) DMP (HC)," "(C) DMP (GA)," "(D) HSP (SA)," and "(E) HSP (HC)"are compared in each metric

♯ of issues	Optimality rate					♯ of shares per agent				
	(A)	(B)	(C)	(D)	(E)	(A)	(B)	(C)	(D)	(E)
3	1.000	0.999	1.000	0.979	1.000	435	307	784	267	201
4	1.000	0.999	1.000	0.966	1.000	1394	1148	3381	1200	511
5	1.000	0.998	1.001	0.952	1.000	3912	2778	8844	3068	901
6	1.000	0.997	0.999	0.936	0.999	8094	5551	17133	6634	1354
7	1.000	0.996	0.999	0.917	0.997	14708	9815	29337	11582	1866
8	1.000	0.991	0.999	0.901	0.996	23508	16142	44498	19647	2434
9	1.000	0.990	0.998	0.888	0.994	35893	24878	63145	30413	3057
10	1.000	0.987	0.997	0.880	0.992	38050	26003	65590	32862	3187

higher optimality than (C) HSP (SA) because SA in the DMP step sometimes stops searching for a worse state than the initial state owing to its random nature. In contrast, HC stops searching for a better state than the initial state.

The number of shares enables us to compare the memory usage of the protocols. That for (C) DMP (GA) increases exponentially. On the other hand, (A) DMP (SA) and (B) DMP (HC) use fewer shares than (C) DMP (GA) because GA searches for more states than SA and HC. The number of shares in the DMP depends on the features of the search protocol. Furthermore, (C) HSP (SA) and (D) HSP (HC) use fewer shares than (A) DMP (SA), (B) DMP (HC), and (C) DMP (GA) because the initial state in the DMP step in the HSP has a higher value than the initial state in the DMP because TOL was performed first. Thus, the HSP can reduce the number of shares more than the DMP can.

5 Secure and Fair Protocol that Addresses Weaknesses in Nash Bargaining Solution

The Nash bargaining solution, which maximizes the product of the agent utilities, is a well-known metric that provably identifies the optimal (fair and social-welfare-maximizing) agreement for negotiations in linear domains [8, 52, 53]. In *nonlinear* domains, however, the Pareto frontier will often not satisfy the convexity assumption required to make the Nash solution optimal and unique [8, 52, 54]. In other words, in nonlinear domains, multiple agreements can satisfy the Nash bargaining solution, and many or all of these will have sub-optimal fairness and social welfare. Therefore, a new approach is necessary to produce good outcomes for nonlinear negotiations.

A secure mediated protocol (SFMP) that addresses this challenge is presented. The protocol consists of two primary steps. In the first step, the SFMP uses a nonlinear optimizer, integrated with a secure information-sharing technique called the secure gathering protocol [55], to find the Pareto frontier without causing agents to reveal private utility information. In the second step, an agreement is selected from the set of Pareto-optimal contracts using a metric called approximated fairness, which measures how equally the total utility is divided across the negotiating agents (e.g., [56]). It shows that SFMP produces better scalability and social welfare than previous nonlinear negotiation protocols.

5.1 Weaknesses of the Nash Bargaining Solution in Nonlinear Negotiation

Working in the nonlinear domain has some important impacts on the types of negotiation protocols that can be effective. First, consider Pareto-optimality, which is widely recognized as a basic requirement for a good negotiation outcome. It is defined as follows: Contract $\mathbf{s} = (s_1, \ldots, s_M)$ is *Pareto optimal* if there is no $\mathbf{s}'$ such that $u_i(\mathbf{s}') > u_i(\mathbf{s})$ for all agents ($u_i(\mathbf{s})$ is agent i's utility value). Pareto-optimality thus eliminates all contracts when others exist that are better for all the parties involved. In a linear negotiation (i.e., where the agent utility functions are defined as the weighted sum of the values for each issue), it is computationally trivial to find the Pareto frontier and the social welfare (sum of agent utilities) for every contract on the Pareto frontier is the same. In fact, the Pareto-optimal frontier for negotiation will be sparse in the proposed model, i.e., the Pareto-optimal contract points will be few and widely scattered.

Next, let us consider fairness. Fairness is critical in bargaining theory because some experimental results suggest that it profoundly influences human decision-making (e.g., [59]) in such contexts as family decision-making (e.g., where will we go on our next vacation?), the less formal economy of consumer transactions (such as ticket scalpers or flea markets), and price setting for consumer purchases. The ultimatum game is a popular example of this effect [60, 61]. People tend to offer "fair" (i.e., 50:50) splits, and offers of less than 20% are often rejected in this game, even though it is irrational to reject any deal because the alternative is a zero payoff. There are many other studies about the relationship between decision-making and fairness in experimental and behavioral economics [29, 62].

The Nash bargaining solution (i.e., the contract that maximizes the Nash product = the product of the agents' utility functions) is widely used for identifying the fairest contract from those that make up the Pareto frontier. As shown in Fig. 6, the Nash bargaining solution divides the utility equally among the negotiating parties in a linear domain. It can be proven that there is a unique Nash bargaining solution for

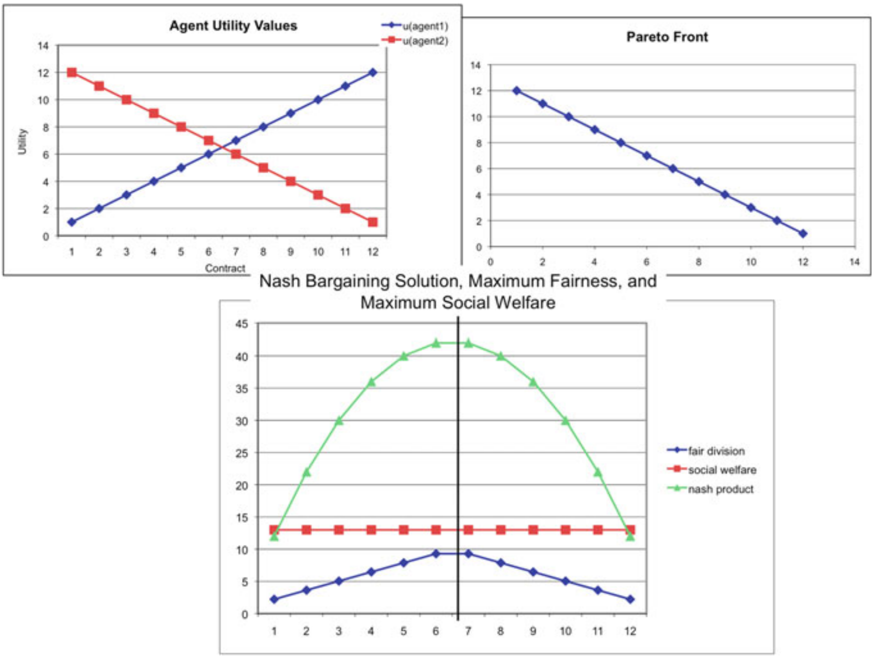


Fig. 6 Relationships among Nash product, fairness, and social welfare in a linear utility function

negotiations with convex Pareto frontiers, which is satisfied trivially for negotiations with linear utilities [8][3].

These properties change radically in nonlinear negotiation. As shown in Fig. 7, when agents have nonlinear utility functions, the Pareto frontier can be non-convex [63]. Multiple Nash bargaining solutions can exist, even with continuous issue domains, and some of them may be non-optimal in terms of social welfare and fair division of utility. It is even straightforward to find nonlinear cases where all the contracts on the Pareto frontier are Nash bargaining solutions, although many diverge widely from maximal fairness and social welfare. The Nash bargaining solution concept, widely used as a basis for negotiation protocols for linear domains, will thus often fare poorly in nonlinear domains. Therefore, it is necessary to find negotiation protocols that can achieve high social welfare and fairness values with nonlinear agent utilities.

[3] In discretized issue domains, multiple Nash bargaining solutions can exist, but they will all be clustered immediately beside each other and thus offer similar fairness values.

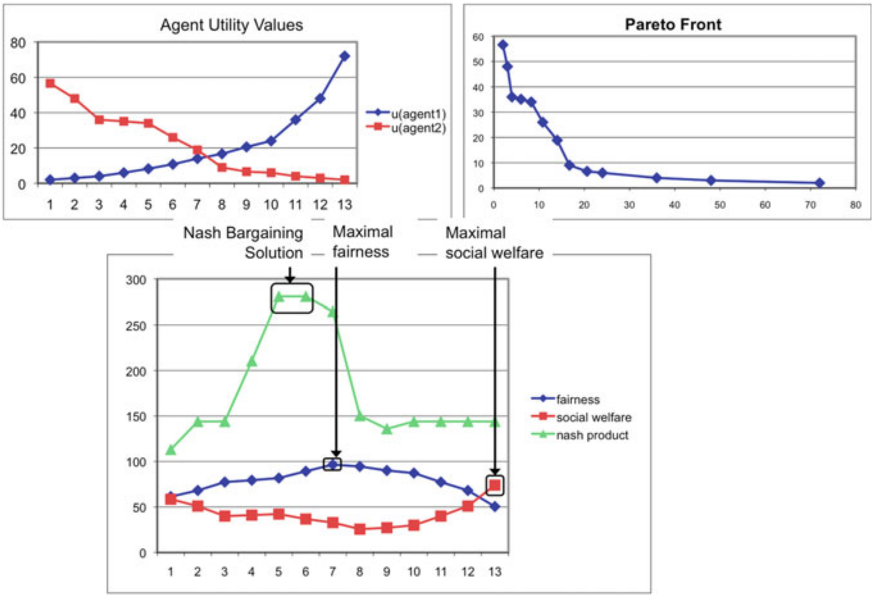


Fig. 7 Relationship among Nash product, fairness, and social welfare in a non-linear utility function

5.2 *Secure and Fair Mediator Protocol with Approximated Fairness*

The SFMP was defined to achieve these goals while protecting agents' private utility information. It consists of two primary steps: (1) finding the set of Pareto-optimal contracts and (2) selecting a fair contract from that set. These steps are defined below.

- Finding the Pareto Frontier: This step is achieved using a mediated approach [64, 65]. The mediators use this preference information to provide the objective function for a non-linear optimization technique such as simulated annealing (SA) or a genetic algorithm (GA). Over the course of multiple rounds, the mediators converge on the set of Pareto-optimal contracts. As is common in negotiation contexts, that agents prefer not to share their utility functions with others in order to preserve a competitive edge. Accordingly, the protocol uses a secure gathering protocol based on a multi-party protocol [55] to ensure that mediators can calculate the sum of the agents' utilities without learning, or revealing, the individual agent's utility information.
- Selecting the Final Agreement: The SFMP selects the final agreement from the Pareto-optimal contract set by calculating the fairest. Several definitions of fair have been identified in social choice and game theory [56]. Suppose that a division $X = X_1 \cup \cdot \cup X_n$ among n agents where agent i receives X_i. "Simple" fair division results if $u_i(X_i) \geq 1/n$ whenever $1 \leq i \leq n$ (each agent gets at least $1/n$.) Another

definition, from game theory, calls a division X is fair if and only if it is Pareto-optimal and envy-free [66]. A division is "envy-free" if no agent feels another has a strictly larger piece of the utility [56].

The simple fair division is considered as the concept of fairness. Contract agreements, in general, rarely fully satisfy this condition. Accordingly, it is measured that how close an agreement is to simple fair division by calculating its "approximated fairness" , i.e., the deviation of each agent's utility from the average of the total utility. The approximated fairness of a contract is formally defined as follows:

$$V(u_1, \ldots, u_n) = \sum_{i=1}^{n} \frac{(u_i - \overline{u})^2}{n}$$

$(u_1, \ldots, u_n$: agent's utility value in contract,
$\overline{u}$: the average of all agent's utility value).

An ideal contract, therefore, has an approximated fairness value of zero, and all other contracts will have larger values. The final agreement selected by the protocol is the contract from the Pareto-optimal set with the smallest approximated fairness value.

Note that the fairness concept is equivalent to the Nash bargaining solution in linear contexts with continuous issue domains. Assume that $u_1 + u_2 + \cdots + u_n = K$ (constant) (where u_i: agent i's utility value). The Nash product is maximized when $u_1 = u_2 = \cdots = u_n = K/n$ (this has been proven mathematically in the field of isoperimetric problems). The approximated fairness does not, however, correspond to the Kalai-Smorodinsky solution because the latter is not always fair [67].

5.3 *Experiments*

A series of negotiation simulation experiments were ran to demonstrate the weaknesses of the Nash bargaining solution in non-linear domains and to compare the performance of the SFMP protocol with that of previous approaches. The sub-sections below describe the experimental setup and results.

Detailed Description of Secure and Fair Mediator Protocol (SFMP)

The SFMP uses multiple mediators to help ensure agent privacy. There are $k = mn$ mediators M_j and n agents (A_i), where m is an arbitrary integer. Note that this approach requires that m be relatively high to effectively conceal the agents' private information. If the number of mediators is low, it is more likely that all the mediators will collude and thus compromise the agents' privacy.

(Optional Pre-Negotiation Step) Contract Space Division among Mediators: The mediators divide the contract space between them so that each mediator searches a different sub-region. Suppose, for example, there are two issues whose domain is the integers from 0 to 10. In this case, Mediator 1 can manage the region of values from 0 to 5 for Issue 1 and from 0 to 10 for Issue 2, while Mediator 2 can manage the region of values from 6 to 10 for Issue 1 and from 0 to 10 for Issue 2. This step is optional, but it has the advantage of potentially reducing the time needed to search the contract space by allowing parallel computation.

(Step1) Secure Search to Find a Pareto-optimal Contract Set: Each mediator searches its assigned portion of the contract space using a local search algorithm [58]. The experiments employed hill-climbing (HC), SA, and GA. In HC, an agent starts with a random solution, makes random mutations at each step, and selects the one that causes the most significant utility increase. When the algorithm cannot find any more improvements, it terminates. In SA, each step of the SA algorithm replaces the current solution with a randomly generated nearby contract, with a probability that depends on the change in the utility value and a global parameter T (the virtual temperature) that is gradually decreased during the process. The agent moves almost randomly when the temperature is high but acts increasingly like a hill climber as the temperature decreases. When T is 0, the search is terminated. The advantage of SA is that it can avoid getting stuck in the local optima that occur in non-linear optimization problems and often finds more optimal solutions than HC. GA is a search technique inspired by evolutionary biology, using inheritance, mutation, selection, and crossover techniques. First, many individual contracts are randomly generated to form an initial population. Next, at each step, a proportion of the existing population is selected based on its fitness (i.e., utility values). Crossover and mutation are then applied to these selections to generate the next generation of contracts. This process is repeated until a termination condition is reached. The objective function of all these local search algorithms is social welfare maximization. At each search step, the mediators determine the social welfare values by securely gathering their assigned agents' utility values for the current contract(s). It called as secure value gathering.

(Step 2) Identify Agreement: All mediators share the maximum value in their sub-region of the contract space with all other mediators. On the basis of these values, they identify the Pareto-optimal contract set. The mediators then select the contract in that set that minimizes the approximated fairness metric. This represents the final agreement for that negotiation.

Nash Product Maximization Search (NPMS)

For a comparison case, the Nash Product Maximization Search (NPMS) is used to find the Nash bargaining solution for the tests [58]. The implementation used SA to maximize the Nash product for the negotiating agents, gathering their utility values using the secure gathering protocol. SA has been shown to be very effective for nonlinear optimization tasks [43]. NPMS can solve to assess the scale of the performance decrement caused by using the Nash bargaining solution concept in nonlinear domains.

Experimental Setting

Five experiments were conducted to evaluate the effectiveness of the approach. 100 negotiations between agents in each experiment with randomly generated utility functions were ran. The number of agents was six, and the number of mediators was four. The mediators could calculate the sum of the agents' utilities. The search space was divided equally amongst the mediators. The domain for the issue values was [0, 9]. The constraints included 10 unary constraints, 5 binary constraints, 5 ternary constraints, and so on (a unary constraint relates to one issue, a binary constraint relates to two issues, and so on). The maximum value for a constraint was 100 $\times$ (Number of issues). Constraints that satisfy many issues thus have, on average, larger utility, which seems reasonable for many domains. In scheduling meetings, for example, higher-order constraints affect more people than lower-order constraints, which are more important. The maximum width for a constraint was 7. The following constraints, for example, are both valid: Issue 1 = [2, 6] and Issue 3 = [2, 9].

The following negotiation protocols were compared: SFMP (SA), SFMP (HC), SFMP (GA), Nash Product Maximization Search (NPMS), Basic Bidding protocol, and Exhaustive Search.

- **(A) SFMP (SA):** This is SFMP using SA as the optimization algorithm. The initial temperature was 50. For each iteration, the temperature decreased by 0.1, and so 500 iterations were performed. 20 + (Number of issues) $\times$ 5 searches were conducted, randomly changing the initial start point for each search.
- **(B) SFMP (HC):** This is SFMP using HC as the optimization algorithm. The random-restart HC mechanism [58] is employed. 20 + (Number of issues) $\times$ 5 searches were conducted, randomly changing the initial start point for each search.
- **(C) SFMP (GA):** This is SFMP using a GA as the optimization algorithm. The population size was *20 + (Number of issues)* $\times$ *5*. A basic crossover method is conducted combining two parent individuals to produce two children (one-point crossover). The fitness function was the sum of all the agents (declared) utility. 500 iterations were conducted. Mutations occurred with a tiny probability. In a mutation, one of the issues in a contract vector was randomly chosen and changed.
- **(D) Nash Product Maximization Search (NPMS):** NPMS used SA to search for the Nash bargaining solution(s), i.e., for contracts that maximize the Nash product., i.e., for contracts that maximize the Nash product. The initial temperature was 50 degrees. The temperature decreased by 0.1 degrees for each iteration, so 500 iterations were performed. 20 + (Number of issues) $\times$ 5 searches, changing start point randomly for each search. These settings are the same as those for SFMP (SA).

- **(E) Basic Bidding protocol:** The basic bidding protocol is that proposed [43]. The number of samples taken during random sampling is (Number of issues) × 200. The threshold used to remove contract points that have low utility is 200. The limitation on the number of bids per agent is $\sqrt[N]{6400000}$ for N agents. This method fails to reach an agreement if the number of issues exceeds eight because it is computationally very complex.
- **(F) Exhaustive Search:** An exhaustive search is a centralized brute-force algorithm that traverses the entire contract search space to find the Pareto-optimal contract set. The final agreement is then selected using the approximated fairness measure. This approach was computationally practical only when the number of issues was seven or fewer.

The code was implemented in Java 2 (1.5) and ran on a Core^{TM} 2 Duo processor iMac with 1.0 GB of memory under Mac OS X 10.5.

Experimental Result

Table 3 compares the social welfare, the number of Pareto-optimal contracts, and the variance in the agents' utilities for the final agreements achieved by these six methods.

About the social welfare, (A) SFMP (SA) and (C) SFMP (GA) performed similarly. Neither had fully optimal results, reflecting the difficulty of performing optimization in large non-linear contract spaces. All the SFMP protocols outperformed the basic bidding protocol, which was hampered by the limit on the number of bids per agent necessitated by the combinatorics of winner determination in this protocol. The performance of (B) SFMP (HC) decreased rapidly as the number of issues grew because HC became stuck on local optima. The performance of (A) SFMP (SA) and (C) SFMP (GA) did not decrease appreciably as the number of issues increased.

About success rate in finding Pareto-optimal contracts, (A) SFMP (SA) and (C) SFMP (GA) were better at finding Pareto-optimal contracts than either the NPMS or the basic bidding protocol. It makes sense because the SFMPs((A)–(C)) were explicitly designed to find the entire Pareto frontier before selecting a final agreement, whereas other protocols were not. (A) SFMP (SA) and (C) SFMP (GA) outperformed the basic bidding protocol because the latter often fails to find Pareto-optimal solutions owing to the limit on the number of bids allowed to each agent. As always, the performance of (B) SFMP (HC) decreased rapidly as the number of issues grew. (C) SFMP (GA) showed the highest performance on this measure because GA is inherently more suitable for finding Pareto-optimal contract sets. However, for all the methods, when the number of issues increased, the percentage of Pareto-optimal contracts found drastically decreased.

About the variance in the agents' utilities for the final agreements to assess their fairness, the SFMPs ((A)–(C)) outperformed the basic bidding protocol on this measure because the latter does not consider fairness when finding agreements. (C) SFMP (GA) showed the lowest (best) value among the SFMP variants. (D) NPMS outper-

Table 3 Social welfare, success rate in finding Pareto-optimal contracts. (A) SFMP (SA), (B) SFMP (HC), (C) SFMP (GA), (D) Nash Product Maximization Search (NPMS), (E) Basic Bidding protocol, and (F) Exhaustive Search are compared in each metric. If "–" is expressed, the score can't be obtained in practical time because of the computational complexity. The social welfare was (Social welfare for final agreement from method)/(Social welfare for final agreement from SFMP (SA)). As predicted, SFMP (SA) and SFMP (GA) outperformed NPMS, confirming the claim that the Nash bargaining solution produces sub-optimal outcomes when applied to non-linear negotiation

♯ of issues	Social welfare						Success rate in finding Pareto-optimal contracts					
	(A)	(B)	(C)	(D)	(E)	(F)	(A)	(B)	(C)	(D)	(E)	(F)
3	1.000	1.000	1.004	0.982	0.995	1.006	0.940	0.560	0.990	0.550	0.530	1.000
4	1.000	0.996	1.008	0.988	0.993	1.018	0.170	0.182	0.550	0.181	0.123	1.000
5	1.000	0.970	1.016	0.987	0.952	1.031	0.129	0.131	0.458	0.146	0.070	1.000
6	1.000	0.935	1.004	0.970	0.900	1.038	0.097	0.102	0.351	0.140	0.043	1.000
7	1.000	0.918	0.993	0.951	0.865	1.046	0.092	0.090	0.333	0.125	0.015	1.000
8	1.000	0.873	0.987	0.958	0.832	–	0.087	0.088	0.326	0.119	0.004	–
9	1.000	0.851	1.010	0.961	0.833	–	0.066	0.068	0.275	0.094	0.000	–
10	1.000	0.836	1.025	0.965	0.824	–	0.068	0.067	0.270	0.097	0.000	–
11	1.000	0.797	1.012	0.944	0.800	–	0.060	0.062	0.255	0.077	0.000	–
12	1.000	0.799	1.008	0.967	0.784	–	0.070	0.075	0.272	0.085	0.000	–
13	1.000	0.765	1.029	0.947	0.789	–	0.070	0.082	0.307	0.080	0.000	–
14	1.000	0.755	1.036	0.949	0.777	–	0.066	0.074	0.162	0.075	0.000	–
15	1.000	0.728	1.046	0.924	0.768	–	0.066	0.066	0.035	0.063	0.000	–

formed the SFMPs on this measure. It contradicts that the Nash bargaining solutions to vary widely in their fairness values, causing NPMS to produce sub-optimal fairness values on average.

These results can be explained by considering the allocation of computational effort in non-linear optimization. In an even moderately large non-linear optimization problem, the contract space is too large to explore exhaustively. For example, if there are only ten issues with ten possible values per issue, this produces a space of 10^{10} (10 billion) possible contracts. As a result, with limited computational resources, It is no guarantee of finding the complete Pareto frontier. The SFMP is presumably able to find only a subset of the Pareto-optimal contracts, and those are scattered over the entire frontier. Because the coverage is sparse, the SFMP will often not find the Pareto-optimal contract that optimizes the fairness metric. It will reduce the average fairness score for the SFMP. The NPMS, in contrast, devotes its entire computational effort to finding a single Nash-product-maximizing contract. Even though it is an inferior optimization objective, it has the benefit of a more concentrated application of computing restheces.

This interpretation is supported by Fig. 8, which shows the utility values for the SFMP ((A)–(C)) and (D) NPMS for a case with two agents and five issues with randomly generated non-linear utility functions. The diamond symbols indicate the

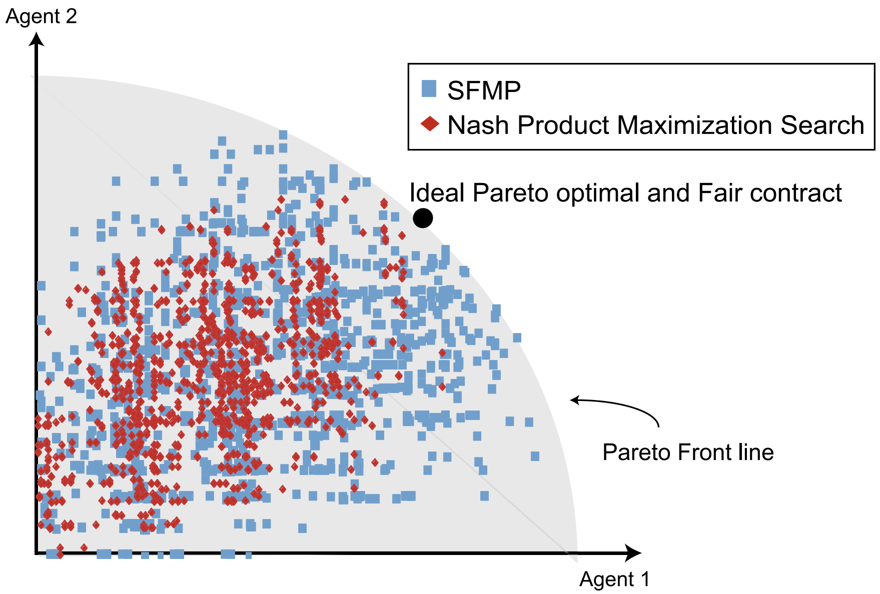


Fig. 8 Comparison of SFMP and NPMS in the outcome space. The diamond symbols indicate the contracts considered by the NPMS, and the square symbols indicate those considered by the SFMP

contracts considered by the NPMS, and the square symbols indicate those considered by the SFMP. Because the SFMP aims to find the entire Pareto frontier, it searches throughout the frontier. The NPMS, by contrast, aims to find the contract that directly maximizes the Nash product; hence, it focuses its search toward the middle of the Pareto frontier. In this case, the SFMP came closer to the Pareto frontier than the NPMS.

6 Decomposing the Contract Space Based on Issue Interdependencies

One of the main challenges in developing effective non-linear negotiation protocols is scalability; it can be challenging to find high-quality solutions when there are many issues owing to computational intractability. One reasonable approach to reducing computational cost while maintaining high-quality outcomes is decomposing the contract space into several independent sub-spaces. A method for decomposing a contract space into sub-spaces is proposed according to the agents' utility functions. A mediator finds sub-contracts in each sub-space based on votes from the agents and combines the sub-contracts to produce the final agreement. It is experimentally demonstrated that the proposed protocol allows highly optimal outcomes with greater scalability than previous efforts.

It is also addressed incentive compatibility issues [68]. Any voting scheme introduces the potential for strategic non-truthful voting by the agents, and the proposed method is no exception. For example, one of the agents may always vote truthfully. In contrast, another exaggerates so that its votes are always "strong." It has been shown that this biases the negotiation outcomes to favor the exaggerator at the cost of reduced social welfare. It is applied the limitation of strong votes to decomposing the contract space into several largely independent sub-spaces. It is investigated whether and how this approach can be applied to contract space decomposition.

6.1 Strength of Issue Interdependency

The *strength* of an issue interdependency is captured by the interdependency rate. A measure is defined for the interdependency between i_j and i_{jj} for agent a ($D_a(i_j, i_{jj})$) as follows:

$$D_a(i_j, i_{jj}) = \sharp\{c_k | \delta_a(c_k, i_j) \neq \emptyset \ \wedge \ \delta_a(c_k, i_{jj}) \neq \emptyset\}.$$

This measures the number of constraints that inter-relate the two issues.

Agents capture their issue interdependency information in the form of interdependency graphs, i.e., weighted non-directed graphs where a node represents an issue, an edge means the interdependency between issues, and the weight of an edge represents the interdependency rate between those issues. An interdependency graph is thus formally defined as:

$$G(P, E, w) : P = \{1, 2, \ldots, |I|\}(\text{finite set}),$$

$$E \subset \{\{x, y\}|x, y \in P\}, w : E \to R.$$

Figure 9 shows an example of an interdependency graph.

The objective function of the proposed protocol can be described as follows:

$$\arg\max_{\mathbf{s}} \sum_{a \in N} u_a(\mathbf{s}). \tag{1}$$

$$\arg\max_{\mathbf{s}} u_a(\mathbf{s}), \ (a = 1, \ldots, N). \tag{2}$$

This protocol, in other words, tries to find contracts that maximize social welfare, i.e., the summed utilities for all agents. Such agreements, by definition, will also be Pareto-optimal. At the same time, all agents try to find contracts that maximize their own welfare.

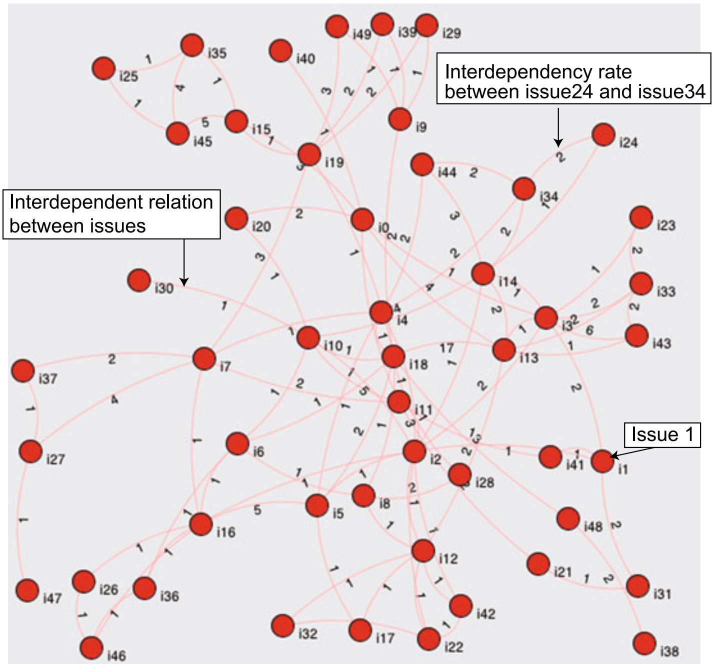


Fig. 9 Example of interdependency graph (50 issues). Agents capture their issue interdependency information in the form of interdependency graphs, i.e., weighted non-directed graphs where a node represents an issue, an edge means the interdependency between issues, and the weight of an edge represents the interdependency rate between those issues

6.2 Decomposing the Contract Space

Analyzing Issue Interdependency

The first step is for each agent to generate an interdependency graph by analyzing interdependencies in its own utility space.

Grouping issues

In this step, the mediator employs a breadth-first search to combine the issue clusters submitted by each agent into a consolidated set of issue groups. For example, if Agent 1 submits the clusters $\{i_1, i_2\}$, $\{i_3, i_4, i_5\}$, $\{i_0, i_6\}$ and Agent 2 submits the clusters $\{i_1, i_2, i_6\}$, $\{i_3, i_4\}$, $\{i_0\}$, $\{i_5\}$, the mediator combines them to produce the issue groups $\{i_0, i_1, i_2, i_6\}$, $\{i_3, i_4, i_5\}$. In the worst case, if all the issue clusters submitted by the agents have overlapping issues, the mediator generates the union of the clusters from all the agents. The details of this algorithm are given in Algorithm 4.

Algorithm 4 Combine_IssueGroups(G)

Ag: A set of agents, G: A set of issue-groups of each agent
($G = \{G_0, G_1, ..., G_n\}$, a set of issue-groups from agent i is $G_i = \{g_{i,0}, g_{i,1}, ..., g_{i,m_i}\}$)

```
SG := G_0, i := 1
while i < |Ag| do
  SG' := ∅
  for s ∈ SG do
    for g_{i,j} ∈ G_i do
      s' := s ∩ g_{i,j}
      if s' ≠ φ then
        SG' := s ∪ g_{i,j}
      end if
      SG := SG', i := i + 1
    end for
  end for
end while
```

Gathering all of the agents' interdependency graphs in one central place and then finding the issue groups using standard clustering techniques is possible. However, it is difficult to determine the optimal number of issue groups or the clustering parameters using central clustering algorithms because the basis of clustering can differ for each agent. The proposed approach avoids these weaknesses by requiring that each agent generates its own issue clusters. In the experiments, agents used the well-known Girvan-Newman algorithm [69], which computes clusters in weighted non-direct graphs. The algorithm's output can be controlled by changing the "number of edges to remove" parameter. Increasing the value of this parameter increases the number of issue dependencies that are ignored when calculating the issue clusters, thereby producing a more significant number of smaller clusters. The running time of this algorithm is $\mathcal{O}(kmn)$, where k is the number of edges to remove, m is the total number of edges, and n is the total number of vertices.

Finding Agreements

A distributed variant of simulated annealing (SA) [58] is used to find optimal contracts in each issue group. In each round, the mediator proposes an agreement that is a random single-issue mutation of the most recently accepted contract (the accepted contract is initially generated randomly). Each agent then votes to accept(+2), weakly accept(+1), weakly reject(−1), or reject(−2) the new contract, depending on whether it is better or worse than the last accepted contract for that issue group. When the mediator receives these votes, it adds them together. If the sum of the vote values from the agents is positive or zero, the proposed contract becomes the currently accepted one for that issue group. If the vote sum is negative, the mediator will accept the agreement with probability $P(\text{accept}) = e^{\Delta U/T}$, where T is the mediator's virtual temperature (which declines over time) and ΔU is the utility change between the contracts. In other words, at higher virtual temperatures and smaller utility decrements, an inferior agreement is more likely to be accepted. If the proposed contract is not accepted, a mutation of the most recently accepted contract is proposed in the next round. This continues over many rounds. This technique allows the mediator to skip past local optima in the utility functions, significantly earlier in the search process in the pursuit of global optima.

Algorithm 5 Simulated_Annealing()

$Value(N)$: the sum of the numeric values mapped from votes to N from all agents

S := initial solution (set randomly)
for $t = 1$ **to** ∞ **do**
  $T := schedule(t)$
  if $T = 0$ **then**
    return $current$
  end if
  $next$ **:= a randomly selected successor of** $current$
  if $next.Value \geq 0$ **then**
    $\Delta E := next.Value - current.Value$
    if $\Delta E > 0$ **then**
      $current := next$
    else
      $current$ **:=**$next$ **only with probability** $e^{\Delta E/T}$
    end if
  end if
end for

Exaggerator Agents

Any voting scheme introduces the potential for strategic non-truthful voting by the agents, and the proposed method is no exception. For example, one of the agents may always vote truthfully, whereas another exaggerates so that its votes are always strong. It has been shown that this biases the negotiation outcomes to favor the exaggerator at the cost of reduced social welfare [36]. An enhancement of the negotiation protocol is necessary that prevents exaggerated votes and maximizes social welfare.

Simply limiting the number of strong votes by each agent can work well. If the limit is very low, it is effectively lost the benefit of voting weight information and obtain lower social welfare values. Limiting the number of strong votes per agent can avoid this; however, if the strong vote limit is set too high, all an exaggerator has to do is save all of its strong votes until the end of the negotiation. At this point, it can drag the mediator toward making a series of proposals that are inequitably favorable to it. The experiments demonstrate that limiting the number of strong votes is effective for finding high-quality solutions.

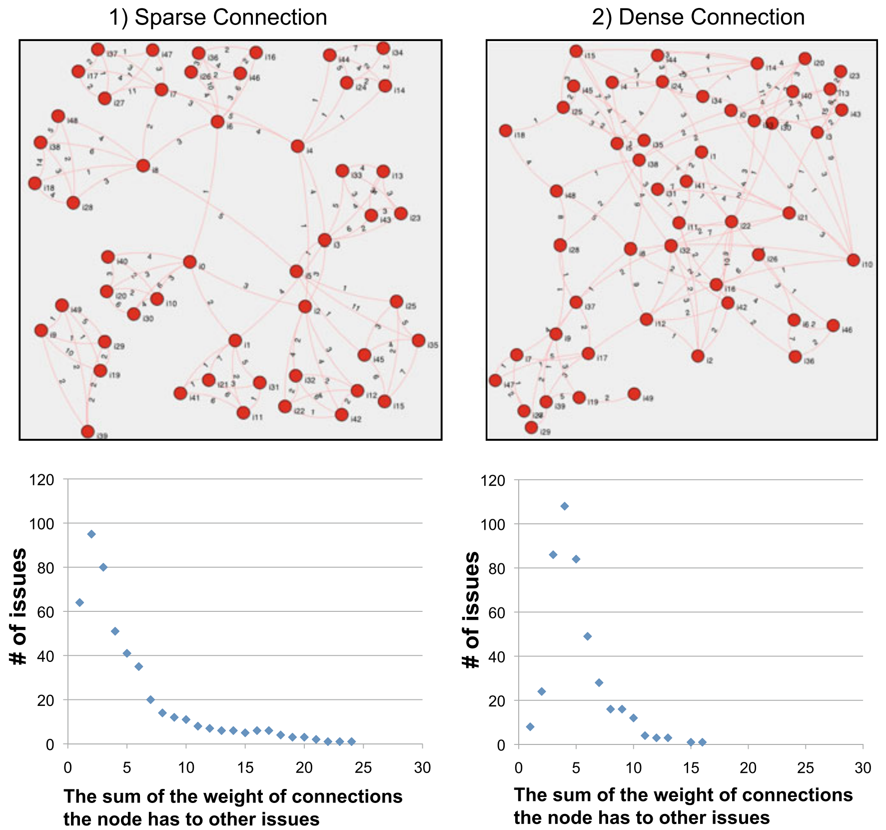


Fig. 10 Issue interdependencies in the experiments. It gives examples of inter-dependency graphs and the relationship between the number of issues and the sum of the connection weights between issues for these two cases. The sparse connection case is closer to a scale-free distribution with power-law statistics, whereas the dense connection condition is closer to a random graph

6.3 Experiments

6.3.1 Experimental Setting

Several experiments were conducted to evaluate the proposed approach. In each experiment, 100 negotiations were ran using the following parameters. The domain for the issue values was [0, 9]. Constraint-based utility functions were employed. Each agent had 10 unary constraints, 5 binary constraints, 5 ternary constraints, and so on (a unary constraint is related to one issue, a binary constraint is related to two issues, and so on). The maximum weight for a constraint was 100 × (Number of issues).

Each agent's issues were organized into ten small clusters with strong dependencies between the issues within each cluster. Then, two conditions were ran: Sparse

Connections and Dense Connections. Figure 10 gives examples of inter-dependency graphs and the relationship between the number of issues and the sum of the connection weights between issues for these two cases. As these graphs show, the Sparse Connection case is closer to a scale-free distribution with power-law statistics, whereas the Dense Connection condition is closer to a random graph.

The following negotiation methods were compared:

- "(A) Issue Grouping (True Voting):" SA is applied based on the agents' votes, and negotiation is performed separately for each issue group. The resulting sub-agreements are combined to produce the final agreement. All agents make truthful votes.
- "(B) Issue-Grouping (Exaggerator Agents):" SA is applied based on the agents' votes with issue grouping. All the agents make exaggerated votes.
- "(C) Issue-Grouping (Limitation):" This is the same as (B) except that a limitation on strong votes is applied. The maximum number of strong votes is 250, the optimal number of limitations in these experiments.
- "(D) Without Issue-Grouping:" This method is presented in [36], using SA based on the agents' votes without generating issue-groups.

In all these cases, the search began with a randomly generated contract, and the SA initial temperature was 50.0 and decreased linearly to 0 throughout the negotiation. In (D), the search process involved 500 iterations. In (A)–(C), the search process involved 50 iterations for each issue group. Therefore, all the cases used the same computation time and are thus directly comparable. In all cases, the number of edges removed from the issue inter-dependency graph when the agents were calculating their issue groups was six.

The centralized SA was applied to the sum of the individual agent's utility functions to approximate the optimal social welfare for each negotiation test run. An exhaustive search was not a viable option because it becomes computationally intractable as the number of issues grows. The SA initial temperature was 50.0 and decreased linearly to 0 throughout 2,500 iterations. The initial contract for each SA run was randomly selected. A normalized optimality rate was calculated for each negotiation run, defined as (Social welfare achieved by each protocol)/(Optimal social welfare calculated by SA).

The code was implemented in Java 2 (1.6) and was run on a Core^{TM} 2 Duo CPU with 2.0 GB of memory under Mac OS X 10.6.

6.3.2 Experimental Result

Figures 11 and 12 compare the optimality rate in the Sparse Connection and Dense Connection cases. (A) achieved a higher optimality rate than (D), which means that the issue-grouping method produces better results for the same amount of computational effort. The optimality rate of the (A) decreased as the number of issues (and therefore the size of the search space) increased. (B) performed worse than condition

(A) because the exaggerator agents reduced the social welfare in multi-agent situations. However, (C) outperformed condition (B); therefore, limiting the number of strong votes is effective for counteracting the reduction in the social welfare caused by the exaggerator agents.

The optimality rates for all methods were almost unaffected by the number of agents, as Fig. 12 shows. The optimality rate for (A) is higher than that for (D) in the Sparse Connections case; this is also true in the Dense Connections case but to a lesser degree. This is because the issue grouping method can achieve high optimality if the number of ignored inter-dependencies is low, which is more likely to be true in the Sparse Connections case. The sparse issue inter-dependencies characterize many real-world negotiations.

It is also assessed a quality factor measure, QF = (Sum of internal weights of edges in each issue group)/(Sum of external weights of edges in each issue group) to assess the quality of the issue groups, i.e., the extent to which issue dependencies occurred only between issues in the same clusters, rather than between issues in different groups. A higher-quality factor should increase the advantage of the issue grouping protocols because fewer dependencies are ignored when negotiation is done separately for each issue group. Figure 13 shows the quality factors when the

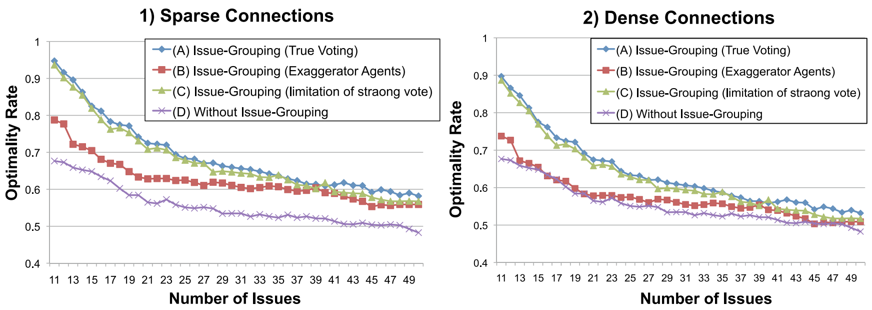


Fig. 11 Comparison of optimality versus number of issues changes in the sparse connection and dense connection cases

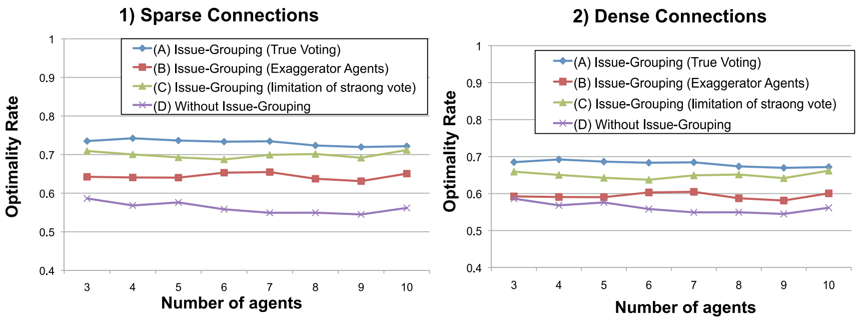


Fig. 12 Comparison of optimality versus number of agents changes in the sparse connection and dense connection cases

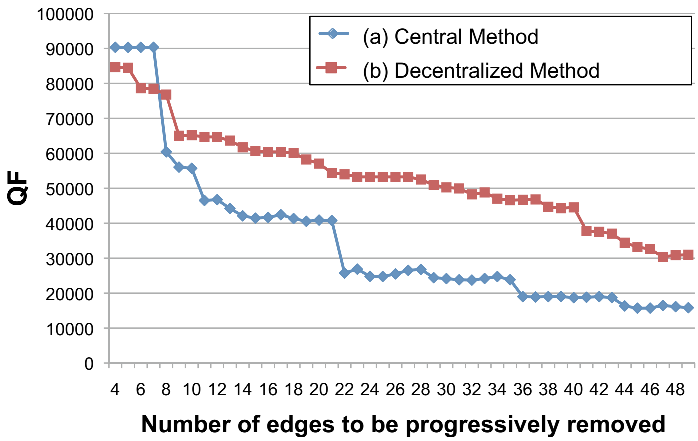


Fig. 13 Number of edges to be progressively removed (clustering parameter) v.s. QF

number of agents is 3 and 20 as a function of the number of edges to be removed, which is the key parameter in the clustering algorithm. For example, the number of issues is 50 in the Sparse Connection case. In the (a) Central Method, all the agents' inter-dependency graphs are gathered in one central place, and then the issue groups are identified using the well-known Girvan-Newman algorithm [69]. In the (b) Decentralized Method, a breadth-first search is employed to combine the issue clusters submitted by each agent into a consolidated set of issue groups.

A comparison of (a) with (b) in Fig. 13 reveals that the decentralized method outperforms the central method. This is because, in the method, all the agents' issues are included in the final issue grouping without a fixed clustering parameter. QF became smaller when the number of edges to be progressively removed grew larger. This is because the number of issue groups generated by each agent increases as the number of edges to be progressively removed becomes larger. A rapid decrease sometimes occurs as the number of edges to be progressively removed increases. These points are good parameters for decomposing the issue groups. In real life, the agents' utilities reflect an adequate concept of issue groups, and agents can determine the optimal issue groups by analyzing the utility spaces.

7 Conclusion and Future Work

7.1 Conclusion

The work described in this chapter makes numerous essential contributions to state of the art in automated negotiation. The contributions of this work can be summarized as follows.

Section 2: A model of nonlinear multi-issue negotiation and a bidding-based negotiation protocol (basic bidding) were described for multiple-issue negotiation among agents with highly nonlinear utility functions. Applying constraints produces a bumpy and highly nonlinear utility function. In the basic bidding protocol, agents generate bids by sampling their utility functions to find local optima and then use constraint-based bids to describe regions with large utility values for that agent compactly. These techniques make bid generation computationally tractable even in large utility spaces. A mediator then finds a combination of bids that maximizes social welfare.

Section 3: A threshold adjustment mechanism for multi-issue negotiations among agents with nonlinear utility functions were proposed. A negotiation with interdependent issues in which the agents' utility functions are nonlinear was assumed. Many real-world negotiation problems are complex and involve multiple interdependent issues. The concept of the revealed area was proposed, which represents the amount of utility information an agent reveals. Moreover, the threshold adjustment mechanism reduces the amount of private information each agent reveals. Additionally, this mechanism could reduce the computational cost of finding a deal with high optimality. Experimental results demonstrated that the threshold adjustment mechanism could reduce the computational cost and provide sufficient optimality.

Section 4: A Distributed Mediator Protocol (DMP) were proposed, which can reach agreements while completely concealing agents' utility information and achieving high scalability concerning utility space. Moreover, the Hybrid Secure Protocol (HSP) was proposed that combines the DMP and the Take it or Leave it (TOL) Protocol. Experimental results demonstrated that the HSP could reduce the required memory with high optimality.

Section 5: It was shown that the Nash bargaining solution, although optimal for negotiations with linear utilities, can lead to sub-optimal outcomes when applied to nonlinear negotiations. Secure and Fair Mediator Protocol (SFMP) was proposed. This negotiation protocol uses a combination of nonlinear optimization, secure information sharing, and an approximated fairness metric. It was demonstrated that it achieves higher social welfare values than a protocol based on searching for the Nash bargaining solution. Finally, it was shown that the SFMP outperforms the own previous efforts to enable multi-lateral negotiations in complex domains.

Section 6: A new negotiation protocol based on grouping issues that can find high-quality agreements in inter-dependent issue negotiation was proposed. In this protocol, agents privately generate their own issue inter-dependency graphs, the mediator identifies issue groups according to these graphs, and multiple independent negotiations proceed for each issue sub-group. It was demonstrated that the proposed protocol has greater scalability than those in previous works and analyzed the incentive compatibility issues.

7.2 Future Work

Future work includes building protocols to find Pareto-optimal contracts more quickly, making them more scalable and increasing fairness performance. One potential approach to this problem is to focus the search efforts of the mediators more closely on the fair portion of the Pareto frontier.

Another possible future work is to analyze the negotiation protocol theoretically. Investigating the incentive compatibility issues can ensure that the protocol cannot be gamed by agents seeking to gain disproportionate influence or sabotage the outcomes. Enhancing the negotiation protocol that incentivizes truthful bidding can preserve equity and maximize social welfare. In the bilateral case, it can be done using a type of Clarke tax [70], wherein each agent has a limited budget from which it has to pay other agents before the mediator accepts a contract that favors that agent, but reduces the utility for others. This approach incentivizes agents to avoid exaggeration because it will cause them to spend their limited budget on contracts that do not strongly affect their true utility values.

In this chapter, cardinal utilities in constraint-based utility functions were considered; however, other utility functions based on cardinal utilities and ordinal utilities are essential factors to apply to the real-world setting [9].

References

1. Wooldridge M, Jennings NR (1995) Intelligent agents: theory and practice. Knowl Eng Rev 10(2):115–152
2. Castelfranchi C (1998) Modelling social action for Ai agents. Artif Intell 103:157–182
3. Hopmann PT (1996) The negotiation process and the resolution of international conflicts (Studies in international relations). University of south Carolina Press
4. Conry SE, Meyer RA, Lesser VR (1988) Multistage negotiation in distributed planning. In: Bond AH, Gasser L (eds) Distributed artificial intelligence. Morgan Kaufmann Publishers Inc., pp 367–384
5. Egels-Zandn N (2009) Tnc motives for signing international framework agreements: a continuous bargaining model of stakeholder pressure. J Bus Ethics 84(4):529–547
6. Elizabeth H, Kevin M, Keith S, Vernon S (1994) Preferences, property rights, and anonymity in bargaining games. Games Econ Behav 7(3):346–380
7. Fisher R (1987) Getting to yes : how to negotiate agreement without giving in (AUDIO CASSETTE). Sound Ideas
8. Nash J (1950) The bargaining problem. Econometrica 18:155–162
9. Raiffa H (1982) The art and science of negotiation. Harvard University Press
10. Rubinstein A (1982) Perfect equilibrium in a bargaining model. Econometrica 50(1):97–109
11. Smith RG (1980) The contract net protocol: high-level communication and control in a distributed problem solver. IEEE Trans Comput 29(12):1104–1113
12. Ito T, Hattori H, Zhang M, Matsuo T (2005) Rational, robust, secure negotiations in multiagent systems. IEEE Computer Society
13. Ito T, Zhang M, Robu V, Fatima S, Matsuo T, Yamaki H (2008) Rational, robust, and secure negotiations in multiagent systems. Series of studies in computational intelligence. Springer
14. Ito T, Zhang M, Robu V, Fatima S, Matsuo T, Yamaki H (2010) Advances in agent-based complex automated negotiations. Series of studies in computational intelligence. Springer

15. Ito T, Zhang M, Robu V, Fatima S, Matsuo T, Yamaki H (2011) Innovations in agent-based complex automated negotiations. Series of studies in computational intelligence. Springer
16. Kersten G, Lai H (2007) Negotiation support and e-negotiation systems: an overview. Group Dec Negotiation 16:553–586
17. Lomuscio A, Wooldridge M, Jennings NR (2001) A classification scheme for negotiation in electronic commerce. In: Agent mediated electronic commerce, the European AgentLink perspective. Springer, pp 19–33
18. Wellman MP, Greenwald A, Stone P (2007) Autonomous bidding agents: strategies and lessons from the trading agent competition (Intelligent robotics and autonomous agents). The MIT Press
19. Kraus S (2001) Strategic negotiation in multiagent environments. Cambridge University Press
20. Ren F, Zhang M, Miao C, Shen Z (2009) A market-based multi-issue negotiation model considering multiple preferences in dynamic e-marketplaces. In: Proceedings of 12th international conference principles of practice in multi-agent systems (PRIMA-2009), pp 1–16
21. Sandholm T (2000) Agents in electronic commerce: component technologies for automated negotiation and coalition formation. Auto Agents Multi-Agent Syst 3:73–96
22. Sandholm T, Lesser V (1995) Issues in automated negotiation and electronic commerce: extending the contract net framework. In: First international conference on multi-agent systems (ICMAS-95), pp 328–335
23. Klein M, Iandoli L (2008) Supporting collaborative deliberation using a large-scale argumentation system: The mit collaboratorium. In: Proceedings of directions and implications of advanced computing symposium conference on online deliberation (DIAC-2008/OD2008)
24. Malone TW, Klein M (2007) Harnessing collective intelligence to address global climate change. Innov J 2(3):15–26
25. Jin Y, Geslin M (2008) Roles of negotiation protocol and strategy in collaborative design. In: Gero JS, Goel AK (eds) Design Comput Cognit ’08. Springer, Netherlands, pp 491–510
26. Jin Y, Lu SY (2004) Agent based negotiation for collaborative design decision making. CIRP Ann—Manuf Technol 53(1):121–124
27. Cappiello C, Comuzzi M, Plebani P (2007) On automated generation of web service level agreements. In: Krogstie J, Opdahl A, Sindre G (eds) Advanced information systems engineering, vol 4495. Lecture Notes in Computer Science. Springer, Berlin/Heidelberg, pp 264–278
28. Koumoutsos G, Thramboulidis K (2009) A knowledge-based framework for complex, proactive and service-oriented e-negotiation systems. Electron Commerce Res 9:317–349
29. Keeney RL, Raiffa H (1993) Decisions with multiple objectives. Cambridge University Press
30. Luo X, Jennings NR, Shadbolt N (2006) Acquiring user strategies and preferences for negotiating agents: a default then adjust method. Int J Human Comput Stud 64(4):304–321
31. Sandholm T, Boutilier C (2006) Preference elicitation in combinatorial auctions. MIT Press
32. Jennings N, Faratin P, Lomuscio A, Parsons S, Wooldridge M, Sierra C (2001) Automated negotiation: prospects, methods and challenges. Group Dec Negotiation 10:199–215
33. Zhang X, Lesser V, Podorozhny R (2005) Multi-dimensional, multistep negotiation. Auto Agents Multi-Agent Syst 10:5–40
34. Faratin P, Sierra C, Jennings NR (1998) Negotiation decision functions for autonomous agents. Int J Robot Auto Syst 24:3–4
35. Faratin P, Sierra C, Jennings NR (2002) Using similarity criteria to make issue trade-offs in automated negotiations. Artif Intell 142:205–237
36. Klein M, Faratin P, Sayama H, Bar-Yam Y (2003) Negotiating complex contracts. Group Deci Negotiation 12(2):58–73
37. Fatima SS, Wooldridge M, Jennings NR (2004) Optimal negotiation of multiple issues in incomplete information settings. In: Proceedings of third international joint conference on autonomous agent and multi-agent systems (AAMAS-2004), pp 1080–1087
38. Lau RK (2005) Towards genetically optimised multi-agent multi-issue negotiations. In: Proceedings of 38th Hawaii international conference on system sciences (HICSS-2005)
39. Soh LK, Li X (2004) Adaptive, confidence-based multiagent negotiation strategy. In: Proceedings of third international joint conference on autonomous agent and multi-agent systems (AAMAS-2004), pp 1048–1055

40. Chevaleyre Y, Endriss U, Estivie S, Maudet N (2008) Multiagent resource allocation in -additive domains: preference representation and complexity. Annals OR 163(1):49–62
41. Grabisch M (1997) k-order additive discrete fuzzy measures and their representation. Fuzzy Sets Syst 92(2):167–189
42. Boutilier C, Hoos HH (2001) Bidding languages for combinatorial auctions. In: Proceedings of the 17th international joint conference on artificial intelligence, vol 2, pp 1211–1217
43. Ito T, Hattori H. Klein M (2007) Multi-issue negotiation protocol for agents: exploring nonlinear utility spaces. In: Proceedings of 20th international joint conference on artificial intelligence (IJCAI-2007), pp 1347–1352
44. Lai KR, Lin MW (2004) Modeling agent negotiation via fuzzy constraints in e-business. Comput Intell 20(4):624–642
45. Luo X, Jennings NR, Shadbolt N, Leung H, Lee JH (2003) A fuzzy constraint based model for bilateral, multi-issue negotiations in semi-competitive environments. Artif Intell 148:53–102
46. Ito T, Klein M, Hattori H (2008) A multi-issue negotiation protocol among agents with nonlinear utility functions. Multiagent Grid Syst 4(1):67–83
47. Hattori H, Klein M, Ito T (2007) A multi-phase protocol for negotiation with interdependent issues. In: 2007 IEEE/WIC/ACM international conference on intelligent agent technology (IAT-2007), pp 153–159
48. López-Carmona MA, Velasco JR (2006) An expressive approach to fuzzy constraint based agent purchase negotiation. In: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems (AAMAS-2006), pp 429–431
49. Eppinger SD (1991) Model-based approaches to managing concurrent engineering. J Eng Des 2(4):190–283
50. Eppinger SD, Whitney DE, Gebala DA (1992) Organizing tasks in complex design projects: development of tools to represent design procedures. In: NSF design and manufacturing systems conference, pp 301–309
51. Tamura H, Nakamura Y (1983) Decompositions of multiattribute utility functions based on convex dependence. Oper Res 31(3):488–506
52. Kaneko M (1980) An extension of the Nash bargaining problem and the Nash social welfare function. Theory Dec 12(2):135–148
53. Binmore K, Rubinstein A, Wolinsky A (1986) The Nash bargaining solution in economic modeling. The RAND J Econ 17:176–188
54. Denicolo V, Mariotti M (2000) Nash bargaining theory, nonconvex problems and social welfare orderings. Theory Dec 48(4):351–358
55. Shamir A (1979) How to share a secret. Commun ACM 22(11):612–613
56. Robertson J, Webb W (1998) Cake-cutting algorithms: be fair if you can. A K Peters Ltd
57. Lindell Y (2003) Composition of secure multi-party protocols: a comprehensive study. Springer
58. Russell SJ, Norvig P (2002) Artificial intelligence: a modern approach. Prentice Hall
59. Ken B, Shaked A, Sutton J (1985) Testing noncooperative bargaining theory: a preliminary study. Am Econ Rev 75:1178–1180
60. Alvard MS (2004) 14. The ultimatum game, fairness, and cooperation among big game hunters. Found Human Sociality 1(9):413–451
61. Bolton GE (1991) A comparative model of bargaining: theory and evidence. Am Econ Rev 81(5):1096–136
62. Werner G, Rolf S, Bernd S (1982) An experimental analysis of ultimatum bargaining. J Econ Behav Organ 3:367–388
63. Myerson R (1991) Game theory: analysis of conflict. Harvard University Press
64. Fujita K, Ito T, Klein M (2008) Preliminary result on secure protocols for multiple issue negotiation problems. In: Proceedings of The 11th pacific rim international conference on multi-agents (PRIMA-2008)
65. Fujita K, Ito T, Klein M (2009) Approximately fair and secure protocols for multiple interdependent issues negotiation. In: Proceedings of The eighth international joint conference on autonomous agents and multi-agent systems (AAMAS-2009), pp 1287–1288

66. Chevaleyre Y, Endriss U, Estivie S, Maudet N (2007) Reaching envy-free states in distributed negotiation settings. In Proceedings of 20th international joint conference on artificial intelligence (IJCAI-2007), pp 1239–1244
67. Thomson W (1992) Cooperative models of bargaining. Handbook of game theory with economic applications 2:1238–1284
68. Nisan N, Roughgarden T, Tardos, E (eds) (2007) V.V.V.: algorithmic game theory. Cambridge University Press
69. Girvan M, Newman MEJ (2002) Community structure in social and biological networks. Proc Natl Acad Sci 99(12):7821–7826
70. Sandholm TW (1998) Distributed rational decision making. In: Weiss G (ed) Multi-agent systems

Exploiting Smart Systems for Monitoring and Assisting Elderly People at Home

Masahide Nakamura

Abstract In a super-aging society, the government is changing the policy from conventional facility care to in-home care, where elderly people live in their homes as long as possible. A major challenge is that in-home care relies heavily on the self-aid of individual elders as well as the assistance of family caregivers. Our research group has been studying service-oriented smart systems that support elderly people at home. In this chapter, we introduce two kinds of technologies for monitoring in-home elderly people. The first technology is non-intrusive environmental sensing, which monitors the daily living of elderly people. Using the time-series environmental data, automated activity recognition is also conducted. The second technology is called mind sensing (called "kokoro" sensing), which monitors the internal states of elderly people. An animated virtual agent and a text-based chatbot actively talk to elderly people to externalize their internal states as words, and then record the words in order to monitor the minds that cannot be captured by conventional sensors.

1 Introduction

1.1 Background

Japan is facing a super-aging society. According to the research of the Japanese government, the proportion of people over 65 years old in the total Japanese population was less than 5% in 1950, but it increased to 28.4% in 2019 [1]. Under these circumstances, there is a chronic shortage of nursing facilities and care workers. To cope with the problem, the Japanese government is shifting the policy from conventional facility-based care to in-home long-term care.

The Ministry of Health, Labor, and Welfare in Japan declares the Community-based Integrated Care System [2], which ensures the provision of health care, nursing

M. Nakamura (✉)
Center of Mathematical and Data Sciences, Kobe University, 1-1, Rokkodai-cho Nada, Kobe 657-8501, Japan
e-mail: masa-n@cmds.kobe-u.ac.jp

Y. Murakami and K. Kimura (eds.), *Human-Centered Services Computing for Smart Cities*, https://doi.org/10.1007/978-981-97-0779-9_7

care, prevention, housing, and livelihood support. The system relies on four kinds of aids: self-aid, mutual voluntary aid, insurance aid, and public aid. Among them, the insurance aid and the public aid are no longer expandable due to the limitation of the social security budget. Hence, the government especially encourages elderly people to conduct the self aid as well as mutual voluntary aid under the system.

However, it is not easy for most elderly people to keep self-aid and independent living at home. As their physical abilities and cognitive functions are declined, external support must be needed. With the declining birthrate and increasing prevalence of nuclear families, support from family members inevitably has its limitations. When elderly people are tired of self-aid, it is almost impossible to take care of others, which makes mutual voluntary aid quite challenging.

Under this situation, the use of technology is promising to alleviate various problems of in-home care. The research and development of assistive technologies for elderly people has been thriving in the world. The book [3] summarizes the practice of assistive technology to support people with dementia. Also, the term gerontechnology appears as a multidisciplinary academic and professional field combining gerontology and technology. In [4], a lot of researchers and practitioners from various fields gather and form communities.

1.2 Research Goal and Approach

Our research group has been studying service-oriented architecture (SOA) [5], and its application to smart systems (also called cyber-physical systems), including smart home (e.g., [6–8]) and smart city (e.g., [9–11]). In general, every smart system consists of heterogeneous things and software components communicating over the network. Wrapping such heterogeneous components by Web services implements glue between the components, which achieves flexible integration and orchestration. Thus, all the distributed and heterogeneous components are considered as services, and can be connected or disconnected easily, based on the principle of loose-coupling.

At first, our research of the service-oriented smart home had been motivated by technical interests. However, we began to think that it was important to use it as gerontechnology. Although smart devices and information on the Internet are quite promising to help elderly people, it is yet difficult for most elderly people to make full use of them. Therefore, we considered it essential to make these devices and information easy to use for the elderly. Concerning wide acceptance and sustainable use, it is also important that the technologies must be affordable for general households, and be non-intrusive for daily living. To realize such a smart system that is really useful for in-home elderly care, we obtained a Grant-in-Aid for Scientific Research (JSPS Kaken-hi) in 2016 [12] and in 2019 [13]. Using the research budget, we have invited collaborators from various research fields.

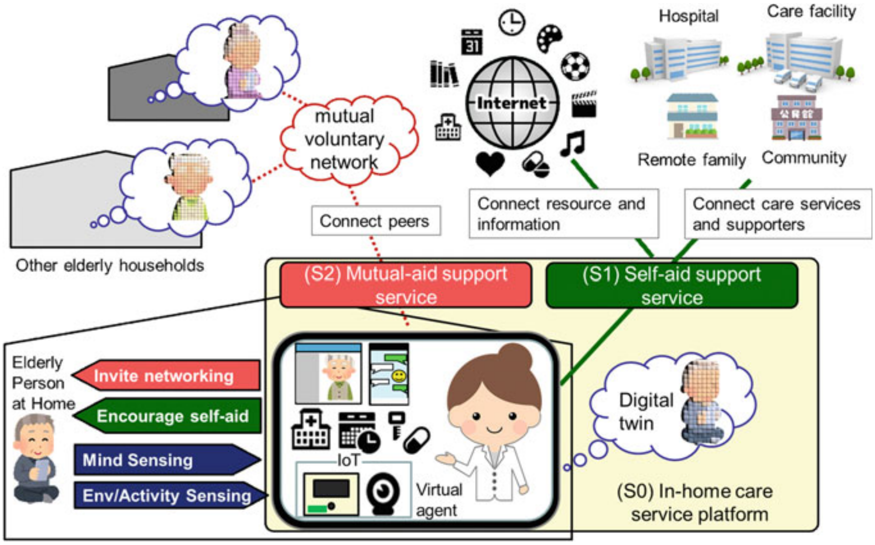


Fig. 1 Conceptual architecture of elderly support system

Our research goal is to design smart services that support and encourage elderly people at home to conduct the self-aid and mutual voluntary aid, and to implement services with devices and systems affordable for general households.

Figure 1 shows a conceptual architecture of the whole system. In the proposed system, a virtual agent (hereinafter referred to as "VA") mediates between the "mind" of the elderly person, such as his/her concerns and wishes, and the support services necessary to resolve or realize them, and provides self-aid and mutual-aid support without requiring complex operations from the elderly person. It consists of three parts.

(S0) In-Home Care Service Platform

It is a platform that monitors the subject's daily living and provides support services. In addition to general environmental and activity sensing using IoT, the system performs Mind Sensing by interacting with the VA, and records the subject's mental state (physical condition, mood, anxiety, hopes, problems, etc.) that cannot be observed by sensors by externalizing them into words. From the sensing data, the system constructs a digital twin, a data object that maps the subject's observable behavior and mental state in cyberspace.

(S1) Self-Aid Support Service

It is a service implemented by applications that support subjects to solve problems by themselves. The system understands the physical and environmental conditions of the subject in real time through the digital twin. The system understands the subject's physical and environmental conditions in real time through the digital twin. The system also attempts to detect signs of mild cognitive impairment (MCI) and dementia, by extracting the number of problems, failed behaviors, and anxious discourse revealed by the externalization of the subject's mind. The system then supports self-aid in the healthy elderly to the MCI stage, by actively connecting to information and services on the Internet, related organizations, and supporters.

(S2) Mutual-Aid Support Service

It is a service implemented by applications that create opportunities where elderly people are connected to help each other. Using information from the digital twin, the system matches elderly people who share the same concerns and interests and the VAs communicate with them. Once a relationship of mutual trust is established, the VAs contact each other directly via chat or videophone applications, forming a network of mutual assistance. The VAs also share the externalized "mind" information with the person who has opted in and achieve safety confirmation, peer counseling, and voluntary living assistance.

1.3 Scope of Chapter

We have been studying various methods, applications, and services to implement the whole system shown in Fig. 1. In this chapter, however, due to the limited pages, we especially focus on the sensing technologies provided by the (S0) in-home care service platform.

What we consider most challenging in in-home care is the individuality of the household. That is, situations and circumstances are quite different from one household to another. It is therefore important for the system to understand first how the individual elderly person is living, and then to provide appropriate (ideally personalized) care and support for the person.

In the following sections, we introduce our research achievements related to sensing technologies for elderly people at home. These sensing technologies are used to monitor in-home elderly people from two different dimensions. The first dimension is to monitor the living of elderly people. As the first step to address individuality, we should observe and understand the physical life environment of individual elderly people. In Sect. 2, we introduce non-intrusive environmental sensing using an IoT sensor device, called Autonomous Sensor Box. We then present an activity recognition method using the environmental sensing data in Sect. 3.

The second dimension is to monitor the minds of elderly people. The ordinary sensing technologies have a limitation that sensors can detect externally observable events only. Thus, it cannot observe the internal state of the elderly person. To cope with the limitation, we proposed the Mind Sensing technology, which externalizes the internal states as words, through conversation with the virtual agent. In Sect. 4, we first introduce the agent technologies and services used for Mind Sensing. Then, in Sect. 5, we introduce the Mind Monitoring Service, which supports healthy daily living based on daily self-assessment with a LINE chatbot.

2 Monitoring Elderly Living by Environmental Sensors

2.1 Autonomous Sensor Box

To provide appropriate support for individual elderly people, it is important to first observe their living and environment physically, and to understand their current situation. Since it is impossible for family caregivers to manually observe and record the situation 24 h a day, deploying IoT sensor devices is a promising method. Recently, IoT has been actively studied in ubiquitous computing and pervasive computing. In the research fields, a lot of sophisticated devices and methods have been developed (e.g., [14–18]).

In the context of monitoring in-home elderly people, however, the sensing devices must be affordable enough, and should not be intrusive to their daily living as well and the house properties. Therefore, we have decided to avoid wearable sensors or expensive indoor positioning systems. Instead, we have developed inexpensive stationary environmental sensing devices, called Autonomous Sensor Box [10].

The Autonomous Sensor Box is an IoT device that consists of a box with seven kinds of environmental sensors, and a single-board computer Raspberry Pi. Figure 2 shows the actual implementation assembled with seven kinds of Phidgets sensors [19] (light, temperature, humidity, sound volume, gas pressure, motion, and vibration).

A user simply puts the sensor box in a location, where the box does not interfere with daily life, and connects the box to a power source. Then, the sensor box automatically starts measuring the surrounding environment at 10 s intervals. The data is uploaded via the Internet to a private cloud in our laboratory. In the private cloud, we implement services that manage data collection, device settings, and deployment information. Communicating with the cloud services, the software running on the Raspberry Pi automates all the processes of environmental sensing. Thus, the operation required at the elderly house is minimized to switch on/off the power.

Using the Autonomous Sensor Box, we have implemented a service platform for in-home environment sensing, as will be introduced in the following sections.

Fig. 2 Autonomous sensor box

2.2 *Service Platform for In-Home Environment Sensing*

System Architecture

Figure 3 shows the system architecture of the proposed environment sensing service. The system consists of the following four components.

C1: Autonomous Sensor Box With power and network connection, this device starts environment sensing autonomously and uploads the data to the cloud.

C2: Sensor Box Management Service This service manages the configuration and deployment information of all the sensor boxes deployed in the experimental area.

C3: Log Collection Service This service collects the environmental data from the sensor boxes, attaches the timestamp, and stores as sensor log in a large-scale database.

C4: State Cache Service This service caches the newest data from every sensor box and provides the data as the current state of the sensor box for external applications.

C5: Sensor Box Log Service This service provides the stored sensor log to external authorized applications.

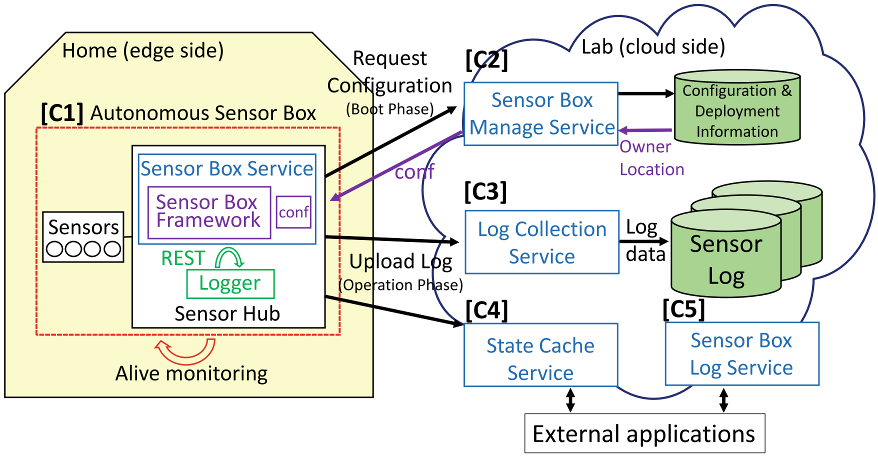


Fig. 3 System architecture of proposed service

We integrate the above five components with the principle of Service-Oriented Architecture (SOA). The detailed implementation of each component is described in the following sections.

(C1) Autonomous Sensor Box

Autonomous Sensor Box is an IoT device conducting indoor environment sensing at one location in an elderly house. The hardware of the sensor box consists of environmental sensors and a sensor hub that controls them.

As shown in Fig. 3, the sensor hub is equipped with Sensor Box Framework, which abstracts concrete sensor devices as sensor objects. More specifically, the framework takes a configuration file as input, declaring the name, the owner, and the location of the sensor box, and the type and implementation of each of the sensors in the box. Based on the configuration, the framework dynamically creates sensor objects and binds each object to the sensor implementation class. Thus, the framework allows developers to install various kinds of sensors in the box.

The simplest way to manage the sensor box configuration is to put the configuration locally in each sensor box. However, this approach lacks scalability as we manage more and more boxes. Therefore, we manage the configuration and deployment information in the central database on the cloud, so that every sensor box downloads its own configuration on the boot phase. This is shown in Fig. 3 as the interaction between C1 and C2.

In our implementation, the Sensor Box Framework is further wrapped by Sensor Box Service, by which the external application can get the data from the sensor objects via REST API. The logger application in the sensor hub periodically calls Sensor

Box Service by REST to acquire sensor data and upload it to the Data Collection Service. The default sampling interval is 10 s.

To minimize the manual operation at home, we implement the following features by which the sensor box can autonomously start the environment sensing.

Auto connection to the network: When switched on, the sensor box automatically connects to the pre-set network and prepares for connection to the cloud services.

Auto configuration of sensor box: When prepared, the sensor box confirms its own ID and requests its own configuration to the Sensor Box Manage Service. Based on the given ID, the Sensor Box Manage Service retrieves configuration and deployment information for the sensor box. Upon receiving the configuration, the sensor box creates sensor objects and launches the Sensor Box Service.

Auto launch logger: When the service is ready, the sensor logger is launched automatically. The logger uses REST API of the Sensor Box Service and obtains the current values of the connected sensors. The logger finally uploads acquired values to Log Collection Service, and State Cache Service. The data sampling and upload are executed at every pre-determined interval (10 s by default).

Alive monitoring of logger: While the sensor box is running, the system periodically checks if the logger, the service, and the network are all alive. If any critical error is observed, the system is rebooted automatically.

With the above autonomous functions, the user only needs to put the sensor box in a location and turn on the power, which automatically starts the environmental sensing. This minimizes the time and effort required to set up and manage the installation.

(C2) Sensor Box Manage Service

Sensor Box Manage Service manages the configuration and deployment information of all the sensor boxes deployed for the experiment. Each sensor box is identified by ID. The configuration information of a sensor box declares its name and the list of sensors installed in the box. Each sensor is defined by sensor type (e.g., temperature, light, humidity, etc.), device (i.e., the reference to a concrete device class), and binding information (parameters passing to the device class). The deployment information manages where the sensor box is deployed, including the location (house, room, position) and owner.

When booted, a sensor box accesses this service to retrieve its own configuration and deployment information. Furthermore, the service also manages network connection information (IP address and others) of every sensor box. The system administrator uses this information for remote testing and maintenance.

Table 1 Data schema of environmental sensor data

	Key name	Value description	Key-value example
Data	Light temperature ...	Measured sensor values	Light:82 Temperature: 10.667 ...
Info	Date	The date of log obtained	Date: 2016-02-04
	TmeOfDay	The time (of day) of log obtained	TimeOfDay: 16:07:39
	Time	The time of log obtained	Time: 2016-02-04-T16:07:39+09:00
	Boxid	Sensor box ID	Boxid: sbox-phidget-406364
	Owner	Owner of sensor box	Owner: sakakibara
	Location	Installation location of sensor box	Location: Kobe/Kobe-Univ./S101/desk

(C3) Log Collection Service

Log Collection Service receives environmental data from every sensor box and stores the data as Sensor Log (i.e., time-stamped senor values). To achieve efficient retrieval and aggregation of sensor data, the log collection service defines the data schema shown in Table 1. In the table, `data` is the sensor values measured by the sensor box, and `info` is metadata describing the sensor values. In order to handle data with various combinations of sensors in a unified manner, the sensor data itself is represented by the Key-Value of attribute names and values, without a strict schema.

On the other hand, the metadata is defined by common attributes that are independent of specific types of sensors. This enables cross-sectional search and aggregation of all sensor data. More specifically, we identified data items that explain when, who, and where the measurements were taken since these aspects are independent of specific environmental sensing. The `date, timeOfDay, time` are data items related to when. The `boxId, owner` are data items related to who. The `location` is a data item related to where.

The logger in the sensor box generates data based on the schema for each measurement, represents the data in JSON-formatted text, and uploads it to the Log Collection Service.

(C4) State Cache Service

State Cache Service caches only the latest data sent one after another from the sensor box and provides applications with fast access to the current values of the sensor box. The sensor log stored in Log Collection Service is good for applications that use past values. However, for applications that need only the current values, the overhead of retrieving the latest values from the stored data is not ignorable.

The state cache service always keeps the latest measured value in memory with the sensor box ID as the Key and the current measured value as the Value to realize fast access to the current value of any sensor box. Autonomous sensor box uploads the measured sensor values to both the log collection service and the state cache service to realize efficient data provision to both applications that use past and current data.

(C5) Sensor Box Log Service

Sensor Box Log Service provides the stored sensor log for external applications. Through REST API with `info` attributes in Table 1, external applications can retrieve the sensor data by JSON or XML format.

2.3 Implementation

Service Platform

We have implemented the service platform for in-home environmental sensing.

First, the autonomous sensor box has been implemented by assembling commercial sensors manufactured by Phidgets Inc [19]. More specifically, the following seven sensors were used:

- Temperature Sensor 1125
- Humidity Sensor 1125
- Absolute Pressure Sensor 1141
- Vibration Sensor 1104
- Sound Sensor 1133
- Light Sensor 1127
- Motion Sensor 1111.

These seven sensors were connected with a Phidget Interface Kit, which exposes the sensor values to USB interface. For the sensor hub, we used Raspberry Pi 2 (Model B, Raspbian Jessie) single-board computer. As shown in Fig. 2, the box case contains the seven sensors and the interface board, and a USB cable is connected to the Raspberry Pi.

The sensor logger was implemented by Perl, and the alive monitoring system was implemented by shell script and cron. Also, Sensor Box Framework and Service were implemented in Java, and deploy as Web service using Apache Axis2.

The Sensor Box Manage Service was implemented in Perl CGI and HTML::Template library. The Log Collection Service was implemented in the Fluentd log collection framework. For the database, we used MongoDB and HBase. Finally, the State Cache Service and Sensor Box Log Service were implemented in Java and were deployed as RESTful Web service using Jersey.

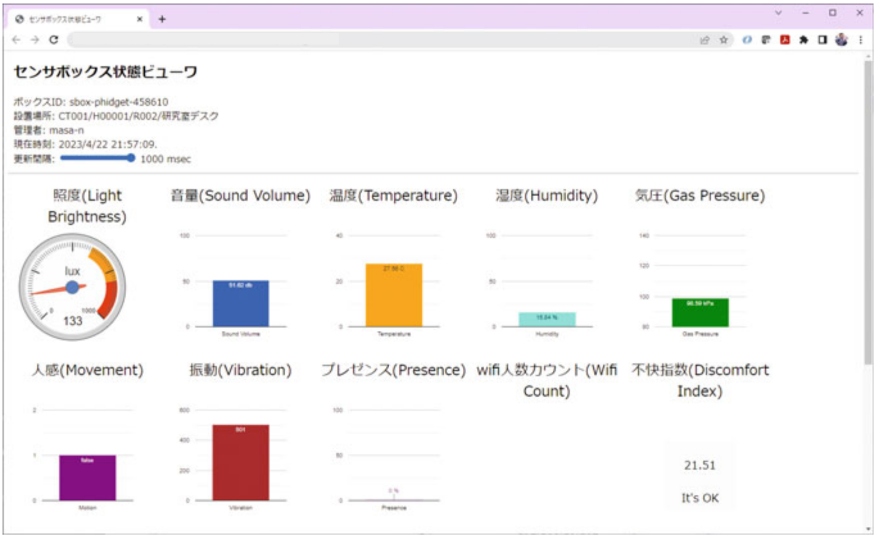


Fig. 4 Sensor box dashboard

Applications

As examples of external applications of the proposed system, we introduce two Web applications. Figure 4 shows a Web application, called Sensor Box Dashboard. Connecting to the Sensor Box Service on the Autonomous Sensor Box, the application displays the current values of installed sensors. Using the application, an administrator of the sensor box can check if the sensor box works correctly.

Figure 5 shows a Web application, called Sensor Box Log Viewer. Connecting to the Sensor Box Log Service, the application displays the daily time-series data of a given sensor box. Using the application, the user can review how the environment has been changed during the day.

With the consent of the elderly person, the data on these applications can be shared with family members and acquaintances to create opportunities for mutual assistance. Thus, the applications can implement the first step of the (S2) Mutual-aid support service in our conceptual architecture (see Fig. 1).

2.4 Deploying Autonomous Sensor Box in Actual Elderly Home

Currently, the autonomous sensor boxes are installed in 20 locations in the houses of research collaborators. The sensor log has been collected for several years. Let us see an example of how the sensor box can be used for daily monitoring of an elderly person.

Fig. 5 Sensor box log viewer

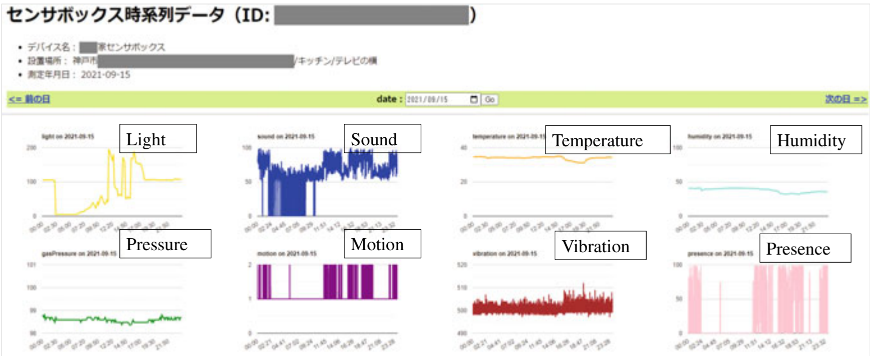


Fig. 6 Visualized sensor data of an elderly person

Figure 6 shows time-series sensor data recorded on September 15th, 2021 at the home of an elderly woman. She was in her 80 s, and was living alone. The sensor box was placed beside her television in the dining kitchen. In the figure, the eight line plots represent the data of light, sound, temperature, humidity, gas pressure, motion, vibration, motion, and human presence likelihood (derived by the integration of motion value), respectively. In each graph, the horizontal axis represents the time (from 0:00 to 23:59), while the vertical axis plots the sensor value.

From the graphs, we can infer the woman's approximate daily life. First, the values of the motion and presence indicate that she went to bed at 2:20 and woke up at 11:30 During the sleeping period, she woke up once around 5:30. The value of the sound volume indicates that the TV was turned on at 11:50. The volume dropped and the human presence did not respond around 14:15, indicating that she went out somewhere. Then, she returned home at 15:30. After that, she turned on the air conditioning since the temperature and humidity changed from 15:30 to 19:30. From 19:50 no motion was detected, indicating she was taking a nap. Then, she woke up at 22:40, and sat up late until 2:00 the next day. The gas pressure was low, as it was raining at this date.

Thus, the autonomous sensor box accumulates multiple sensor data 24 h and 365 days, which can characterize the home environment of an elderly person from multiple perspectives. Remote family members, who know the elderly person well, can view the data over the Internet, and imagine what he or she is doing. In other words, the family can keep a loose watch over the elderly person without intruding too deeply into his or her privacy.

3 Recognizing Daily Activities from Environmental Sensor Data

3.1 Can System Recognize Activities from Environmental Data?

As seen in the previous section, the time-series data collected by the autonomous sensor box (i.e., sensor log) characterizes the living environment of an elderly person. The sensor log would be useful for remote families to monitor if the elderly person is getting along well as usual. Basically, the environmental sensing by the autonomous sensor box is easy to introduce and is not intrusive too much to daily living, which is a great advantage. On the other hand, due to the nature of environmental sensing, it is not easy to recognize what the person is exactly doing from the data. A family member, who knows the elderly person well, may be able to guess it manually. However, if the system can do this, it helps a lot.

Our research question here is: "Using the sensor log collected by an autonomous sensor box, can the system automatically recognize the daily activities of an elderly person?" The daily activities refer to in-home activities regularly performed in the daily life, including sleeping, eating, cooking, cleaning, bathing, etc.

The problem is generally called sensor-based activity recognition [14], which has been studied for a long time in the fields of ubiquitous and pervasive computing. Related work for recognizing the in-home daily activities are summarized as follows. Kusano et al. [15] proposed a system that derives life rhythm by tracking the movement of the elderly by using RFID positioning technology. Munguia-Tapia et al. [20] installed state-change sensors on regular items such as a door, a window, a refrigerator, a key, and a medicine container, to collect interactions of a resident with an object. Philipose et al. [21] attached an RFID tag to items to collect interactions. Pei et al. [22] combined a positioning system and motion sensors of a smartphone to recognize human movements.

Although there were many existing works, we did not find any method that can answer directly to our research question.

3.2 Proposed Activity Recognition Method

To answer the research question in Sect. 3.1, we have developed a new activity recognition system in [23]. Since the autonomous sensor box cannot distinguish multiple residents at home, we focused our methodology on one-person households (OPH, for short). Although the target is limited, there still exists a strong demand to monitor elderly people in OPH.

In the proposed system, we apply supervised machine learning extensively to the sensor log collected by the autonomous sensor box. Given the proposed method based on supervised machine learning, the proposed system requires initial training, where the resident manually records activities using a designated lifelog tool. The initial training is supposed to be performed over several days, to associate labels of activities with sensor data.[1]

In the proposed system, we define seven daily activities (cooking, PC working, cleaning, bathing, sleeping, eating, and going out), which are the most typical activities for maintaining a life rhythm. For the labeled dataset, supervised learning algorithms are applied to construct a model of activity recognition for the house. For this purpose, careful feature engineering is performed to determine essential predictors that best explain the activities in OPH. Furthermore, we try several classification algorithms to compare performance.

Figure 7 shows the outline of the proposed system, where we explain the proposed system from left to right. The system is initially set up within a target OPH. A single (or multiple if necessary) autonomous sensor box is deployed in a position where the daily activities are well observed as environmental measures. A software called LifeLogger is then installed on the user's PC, which is used to attach correct labels of activities to the environmental sensing data. The autonomous sensor box uploads the measured data to Log Collection Service (see Sect. 2.2), whereas LifeLogger records time-stamped activities as lifelog. The sensor log and the life log are joined by the timestamp, to form the training data. For the training data, we then apply the feature engineering and a machine learning algorithm, in order to construct a prediction model of activity recognition.

Once the trained model is constructed, the system moves to the operation phase. Taking environmental sensing data as input, the trained model outputs recognition result, reasoning the current activity.

[1] Note that the system was developed to see the feasibility of our approach. We have not yet evaluated the acceptance of the method for elderly people.

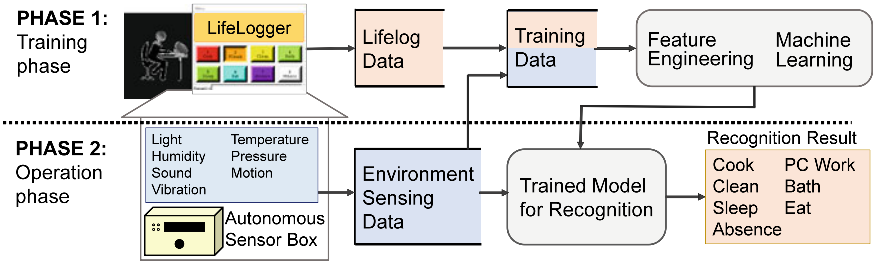


Fig. 7 Outline of the proposed system

3.3 Collecting Data for Activity Recognition

Collecting Environmental Sensor Data

In the proposed system, we use the environment sensing platform with the autonomous sensorbox, which was described in Sect. 2.2.

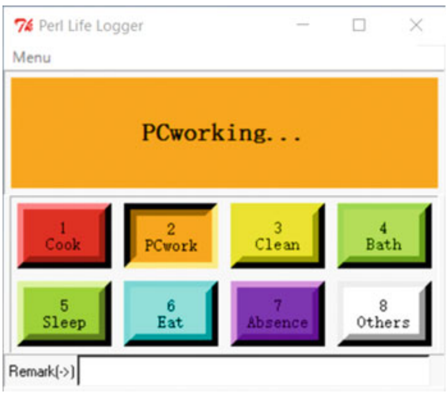
To be able to recognize daily activities by environment attributes, the sensor box should be put on where the resident's activities are frequently conducted. Note that the room layout and living circumstances of every single resident are different among households. Hence, the sensor log collected in a household can be used only for activity recognition within that household.

Recording Lifelog for Correct Labels

During the initial several days, the resident needs to input the correct labels for activities, so that the system can learn these activities from the environmental sensing data. For this purpose, the residents were asked to use LifeLogger.

Figure 8 shows the user interface of LifeLogger. As shown in the figure, LifeLogger has 8 Buttons, each of which corresponds to an activity. When the resident

Fig. 8 Screenshot of life logger tool



```
[2017/02/19] 00:04:52:<user@DESKTOP-G3JFFRR>:Button 2: Starting Pcwork
[2017/02/19] 01:16:36:<user@DESKTOP-G3JFFRR>:Button 2: Ending PCwork
[2017/02/19] 01:16:36:<user@DESKTOP-G3JFFRR>:Button 4: Starting Bath
[2017/02/19] 01:31:08:<user@DESKTOP-G3JFFRR>:Button 4: Ending Bath
[2017/02/19] 01:31:01:<user@DESKTOP-G3JFFRR>:Button 8: Starting Others
[2017/02/19] 01:41:31:<user@DESKTOP-G3JFFRR>:Button 8: Ending Others
[2017/02/19] 01:41:31:<user@DESKTOP-G3JFFRR>:Button 5: Starting Sleep
[2017/02/19] 07:36:52:<user@DESKTOP-G3JFFRR>:Button 5: Ending Sleep
[2017/02/19] 07:36:52:<user@DESKTOP-G3JFFRR>:Button 8: Starting Others
:
```

Fig. 9 Raw data of life log

initiates an activity, he/she simply presses the corresponding button to record the current activity.

Based on relevant studies [24, 25], 8 types of daily activities were chosen (sleeping, eating, bathing, cooking, PC working, cleaning, going out, and others), and registered in LifeLogger. When the button is pressed, the system records the starting time of the activity. When the button is pressed again or another button is pressed, the system records the ending time, and the starting time of the new activity if any. Figure 9 shows a part of the raw data recorded by LifeLogger. From the data, we can see that on February 19, 2017, the user did PCwork, Bath, Others, Sleep, and Others in this order.

Joining Sensor Data and Lifelog Data

For the supervised learning, the system requires training data that have a correspondence between the activities and the sensor log in advance. To establish the training data, we join the two time-series data collected by SensorBox and LifeLogger by timestamp. Activity data labeled as 'other' was deleted, since it was beyond the scope of the activity recognition.

The sensor log is time-series data with fixed interval (10 s by default), while the life log data is event data recording the starting and ending time of every activity. Hence, we first convert the life log data into time-series data with fixed interval, by filling the activity ID between the starting and the ending time. Then, we join the two data with the timestamp.

Table 2 shows the part of the resulting data, which represents the sensor log from 3:33:02 to 3:33:32 on February 19, 2017. We can see that activity ID 5 (i.e., Sleep) is attached in the last column. Thus, this environmental data is used as training data to characterize the Sleep activity of the user.

Table 2 Training data

Date time	Vibration	Light	Motion	Gas pressure	Temperature	Humidity	Sound	Activity ID
2017/2/19 3:33:02	495	1	0	98.8	13.33	35.84	50.15	5
2017/2/19 3:33:12	494	1	0	98.8	13.33	36.04	0	5
2017/2/19 3:33:22	494	1	0	98.8	13.33	36.04	51.62	5
2017/2/19 3:33:32	494	1	0	98.8	13.33	36.04	0	5

3.4 Constructing Machine Learning Recognition Model

Choosing Relevant Environmental Attributes

For accurate activity recognition, it is essential to identify the relevant environmental attributes that best predict activity. From the seven environmental attributes of the sensor log, only temperature, humidity, light, sound volume, and motion were chosen because the remaining attributes (vibration and gas pressure) seem irrelevant to the target activities. According to compared about 20 recognition models based on different combinations of environmental attributes, the determination was made that sensing data of gas pressure and vibration was almost not affected by the resident's activity.

Feature Engineering

Feature value is the data that is effective in the identification of the activities. In this study, the feature values are obtained from training data according to the following process.

The size of time window is first determined. To enhance the features of the time-series data, the raw data within the same time window is aggregated into one data. In this case, the window size affects the accuracy. If the size is too large, the window is likely to contain different activities. If it is too small, the window will not contain sufficient data to reason and predict an activity.

Finally, for each of the five environmental attributes chosen, an aggregation function was determined. An aggregation function aggregates all the data within the same time window. Typical, aggregation functions include maximum value (MAX), minimum value (MIN), average value (AVG), and standard deviation (STDEV). Based on the nature of each environment attribute, an appropriate function was carefully chosen. Figure 10 shows the process of the feature engineering. The fine-grained time-series data is aggregated based on designated time windows, which characterizes features of activities.

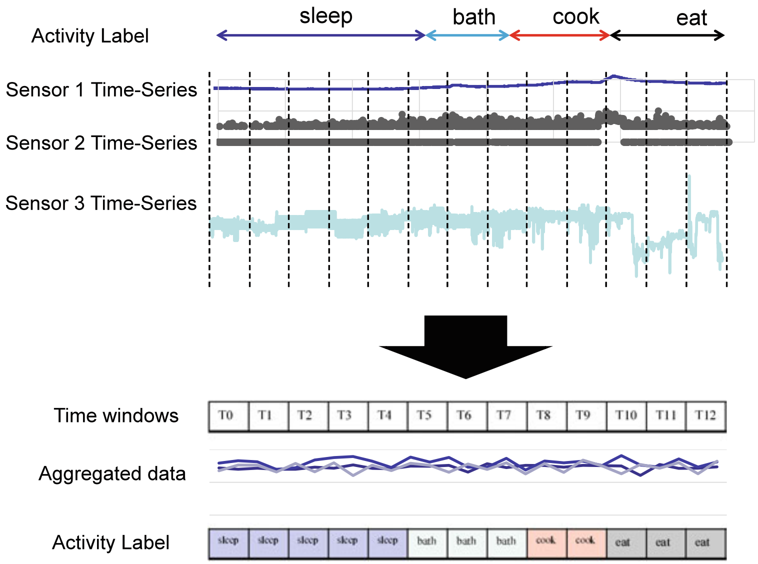


Fig. 10 Feature engineering

Table 3 Nine groups of aggregation functions

Groups	Light	Motion	Temperature	Humidity	Sound
G1	MIN	MAX	AVE	AVE	MAX
G2	MAX	MAX	STD	STD	STD
G3	AVE	AVE	STD	STD	MAX
G4	MAX	AVE	AVE	AVE	MAX
G5	MIN	AVE	AVE	STD	AVE
G6	AVE	AVE	AVE	AVE	STD
G7	MAX	MAX	STD	AVE	AVE
G8	AVE	MAX	AVE	AVE	STD
G9	MIN	AVE	STD	STD	AVE

Note that it is non-trivial to know what aggregate function is best for each environmental attribute. Hence, different aggregation functions must be tested for each environmental attribute. By analyzing all the tests, the optimal combination of aggregation functions can be determined. However, if all situations need to be tested, then hundreds of rounds of tests need to be performed, which is time-consuming.

To effectively test all cases of function combination, a tool called PICT [26] was used. PICT generates a compact set of parameter value choices that represent the test cases required to achieve comprehensive combinatorial coverage of the parameters. Table 3 shows the 9 cases of combinations generated by PICT.

Establishing Recognition Model

For the developed features of the training data, machine-learning algorithms are applied, to construct a prediction model for activity recognition. Popular classification algorithms are then used, including Logistic Regression, Decision Forest, and Neural Network. Using these algorithms, it is possible to construct prediction models that classify given environmental sensor data into one of the seven activities.

The performance of a prediction model is evaluated by a confusion matrix to see how much percentage of the time windows is classified as the correct or wrong activities. The parameters to construct a prediction model are (1) the size of time window, (2) the selection of aggregate functions, and (3) the choice of the machine learning algorithm. We test as many variations of parameters as possible and determine the best combination that yields the most accurate prediction performance.

3.5 Experimental Evaluation

Setup Experiment

The proposed system was deployed in an actual apartment of a single resident. As shown in Fig. 11, the apartment is an ordinary condominium in Japan, consisting of a bed/living room, a bathroom and a kitchen. Two autonomous sensor boxes were positioned as indicated by the red triangles in Fig. 11, one in the kitchen and one in the living room.

A total of 645,705 rows of raw sensor data was collected from the kitchen SensorBox. The living room SensorBox collected 483,862 rows of raw data. We used *Multiclass Decision Forest (DF)*, *Multiclass Logistic Regression (LR)* and *Multiclass Neural Network (NN)* algorithms of Microsoft Azure Machine Learning Studio [27], in order to build the activity recognition model.

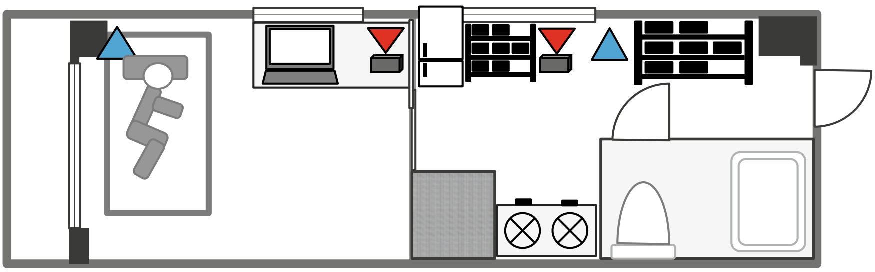


Fig. 11 Apartment for the experiment

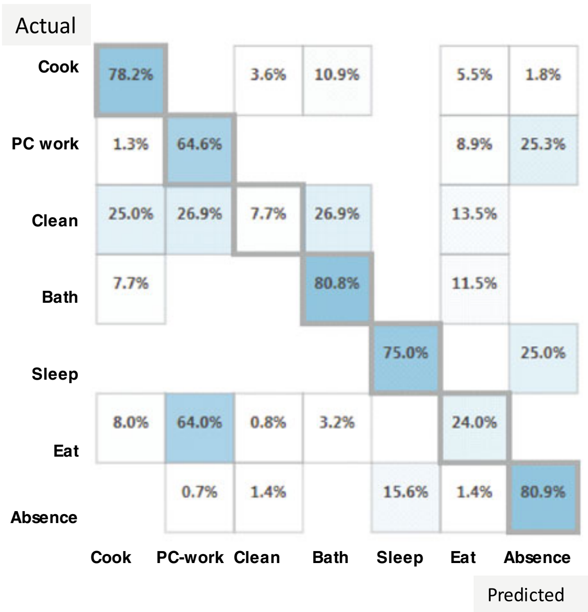


Fig. 12 Confusion matrix of activity recognition with environmental sensing

Result

We have tested many combinations of the parameters to build the activity recognition model. As a result, we found that the best parameters were 30 s for the time window size, the tuple of [Min(light), Ave(motion), Std(temperature), Std(humidity), Ave(sound)] for the aggregation functions, and the decision forest for the machine learning algorithm.

Figure 12 shows a confusion matrix, where each row represents the actual class of activity and each column represents the predicted class of activity. From the matrix, we can see that the accuracy of the activity recognition depends on the class of activity. In this experiment, Cook, Bath and Absence marked high accuracy around 80%, PC work and Sleep marked middle around 60%, and Clean and Eat was quite low.

We investigate the result in more details. The activities PC work and Sleep were often misidentified as Absence. The reason is that the three activities were done in similar environmental condition, where the light was dark, there was no sound or motion. Eat was quite often misidentified as PC work, since the subject often ate meals on the PC desk. Hence, the proposed system is not good at recognizing activities that have similar impact to the environment. In other words, using environmental data only cannot distinguish environmentally similar activities, which is the limitation of the proposed system. Clean was misidentified as Cook, PC work, or Bath. A reasonable interpretation is that for the cleaning the user had to move around whole area, and the duration of each cleaning was short, therefore, the system could not learn unique characteristic of the cleaning.

3.6 Introducing BLE Beacons to Improve Accuracy

As seen in the previous section, the activity recognition with the environmental sensing only was not satisfactory for some activities. This means that the environmental data did not contain sufficient information to identify the activities. Although introducing cameras or wearable devices would provide much richer information, they interfere with the daily life.

As an idea to improve accuracy with preserving the non-intrusiveness, we have attempted additional experiment by deploying BLE (Bluetooth Low Energy) beacons in [28]. A BLE beacon a small device that repeatedly transmits a constant signal that other smart devices (e.g., smartphones) can see. On receiving the signal, a smart device can obtain ID of the beacon as well as RSSI (Received Signal Strength Indicator), by which the device can estimate the distance to the beacon. Using this principle, it is possible to estimate approximately which room the resident is in. As some activities are strongly related to the location, adding the location information to the sensor log is promising to improve the recognition accuracy.

We have implemented a small smartphone application called Blue PIN. When a smartphone receives a signal from a BLE beacon, Blue PIN sends the beacon ID and RSSI to a designated server. The server stores the data in a database.

3.7 Additional Experiment with BLE Beacon Data

Overview of Additional Experiment

In parallel with the previous experiment in Sect. 3.5, we asked the subject to carry a smartphone with Blue PIN. Two BLE beacons were deployed in the kitchen and the living room, as indicated by the blue triangles in Fig. 11. During the experiment, 368,047 rows of data were collected from the living room, while 370,372 rows were collected from the kitchen.

Feature engineering for the beacon data is similar to that of the environmental sensing data. For each of the two beacons, we apply aggregation functions MIN and AVE to data within every time window.

The aggregated sensor data and beacon data are then integrated based on this consistent time window. Finally, training data is created by joining the time-series activity log data and integrated data based on the timestamp. Table 4 show a part of the real training data. In the table, b2 and b3 represent two beacons placed in the living room and the kitchen, respectively. The 'bi.ave' or 'bi.min' respectively represents the average or the minimum of RSSI value of beacon bi.

Table 4 Training data for additional experiment

Datetime	Light	Sound	Temperature	Humidity	Presence	b2.ave	b2.min	b3.ave	b3.min	ADLid
2017/5/29 1:20:00	5.00	86.94	0.10	0.08	88.00	−58.35	−55	−66.94	−63	5
2017/5/29 1:20:30	6.00	88.22	0.00	0.00	67.00	−57.62	−57	−64.75	−62	5
2017/5/29 9:55:30	3.00	88.16	0.00	0.08	92.33	−56.35	−55	−67.60	−64	5
2017/5/29 9:57:00	163.00	17.65	0.00	0.08	78.33	−79.65	−686	−61.44	−52	4
2017/5/29 10:33:30	193.00	68.54	0.00	0.41	3.33	−78.37	−73	−63.00	−56	4

From the data in Table 4, we can estimate approximately which room the resident was in. The first three rows where the values of b2 is larger indicate that the resident was in the living room. This is consistent with the fact that he was sleeping (activity No. 5). The last two rows where the values of b3 is larger indicate that the resident was in the kitchen. It is also consistent with the fact that he was taking bath (activity No. 4). Thus, the training data was integrated with the location information.

Using the integrated training data we constructed a prediction model. As for the feature engineering of the environmental data, we took the same parameters as those of the previous experiment.

Result

Figure 13 shows the confusion matrix. Compared with the previous result in Fig. 12, the accuracy is significantly improved. Thanking to the location information, Sleep and Absence were clearly distinguished. Cook was no more identified as Eat. Clean was improved but yet unsatisfactory. PC work and Eat were still confused, since these activities were performed in the same place. Thus, the location information did not contribute at all.

Table 5 compares the results of the two experiments. We can see that using the beacon data together with the environmental sensor data significantly improve the performance of the activity recognition.

The proposed system enables the automatic activity recognition from non-intrusive sensing, which is promising for the future in-home care. A major drawback is that it requires the activity labeling to the sensor data, which is so tedious that most people may not accept it. How to improve the acceptance of the system is left for our future work.

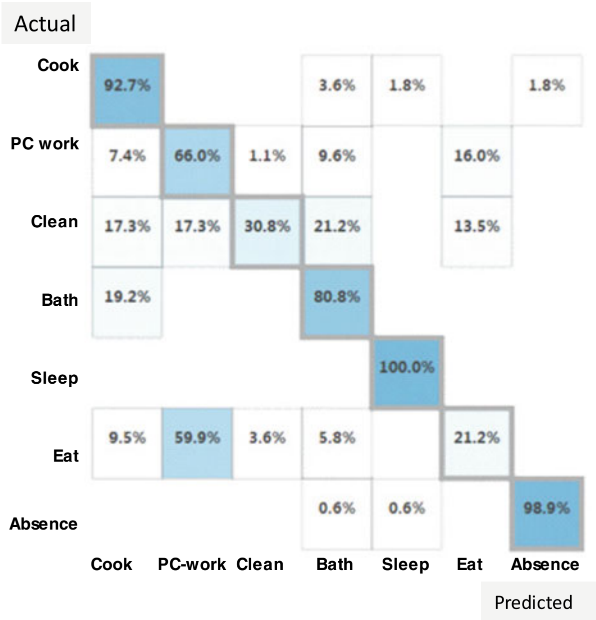


Fig. 13 Confusion matrix for proposed system using integrated data

Table 5 Overall comparison of experimental results

Metrics	Env. only (%)	Env. and Beacon (%)	Improvement (%)
Overall accuracy	55.34	70.96	+15.62
Average accuracy	87.25	91.70	+4.46
Micro-averaged precision	55.35	70.96	+15.61
Macro-averaged precision	50.10	66.85	+16.75
Micro-averaged recall	55.35	70.96	+15.61
Macro-averaged recall	68.72	70.04	+11.32

4 Monitoring Elderly Mind by Agents

4.1 Understanding Internal States

As seen in the previous sections, the environmental sensing achieves automatic and non-intrusive monitoring of physical living environment of elderly people. On the other hand, monitoring with sensors has a limitation that the system can detect only externally observable events. For example, suppose an elderly person sitting in his living room and is concerned about his back pain. Then, the sensors can detect that he is in the living room, but not that his back pain.

When we discussed the activity recognition system with a research collaborator, who was a professional speech therapist specializing in dementia care, he said: "Understanding elderly people by sensors is technologically interesting. However, why you don't ask the person directly how he/she is? Elderly people do not want machines to guess their activities. They are happy that you care about them!". His statement shocked us to realize that we needed a new method that was different from the conventional sensing technology.

The point is how to understand the internal state of elderly people. Here, the internal state refers to a status of a person that cannot be observed externally, including moods, pains, conditions, desires, and intentions. Since the internal state is directly linked to human health, it is important to monitor within the home care [29]. The internal state is usually obtained by conversation, and is technically assessed through inquiries and counseling by clinicians or counselors. However, it is not realistic to request human professionals to monitor the internal state regularly at home.

The episode brought us an idea to utilize the agents technologies. An agent here refers to any software robot that can talk to a human user. It includes animated virtual agents (e.g., MMDAgent [30]) that interact with voice and chat bots that converse via text messages (e.g., LINE Bot [31]).

The key idea is to let an agent talk to the elderly person in the daily life, externalize his/her internal state as words, and record the state with timestamp. We named this idea as Mind Sensing ("Kokoro" Sensing, in Japanese), in the sense that the system is trying to capture the internal mind of elderly people.

4.2 Agent Technologies Developed for Mind Sensing

Using the existing agent technologies, we have developed two kinds of agent systems for the Mind Sensing.

PC Mei-chan

PC Mei-chan is an animated virtual agent implemented with MMDAgent [30, 32]. MMDAgent is a toolkit for building voice interactive software system, which was originally developed in Nagoya Institute of Technology, Japan. MMDAgent contained a variety of modules including text-to-speech (TTS), speech-to-text (STT), voice interaction control, and avatar representation. A virtual agent Mei-chan was contained as a default avatar of MMDAgent.

Since we wanted to integrate Mei-chan with our service-oriented smart home (see Sect. 1.2), we de-coupled the voice interaction control and avatar representation modules from the system, and wrapped them by Web services [33]. By doing this,

Fig. 14 Virtual caregiver system: (C) 2009–2018 Nagoya Institute of Technology (MMDAgent Model “Mei”)

Mei-chan was controllable via Web-API, and was orchestrated with sensors and home appliances within our smart home.

Using the MMDAgent re-engineered, we developed a system called Virtual Caregiver (VCG) [34], where Mei-chan talks to elderly people in accordance with personalized care scenarios. Integrated with a Web browser, Mei-chan can also present Web contents such as texts, control buttons, pictures and videos. Figure 14 shows the screen of VCG, where Mei-chan is asking “Do you regularly take medicine?” Fig. 15 shows a scene of an experiment, where an elderly person was enjoying her favorite music played by Mei-chan.

We then implemented dialog scenarios in which Mei-chan actively listens to the elderly. Mei-chan asked the elderly about their physical condition and mood in response to the motion sensor, and then listened to them, thus externalizing their internal state into the conversation. At the time, Mei-chan was proven to be a powerful means for the Mind Sensing. We named the system PC Mei-chan for simplicity, in the sense that Mei-chan was working on PC for elderly people.

Figure 16 shows PC Mei-chan, in active listening mode. Until present, various extensions have been made to PC Mei-chan (e.g., [35–37]). Also, it has been deployed in actual elderly household to see if elderly people can accept PC Mei-chan in their daily life. Figure 17 shows some scenes taken from demonstration experiments.

Fig. 15 An elderly person talking to VCG

Fig. 16 PC Mei-chan in active listening: (C) 2009–2018 Nagoya Institute of Technology (MMDAgent Model "Mei")

LINE Mei-chan

In order to achieve portable Mind Sensing, we implemented another version of Mei-chan as a LINE chatbot, which is called LINE Mei-chan. Integrated with LINE Messaging API [31], LINE Mei-chan sends questions for Mind Sensing via a well-known smartphone application LINE. Since the conversation is asynchronous based on text messages, the elderly person can answer the question at any time convenient.

Fig. 17 Elderly people operating PC Mei-chan

Fig. 18 LINE Mei-chan asking questions for mind sensing: (C) 2009–2018 Nagoya Institute of Technology (MMDAgent Model “Mei”)

Also, it can send questions even if the elderly person goes outside. Thus, LINE Mei-chan and PC Mei-chan complement each other, and they are chosen appropriately for the purpose of Mind Sensing.

Figure 18 represents the screenshots of LINE Mei-chan on smartphones. The left figure shows Memory-Aid Service [38], with which the elderly person actively chats to LINE Mei-chan to memorize the current internal state in the system. In the figure, LINE Mei-chan was asking the elderly person what he was doing in the room at 22:19 on June 24, based on the event detected by the activity recognition with the environmental sensing (see Sect. 3). The elderly person answered what he was doing at that time. The answer (i.e., the internal state) was recorded in the system. The service then provides the retrospective process, where the person can review, correct, classify, and search the recorded information of own at any time. Thus, the service is designed for the memory-aid purpose of healthy elders as well as people with cognitive impairment. In [39], we extended the Memory-Aid Service so that LINE Mei-chan asks and records daily health status (e.g., blood pressure, weights, body temperature, mood, etc.).

The right screen of Fig. 18 shows Mind Monitoring Service [40], where LINE Mei-chan periodically sends questions to monitor the internal state of the elderly people for long-term assessment. The Mind Monitoring Service will be described in details in Sect. 5.

4.3 Mind Sensing Service: Rule-Based Service for Systematic Mind Sensing

As we developed various applications using PC Mei-chan and LINE Mei-chan, similar features for Mind Sensing were implemented as different software code within individual applications. Thus, the way of Mind Sensing was tightly coupled with each application, which increased the software complexity, and decreased the flexibility and the scalability.

For instance, the Memory-aid Service introduced in the previous section was tightly coupled with the activity recognition system and LINE Mei-chan. That is, the Mind Sensing can be only triggered by the specific activity recognition, and the inquiry is performed only by the LINE Mei-chan. Also, all the questions were hard-coded within the program. Thus, the service lacked the flexibility, where it was quite difficult to add or change the configuration of Mind Sensing, adapting to individual elderly people.

To cope with the limitation, we developed Mind Sensing Service. The proposed service exploits a rule-based system which allows individual users to define custom mind sensing methods. The key idea is to de-couple the definitions of the mind sensing from the surrounding systems.

System Architecture

Figure 19 shows the system architecture of the Mind Sensing Service. In the proposed service, each mind sensing is defined by a rule, specifying which question is sent, to whom, at when, by what event, and with which message service. Once a rule is defined, the service automatically sends the questions to the target users, and

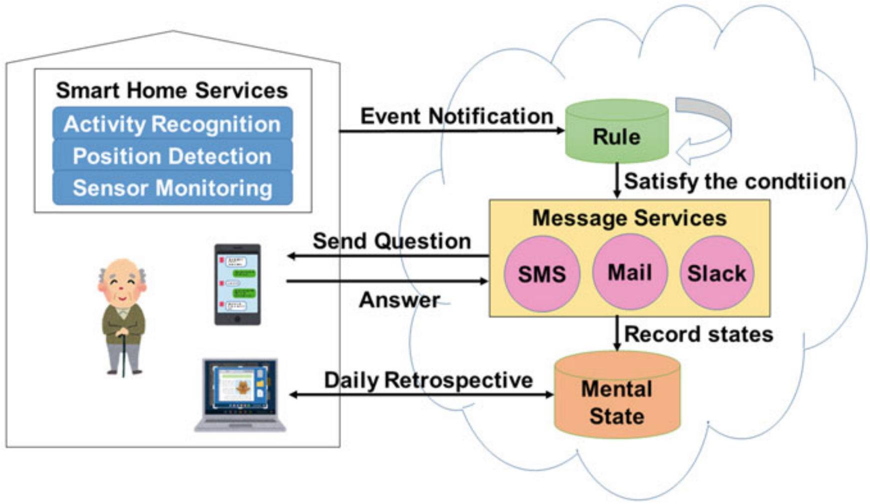


Fig. 19 System architecture of mind sensing service

collects the answers. In the figure, we assume there are various smart home services that support the elderly person at home, including the activity recognition service, the position detection service, the change detection service, and so on. Each of these services generates and manages events. The Mind Sensing Service is supposed to receive event notifications from these services, and ask questions to designated users based on the pre-determined rules.

A rule specifies an enabling condition when the mind sensing should be executed. The condition is based on either time or event. A time-based rule is triggered when the designated time is arrived, while an event-based rule is executed when an event matching the condition is notified. Each rule is associated by a set of actions. An action corresponds to an inquiry to a user, consisting of an address of the user, a question to ask, and a message service to deliver the question. We adopt various messaging services, including SMS (short messaging service), Email, and Slack, to inquiry the questions to the target user. By supporting interaction with various devices such as smartphone and PC, we can perform the mind sensing, according to the lifestyle of individual user.

As the user responds to the question in the natural language text, the answer is then recorded in the database with a timestamp. The stored conversations between a user and a chatbot are later used in services such as Memory-Aid Service. This allows users to review, correct, classify, and search the information they recorded themselves. In addition, through appropriate access control, they can be accessed by third parties such as doctors and caregivers for person-centered care treatments.

Action

An action defines a configuration of concrete inquiry of the mind sensing. The configuration includes three items: `targets` specifies target user(s) ID to inquiry, `messageBody` specifies the content of a question message, and `serviceType` specifies a service to deliver the message.

It is possible to specify multiple users in `targets`, and a designated question can be sent to the multiple users simultaneously. When an action is executed, the system looks up a name aggregation table, which maps a user ID within the proposed service to a user ID of the concrete message service specified in `serviceType`. After resolving the user ID, the service invokes Web-API of the message service, passing the text described in the `messageBody` to the destination address of the user.

For example, suppose that we define the following action: `act1 = {targets: ["Maeda"],messageBody: "How is your current condition?", service: "LINE"}`. The act1 defines an action that the LINE chatbot send a message "How is your current condition?" to a LINE user ID corresponding to "Maeda".

Time-Based Rule

A time-based rule (TBrule, for short) is a rule that repeatedly executes actions at time interval within a designated period. It defines an inquiry without depending on any event from external services. The TBrule can be used when asking questions regularly scheduled or when sending messages at a fixed time of a day. A TBrule is defined by four parameters: `actions` specifies a list of actions to execute, `since` specifies the start time, `until` specifies the end time, and `interval` specifies minutes of the repetition interval.

For example, suppose that we define the following TBrule: `tbrule1 = {actions: ["act1"], since:"10:00", until:"16:00", interval: 60}`. The tbrule1 defines a TBrule that action act1 is executed every hour from 10 o'clock to 16 o'clock every day.

When the service is started, all TBrules in the database are lorded. Each TBrule creates a timer task that periodically checks, for every `interval` minutes, if the current time is between `since` and `until`, and executes the designated actions.

Event-Based Rule

An event-based rule (EBrule, for short) is a rule that is triggered by an event notified from an external service, based on when, where, and what event is notified.

An EBrule is defined by three items: `actions` specifies a list of actions to execute, `conditions` specifies one or more conditions to be satisfied by the event, and `breakTime` specifies minutes of cooling time to the next execution.

When an event is notified, an EBrule is triggered only if all the conditions are satisfied. Each condition is defined by the 5W perspective (i.e., WHO, WHOM, WHAT, WHEN, WHERE). This perspective can cover most events issued by external systems. Each condition is defined by six items: `from`, `to`, `event`, `since`, `until`, `location`:

cid	: condition ID
from	: The subject of event
to	: The object of event
since	: Whether the event took place after this time
Until	: Whether the event took place before this time
Event	: The contents of event
Location	: The location of event
description	: The description of event

For example, suppose we define the following condition: `con1 = {from: "Activity recognition", to:"Maeda", since:"06:00", until: "10:00", event:"Waking up", location:"Bedroom", }` The `con1` defines a condition that activity recognition service detects user Maeda's waking up in the bedroom.

Next, let us define the following EBrule: `ebrule1 = {actions: ["act1"], conditions: ["con1"], breakTime: 30}` . The ebrule1 defines a rule that action act1 is executed only when the condition `con1` is fulfilled. That is, when that activity recognition service detects that Maeda wakes up, then send him a question "How is your current condition?" by LINE. Once ebrule1 is executed, it will not run for the next 30 minutes.

To receive the event notification from external systems, the proposed service exposes REST API, with a method postEvent(from, to, event, time, location). When the external system executes the API, the service evaluates conditions of every EBrule against the given values of the parameters. For example, when postEvent("Activity recognition service", "maeda", "10:42:24", "Waking up", "bedroom") is executed for the above con1, it returns false because the perspective of the WHEN is not met.

4.4 Case Study

Collecting Mental State by LINE Chatbot

As a case study, we conduct an experiment that obtains user's mental state by sending questions using Mind Sensing Service. The purpose of this case study is to confirm if the proposed service works as expected, and to see how effectively the system can collect user's mental states.

In the experiment, a questioner, who is a professional speech therapist, created 42 questions by referring to the mental illness questionnaire sheets. Then, the questioner wanted to ask each subject three questions at a time, twice a day at 6:30 and 21:30.

Since each question was a bit technical, sending the question by text message was more understandable than sending it by voice message. Therefore, we chose LINE chatbot as the message service.

As receiving a question, a subject answered the question by four-level evaluation: (0) not at all, (1) don't think so, (2) think so, and (3) absolutely think so. Then, the answer was stored in a database for later analysis. After the user answers 42 questions in 7 days, we finally sent the review of the week, and questionnaire that asked the actual mental state of the subject.

Creating Rules with Mind Sensing Service

In order to start the experiment, the questioner had to register actions and rules to the proposed system, so that three questions were sent to designated subjects every morning and evening. Additionally, the questions had to be updated to cover the total 42 questions. Therefore, we implemented a Web application that allows the questioner to easily create and update actions and rules within a Web browser. Figure 20 shows a snapshot of the Web application. In the screen, a list of actions and rules registered for each user is shown. With the application, the questioner can easily register, update, and delete them.

First, we set the user information on the subject. The subjects are 6 people in their 20s–60s and we register their user ID and account information of LINE. Because the content of the question is somewhat technical and the SMS text message is easy to convey, we adopt LINE application as the message service we use to send questions.

Second, we set actions to send a question to the subject. In this experiment, we send four messages at a time, twice a day, at specific times in the morning and evening. We accordingly register 8 actions: `MorningAction0,1,2,3` and `EveningAction0,1,2,3`. `MorningAction0` and `EveningAction0` are greeting messages to start an inquiry. `MorningAction1,2,3` and `EveningAction1,2,3` define concrete three questions asked in the morning (orevening, respectively) inquiry. For example, `MorningAction1` is described as follows:

`MorningAction1 = {targets: ["maeda", "yasuda",…, "nakamura"], messagebody: "[Question#1]Do you think you feel satisfied with your daily life?", service: "LINE"}`.

Finally, we set two TBrules to execute actions: `MorningRule` and `EveningRule`. In both rules, the interval is set to 1440 minutes, so that MorningRule and EveningRule are executed exactly once a day. Thus, each target subject receives three questions following a greeting message every morning at 6:30, as well as every evening at 21:30. For example, `MorningRule` is described as follows: `MorningRule = {actions: ["MorningAction0", "MorningAction1", "MorningAction2", "MorningAction3"], since: "06:30", until: null, interval:1440}`.

Fig. 20 Web application managing actions and rules

Result and Feedback

Figure 21 shows the LINE chatbot interacting with the subject. The chatbot sends a message asking about the mental state by `MorningAction0,1,2,3` at 6:30, defined in `MorningRule`. Subjects responded to these questions at any time. In this way, by using the proposed service, it is possible to make a rule-based question as a basic service for Mind Sensing.

In this case study, the questioner, who made the question from some existing questionnaire sheets for mental illness, set some actions and rules through GUI on PC. She said this rule-based talking service was useful and more efficient than manual transmission. On the other hand, she pointed out a lack of usability of GUI.

5 Design and Evaluation of Mind Monitoring Service

5.1 Monitoring Internal State for Long Term

Our next challenge is how to monitor the physical and mental health in-home elderly people for a long term through the Mind Sensing. In general, it is not easy to obtain the physical and mental health condition at home by external observation by non-intrusive sensors. Thus, the proposed mind Sensing with the agent is a promising approach. However, how to observe and assess the health condition by the Mind Sensing is still an open question.

According to the World Health Organization (WHO), the concept of health is defined as follows [41]:

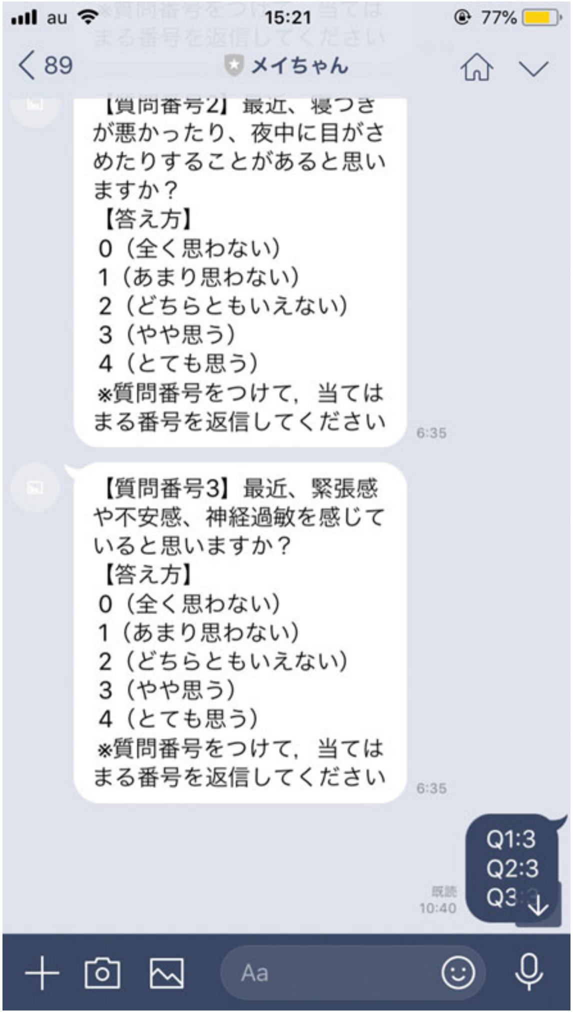


Fig. 21 Interaction between a subject and LINE chatbot

> Health is a state of complete physical, mental and social well-being and not merely the absence of disease or infirmity.

By this definition, health is a state that can be characterized by three aspects: physical, mental and social aspects.

As the physical and cognitive functions decline, elderly people easily develop not only physical illness, but also mental illness. Typical mental illnesses elderly people tend to develop include depression [42] and anxiety disorder [43]. A major factor that causes such mental illnesses lies in their experiences of loss. The experiences include the deterioration of physical ability due to aging, the loss of social role by the retirement, and the bereavement of familiar people.

In clinical scenes, psychological assessment tools, including tests, scales, and questionnaires, are used to quickly assess the mental state of the person. Representative tools include, GDS-15 (Geriatric depression scale 15) [44]: the depression scale for the elderly, PHQ-9 (Patient Health Questionnaire-9) [45]: assessment of general depression, GAD-7 (Generalized Anxiety Disorder-7) [45]: measuring the degree of anxiety disorder, and GHQ (General Health Questionnaire) [46]: assessment of neurosis. However, it is unrealistic for in-home elderly people to use these tests regularly at home.

5.2 Concept of Mind Monitoring Service

Exploiting the Mind Sensing Service with LINE Mei-chan, we have developed a new service named Mind Monitoring Service [47–49]. The service aims to visualize and monitor the mental states of elderly people at home, through a continuous interaction with LINE Mei-chan. The service also provides appropriate supports based on the acquired mental states data to encourage user's self-reflection and spontaneous self-care of mental health.

The concept of Mind Monitoring Service is to grasp mental states of elderly people at home, which have been difficult to obtain so far. It also tries to provide appropriate supports according to the mental states. For this, we utilize LINE Mei-chan, in order to establish continuous interaction platform with elderly people at home. Moreover, we develop specific questions to acquire mental states of the elderly person. We also introduce scoring methods for evaluating answers of the questions and visualizing mental states numerically.

5.3 System Architecture

Figure 22 shows the overall system architecture of the Mind Monitoring Service. As seen in the figure, the proposed service consists of three methods.

M1: Interaction with LINE Mei-chan using Mind Sensing Service: We utilize the Mind Sensing Service (see Sect. 4.3), and let LINE Mei-chan ask questions to an elderly person every day. In stead of human caregivers, the chatbot listen to and record the internal minds of the elderly person continuously.

M2: Inquiry method specialized for acquisition of mental state: We develop inquiries specific for acquiring mental states of the elderly person. The inquiries are stored in a database. The inquiries are then encoded by actions and rules of the Mind Sensing Service.

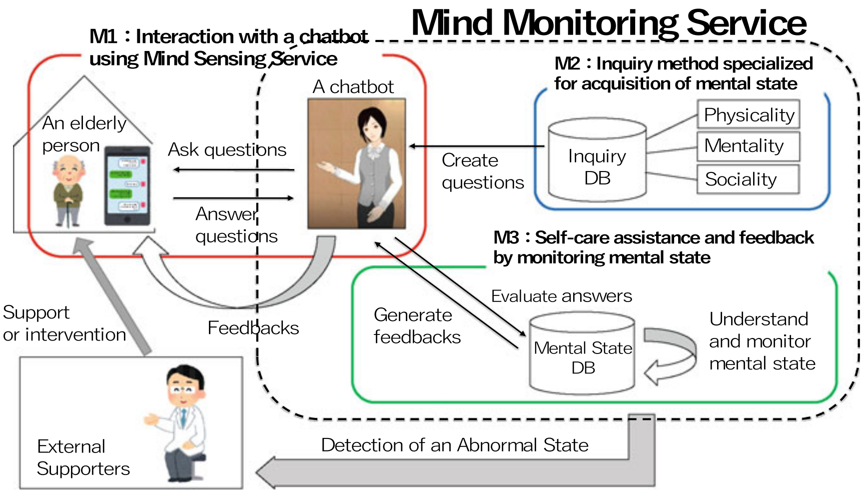


Fig. 22 System architecture of mind monitoring service

M3: Self-care assistance and feedback by monitoring mental state: Every time the elderly person answers a question, the answer is stored in a database with timestamp. With an appropriate period, the service then analyzes the answers and evaluates his/her mental states. According to the result, the service produces feedback including further questions and advices.

5.4 M1: Interaction with LINE Mei-Chan Using Mind Sensing Service

Using the Mind Sensing Service, we let LINE Mei-chan ask questions to elderly people triggered by time or external events. Since the internal state should be obtained within a daily routine, we send questions to elderly people at a fixed time every day. For this, we apply the time-based rule of Mind Sensing Service. The time for sending the questions should be set in consideration of the person's daily rhythm and lifestyle. It is also necessary to change questions every day, and to send encouraging messages, so that the person does not get tired and quit answering.

To make the interaction with LINE Mei-chan easier, we extensively use LINE template message, where we embed a question and the list of choices for the answer within a pre-defined layout. Figure 23a shows a screenshot of a template message. In the figure, the question is written in the middle part of the template message, and the answer choices are given by buttons at the bottom of the message. To answer the question, the user only has to push one of the two buttons. Since this way of answering does not need entering any text, it makes the elderly answer questions easily.

(a) Template Message (b) Reply Message

Fig. 23 Interaction with LINE Mei-chan in mind monitoring service

We also treat the user's answer as an event, and command LINE Mei-chan to send a reply message using the event-based rule. For instance, when the template message provides two buttons meaning "yes" or "no", the user's answer can be classified either positive or negative. Therefore, by defining two kinds of replies in advance, we can make LINE Mei-chan reply a different reply depending on the answer.

Figure 23b shows the actual interaction between the chatbot and the user. In Fig. 23b, the chatbot replies to user's positive answer to the question "Have you slept well in the past week?" After understanding a good sleep condition of the user, the chatbot sends an additional question to ask any concerns regarding sleep. The user can input any text messages and externalize his or her minds as words.

Design of these questions and reply messages will be described in the next section.

5.5 *M2: Inquiry Method Specialized for Acquisition of Mental State*

Monitoring Internal States from Three Aspects

Integrating the definition of health in Sect. 5.1, we monitor the internal status of elderly people in the following three perspectives: Physicality, Mentality, and Sociality.

- Physicality corresponds to the physical aspect of health. It targets physical symptoms that can be explained by objective factors. We try to grasp the health, according to the presence or absence of the physical symptoms.
- Mentality corresponds to the mental aspect of health. It covers subjective feelings such as emotions and moods. We characterize the mental health by subjective assessment.
- Sociality corresponds to the social aspect of health and sickness. It covers the self-evaluations and behaviors such as happiness, self-esteem, motivation, or social behaviors. We try to understand the health from social aspects.

Preliminary Experiment

In our preliminary experiment [48], we developed questions by referring to the psychological assessment tools (see Sect. 5.1). Specifically, based on GDS-15, PHQ-9, GAD-7, and GHQ60, we created 42 questions in total. We then classified the 42 questions into the above three categories. Table 6 shows 21 questions assessing Mentality. Each question was supposed to be answered by 4-level scales:"Yes, I really think so", "Yes, I might think so", "No, I might not think so", "No, I definitely do not think so". The mind sensing was performed twice in the morning and in the evening, and, for each time, three questions out of the 42 were sent to each elderly person by LINE Mei-chan.

In fact, however, the preliminary experiment did not work well. Each question was so technical that the elderly people could not understand well the meaning and the intention of the question. It was also too much to ask three technical questions twice a day, which was a burden for the subjects.

Simplifying Questions and Interactions

With the help of experts, we re-drafted the questionnaire into seven questions shown in Table 7, so that the elderly people can easily answer the questions. The seven questions were intended to grasp approximate state within the week from seven fundamental aspects of daily living: Sleep, Health, Emotion, Memory, Psychology, Motivation, and Socialization. Each question asks the state of past one week, and

Table 6 Questions for assessing mentality, created in the preliminary experiment

Questions (mentality)	Inferred symptom
Do you think your daily activities and interests have declined?	Deline of activity and interests
Do you think you are often driven by vague anxiety in the future?	Future anxiety
Do you think you want to stay home rather than going out or doing new things?	Decline of activity
Do you think you are more worried about forgetting things than anything else?	Concern over forgetfulness
Do you think you feel that there is no hope?	Despair
Do you think you have little interest or enjoyment of things?	Decline of interests
Do you think you feel depressed and hopeless?	Despair
Do you think you feel nervous or anxious recently?	Tension, anxiety, oversensitive
Do you think you have been too worried recently?	Worry
Do you think it is difficult to relax?	Difficulty of relaxing
Do you think you feel restless?	Restlessness
Do you think you feel annoyed and angry recently?	Anger
Do you think you might be afraid that something terrible will happen?	Fear
Do you think you cannot sleep because of worries?	Worry
Do you think you always feel stress?	Chronic stress
Do you think you might get frustrated and angry?	Anger
Do you think you are scared of something for no particular reason?	Fear
Do you think everything is more burdensome for you than usual?	Stress
Do you think you feel anxiety or tension?	Anxiety, tension
Do you think you often seem to be in a good mood?	Bad mood
Do you think you cannot stop worrying?	Worry

each elderly person is supposed to answer it with simply “Yes” or “No”, instead of the 4-level scale. “Survey item” in Table 7 indicates what to investigate by the question. For example, the question “Have you slept well in the past week?” investigates the condition of sleeping. Besides, “Category” shows a class of the three perspectives.

We also configured the time-based rule so that LINE Mei-chan sent only one question per day to the elderly person. The time of the message delivery was determined

Table 7 New seven questions

Question	Survey item	Category
Have you slept well in the past week?	sleep	Physicality
Have you felt sick, pain, or tierd during the past week?	Health	Physicality
Have you had something fun in the past week?	Emotion	Mentality
Have you felt you could not remember something, or forgotton something in the past week?	Memory	Mentality
Have you felt anexity or unwell during the past week?	Psychology	Mentality
Have you felt not motivated or appetite in the past week?	Motivation	Sociality
Have you had many opportunities to go out, to talk and to have hobbies in the past week?	Socialization	Sociality

according to the person's life rhythm. Since one question was sent once a day, all of the seven questions were covered in a week. In the next week, LINE Mei-chan sent the first question again.

To keep the motivation of elderly person answering the question, we let LINE Mei-chan to send reply messages as well as LINE stamps. Table 8 shows an example of the pre-defined reply messages. "Positive Reply" and "Negative Reply" were sent when the user answered the question positively and negatively, respectively. Each reply message was an open question asking why the user selected the choice, externalizing any concerns related to the question.

As shown in Fig. 23b, when the elderly person sends the details of the additional question, LINE Mei-chan sends LINE stamp back to the elderly. We implemented these interactions with the LINE reply messages and event-based rules of the Mind Sensing Service, regarding each answer as an event.

5.6 *M3: Self-Care Assistance and Feedback by Monitoring Mental State*

Based on collected answers from each elderly person, the Mind Monitoring Service then evaluates his/her mental state. According to the result, the service produces feedback including further questions and advices. The service also provides a Web application with which the user can review the past answers.

Table 8 Reply messages according to the user's choice

Question	Positive reply	Negative reply
Sleep	I'm glad to hear that! Do you have any concerns about sleep? If there is anyting else other than sleep, please talk to me about it	I see. Do you have any idea why you're not sleeping? Please tell me the details if you would like
Health	I'm glad to hear that! If you have any concerns, not just about your physical condition, plsease feel free to talk to me	I see. Where in your body are you feeling discomfort or pain? Please tell me more about it if you would like
Emotion	That's good to know! What fun did you have? Please let me know if you would like!	I see. In contrast, were there anything sad or frustrating? You can tell me what kind of things happened to you. if you like
Memory	I see. Even though you don't have forgetfulness, please talk to me if you feel your memory is deteriorating	I see. What kind of things did you have trouble remembering or forgetting? Please let me know if you don't mind
Psychology	Glad to hear you're feeling good! If you have any other concerns about your mood, please talk to me	I see. What kind of things make you feel anxious or upset? If you would like, please tell me about it
Motivation	i'm relieved to hear that. But please don't strain yourself too much! If you have any other concerns, please talk to me	I see. Do you have any idea what's causing you to feel unmotivated? Please tell me the details if you would like
Socialization	That's good! Where you been , and who have you been talking to? Please let me know if you would like!	I see. Are there any reasons why you are not going out or talking to someone? If you have any concerns, please talk to me

Quantifying Answers for Assessment of Mental State

Since each elderly person answers each of the seven questions with "Yes" or "No", the mental state of the week with respect to a category can be assessed to be positive or negative. We also take care of how the state of the week was changed from that of the previous week. If the state remained negative, the situation is bad. If it changed from negative to positive, it is a good sign but still needs to be observed. Finally, we should take the answer for the open question into account. Based on these consideration, we have proposed a method that quantifies the mental state by the following three kinds of scores:

(i) **Score_answer:** The score directly obtained from the answer. We assign 1 point for a positive answer and −1 point for a negative answer.

(ii) **Score_observation:** The score obtained by observing how the answer has changed from the previous week. When the user answered positively in the previous week, if the answer remains positive in the target week, 1 point is assigned. If the answer turns negative, -0.5 points is assigned. Similarly, when the user answered negatively in the previous week, if the answer remains negative in the target week, -1 points is assigned to the answer. If the answer turns positive, 0.5 points is assigned.

(iii) **Score_sentiment:** The score obtained by sentiment analysis of user's answer to the additional open question. Using Microsoft Azure Text Analytics API [50], the service calculates a sentiment value (from negative to positive) from the given text sentence. The score is then normalized so that it takes a value from -1 to 1 (1 points means the most positive).

Finally, we calculate the total score of the answer by the weighted sum of the above three scores.

$$S_{\text{total}} = w_1 \cdot S_{\text{answer}} + w_2 \cdot S_{\text{observation}} + w_3 \cdot S_{\text{sentiment}}$$

Currently, we calculate the total score as the average of the three scores, where $w_1 = w_2 = w_3 = \frac{1}{3}$.

Generating Weekly Feedback for Spontaneous Self-Care

Based on the score of the mental state, Mind Monitoring Service generates a weekly feedback to promote the user's self-reflection and spontaneous mental health care. This feedback generation is intended to implement an instance of (S1) Self-aid support service in our conceptual architecture (see Fig. 1).

In the feedback, the service firstly selects one question whose score is the worst in a week. Secondly, the service creates a concrete feedback message to be sent by LINE Mei-chan. In order to generate natural sentences, we structured a feedback message by four paragraphs: Greeting, Reflection, Advice, and Conclusion.

More specifically, in the greeting paragraph, the chatbot greets the user according to the season or climate. In the refection paragraph, the chatbot shows how the user answered the selected question in order to get the user to look back him- or herself. The advice paragraph gives the user useful information about the content of the question. We refer to the information of the "Kenko-Choju Net" [51], which provides a lot of information about health and longevity for Japanese elderly people. Lastly, in the conclusion paragraph, the chatbot gives a closing remark, such as "Let's do our best again this week."

Figure 24 shows an example of a feedback message. In this feedback, the question about psychology was picked up. Since this feedback was created in June, the chatbot firstly mentioned the climate in June. The chatbot secondly indicated that the user had been feeling anxiety, and suggested to have her family or friends listen to the anxiety.

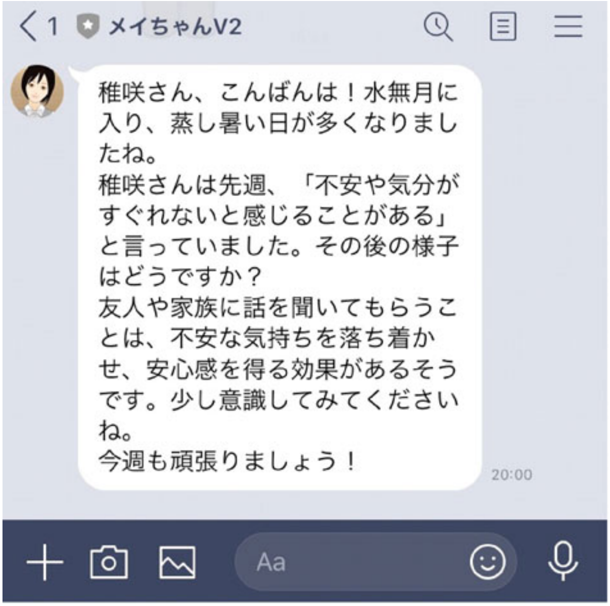


Fig. 24 Example of a weekly feedback message

The sentences of each paragraph are pre-defined, and the service are combining these paragraphs to create the complete feedback message.

Developing Web Application for Visualization

To realize effective mind monitoring, we have developed a web application that visualizes the score of the mental states. Using the application, the elderly person can review his/her mental states. Furthermore, upon the consent of the elderly person, remote supporters (family members, caregivers, doctors, etc.) can watch the target person's mental states by data.

Figure 25 shows the developed application. As shown in the left figure, LINE Mei-chan sends the URL of the application after the weekly feedback. When the elderly person taps the URL, the application shows the weekly score for each of the survey item, as shown in the middle of Fig. 25. As shown in the right figure, the application can also display the time-series score with respect to Physicality, Mentality, and Sociality. Thus, the elderly person and the external supporters can conduct long-term monitoring of the internal state.

Fig. 25 Web application visualizing the mental state

5.7 *Operating Mind Monitoring Service in Actual Households*

Long-term Monitoring Experiment

The Mind Monitoring Service has been deployed on actual households, and been operated for long-term monitoring of their internal states. We recruited 8 elderly subjects (4 men, 4 women in the 50 s–80 s), who were able to use LINE application. The operation period was from November 1, 2019 to January 31, 2021, one year and two months (14 months) in total. The experiment has been approved by the research ethics committee of Graduate School of System Informatics, Kobe University (No. R01-02). Written informed consent was obtained from subjects for publication and accompanying images.

Two elderly subjects dropped out from the experiment within a few months. As for one elderly (male in the 70s), it was difficult for him to use the service because he did not use his smartphone frequently in his daily life. The other person (male in his 70s) had been using the service for the first three months, but he eventually stopped using it because his asthma worsened, and made him difficult to continue to answer the questions from LINE Mei-chan every day.

The remaining 6 subjects kept using the Mind Monitoring Service. Table 9 shows the response rate of each subject, which is the ratio of the number of responses (i.e., the answers) the subject made to the total number of questions from LINE Mei-chan during the 14 months.

Table 9 Total response rate of elderly subject

Subject	Age	Gender	Rate (%)
A	70∼79	M	91
B	60∼69	M	92
C	80∼89	F	30
D	70∼79	F	90
E	70∼79	F	54
F	50∼59	F	95

From Table 9, we can see four out of six elderly subjects responded to more than 90% of the questions from a chatbot. For subject C and E, the overall response rate was low, not because they had stopped using the service, but simply because they responded less frequently. In other words, we could not get high frequency of responses from subject C and E, but we were able to get them to answer the questions periodically.

Analyzing Time-Series Data in Detail

Through the 14 months of the operation, the Mind Monitoring Service collected a large amount of mental state data. Figures 26, 27 shows graphs of the mental state scores of two elderly subjects, subject A (male in 70 s) and subject D (female in 70 s), in 2020. In the graph, the vertical axis represents the average score value and the horizontal axis represents months. The blue, yellow, and green lines represent the scores of Physicality, Mentality, and Sociality, respectively.

In Fig. 26, we can see that subject A's scores of each perspective are generally positive. This means that his health in terms of Physical, Mental and Social was maintained stable throughout the year. However, we can see that his Physicality score dropped sharply in the middle of May. When we asked subject A about the reason, he said that he had hurt his leg by walking too much at that time. Afterward, thanks to treatment and rehabilitation, his leg finally started to get better around July.

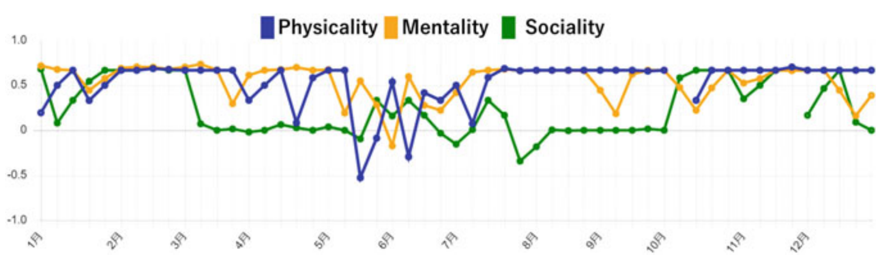


Fig. 26 Transition of scores of subject A

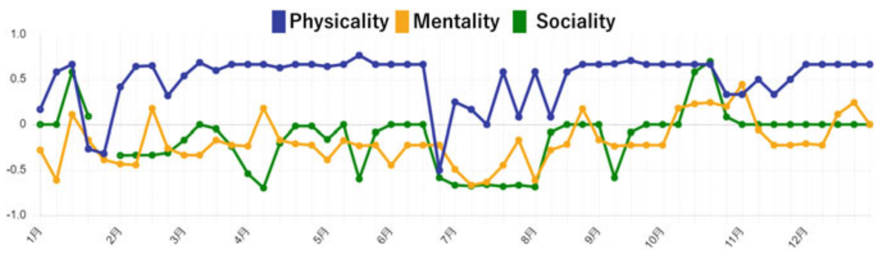


Fig. 27 Transition of scores of subject D

Also, we can find that his Sociality score dropped in early March. This was because the spread of the coronavirus (covid-19) reduced his opportunities to go out. For a while after that, he stopped exercising at the gym, and his Sociality score continued to stagnate. However, he started to go to the gym again around October, and his Sociality score started to increase.

In Fig. 27, it can be seen that the Mentality score and Sociality score of Subject D were much lower than Physicality score. When we asked Subject D about her situation in 2020, she told us that her elder sister had passed away in January and she had been experiencing terrible sense of loss. This sense of loss continued until around October, and her Mentality went through a series of manic-depressive cycles. She also had a lifestyle in which her days and nights were reversed. In contrast, her Physicality score tended to be relatively positive, but we can also find a sudden decrease in her Physicality score around July. In fact, at that time, she was suffering from dizziness caused by otolith detachment. Later, as she learned how to live well with her illness, her Physicality score gradually recovered.

Internal Mind Externalized as Words

During the experiment, elderly people sometimes answered the additional open question by their own words. We found that these words well characterized the internal mind of each elderly person, which was never captured by the conventional sensors. It seemed that elderly people externalized their minds through conversations with LINE Mei-chan.

Figure 28 shows a part of conversation log of Subject D, where the log is listed from the new to the old. As we read the log from the bottom, she answered fine for the question No. 4 of Physicality, but she recognized that she got tired easily because of age. Related to the question No. 5 about the memory, she was anxious to forget what she ate, promised, shopped, etc. As for the question No. 6 about the socialization, she answered negatively, and she missed her friends as she grew older and they passed away. Finally, she answered negatively for the question No. 7 about anxiety, because of the Covid-19.

2020-02-23T22:10:34+0900
コロナウイルスのせいですかネ。
2020-02-23T22:09:17+0900
【質問番号7】*********さん、ここ１週間不安や気分がすぐれないなどを感じることはありますか？ -> ありります
2020-02-22T21:24:42+0900
年を取ると　友達が消えます。致しない事ですネどんなに沢山の仲間や友達がいたにしても。
2020-02-22T21:18:40+0900
【質問番号6】*********さん、ここ１週間外出、会話、趣味などをする機会は多いですか？ -> あまり多くないです
2020-02-21T22:42:04+0900
食べた物　約束した事　買い物の買い忘れ、等等です。どうしたらいいでしょうネ。
2020-02-21T22:37:08+0900
【質問番号5】*********さん、ここ１週間何か思いだせなかったり、忘れたりしたことがありますか？ -> もの忘れが気になります
2020-02-20T21:37:08+0900
疲れ安いのは、加齢のせいと諦めてます。
2020-02-20T21:34:23+0900
【質問番号4】*********さん、ここ１週間体調不良、体の痛み、疲れやすいなどはありましたか？ -> 元気です

Fig. 28 A part of conversation log of subject D (in original text)

Currently, we evaluate these words by the simple sentiment analysis as described in Sect. 5.6. More sophisticated analysis should be considered to detect the severe situation, which is left for our future work.

6 Conclusion

In this chapter, we have introduced our research achievements of sensing technologies for monitoring in-home elderly people. In the first half of the chapter, we presented technologies of monitoring daily living of elderly people. Equipped with seven kinds of environmental sensors, the developed Autonomous Sensor Box enables non-intrusive environmental sensing 24 h 365 days with minimized maintenance effort at the edge side. The collected time-series data allows the elderly person as well as remote supporters to reason how the person is living. We have also presented a method of automatic activity recognition with the environmental sensing data. Using the time-series sensor values labeled by the LifeLogger tool, we have shown that the supervised machine learning was able to recognize the seven kinds of daily activities to some extent of accuracy. It was also shown that the accuracy was improved using location information collected by BLE beacons together with the environmental sensing data.

In the latter half of the chapter, we presented technologies for monitoring internal minds of in-home elderly people. The proposed concept of Mind Sensing (Kokoro Sensing) aimed to externalize the internal minds as words through the conversation with virtual agents. By wrapping the sophisticated MMDAgent components with Web services, we implemented an animated virtual agent PC Mei-chan, who talks to the in-home elderly person to obtain the internal state via voice. We also implemented a LINE chatbot, LINE Mei-chan, who provides asynchronous text communication to obtain the internal state. To achieve the efficient and flexible mind sensing, we introduced Mind Sensing Service, by which users can define custom mind sensing methods by time-based and event-based rules. Finally, we presented the Mind Monitoring Service to achieve long-term monitoring of the internal minds. It was shown, in the long-term experiment, that the scores of the states with respect to Physicality, Mentality, and Sociality characterized well the situation of the target elderly person, and that the internal minds were externalized as words in the answers of open questions.

Our research and development for assisting elderly people are still ongoing, and there are many other achievements that could not be introduced in this chapter (e.g., [52–56]). Although there are still many challenges, we believe that the idea of using smart technologies and big data for person-centered elderly assistance and care is crucial in this super-aging society. The integration with the latest AI technologies such as deep learning and large language model (LLM) is also promising for our future work.

References

1. Government of Japan: annual report on the aging society (2020). http://www.cao.go.jp/ (2020)
2. The Ministry of Health, Labor and Welfare in Japan: Community-based integrated care system (2013). https://www.mhlw.go.jp/stf/seisakunitsuite/bunya/hukushi_kaigo/kaigo_koureisha/chiiki-houkatsu/
3. Yasuda K (2022) Rehabilitation for MCI and dementia: using assistive technology to support daily activities. Union Press, Japan
4. International society for gerontechnology (2017). http://www.gerontechnology.org/
5. Papazoglou MP, Heuvel WJVD (2007) Service oriented architectures: approaches, technologies and research issues. The Vldb J. https://doi.org/10.1007/s00778-007-0044-3
6. Nakamura M, Matsuo S, Matsumoto S (2012) Supporting end-user development of context-aware services in home network system. In: Lee R (ed) Studies in computational intelligence. Springer, pp 159–170
7. Nakamura M, Tanaka A, Igaki H, Tamada H, ichi Matsumoto K (2008) Constructing home network systems and integrated services using legacy home appliances and web services. Int J Web Serv Res 5(1):82–98
8. Watanabe Y, Matsumoto S, Saiki S, Nakamura M (2015) Applying log data to context-awareness in home network system. EAI Endorsed Trans Context-aware Syst Appl 15(3):e3
9. Kagawa T, Saiki S, Nakamura M (2018) PRISM: visualizing personalized real-time incident on security map. Int J Softw Innov (IJSI) 6(4):46–58
10. Sakakibara S, Saiki S, Nakamura M, Matsumoto S (2016) Indoor environment sensing service in smart city using autonomous sensor box. In: 15th IEEE/ACIS international conference on computer and information science (ICIS 2016). Okayama, Japan, pp 885–890
11. Yamamoto S, Matsumoto S, Saiki S, Nakamura M (2014) Design and evaluation of materialized view as a service for smart city services with large-scale house log. IEICE Trans Inf Syst E97-D(7):1709–1718
12. Nakamura M, Hatano K, Miyazaki J, Matsumoto S, Saiki S (2016) Implementing person-centered home dementia care using IoT and cloud computing. JSPS Kaken-hi, Grant-in-Aid for Scientific Research (B) No. 16H02908

13. Nakamura M, Hatano K, Miyazaki J, Yasuda K, Kuwahara N, Kazui H, Saiki S, Tokunada S, Otake M, Kodama N, Kosugi N, Greinar C (2019) Assisting self care and mutual aid of elderly people with dementia at home based on mind externalization. JSPS Kaken-hi, Grant-in-Aid for Scientific Research (A) No. 19H01138
14. Chen L, Hoey J, Nugent CD, Cook DJ, Yu Z (2012) Sensor-based activity recognition. IEEE Trans Syst, Man, Cybern, Part C (Appl Rev) 42(6):790–808. https://doi.org/10.1109/TSMCC.2012.2198883
15. Kusano K, Muro H, Hayashi T, Harada F, Shimakawa H (2011) Derivation of life rhythm from tracing elderly movement. In: The 10th forum on information technology (FIT2011), vol 10, pp 891–892
16. Kwapisz JR, Weiss GM, Moore SA (2011) Activity recognition using cell phone accelerometers. ACM SigKDD Explorations Newslett 12(2):74–82
17. Ueda K, Tamai M, Yasumoto K (2015) A method for recognizing living activities in homes using positioning sensor and power meters. In: 2015 IEEE international conference on pervasive computing and communication workshops (PerCom workshops), pp 354–359. https://doi.org/10.1109/PERCOMW.2015.7134062
18. Wang Z, Yang Z, Dong T (2017) A review of wearable technologies for elderly care that can accurately track indoor position, recognize physical activities and monitor vital signs in real time. Sensors 17(2). https://doi.org/10.3390/s17020341, https://www.mdpi.com/1424-8220/17/2/341
19. Phidgets Inc.: unique and easy to use USB interfaces. http://www.phidgets.com/
20. Munguia-Tapia E, Intille S, Larson K (2004) Activity recognition in the home using simple and ubiquitous sensors. In: Proceedings of PERVASIVE, pp 50–57
21. Philipose M, Fishkin KP, Perkowitz M, Patterson DJ, Fox D, Kautz H, Hahnel D (2004) Inferring activities from interactions with objects. IEEE Pervasive Comput 3(4):50–57
22. Pei L, Guinness R, Chen R, Liu J, Kuusniemi H, Chen Y, Chen L, Kaistinen J (2013) Human behavior cognition using smartphone sensors. Sensors 13:1402–1424
23. Niu L, Saiki S, Nakamura M (2018) Using non-intrusive environmental sensing for ADLS recognition in one-person household. Int J Softw Innov (IJSI) 6(4):16–29
24. Fiore L, Fehr D, Bodor R, Drenner A, Somasundaram G, Papanikolopoulos N (2008) Multi-camera human activity monitoring. J Intell Robot Syst 52(1):5–43
25. Fujino Y (2008) A prospective cohort study of shift work and risk of ischemic heart disease in Japanese male workers. J Univ Occupational Environ Health 30(1):104
26. Czerwonka J, Pairwise independent combinatorial testing tool (PICT). https://github.com/microsoft/pict
27. Microsoft: machine learning—initialize model—classification (2018). https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/machine-learning-initialize-model-classification
28. Niu L, Saiki S, Nakamura M (2017) Integrating environmental sensing and BLE-based location for improving daily activity recognition in OPH. In: The 19th international conference on information integration and web-based applications & services (iiWAS2017). Salzburg, Austria, pp 330–337
29. Di Cesare G, Di Dio C, Marchi M, Rizzolatti G (2015) Expressing our internal states and understanding those of others. Proc Natl Acad Sci 112(33):10331–10335. https://doi.org/10.1073/pnas.1512133112
30. MMDAgent Project Team: MMDAgent—toolkit for building voice interaction systems (2018). http://www.mmdagent.jp
31. LINE Cooporation: using LINE Bot designer (2021). https://developers.line.biz/en/docs/messaging-api/using-bot-designer/
32. Lee A, Oura K, Tokuda K (2013) MMDAgent—a fully open-source toolkit for voice interaction systems. In: 2013 IEEE international conference on acoustics, speech and signal processing, pp 8382–8385. https://doi.org/10.1109/ICASSP.2013.6639300
33. Horiuchi H, Saiki S, Matsumoto S, Nakamura M (2015) Virtual agent as a user interface for home network system. Int J Softw Innov 3(2):24–34

34. Tokunaga S, Tamamizu K, Saiki S, Nakamura M, Yasuda K (2016) VirtualCareGiver: personalized smart elderly care. Int J Softw Innov (IJSI) 5(1):30–43. https://doi.org/10.4018/IJSI.2017010103, http://www.igi-global.com/journals/ abstract-announcement/158780
35. Chen S, Nakamura M (2022) Study of multi-modal diary service using spoken dialogue agent for self-care in elderly people. In: 2022 1st international conference on software engineering and information technology (ICoSEIT), pp 120–125
36. Ozono H, Chen S, Nakamura M (2021) Study of microservice execution framework using spoken dialogue agents. In: 22nd IEEE-ACIS international conference on software engineering, artificial intelligence, networking and parallel distributed computing (SNPD2021). Taichung, Taiwan, pp 273–278
37. Tokuda Y, Nakamura M, Saiki S, Yasuda K (2020) Extracting and evaluating personal interests with dialogue agent. In: 11th international conference, DHM 2020, held as part of the 22nd HCI international conference, HCII 2020, vol LNCN 12198. Springer. Copenhagen, Denmark, pp 592–608
38. Maeda H, Saiki S, Nakamura M, Yasuda K (2019) Memory aid service using mind sensing and daily retrospective by virtual agent. In: 10th international conference, DHM 2019, held as part of the 21st HCI international conference, HCII 2019, vol LNCS 11582. Springer, pp 353–364
39. Maeda H, Saiki S, Nakamura M, Yasuda K (2019) Recording daily health status with chatbot on mobile phone—a preliminary study . In: the 12th international conference on mobile computing and ubiquitous networking (ICMU 2019). Kathmandu, Nepal
40. Miura C, Maeda H, Saiki S, Nakamura M, Yasuda K (2020) Implementing mind monitoring service for elderly people at home using line chatbot. In: ISG's 12th world conference of gerontechnology, vol 19(suppl)
41. World Health Organization: constitution of the World Health Organization (1946). https://www.who.int/
42. The Ministry of Health, Labor and Welfare in Japan: Depression prevention and support manual (revised version) (2009). https://www.mhlw.go.jp/topics/2009/05/dl/tp0501-1i.pdf
43. Takeda M, Tanaka T (2007) E de Miru Kokoro no Hokenshitu(Mental clinic with pictures). Alta Publishing Co., Ltd
44. Sugishita M, Asada T (2009) Geriatric depression scale and minus; short version-Japanese, gds-s-j. Jpn J Cognit Neuroscie 11(1):87–90. https://doi.org/10.11253/ninchishinkeikagaku.11.87
45. Muramatsu K (2014) An up-to-date letter in the Japanese version of phq, phq-9, phq-15. https://ci.nii.ac.jp/naid/120005482683/
46. Goldberg DP, Nakagawa Y, Daibo I (1985) The general health questionnaire. Nihon Bunka Kagakusha
47. Chisaki M, Chen S, Saiki S, Nakamura M, Yasuda K (2022) Assisting personalized healthcare of elderly people: developing a rule-based virtual caregiver system using mobile chatbot. Sensors 22(10):3829. https://doi.org/10.3390/s22103829
48. Miura C, Maeda H, Saiki S, Nakamura M, Yasuda K (2019) Prototyping and preliminary evaluation of mind monitoring service for elderly people at home. In: 21st international conference on information integration and web-based applications & services (iiWAS2019). Munich, Germany, pp 439–445
49. Miura C, Saiki S, Nakamura M, Yasuda K (2020) Implementing and evaluating feedback feature of mind monitoring service for elderly people at home. In: The 22nd international conference on information integration and web-based applications & services (iiWAS2020), pp 390–395
50. Microsoft Azure: text analytics. https://azure.microsoft.com/en-ca/services/cognitive-services/text-analytics/
51. The Japan Foundation for Aging and Health: Kenko-Choju Net. https://www.tyojyu.or.jp/net/
52. Akashi T, Nakamura M, Yasuda K, Saiki S (2021) Proposal for a personalized adaptive speaker service to support the elderly at home. In: 22nd IEEE-ACIS international conference on software engineering, artificial intelligence, networking and parallel distributed computing (SNPD2021). Taichung, Taiwan, pp 2–7

53. Chen S, Ozono H, Nakamura M, Yasuda K (2023) Quantitative expression of elderly multimodal emotions with spoken dialogue agent and edge AI. In: 2023 6th IEEE Eurasian conference on educational innovation (ECEI), pp 219–221
54. Chen S, Saiki S, Nakamura M (2020) Nonintrusive fine-grained home care monitoring: characterizing quality of in-home postural changes using bone-based human sensing. Sensors 20(20):5894. https://doi.org/10.3390/s20205894
55. Hirayama K, Chen S, Saiki S, Nakamura M (2021) Toward capturing scientific evidence in elderly care: efficient extraction of changing facial feature points. Sensors 21(20):6726. https://doi.org/10.3390/s21206726
56. Ozono H, Chen S, Nakamura M (2022) Encouraging elderly self-care by integrating speech dialogue agent and wearable device. In: 8th international conference, ITAP 2022, held as part of the 24th HCI international conference, HCII 2022, vol LNCS 13331, pp 52–70