User-Centered Design of Translation Systems

Chunqi Shi

September 2013

Doctor Thesis Series of
Ishida & Matsubara Laboratory
Department of Social Informatics
Kyoto University

Abstract

The goal of this thesis is to design an interactive translation system to support multilingual communication using the user-centered design approach; it details how to select the best machine translation for the user's input message, customize translation for different communication topics, and interact with users to improve translation quality for multilingual communication.

Existing studies on machine translation mediated communication show that mistranslation can lead to ineffective communication. Traditionally, machine translators cannot prevent the transfer of mistranslations, and users do not know how machine translator works, thus translation systems are just transparent channels to the users. We analyze three challenges of users' needs and limitations from the perspective of monolingual and non-computing professional users. The first challenge is that how can users use multiple machine translators. The second is how can users customize translation. The last is how to help users repair the mistranslations. Following the user-centered design of interactive translation systems, we present three contributions toward the above challenges.

1. Selecting the best machine translation for users
   We propose a two-phase evaluation process for selecting the best translation result from multiple machine translation services. The first phase selects one of a number of automatic machine translation evaluation methods, and the second phase uses the selected evaluation method to identify the best translation result. In preparation for machine translation evaluation method selection, the supervised learning approach is used to learn evaluation method selection rules by using

human evaluation results from experts as a supervisory signal. In the first phase, the machine translation evaluation method that best suits the user's input message is selected by using the learned rules. In the second phase, the selected evaluation method is used to evaluate translation results of the user's input messages from multiple machine translation services for selecting the best translation. An experiment on a test set for machine translation evaluation shows that even though the proposed method currently has very simple evaluation method selection rules, it can achieve an improvement from 3.8 to 4.2 (5-point scale of adequacy) compared to using just one evaluation method.

2. Allowing users to flexibly customize translation
   We present a customization method for translating messages across multiple topics The target is to enable the user to flexibly compose the language services of domain resources (dictionaries and parallel texts) with machine translation services so that different domain resources can be selected for different topics. A declarative language is designed for users to incrementally add domain resources into composite services for each topic, and its execution environment is developed by allowing the dynamic identification of a topic by keyword-based topic detection, the generation of all possible composite services by using logic programming, and the selection and execution of the best composite service for translation. A case study of foreign students' communication on multiple topics, such as learning life and graduation procedure, is provided. Following the description of customization, a significant increase in human judgment accuracy is verified.
3. Interacting with users to suggest the repairs of mistranslation
   We propose a translation agent that interacts with users for improving translation quality. The translation agent is designed to detect the mistranslations output by machine translation services. This design enables the translation agent to prevent the transfer of mistranslations and to suggest message alteration for improving translation quality. Thus, the translation agent can reduce the number of user messages needed to address the mistranslation. Through a multilin-

gual communication experiment in which users collaborate on tangram arrangement, this chapter shows that translation agent mediated communication allows users to achieve consensus-building by exchanging 22% fewer messages than the traditional machine translation mediated communication.

In brief, the user-centered design proposal is useful in selecting the best machine translation service for each user's input message, to flexibly apply various language services for customizing translation, and to interact with users for improving translation quality, so as to improve translation quality for multilingual communication.

Acknowledgments

My success in graduation would not have been possible if not for the support of so many people. Time flies like a fleeting amber. Amicabilities are as countless as the stars. You will understand if I forget someone ...

First and foremost, I would like to express my deepest gratitude to my supervisor, Professor Toru Ishida, for his enduring patience, supervision, advice, and guidance throughout the last three years, for pushing me in the right direction, for making me grope my way toward the door for research, and for the chance to live in the amazingly beautiful Kyoto. I will never forget those sincere words on research, communication, and conduct. I am glad you are my professor. I have two words: endless gratitude!

I gratefully acknowledge the members of my adviser committee, Professor Katsumi Tanaka, Professor Sadao Kurohashi, Professor Akihiro Yamamoto, and Professor Qiang Ma, for your advice, supervision, and crucial contribution to my research. I would also like to thank Professor Zhongzhi Shi, for his continuous care and attention of my research progress.

I am very grateful to Assistant Professor Donghui Lin and Yohei Murakami for their practical advice and support. They has paid so much on giving me advice and encouragement, since the first month I was a PhD student. I remember and appreciate their warm guidance all the time.

I would like to thank all faculty members of Ishida & Mastubara Lab: Associate Professor Shigeo Matsubara, Associate Professor David Kinny, Assistant Professor Hiromitsu Hattori, Yohei Murakami, Yuu Nakajima, Masayuki Otani, Rieko Inaba. Prof. Matsubara, Kindy, and Hattori have been giving me valuable suggestions. It calls to mind that, Murakami taught

me how to be a PhD student. Nakajima taught me how to make a diagram.

And all the coordinators, Yoko Kubota, Terumi Kosugi, Yoko Iwama, Hiroko Yamaguchi. Ms. Kubota has been given support to my study since ten months before I came to Kyoto. Each time, Ms. Kubota and Ms. Kosugi kindly prepared me the conference trip.

And all students, many alumni, and project members, Masahiro Tanaka, Takao Nakaguchi, Arif Bramantoro, Bourdon Julien, Huan Jiang, Ari Hautasaari, Xun Cao, Xin Zhou, Andrew W. Vargo, Mairidan Wushouer, Amit Pariyar, Trang Mai Xuan, Kemas Muslim Lhaksmana, Shinsuke Goto, Hiroaki Kingetsu, Hiromichi Cho, Kaori Kita, Daisuke Kitagawa, Yosuke Saito, Takuya Nishimura, Ann Lee, Shunsuke Jumi, Meile Wang, Jie Zhou, Noriyuku Ishida, Jun Matsuno, Nadia Bouz-asal, Wenya Wu and many others. Thank you for studying together, hanami together, lunch together, all the suggestions and comments on research meetings.

In preparing my papers, I would like to thank Hui Hao, Donghui Lin, Andrew (Andy), Ann, Xun, Wenya, Amit, Kun Huang, Yun Gu, and Blackburn for English proof reading. I also would like to thank Amit, Mairidan (Mardan), Kemas, Trang, and Xun for the comments, Goto, Cho and Higuchi for Japanese translation of the abstract and the title.

I also want to thank my friends and family for their recreational and emotional support: my grandparents Fubao Shi and Daiya Zhang, parents Jiayou Shi and Jianfang Zhang, my uncle Jiaqing Shi, my sister Chunliu Shi and brother-in-law Bin Wang, for their love and support over the years; my friends, Xu Zhang, Yinli Zhang, Shuqing Han, Jianjian Li, Qinghua Su, Ye Zhou, Yulei Ding, Kun Huang, Jianjian Gao, Haitao Mi and YongXin Cai, Yingdong Cai, GuoJun Dai, Haiqing Zheng, Yuanfeng Li, Shan Rong, and many others, who have made me laugh through good times and bad times.

Arigato Gozaimasu! Thank you! Xièxie!

My stay in Kyoto University was supported by the Japanese Government (Monbukagakusho) Scholarships (2010.10-2013.9). This research was partially supported by Grant-in-Aid for Scientific Research (A) (18200009, 2006-2008) from Japan Society for the Promotion of Science (JSPS).

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Multilingual communication connects people from different nations, encourages business, and brings transnational cooperation. Given the success of famous companies, such as Facebook and Amazon, the need for multilingual communication is obvious. Multilingual communication supporting tools continue to receive more attention [Inaba, 2007]. Machine translation (MT) plays an important role in the implementation of such tools. For example, machine translation has been integrated in a communication support for multilingual participatory gaming [Tsunoda and Hishiyama, 2010]. Machine translation is promising as a medium for multilingual communication. Multilingual communication among different nations and cultures is really important to the international business, remote education, medical assistance, etc. The progress on natural language processing has given birth to the machine translation.

The success of machine translation brings the promising *machine translation mediated communication*, which makes multilingual communication highly available even among monolingual people. This is novel and important to both the large number of monolingual speakers and the foreign language learners. For the monolingual speakers, it is low-cost but highly available solution to communicate with foreigners. For example, 62% of Eng-

land people cannot speak any foreign language[1], and 99% of Chinese people cannot speak English[2]. For the foreign language learner, MT-mediated communication can lower learner's anxiety, and show no significant difference in reduction of communication apprehension [Arnold, 2007]. The translation system for MT-mediated communication built upon machine translation is really meaningful.

However, machine translation has limits in terms of translation quality [Wilks, 2009]. The translation errors continue to be the barrier for MT-mediated communication. When MT-mediated communication is used for a cooperation task, it is necessary to translate the task-oriented dialog accurately. Generally speaking, a communication dialog can be tagged as task-oriented, emotion-oriented, or both [Lemerise and Arsenio, 2000]. According to social information process theory, emotion-oriented dialog involves not only the cognitive process but also the emotion transfer process. Task-oriented dialog mainly focuses on the acquisition of information in the task domain [Bangalore et al., 2006]. In machine translation of task-oriented dialogs, the accurate translation of concepts is the basis of successful information transfer [Yamashita and Ishida, 2006a]. Considering the limits of high quality translation, it is hard to deal with machine translation errors in MT-mediated communication, even without considering the complex individual emotion-related factors, such as cultural background [Kim, 2002].

1.1 Objectives

In view of the fact that machine translation errors cannot be ignored, the shift from the transparent-channel metaphor to the human-interpreter metaphor (agent metaphor) was originally introduced by [Ishida, 2006a]. Interactivity is suggested as a new goal of the machine translator. Interactivity is the machine initiated interaction among the communication participants; it represents the ability to take positive actions to improve grounding

[1] http://en.wikipedia.org/wiki/Languages_of_the_United_Kingdom

[2] http://en.wikipedia.org/wiki/List_of_countries_by_English-speaking_population

and to negotiate meaning [Ishida, 2006a, Ishida, 2010]. Interactivity makes it clear that translation errors are to be treated as channel noise. This noise can be suppressed through the efforts of the multilingual communication participants.

The objectives of thesis is to design user-centered translation systems for multilingual communication. Users are facing novel and complex translation environment: increasing number of language services, limitations of high quality machine translation, and limitations of users to handle low translation quality. The user-centered design will analyze users' needs and limitations, and provide design machine-aided solutions. There are two motivations for our machine-aided solutions:

1. Help users to make better use of language services to gain better translation. There are two focus from users' perspective. The first focus is *to select the best machine translation for users*. Users need the best machine translation to deploy more machine translators, when one single machine translator cannot guarantee the translation quality. The second focus is *to allow easy integration of domain resources*. For example, communication user might improve the machine translation by retrieving dictionary result of confusing word, or searching parallel text for phrases or sentences. Language Grid allows wrapping language resources into language services. Assuming users can integrate those language services through service computing techniques, such as service selection and service composition, the translation translation will be promoted.
2. Help users to adapt to machine translation to gain better communication efficiency. We focus on how *to motivate users to adapt to machine translator*. The interactivity between communication users is need to make sure each other understands the translated message. Meanwhile, the interactivity between the translation system and users can prevent transferring mistranslation from sender to receivers. To realize the interactions to repair miscommunication, we have to help users to adapt to machine translation.

1.2 Issues and Approaches

We would like to apply *user-centered design* of translation systems to support multilingual communication. From the perspective of users, we postulate the basic mechanism for our hypothesis. Next, the design and implementation are based on the hypothesis. Then, we primarily evaluate the mechanism and propose refinement suggestion. According to the three issues in fulfilling two mentioned objectives, we listed three approaches for our user-centered design of translation systems (see Figure 1.1).

1. Two-phase evaluation to select the best machine translation. There are many machine translation services and more than one evaluation methods available. It is difficult for users to pick up a machine translation, because the variable translation quality of different source messages. Meanwhile, the existing evaluation methods show different inconsistent selection of translations. We design a two-phase evaluation by selecting an evaluation to evaluate multiple machine translations. The architecture contains two phases. In the first phase, data-driven mechanism, a decision tree, is used to select the best evaluation methods according to the features of input source message. In the second phase, according to the selected evaluation methods is used to select the best translation results. Thus, we select the best translation from multiple machine translators using several evaluation methods.
2. Scenario description to allow easy integration of domain resources. Machine translation mediated communication cannot guarantee high accuracy. If available domain resources could be integrated, the accuracy could be promoted. From user's perspective, they can develop their own domain resources, but it is difficult to integrate those resources, such as self-prepared dictionary or parallel text. Traditional domain adaptation needs techniques to train resources, which is too complex to non-computing people. We propose scenario description as a light-weight tool to integrate domain resources. The Language Grid well wraps language resources as language services. The scenario description allows users to mapping resources to the communi-

cation topics. After that, composition of those language services with translations will integrate the resources for better accuracy.

3. Interactivity solution to motivate users to adapt to machine translator. In machine translation mediated communication, translation errors can easily lead to communication break down or miscommunication. Interactivity can promote the communication efficiency by motivating users to adapt to machine translator. We propose interactivity solution, agent metaphor, to implement interactivity between translation system and users to repair translation errors.

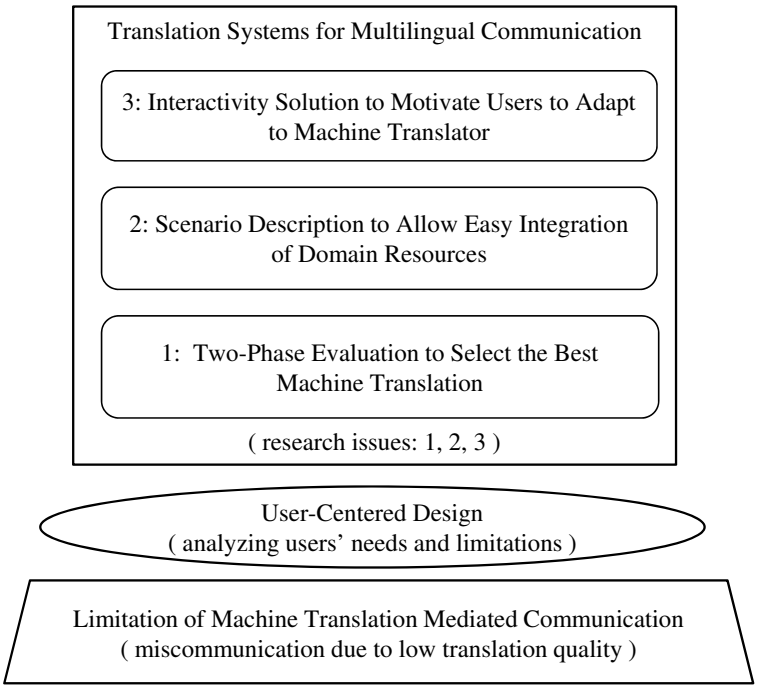

Figure 1.1: Issues in user-centered design of translation systems to support multilingual communication

1.3 Thesis Overview

The goal of this thesis is to design an interactive translation system to support multilingual communication using the user-centered design approach; it details how to select the best machine translation for the user's input message, customize translation for different communication topics, and interact with users to improve translation quality for multilingual communication. This thesis consists of six chapters.

Chapter 1 outlines the thesis, including the research objective, approaches and issues.

Chapter 2 describes the background of this thesis. This chapter studies the previous work on machine translation mediated communication, shows the communication problems caused by machine translation, and clarifies the requirements important in designing interactive translation systems.

Chapter 3 proposes a two-phase evaluation process for selecting the best translation result from multiple machine translation services. The first phase selects one of a number of automatic machine translation evaluation methods, and the second phase uses the selected evaluation method to identify the best translation result. In preparation for machine translation evaluation method selection, the supervised learning approach is used to learn evaluation method selection rules by using human evaluation results from experts as a supervisory signal. In the first phase, the machine translation evaluation method that best suits the user's input message is selected by using the learned rules. In the second phase, the selected evaluation method is used to evaluate translation results of the user's input messages from multiple machine translation services for selecting the best translation. An experiment on a test set for machine translation evaluation shows that even though the proposed method currently has very simple evaluation method selection rules, it can achieve an improvement from 3.8 to 4.2 (5-point scale of adequacy) compared to using just one evaluation method.

Chapter 4 presents a customization method for translating messages across multiple topics. The target is to enable the user to flexibly compose the language services of domain resources (dictionaries and parallel texts)

with machine translation services so that different domain resources can be selected for different topics. A declarative language is designed for users to incrementally add domain resources into composite services for each topic, and its execution environment is developed by allowing the dynamic identification of a topic by keyword-based topic detection, the generation of all possible composite services by using logic programming, and the selection and execution of the best composite service for translation. A case study of foreign students' communication on multiple topics, such as learning life and graduation procedure, is provided. Following the description of customization, a significant increase in human judgment accuracy is verified.

Chapter 5 proposes a translation agent that interacts with users for improving translation quality. The translation agent is designed to detect the mistranslations output by machine translation services, with evaluation support from Chapter 3 and service deployment support from Chapter 4. This design enables the translation agent to prevent the transfer of mistranslations and to suggest message alteration for improving translation quality. Thus, the translation agent can reduce the number of user messages needed to address the mistranslation. Through a multilingual communication experiment in which users collaborate on tangram arrangement, this chapter shows that translation agent mediated communication allows users to achieve consensus-building by exchanging 22% fewer messages than the traditional machine translation mediated communication.

Chapter 6 summarizes the original contributions and future directions. The user-centered design proposal is useful in selecting the best machine translation service for each user's input message, to flexibly apply various language services for customizing translation, and to interact with users for improving translation quality, so as to improve translation quality for multilingual communication.

Chapter 2

Background

2.1 Translation Systems To Support Multilingual Communication

2.1.1 Machine Translation Mediated Communication

Language barrier prevents people from different nations and culture to communicate with each other. To encourages business, and brings transnational cooperation, people have to overcome the language barrier. For non-foreign language learning people, they need translation support. For example, famous transnational companies, such as Facebook and Amazon will be greater success, if their translation support is efficient. Thus, efficient support tools continue to receive more attention [Inaba, 2007]. Without proper support in the multilingual environment, the language barrier will make non-foreign language learning people remain on the surface, unaware of the strangeness and complexity of life beneath the waves [Swift, 1991]. Machine translation plays an important and promising role in the preparation of such tools. For example, machine translation has been integrated in a communication support agent developed for multilingual participatory gaming [Tsunoda and Hishiyama, 2010]. However, machine translation has limits in terms of translation quality [Wilks, 2009]. The translation errors

continue to be the barrier for machine translation mediated (MT-mediated) communication. When MT-mediated communication is used for a cooperation task, it is necessary to translate the task-oriented dialog accurately. Generally speaking, a communication dialog can be tagged as task-oriented, emotion-oriented, or both [Lemerise and Arsenio, 2000]. According to social information process theory, emotion-oriented dialog involves not only the cognitive process but also the emotion transfer process. Task-oriented dialog mainly focuses on the acquisition of information in the task domain [Bangalore et al., 2006]. In machine translation of task-oriented dialogs, the accurate translation of concepts is the basis of successful information transfer [Yamashita and Ishida, 2006a]. Considering the limits of translation quality, it is hard to deal with machine translation errors in MT-mediated communication, even without considering the complex individual emotion-related factors, such as cultural background [Kim, 2002]. Thus, in traditional way of using machine translators, they are just transparent-channel to non-foreign language learners. The multilingual communication will be broken due to the translation errors.

2.1.2 Limitations of Machine Translation Systems

Machine translation has limitation to guarantee high quality translation all the time [Wilks, 2009]. Translation environment involves both translation function and user. From the perspective of translation function, the analysis of machine translation errors is very important for the development of machine translation [David Vilar, 2006, Popović and Ney, 2011]. We focus on the limitations on applying machine translation system by users. Low quality translation leads to translation errors to users. We examined existing works on translation errors from the user perspective.

Users will face low quality translation, which is the main limitation of deploy machine translation system. In MT-mediated communication, translation errors lead to miscommunication. Analyzing miscommunication at the phrase, sentence, and dialog level is popular in machine-mediated communication research [Kiesler et al., 1985, Yamashita and Ishida, 2006a].

These three observations of machine translation errors are picked up according to these levels: *phrase-level*, *sentence-level*, and *dialog-level*.

- Phrase level works, extract and highlight inaccurate words [Miyabe et al., 2008], picture icons as precise translation of basic concepts [Song et al., 2011].
- Sentence level works, examine back-translation for sentence level accuracy check [Miyabe and Yoshino, 2009], Round-trip monolingual collaborative translation of sentence [Hu, 2009, Morita and Ishida, 2009a].
- Dialog level works, examine asymmetries in machine translations [Yamashita and Ishida, 2006b], Predict misconception due to unrecognized translation errors [Yamashita and Ishida, 2006a].

Users are not helped enough to handle translation errors, which is due to the limitation of interactivity between translation system and users. The analysis of translation errors is specific in whether user can manually correct translation output or not. These show several existing works on examining mistranslation problems, providing suggestions and strategies for reducing errors at each level. For example, in phrase level, highlighting inaccurate words will facilitate user modification. In sentence level, round trip translation will provide certain information of translation result. In dialog level, prediction of potential translation inconsistency prevents user using an improper shorten reference of the previous concept. However, such user adaptation only help user to deal with parts of particular translation errors. To help users to handle different translation errors, the interactivity of machine translation system is very important.

2.1.3 Interactivity of Machine Translation Systems

The interactivity was referred in studying the relationship between messages in human to human communication and then human to machine communication [Rafaeli, 1988]. The goal was to understand the influence of how to responses to a message. Meanwhile, the computer mediated communication was first modeled as information transfer, succeeded from Shannon and Weaver's model of signal transmission in telecommunication systems.

However, this information transfer does not count in the users, without any linguistic or social phenomenon. After the birth of the conversational model of computer mediated communication, the importance of interaction and conversation in communication were stressed [Riva and Galimberti, 1998].

As a special computer mediated communication, machine translation mediated (MT-mediated) communication cannot ignore the linguistic and social nature of the users. For example, the level of user's foreign language skill will affect this multilingual communication. The emergence of MT-mediated communication brings promise to the multilingual communication among non-foreign language learning people and puts all the emphasis on the machine translation function. Of course, the accuracy promotion in machine translation function is really important. Due to the limits of current research on machine translation [Wilks, 2009], machine translation itself largely needs human participation to guarantee high accuracy, from general machine translation [Toma, 1977, Berger et al., 1994], to domain adaptation of machine translation [Bertoldi and Federico, 2009, Wu et al., 2008, Koehn and Schroeder, 2007, Sankaran et al., 2012], to human-assisted machine translation, to computer assisted human translation, and to human translation. The availability decreases as the human participation increases. Meanwhile, the availability of translation increases as the human participation decreases. Especially, using Web services technique, for example through Language Grid [Ishida, 2011], the usability of general machine translations has been greatly promoted. Due to the expense and availability of human resource, we cannot count on other bilingual experts, but we need to rely on the participants of the multilingual communication.

Following the paradigm shifting from transparent-channel metaphor to human-interpreter metaphor [Ishida, 2006b], we will not assume that the accuracy of the machine translation is perfect and we turn to interactivity motivation. The transparent-channel metaphor putting weight on accuracy in MT-mediated communication, ignores the users just like the information transfer model in computer mediated communication. However, the interactivity motivation in MT-mediated communication has not been studied as much as the conversational model in computer mediate communication.

Thus, given the assumption of quality limitations of machine translators, we will turn from the accuracy promotion to the interactivity motivation, so as to analyze the interactivity model of machine translation mediated communication, and to design agent metaphor to motivate the interactions, which reduces miscommunication.

2.2 Design of Machine Translation Systems

2.2.1 Availability

Increasing number of translation systems, either online services or softwares, are developed. It includes both machine translation and human translation. Translation quality and availability of translation function play a key role in translation environment (see Figure 2.1). For example, in certain resource-limited languages, it is often that the machine translation works not as well as popular languages. Thus, human translation is used for high quality translation. The common machine translation includes *rule-based* machine translation (RBMT) [Toma, 1977], *statistical* machine translation (SMT) [Berger et al., 1994], *example-based* machine translation (EBMT) [Nagao, 1984], *knowledge-based* machine translation (KBMT) [Nirenburg et al., 1991], and *hybrid* of them, domain adaptation of machine translation [Bertoldi and Federico, 2009, Wu et al., 2008, Koehn and Schroeder, 2007, Sankaran et al., 2012].

More and more machine translation software and resources are wrapped into services. Originally, some owners allow access of their machine translation through networks. But more owners hold their own usage because of technique, policy, or other issues. Meanwhile, those available online services do not share standard interface, which makes it difficult to automatically invoke those MT services. The creation of Language Grid platform helps a lot to promote the availability of MT services in Web services description language (WSDL) standard [Ishida, 2011]. It solves the control and policy for the interests of providers. It wraps ex-

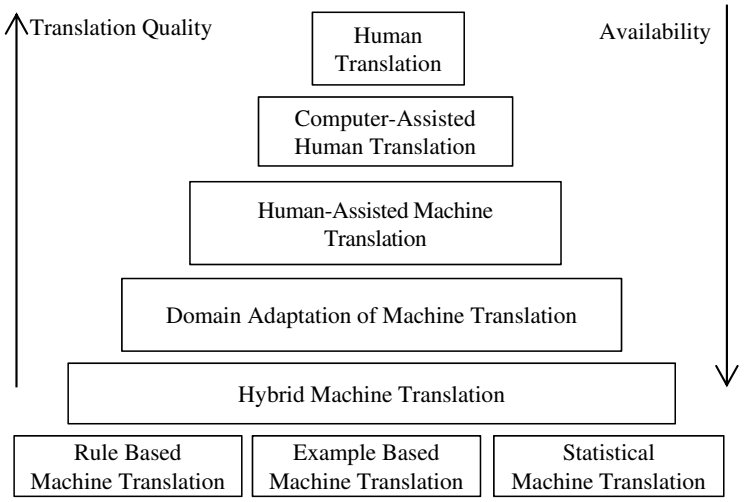

Figure 2.1: Pyramid view of translation functions

isting language related software or resources into services, with standard interfaces. The service architecture becomes open source[1], and different service nodes form federation and share the resources, for example, the Linguagrid of CELI Research [Bosca et al., 2012]. There are other service architectures of providing machine translations as services, such as ScaleMT [Sánchez-Cartagena and Pérez-Ortiz, 2010]. The number of available MT services also increases through language service composition. For one thing, by combining two MT services, a type of composite MT service, which is based on intermediate language, can be reached. For another, by combining dictionary service, morphological analyzer service, and MT service, another type of composite service, can be implemented. Language Grid provides machine translation services, dictionary services, morphological analyzer services, etc. A multi-hop MT service, has been developed as such composite service.

[1] http://langrid.org

2.2.2 Translation Functions

Levin et al. created an interlingua based on the domain act in travel planning domain [Levin et al., 1998]. The interlingua was composed by speaker tag, speech acts, concepts and arguments. The machine translation system was two-step interchange format mapping, from *source concept parse trees* to *target concepts gen trees*. Four sub-domains of travel planning, hotel reservation, transportation, sight seeing, and events, were focused in this task-oriented machine translation system. The system was developed based on the 423 domain actions that cover hotel reservation and transportation. The experiment was mentioned that comparing robustness with other domain acts based on systems including a statistic method and glossary-based approach. Obviously, this interlingua-based system needs large manual work on preparing interchange format mapping rule. The benefit is that the translation will be done by mapping rules, which is robust and consistent.

Bangalore et al. created a finite-state model for task-oriented machine translation [Bangalore and Riccardi, 2000]. The process of machine translation was treated as encoding and decoding process, with integration of constraints from various levels of language processing. The stochastic finite-state machine translation was trained automatically from pairs of source and target dialog utterances. The constraints were decomposed into two levels: *local (phrase-level)* and *global (sentence-level)*. After on-line learning of variable N-gram translation model, this process of phrase-based N-gram statistic machine processed the reordering through variable N-gram stochastic automation. This model has been tested on the Japanese-English translation of call routing task.

Josyula et al. proposed an agent, ALFRED (Active Logic for Reason Enhanced Dialog), for task-oriented dialog translation [Josyula et al., 2003]. It provided the capability design, including *understanding the use-mention distinction*, *using meta-dialog*, *learning new words*, *maintaining context*, and *identify miscommunication*. It also provided an example for explanation of new word and inconsistent reference of new word.

Composition of machine translation services

Bramantoro et al. combine Heart of Gold and Language Grid technology to provide more language resources available on Web [Bramantoro et al., 2008]. Heart of Gold is known as middleware architecture for integrating NLP functions, while Language Grid is an infrastructure of the distributed language services. Having Heart of Gold available as Web services in the Language Grid environment would contribute to interoperability among language services. The interface of Heart of Gold is extended, so that XML string and XPath can handle the result.

Lin et al. and Lewis et al. discussed the combination of human and machine translators for localization process. During this process, different users, monolingual and bilingual translator, are employed. Lewis et al. proposed BPEL4People extension for better support for human translator [Lewis et al., 2009]. The requirement on different QoS properties are calculated, including the translation accuracy, time, and cost [Lin et al., 2010].

Eduardo et al. proposed a composition algorithm of automatic service composition [Eduardo et al., 2007]. First, it assumes that every available service is semantically annotated. Second, a user/developer service request a matching service is composed in terms of component services. Third, the composition follows a semantic graph-based approach, on which atomic services are iteratively composed based on services' functional and non-functional properties. It was implemented based on the architecture of SPICE, which has four main components: including natural language processing, matcher, composition factory, and property aggregator. An example of composition of sendSMS and Translation services is given.

Cooperation of multiple translation agents

Tanaka et al. proposed a coordinator agent for composition of bilingual dictionaries and machine translations. It is a context-based coordination to maintain the consistency of word meanings during pivot translation services [Tanaka et al., 2009].

Selection among multiple translation results

Goto et al. proposed to select a useful service for a specific user and task by using reputation information of other users [Goto et al., 2011]. When direct evaluation of the service quality is much too costly, the reputation information from other users might be obtained at a lower cost. Moreover, the reputation is defined as a judgment of useful or useless according to the triplet (service, user, task). Akiba et al. and Shi et al. proposed to select among the translation results from multiple translation services. To select a translation results, it needs scores of translation quality assessment of each results [Akiba et al., 2002, Shi et al., 2012c]. Akiba et al. calculate the score using the probability of the original language model, and improves the score by highlighting the much better quality translation and suppressing the much lower quality translation. Shi et al. proposed selection of best translation using back translation and multiple evaluation methods with relative score calculated.

Combination of multiple translation results

Algorithms have been proposed algorithms to combine, compute consensus, and improve accuracy based on multiple peer translation results [Macherey and Inc, 2007, veikko I. Rosti et al., 2007, Matusov et al., 2006, Karakos et al., 2008]. First, candidate translation sentences are parsed into words. Second, the mapping word translation are aligned. Third, the better translation are combined either through selection according to automatic evaluation results [Macherey and Inc, 2007, veikko I. Rosti et al., 2007, Matusov et al., 2006, Karakos et al., 2008] or the probability of source-totarget and target-to-source word translation models are recalculated.

Integration of machine translation and auxiliary functions

Heyn proposed to integrate machine translation with translation memory [Heyn, 1996]. It would be simplest tool for machine assisted human translation. Matusov et al. proposed to integrate machine translation with speech

recognition [Matusov et al., 2005]. ASR word lattices was used to replace statistical translation system. So that, coupling of speech recognition and machine translation can be implemented together.

2.2.3 Translation Users

Different Types of Translation Users

Most common translation users include monolingual or bilingual people. The difference of contribution between monolingual and bilingual have been noticed [Lin et al., 2010, Resnik et al., 2010]. Lin et al. quantified the difference into the QoS properties: accuracy, time, and cost [Lin et al., 2010]. Resnik et al. constrained the translation ability of monolingual or treated as the baseline of user ability to improve translation, and examined the contribution of monolingual users in promoting the translation quality by paraphrasing [Resnik et al., 2010]. For another example, experienced/novice translation user. Narayanan et al. noticed that the user interface had two versions: one version allows no customization thus being appropriate for the novice user and the other allows for a range of options [Narayanan et al., 2006]. Somers and Jones described two scenarios, for an experienced user and for a less experienced user, because the operation of the system depends somewhat on the expertise of the user [Somers and Jones, 1992]. Meanwhile, there is a intentional model for the experienced user to input text, and a predictive model for the less experienced user. Estrella described that superior and novice provide different quality characteristics [Estrella, 2008]. The superior represents author's proficiency in source language, while the novice represents user's proficiency in source language. The superior can provide dictionary level quality, while novice can provide fidelity level quality.

User Repair of Translation Errors

Agent has been proposed for human repair of machine translation [Miyabe et al., 2008, Miyabe et al., 2009]. It extracts nouns and verbs that

exist in the input sentence and do not exist in the back-translated sentence. Such difference is helpful to support translation repair. Shahaf and Horvitz examined three translation scenarios, and repair based on the result of machine translation is a typical scenario [Shahaf and Horvitz, 2010]. Naruedomkul and Cercone suggested a architecture allows repair and iterative improvement [Naruedomkul and Cercone, 2002]. Kay proposed to adopt the kinds of solution that have proved successful in other domains, namely to develop cooperative man-machine systems [Kay, 1998]. For example, paraphrases could be a repair technique for inaccurate translated phrase. Callison-Burch et al. proposed to learn the paraphrase from bilingual corpus [Callison-Burch et al., 2006]. Resnik et al. proposed a process of paraphrase to eliminate translation errors with only monolingual knowledge of the target language [Resnik et al., 2010]. It is possible to generate alternative ways to say the same thing with only monolingual knowledge of the source language. Another example, pre(post)-editing could be a repairing technique for influent translation. Plitt and Masselot compared the productivity increase of statistical MT post-editing with traditional translation, the result show a productivity increase for each participant, with significant variance across individuals [Plitt and Masselot, 2010]. Lehmann et al. clarified the details of pre(post)-editing [Lehmann et al., 2012]. Pre-editing covers these aspects: Spelling and Grammar, Terminology, Style. Moreover, it identified seven rule of pre-editing and seven rules of post-editing. Hutchins described the pre(post)-editing as the main functions of human assisted machine translation [Hutchins, 2005].

Interface for User Repair

To facilitate user repair, a number of interfaces have been researched for translation errors, such as protocol, interface language, etc. For example, collaborative translation system has been proposed to improve translation quality over a poor translation channel by negotiation between two participants with imbalanced language skills [Hu, 2009, Hu et al., 2011]. It provided two hypotheses: (1) editing by monolingual users improves transla-

tion quality; (2) redundancy improves translation quality. Morita and Ishida proposed collaborative translation and designed a protocol for collaboration [Morita and Ishida, 2009a, Morita and Ishida, 2009b]. It analyzed two problems: misinterpretation and incomprehension of the meaning of translated sentences. The design of protocol will promote both fluency through post-editing and adequacy through back-translation. Flickinger et al. proposed a grammar-specific semantic interface to facilitate the construction and maintenance of a scalable translation engine [Flickinger et al., 2005]. The SEM-I is a theoretically grounded component of each grammar, capturing several classes of lexical regularities while also serving the crucial engineering function of supplying a reliable and complete specification of the elementary predications the grammar can realize.

Chapter 3

Two-Phase Evaluation for the Best Machine Translation

Users have to select the best machine translation for using more than one machine translators. From the perspective of monolingual users in multilingual communication, automatic selection of best machine translation is needed. This chapter proposes a two-phase evaluation process for users to use automatic evaluation method service and machine translation service for automatic selection of the best machine translation [Shi et al., 2012c].

3.1 Introduction

3.1.1 Evaluation of Translation Quality

Various machine translations provide divergent translation quality to the users. Different providers have implemented their machine translations based on different mechanisms. The main mechanisms include *rule-based* machine translation (RBMT) [Toma, 1977], *statistic* machine translation (SMT) [Berger et al., 1994], *example-based* machine translation (EBMT) [Nagao, 1984], *knowledge-based* machine translation (KBMT) [Nirenburg et al., 1991], and *hybrid* of them. For example, the oldest and

well-known Systran[1] is a typical rule-based machine translation. The Google translate[2] and Bing translator[3] use both statistic mechanism and rule-based mechanism, such as Chinese-English or Arabic-English translation using former mechanism, which requires huge amount of empirical training data, and resource-limited languages translation using the latter mechanism. Different providers have different focus and superiority on certain languages or domains. For example, Systran and J-Server[4], both are based on the rule-based mechanism, but Systran focuses on the translation between European languages, such as German and French, while J-Server focuses on the Asian languages, such as Chinese and Japanese. Also, there are many domain-specialized machine translations in domains such as medical services, airline services, technique manuals, etc. People are facing increasing numbers of machine translation systems. Thus, the problem, which machine translation is more competitive for the translation requests, makes the evaluation of translation quality extremely important. To relieve people from toilsomeness of human evaluation, the automatic evaluation methods, such as BLEU [Papineni et al., 2002], and NIST [Doddington, 2002], have been developed.

Currently, available automatic evaluation methods have limitations in correlation with human evaluation, and human evaluation is still the final standard. On the one hand, automatic evaluation of translation quality is necessary. It is tedious for human beings to assess machine translations. Current machine translation has limitations in providing high-quality translation [Wilks, 2009]. It means, sometimes, that the translation result is unreadable or meaningless, which makes people feel it uninteresting and dreary. Meanwhile, people have limited time, energy and consistency to provide manual evaluation. Compared with human evaluation, automatic evaluation methods, such as the famous BLEU [Papineni et al., 2002], NIST [Doddington, 2002], have the advan-

[1] http://www.systran.de/

[2] http://translate.google.com/

[3] http://www.microsofttranslator.com/

[4] http://www.j-server.com/

tages of faster processing, cheaper cost, and higher availability, but have the disadvantage of insufficient correlation with human evaluation. The birth of automatic evaluation method, especially the success of BLEU, transfers the manual judgment into comparison against references, which are the correct human translations. On the other hand, it is still an ongoing problem to find out the high qualified evaluation method, which has the highest correlation with human evaluation. Amigó et al. proposed that the reliability of evaluation methods are highly corpus-dependent [Amigó et al., 2011]. Pado et al. suggested that evaluation methods lack crucial robustness, and affected considerably across languages and genres [Pado et al., 2009]. Liu et al. and Cer et al. showed that for phrase-based SMT in several language pairs, the best evaluation method was picked out empirically [Liu et al., 2011, Cer et al., 2010a]. Even though multiple evaluation methods are available, none of them are outstanding enough to replace human evaluation. The correlation to the human evaluation is calculated to show its efficiency, such as Pearson's correlation coefficient, and Spearman's rank correlation coefficient [Callison-Burch et al., 2008]. The most popular human evaluation of translation quality is interpreted as adequacy and fluency. For example, five-level scales of manual assessment scores, {5:All , 4:Most , 3:Much, 2:Little 1:None} for adequacy, and {5:Flawless, 4:Good, 3:Non-native, 2:Disfluent, 1:Incomprehensible} for fluency, are used to quantify the translation quality in DARPA TIDES projects at University of Pennsylvania. As better evaluation leads to better translation quality, better automatic evaluation of translation quality is still an ongoing issue.

Involving the translation quality of machine translation and existent automatic evaluation methods, current researches are in different directions (see Figure 3.1). First, in the novel mechanism direction, *distinctive design* creates original and novel mechanism, which is different from any existent evaluation methods. For example, after BLEU [Papineni et al., 2002], other n-gram precision mechanisms evaluation methods NIST [Doddington, 2002], METEOR [Banerjee and Lavie, 2005], and ROUGE-N [Lin, 2004] have been proposed. Besides n-gram precision mechanisms, there are the edit distance mechanisms, such as WER

[Nießen et al., 2000], TER [Snover et al., 2006], and the length of the least common sub-string (LCS) mechanism, such as ROUGE-L, ROUGE-W. Besides these lexical level mechanism, syntactic level, and semantic level mechanisms have been designed [Amigó et al., 2009]. Second, *combination design* tightly combines features of well-chosen evaluation methods to reach robust assessment. For example, Paul et al. suggested taking into account of feature sets from existent evaluation methods, and making use of combined binary classifiers for classification [Paul et al., 2007]. Pado et al. suggested promoting robustness of evaluation methods, not only based on the combination of ensemble lexical evaluation methods, but also based on the combination of syntactic level, and semantic level features [Padó et al., 2009]. Amigó et al. suggested increasing the reliability of machine translation evaluation through the corroboration of heterogeneous evaluation methods [Amigó et al., 2011]. Third, *adaptive design* meets with extensive application of available evaluation methods. For example, from developers' view angle, Gimenez et al. suggested a framework for machine translation developers to locate weakness based on existent evaluation methods [Giménez and Amigó, 2006]. From human translators' view angle, Sankaran et al. showed the application of BLEU to reduce manual post-editing in machine assisted translation domains [Sankaran et al., 2012].

Our problem is that, as there are multiple evaluation methods, their efficiency is not unanimous, in consideration of different languages and domains, how to select machine translation by taking advantage of existent evaluation methods. To research on this problem, *creation design* direction will create a novel mechanism to beat all the existent evaluation methods. While, the *combination design* will generate a combination of robust assessment to supersede any of its constituents. But there have been limited breakthroughs in these two directions in recent years. We focus on the *application design* direction. Especially, we are from the perspective of the users, which is different from the previously mentioned two types: the perspective of machine translation developers [Giménez and Amigó, 2006], or the perspective of concrete domain application of machine translation [Sankaran et al., 2012]. We propose an architecture to enable the user to

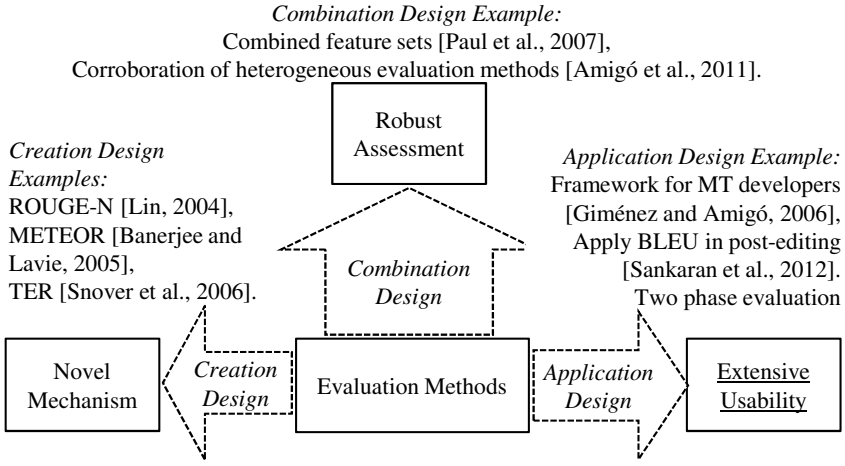

Figure 3.1: Existing evaluation methods and main research directions

select among the machine translations by taking advantage of available evaluation methods. The proposed architecture is collective adaption, *loosely* depending on the available evaluation methods, with following considerations:

- *Service availability*: the architecture makes the machine translation and evaluation methods available to the users through service-oriented platform, Language Grid. It encourages the providers to publish their machine translation as services, and attracts the users to make use of different machine translations through the same interface.
- *Improved selection*: the architecture promotes machine translation quality, by dynamically choosing proper evaluation methods for each translation request. Better evaluation methods lead to better translation quality. The user will receive higher sum of translation quality for all the translation requests.
- *Selection assessment*: the architecture offers a comparable assessment score, representing the contribution of selecting machine translations for users. Due to the different metrics of different evaluation methods, their evaluation scores are not comparable.

3.1.2 Examples of Multiple Machine Translators

As mentioned before, more and more machine translation services are usable. Similarly, increasing number of evaluation methods are available with standard interface, which greatly promoted their availability. Project Asiya, offered a rich repository of evaluation methods, [Giménez and Màrquez, 2010]. Stanford Phrasal Evaluation project provided uniform Java interface [Cer et al., 2010b]. Eck et al. [Matthias Eck and Waibel, 2006], from Carnegie Mellon University, provided online services of multiple evaluation methods. With standard interface defined, Language Grid wraps existing evaluation method software into service [Ishida, 2011].

Efficient evaluation method leads to better machine translation. Currently, the correlation to the human evaluation (manually judgment) is used for judging the efficiency of evaluation method [Zhang and Vogel, 2010]. Human evaluation is still the only high standard assessment of machine translation.

Table 3.1: Parallel text sentences in Japanese, Chinese, and English

Language	Parallel Text
Japanese (ja)	東京ディズニーランドを目一杯楽しむための攻略法を掲載しています。
English (en)	Strategies for enjoying Tokyo Disneyland to the fullest are provided.
Chinese (zh)	正在公布尽情享受东京迪斯尼乐园的攻略。

For an example, there are four translations from four different machine translations (Bing, Google, J-Server, Web-transer) for a user (see Table 3.1,3.2,3.3).

- Both machine translation systems and evaluation methods are not unanimous. For different translation request, such as from Japanese-English translation to Japanese-Chinese translation (see Table 3.1,3.2), the translation quality (adequacy by human) ranking is not the same. Meanwhile, different evaluation methods have their

Table 3.2: Translation output of multiple machine translators

(Source is the Japanese sentence in Table 3.1)

Translate	MT	Translation Results
ja→en	Bing	It includes strategies for Tokyo Disneyland to enjoy utmost.
	Google	Has posted a capture method for enjoying a glass eye to Tokyo Disneyland.
	J-Server	The capture way to enjoy Tokyo Disneyland fully is carried.
	Web-Transer	I place the capture method to enjoy Tokyo Disneyland at the full blast.
ja→zh	Bing	它包括东京迪斯尼乐园，享受最大的战略。 (It includes Tokyo Disneyland, enjoy the biggest strategy.)
	Google	已经发布了东京迪斯尼乐园享受玻璃眼的捕获方法。 (Have published Tokyo Disneyland enjoying the catching method of glass eyes.)
	J-Server	刊登了为了享受东京迪斯尼乐园眼一杯的攻占法。 (To enjoy Tokyo Disneyland, have published the occupying method of eye one cup)
	Web-Transer	正刊登攻占给最大限度享受东京迪士尼乐园的方法的。 (Have been publishing occupying the mothod of enjoying the most of Tokyo Disneyland)

own ranking too. BLEU,NIST, WER, and METEOR show different preferred MT systems (see Table 3.3).

- Proper evaluation methods lead to better translation quality. Carrying any evaluation method through the two requests, does not produce the best correlation with human judgment (see Table 3.3).

Thus, even though multiple evaluation methods are available to people, to choose a machine translation for a translation request is still a problem for people. Previously, We provide an architecture to help users take advantage

Table 3.3: Evaluation results of automatic evaluation methods are not unanimous, and human evaluation is used as standard

(Adequacy is the average of four human evaluation results)

Translate	MT	Evaluation Score				Average Adequacy (Human)
		BLEU	NIST	WER	METEROR	
ja→en	Bing	**0.3**	1.27	-0.89	0.33	2
	Google	0.19	**1.38**	-0.92	0.30	1.5
	J-Server	0.19	1.00	-1.00	**0.44**	2
	Web-Transer	0.15	1.03	**-0.85**	0.35	**3**
ja→zh	Bing	0.23	**1.33**	-1.00	**0.34**	**3.5**
	Google	0.19	1.30	-0.82	0.27	2
	J-Server	**0.27**	1.30	**-0.64**	0.29	2
	Web-Transer	0.12	0.51	-0.79	0.17	3

of multiple evaluation methods, so as to make good use of multiple machine translations.

3.2 Quality Evaluation Architecture

As mentioned before, the goal of our architecture is to qualify *service availability*, *improved selection*, and *selection assessment*. Firstly, according to the above example, four machine translations are from different providers, and of different interfaces. Bing, Google, and J-Server provide online services, but Web-transer are not provided with online-access by the providers. To select one among multiple machine translations automatically, we collect different machine translations and provide a unified interface. Secondly, the evaluation results show that if you could pick a proper evaluation methods for each of the two translation requests, it will select the machine translation of higher quality. For example, if WER can be selected for Japanese-to-English translation request and METEOR for Japanese-to-Chinese request, the translation results of J-Server and Bing, which have the highest adequacy (human evaluation), will be selected (see Table 3.3). It becomes explicit that, for each translation request, an application design is to pick out a proper evaluation method in the first place. Lastly, after selecting an eval-

uation method and the target MT system, assessment of selection is also needed to inform the users of selecting benefits. Thus, for each translation request from a MT user, there are three processes: *selecting evaluation method*, *selecting MT system*, and *assessing selection* (see Figure 3.2).

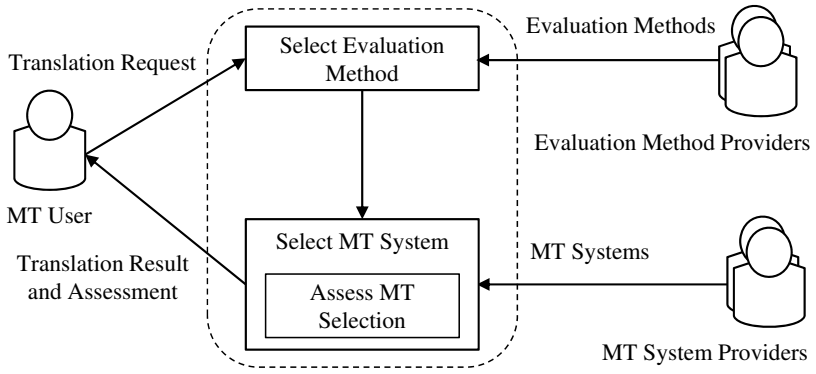

Figure 3.2: Process of machine translation selection

- *Selecting evaluation method*: multiple evaluation methods have become the candidate for evaluating the quality of MT system. However, they are not unanimous. Considering about different languages, domains, or the length of request [Och, 2003], the most proper evaluation method can be different. The problem is about how to pick out the proper evaluation method for each translation request.
- *Selecting MT system*: the candidate MT systems are prepared according to the functions, such as the proper translation languages. According to the selected evaluation method, the MT system of the highest evaluation score is to be selected as one of the highest translation quality.
- *Assessing selection of MT system*: because the selection of MT system will take time or other costs (for example, service prices), the selection efficiency of translation quality should be known to the MT user. Thus, assessing selection can inform the user of the benefits of MT selection. The problem is about how to calculate an assessment score, which is not tied up to the metric of each evaluation method.

Based on this process of MT service selection, we will design a two-phase selection architecture for MT selection. After that, we will explain the empirical way to select an evaluation method, and the novel assessment of MT selection for the users in the end.

3.2.1 Two-phase Selection Architecture

In view of an extensive application for the benefit of the users, we suggest designing the machine translation selection as a broker (see Figure 3.3), which is inspired by Web service selection [Tian et al., 2004, Serhani et al., 2005]. It will receive the request from MT user and reply the selected machine translation to the user. It has accesses to both the machine translations and evaluation methods from different providers. Three important components of Language Grid, *Service Wrapper*, *Service Registry*, and *Service Invoker*, will finish collecting and invoking the machine translation and evaluation from different providers and of different interfaces.

- *Service Wrapper*: both MT systems and evaluation methods are to be accessed as Web service, and to be separately categorized. As mentioned before, MT systems are more and more available as MT services. Meanwhile, it is easy to wrap existing evaluation methods into Web services through service grid[5] of Language Grid project.
- *Service Registry*: Language Grid has successful experience in solving various register, and management issues [Ishida, 2011]. Especially, a broker itself can be a translation service, which can be published through the service registry.
- *Service Invoker*: Language Grid provides a client as service invoker. After employing it in this broker, it is easy to invoke those services through categorized service identities.

To extend this broker (see Figure 3.3) to our machine translation selection architecture (see Figure 3.4), one evaluation method will be picked out using data-driven strategy, then one MT result will be picked out according

[5] http://servicegrid.net

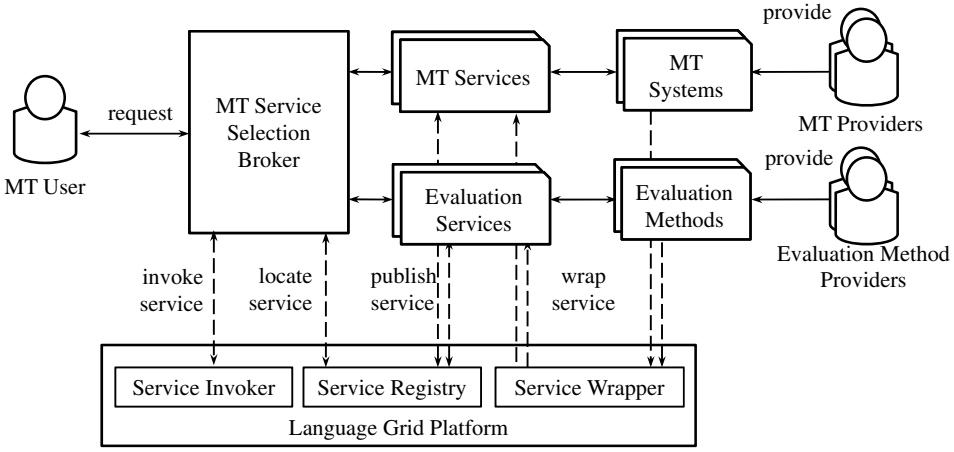

Figure 3.3: Service broker for selecting the best machine translation

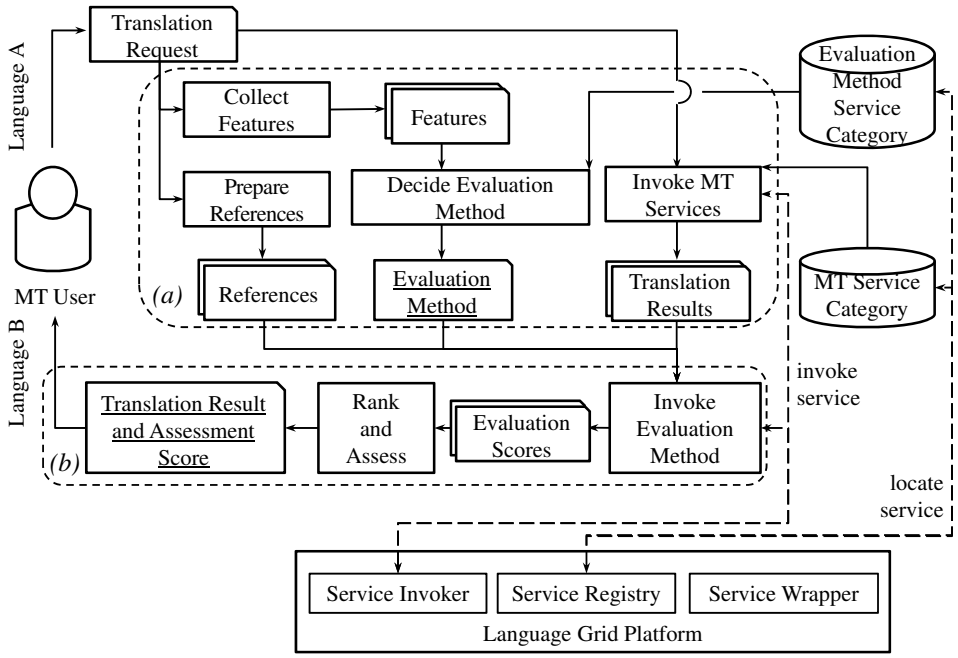

Figure 3.4: Architecture of machine translation service selection broker *(a)* selecting and preparing a proper evaluation method, *(b)* selecting a MT result and providing the assessment of selection.

to the selected evaluation method. They make up a two-phase selection, to fulfill the MT service selection process. The two phases to realize MT selection functionality are as follows.

- *Selecting and preparing a proper evaluation method*: previously, we have learned that, when in different translation language pairs, different domain corpus, or different translation length, there are researches showing that evaluation methods show different efficiency. We apply data-driven approach to build classification model by using these features, including *language pair*, *domain information*, and *length of translation request*. Because data-driven is the choice for implicit or dynamic causal relationship between these features and evaluation methods. With the trained classification model, it will empirically select an evaluation method.
- *Selecting a MT result and providing the assessment*: it becomes easy to make a choice among MT result when evaluation method is selected. But it is not easy for the user to understand the necessity of such selection. We suggest a novice assessment for the user to understand the contribution of MT selection. The process of ranking and calculation will be explained in detail later.

3.2.2 Components and Implementation

More details of the two phase architecture (see Figure 3.4), the main components in the first phase (selecting and preparing a proper evaluation method phase) include: *collect features*, *prepare references*, *decide evaluation method*, *invoke MT services*. In the second phase (selecting a MT result and providing the assessment phase), there are *invoke evaluation method*, and *rank and assess*.

To provide an applicable architecture, we will explain the components and their deployment currently. The ideal of our deployment is to make use of existing software or functions in the most. Then we will focus on the mechanism and algorithm realization. We deploy the MT systems and evaluation methods easily available into our architecture, and we will do

experiment based on this deployment.

- *Language Grid platform*: After wrapping and registering MT systems and evaluation methods into language services, the service invoker invokes either the MT system or evaluation method through a unique identity, such as Google Transalte, J-Server, Web-transer, and YakushiteNet.
- *MT service category*: We use a simple data base MySQL[6] to store the unique identity of service, service name, the URL, operation names and types, parameter names and types, and pre-setting values.
- *Evaluation method service category*: Four evaluation methods are deployed, *BLEU*, *NIST*, *METEOR*, and *WER* metrics of Stanford Phrasal project [Cer et al., 2010b]. We wrap them into WSDL services and register them to evaluation method service category.

The features and evaluation methods selection strategy are set in this phase, the deployments are listed as follows.

- *Invoking MT services*: we invoke the service based on the Language Grid client, which is a JAX-RPC service invoker[7]. It easily calls a Web service by unique service identity, operation name and type, parameter name and type, which can be indexed according to service identity in the service category.
- *Collecting features*: It analyzes the translation request and collect attribute-value pairs. In our situation, we collect three properties: translation language pairs, domain information, and length of translation request.
- *Deciding evaluation method*: The data-driven strategy, *decision tree* is applied for this purpose. First, how the translation features exactly affect the efficiency of evaluation methods is still too complex currently, for example, English-Chinese translation can be different from English-German translation, and spoken language translation can be different from written language. Data-driven approach can build classification model from input-data automatically, after that it becomes

[6]http://www.mysql.com/

[7]http://java.net/projects/jax-rpc/

convenient to make a decision. Second, we suggest decision tree learning for this purpose, because it easily transforms decision tree result into rules. Then it is convenient to test and verify a rule manually. C4.5 is a popular decision tree algorithm for classification tasks [Quinlan, 1993]. It has other merits such as handling missing values, allowing presence of noise, categorizing continuous attributes. It should be noticed that we treat C4.5 as a "black box", a tool for the task of deciding a target evaluation method based on feature of a translation request. No attempt is made to modify its function. We use J48 decision tree, a Java implementation of C4.5 algorithm from Weka data mining tool[8]. The name-value feature pairs will be its input, and its output is the identity of the evaluation method service.

- *Preparing References*: Reference preparation is one of the key issues for evaluation methods of lexical level mechanism. Reference is the wanted standard result to be compared with the result of the translation candidate. The similarity between the reference and translation candidate is calculated as the quality of this translation candidate according to the reference. We consider incorporating automatic reference preparation, thus the unsupervised process is important. Currently there are two ways of reference preparation process: the parallel text way and the round-trip translation way (see Figure 3.5).
  (a) *Parallel text way*: parallel text is the most common way to prepare reference. The parallel text service from Language Grid provides searching function. Thus, it is easy to prepare references.
  (b) *Round-trip translation way*: round-trip translation is a controversial way to be used as reference preparation. As it is widely known and tried by the users, we include it as one way for preparing references.

For this phase the most important deployment is to implement the algorithm for ranking and assessment, which will be explained in next section.

- *Invoking evaluation method*: similar to the component "*invoke MT*

[8]http://weka.sourceforge.net/

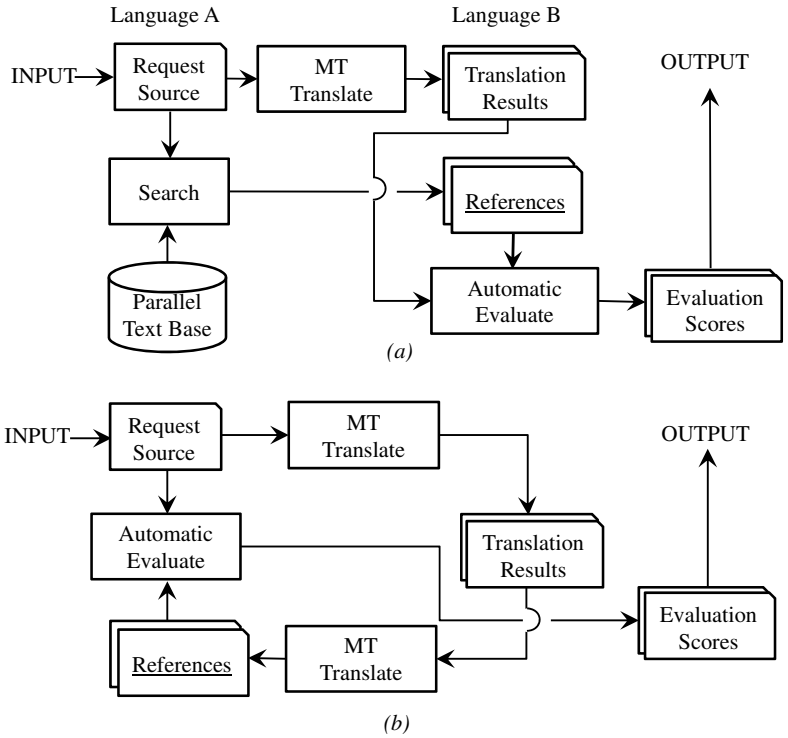

Figure 3.5: Two ways to prepare references
(a) parallel text for reference preparation, (b) round trip translation way for reference preparation.

services", it will prepare parameters and call Language Grid client to invoke the wrapped evaluation method service.

- *Ranking and assessing*: we implement the ranking and assessment algorithm in Java. The input is the evaluation scores after evaluating the translation results. After ranking, the selected translation result and assessment score are returned to the user.

3.3 Quality Evaluation Process

After the description of the architecture and component deployment, we will explain the selection and assessment in detail. Finally, the mapping

algorithm will be provided and an example will be taken for explanation.

3.3.1 Definitions and Process Description

Let S denote the machine translation systems developed by different MT system developers. Similarly, let E denote the evaluation methods developed by different evaluation method developers. Then, the available MT and evaluation methods will be represented as follows:

- $S = \{s_1, s_2, \ldots, s_n\}$: n candidate machine translations available.
- $E = \{e_1, e_2, \ldots, e_m\}$: m candidate evaluation methods available.

According to our selection process (see Figure 3.2), let denote:

- $R = \{r(1), r(2), \ldots, r(p)\}$: p times translation requests from one user.
- $trans(s_i, r(t))$: the tth request $r(t)$ is translated by MT service s_i.

For each translation request source $r(t)$, a proper evaluation method $e(t)$ is to be selected . As m evaluation methods and n MT services are available, for each request $r(t)$, there are:

- $TR(t) = \{tr_1(t), tr_2(t), \ldots, tr_n(t)\}$:$n$ translation results are generated, and each translation result is $tr_i(k) = trans(s_i, r(k))$.
- $eval(e_j, tr_i(t))$: the tth translation result by s_i is evaluated by method e_j.
- $V(t)$: all evaluation scores (see Equation (3.1)). If each translation result $tr_i(t)$ is evaluated by each evaluation method e_j automatically, each evaluation score will be $v_{ij}(t) = eval(e_j, tr_i(t))$.

$$V(t) = \begin{Bmatrix} v_{11}(t) & v_{12}(t) & \cdots & v_{1n}(t) \\ v_{21}(t) & v_{22}(t) & \vdots & v_{2n}(t) \\ \vdots & \cdots & \ddots & \vdots \\ v_{m1}(t) & v_{m2}(t) & \cdots & v_{mn}(t) \end{Bmatrix} \tag{3.1}$$

Selecting Evaluation Method

The empirical way, a data-driven strategy to decide target evaluation method according to the features like translation languages, and translation length,

will select the evaluation method $e_k(t)$ for each request $r(t)$. There are limited research results clarifying the effect of translation context and the relationship between all the evaluation methods, different efficiency between the evaluation methods has been showed empirically most of the time. Och showed their different efficiencies is affected by the length of input [Och, 2003]. Callison-Burch et al. showed different language pairs affect the evaluation efficient empirically [Callison-Burch et al., 2008], and Amigó et al. showed similar situation [Amigó et al., 2011]. Thus, we want to empirically check such features in selecting evaluation methods using data-driven strategy.

For the process to select evaluation method, let denote

- F=$\{f_1, f_2, \ldots, f_c\}$:the c feature collectors. For each request $r(t)$, its features are collected as $F(r(t)) = \{f_1(r(t)), f_2(r(t)), \ldots, f_c(r(t))\}$.

To make a decision about the selected evaluation method $e^*(t)$, a decision rule should be created as follows.

$$\begin{aligned}(\theta_1^{low} < f_1(r(t)) < \theta_1^{up}) \wedge \ldots \wedge (\theta_t^{low} < f_t(r(t)) < \theta_t^{up}) \\ \wedge \ldots \wedge (\theta_c^{low} < f_c(r(t)) < \theta_c^{up}) \rightarrow e^*(t)\end{aligned} \tag{3.2}$$

Data-driven strategy, such as the decision tree classification, is effective for such purpose, after a training. Thus, the first process to select an evaluation method, which is further divided into several parts: collecting the features, training a classifier, and testing the decision rules. Then the target evaluation method $e^*(t)$ will be decided for the request $r(t)$.

Selecting Machine Translation

For the process to select MT service, assuming the selected evaluation method $e^*(t) = e_k$, then evaluation scores of $r(t)$'s translation results are $\{v_{k1}(t), v_{k2}(t), \ldots, v_{kn}(t)\}$ (see Equation 3.1). Then, the selected MT service for request $r(t)$ will be $s^*(t)$ as follows:

$$s^*(t) = \arg\max_{s_i} eval(e_k, trans(s_i, r(t))) \tag{3.3}$$

Assessing Selection of Machine Translation

For the process to assess MT selection, the contribution of this selection should be reported to the users. Here are the considerations about why we need this assessment.

- One problem is that the evaluation score $eval(e^*(t), trans(s^*(t), r(t)))$ cannot reflect whether the selection is necessary or not. Give an extreme example, if all the translation results are the same, the evaluation score will be the same, thus the selection seems not contributive, but it does not matter whether the score is high or low.
- Another problem is that multiple evaluation methods have different metrics, therefore, they cannot be directly compared.

We propose a new assessment strategy, which calculates the relative quality promotion. Then, we use the change of average score to represent the relative quality promotion. It compares the average score of two status, counting in a selected MT service and not counting in. Thus, the higher the changing ratio is, the bigger this selection contributes [Shi et al., 2012b].

First, we calculate the change ratio of the average evaluation score, counting the MT service $s^*(t)$, which is $\frac{1}{n}\sum_j eval(e^*(t), trans(s(j), r(t)))$, to not counting in, which is $\frac{1}{n-1}(\sum_j eval(e^*(t), trans(s(j), r(t))) - eval(e^*(t), trans(s^*(t), r(t))))$. Assuming $s^*(t) = s_i$, $e^*(t) = e_k$ and $avg(t) = \frac{1}{n}\sum_j v_{kj}(t)$, this ratio of a change in average score, $contri_i(t)$, representing the contribution of selecting this MT service s_i, will be calculated as follows:

$$\begin{aligned} contri_i(t) &= \frac{\frac{1}{n}\sum_j eval(e^*(t), trans(s(j), r(t)))}{\frac{1}{n-1}(\sum_j eval(e^*(t), trans(s(j), r(t))) - eval(e^*(t), trans(s^*(t), r(t))))} \\ &= \frac{\frac{1}{n}\sum_j v_{kj}(t)}{\frac{1}{n-1}(\sum_j v_{kj}(t) - v_{ki}(t))} \\ &= \frac{(n-1)avg(t)}{n \cdot avg(t) - v_{ki}(t)} \end{aligned} \tag{3.4}$$

After that, we want to normalize the $contri_i(t)$ into range $[0,1]$. We choose an easy function to do that, function $f(x) = x/(x+1)$. We can calculate the quality score $contri'_i(t)$, where $avg(t) = \frac{1}{n}\sum_j v_{kj}(t)$, as follows:

$$contri'_i(t) = \begin{cases} \dfrac{(n-1)avg(t)}{(2n-1)avg(t) - v_{ki}(t)} & \text{if } v_{kj}(t) \geq 0 \\ \dfrac{n \cdot avg(t) - v_{ki}(t)}{(2n-1)avg(t) - v_{ki}(t)} & \text{if } v_{kj}(t) < 0 \end{cases} \tag{3.5}$$

Finally, the assessment $contri'_i(t) \in [0,1]$ will be reported to the MT user.

3.3.2 Machine Translation Selection Algorithm

Algorithm and Explanation

After the strategy analysis, here we provide the algorithm in detail (see Algorithm 1). The algorithm works in the broker for MT service selection. It includes two-phase execution. In the first phase, if no decision rules exist, we need to train the decision tree, and generate decision rules. Next, we will calculate attributes $\{f_1(r), f_2(r), \ldots, f_c(r)\}$ from request translation source r by attribute collector functions, then their values are checked by decision rules. If decision rules exist, we can select a target evaluation method *selected_evaluation*, which completes the first phase.

In the second phase, it invokes the MT services S to translate current request r, and get the translation results $tr_i = trans(s_i, r)$, evaluate translation results by the selected evaluation method e^*, and get evaluation scores $v_{ki} = eval(e^*, tr_i(r))$ from evaluation results. Then it is easy to rank for the target result $tr^*(r)$. After that, the assessment of selection is calculated according to equation (3.5). Finally, MT service s^*, translation result $tr^*(r)$, and assessment $contri'$ are returned.

Algorithm 1: machine-translation-select(E,S,r,F)

Input: E=$\{e_1, e_2, \ldots, e_m\}$: the m evaluation methods;
S=$\{s_1, s_2, \ldots, s_n\}$: the n MT services;
r: current request translation source ;
F=$\{f_1, f_2, \ldots, f_c\}$: c feature collectors ;

/** phase 1: select evaluation method **/
if decision rules *not exist* **then**
    train decision tree by J48, and generate decision rules.
/* collect attribute values */
process translation source r by $\{f_1, f_2, \ldots, f_c\}$, and get $\{f_1(r), f_2(r), \ldots, f_c(r)\}$;
/* check decision rules, and select evaluation method */
$e^* \leftarrow \{e_k \| (\theta_1^{low} < f_1(r) < \theta_1^{up}) \wedge \ldots \wedge (\theta_c^{low} < f_c(r) < \theta_c^{up}) \rightarrow e_k\}$;
/** phase 2: select MT result and assess selection **/
$max \leftarrow 0$, $avg \leftarrow 0$;
/* evaluate MT results */
foreach $s_i \in S$ **do**
    translate r by execute service s_i, and get translation result;
    evaluate translation result by e^*, and get v_{ki};
    $tr_i(r) \leftarrow trans(s_i, r)$;
    $v_{ki} \leftarrow eval(e_k, tr_i(r)) \leftarrow eval(e^*, tr_i(r))$;
/* rank the best MT */
foreach $i \in \{1, 2, \ldots, n\}$ **do**
    /* select max quality score */
    if $max < v_{ki}$ **then**
        $max \leftarrow v_{ki}$, $s^* \leftarrow s_i$, $tr^*(r) \leftarrow tr_i(r)$;
    $avg \leftarrow avg + v_{ki}$;
$avg \leftarrow avg/n$;
/* assess selection */
calculate $contri'$ according to equation (3.5);
return s^*, $tr^*(r)$, $contri'$;

Example

Previously, we have four MT systems, $\{s_1$:Bing, s_2:Google, s_3:J-Server, s_4:Web-Transer$\}$, and 4 evaluation methods, $\{e_1$:BLEU, e_2:NIST , e_3:WER, e_4:METEOR$\}$. The Japanese-to-English and Japanese-to-Chinese translation are two request from the user $\{r(1)$:(ja→en), $r(2)$:(ja→zh)$\}$. We explain our ranking and assessment by a simple example. Assuming data-driven method, a decision tree, is trained. Two features, (*language pair*, *length of translation request*), are used for training. Finally, eight rules are generated. Two of them are listed as follows:

- (*language pair*==”ja→en”) $\wedge$ (0<*length of machine request* ≤ 24) $\rightarrow$ WER.
- (*language pair*==”ja→zh”) $\wedge$ (12<*length of machine request* ≤ 24) $\rightarrow$ NIST.

In the process of selecting evaluation method, the features of two requests $r(1)$ and $r(2)$, are collected and the above rules are checked.

- Features (“ja→en”, 22) leads to WER, thus for $r(1)$, WER is selected, and $e^*(1) = e_3$.
- Features (“ja→zh”, 22) leads to NIST, thus for $r(2)$, NIST is selected, and $e^*(2) = e_2$.

In the process of selecting machine translation, assuming evaluation scores are as follows:

- WER scores: $\{v_{31}(1)$:$-0.89, v_{32}(1)$:$-0.92, v_{33}(1)$:$-1.00, v_{34}(1)$:$-0.85\}$.
- NIST scores: $\{v_{21}(2)$:$1.11, v_{22}(2)$:$0.85, v_{23}(2)$:$1.13, v_{24}(2)$:$0.35\}$.

Then, for $r(1)$ and $r(2)$, $v_{34}(1)$ and $v_{21}(2)$ are the highest and selected. For translation request $r(1)$, there is no obvious translation quality difference among all results, thus the selection does not contributes too much. For translation request $r(2)$, MT service s_1 translates with obvious high score, when its result is selected, the user will receive higher quality translation than selected by random.

In the process of assessing selection of machine translation, the $avg(1) = -0.91$ and $avg(2) = 0.86$ are calculated as $avg(t) = \sum_j v_{kj}(t)/4$. For $r(1)$, $k = 3$, and for $r(2)$, $k = 2$. We calculate translate quality by Equation (3.5).

- $\{contri'_1(1)$:0.502, $contri'_2(1)$:0.499, $contri'_3(1)$:0.492, $contri'_4(1)$:0.506$\}$
- $\{contri'_1(2)$:0.525, $contri'_2(2)$:0.499, $contri'_3(2)$:0.527, $contri'_4(2)$:0.455$\}$,

From these results, for translation request $r(1)$, there are no large difference among $contri'_1(1)$, $contri'_2(1)$, and $contri'_3(1)$. But for translation request $r(2)$, $contri'_3(2)$ is obviously higher than $contri'_4(2)$. Among all these results, $contri'_3(2)$ is the highest. Thus, our calculation results expressed the logic which assesses the contribution of selection for the user, so that it will balance different metrics of multiple evaluation methods and provide new comparable assessment.

Finally, in this example, for the translation request $r(1)$, $s^*(1)$:*Web-Transer*, $tr^*(r(1))$: *Web-Transer*'s translation of $r(1)$, and $contri'(1)$:0.506 are returned. For the translation request $r(2)$, $s^*(2)$:*Bing*, $tr^*(r(2))$:*Bing*'s translation of $r(2)$, and $contri'(2)$:0.527 are returned.

3.4 Experiment and Analysis

After the emphasis of *service availability* in the architecture section, and the explanation of *selection assessment* in the algorithm section, we want to show the *improved selection* empirically. As mentioned before, we focus on the collective adaptation, loosely depending on the available evaluation methods. Thus, we compare the results of our proposed strategy to two situations, "selecting MT systems with one evaluation method", and "using one MT system without selection". In order to show the adaptive application, we change the translation languages and domains of translation requests. Then we calculate the human adequacy and correlation to human evaluation, because it is assumed that our proposal will promote the *average human adequacy* and have a higher *correlation to human evaluation* for the total translation requests, in comparison to the two situations mentioned above. Besides, two important issues about experience setting are noted as follows:

- Parallel text as translation requests: lower quality reference makes the comparison more complex to explain. To show how the user can enjoy

better result by our application design, we have to make assessment more accurately on the relationship between quality changes and application of more resources. So we need to avoid being affected by lower quality references.

- Human evaluation as final judgment standard: it is often the final standard for empirically comparison of the evaluation methods. With standard human evaluation score, we can calculate to which percent the proposed strategy correlated to human evaluation empirically.

3.4.1 Experiments Setting

Corpus for Experiments

We experiment on 3 Japanese-English corpora and 2 Japanese-Chinese corpora:

- Japanese-English parallel text corpus:
  1) NTT Communication Science Lab corpus (NTT): it is *everyday life* material, and 100 pairs are sampled from total 3 715 pairs.
  2) Medical corpus is used (Medical): it is *medical* information material, and 100 pairs are sampled from 2 001 pairs.
  3) Tanaka corpus[9](Tanaka): it mainly is *textbook* material, which are from English textbook for Japanese students, and 100 pairs are sampled from 150 127 pairs.
- Japanese-Chinese parallel text corpus:
  1) School guidance parallel text corpus[10](School): it is *school* guidance material, and 100 pairs are sampled.
  2) Disaster information parallel text corpus[10](Disaster): this is *disaster* handbook material, and 100 pairs are sampled.

Totally, there are 500 pairs to be the translation requests in the experiments.

[9]http://www.edrdg.org/wiki/index.php/Tanaka_Corpus
[10]http://langrid.org/playground/parallel-text.html

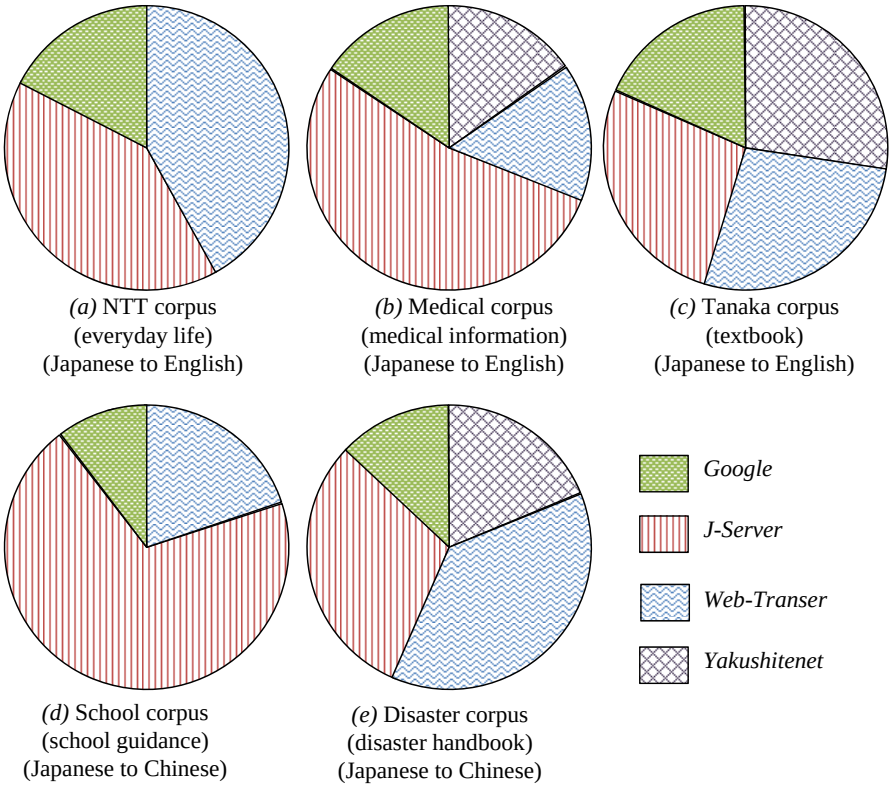

Figure 3.6: Percentage of best machine translations in each domain based on human score: adequacy

Planned Experiments

To check the quality promotion, we have planned two experiments in different adaptive applications.

- Translation requests from the same corpus (same domain): translation requests have the same language pair and domain information. Given requests $R = \{r(1), r(2), \ldots, r(p)\}$, the features, only *length of translation request* of each $r(t)$, are changing as the number of requests t increases from 1 to p.
- Translation requests from different corpora (mixed domains): it is the situation of dynamic translation requests. Given requests $R = \{r(1),$

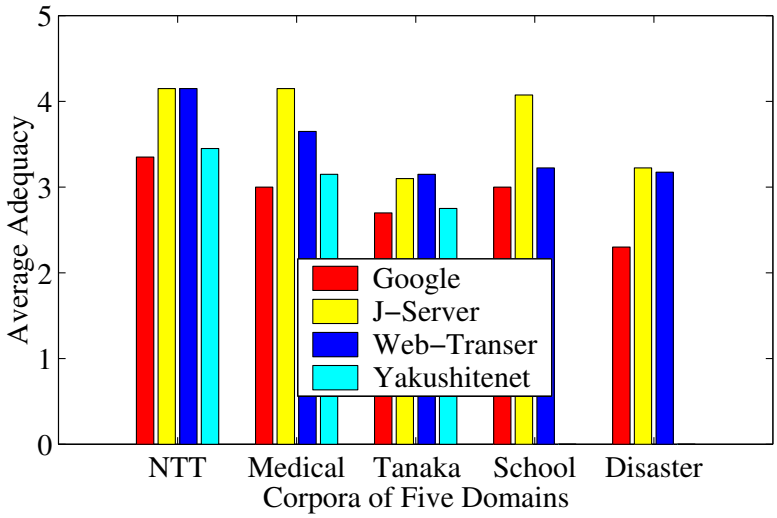

Figure 3.7: Average adequacy of each machine translation in five domains (adequacy is human score)

$r(2), \ldots, r(p)\}$, as the number of requests t increasing from 1 to p. The feature *length of translation requests*, and two other features of $r(t)$, *language pair* and *domain information*, are changing between Japanese-English pair and Japanese-Chinese pair, and from different domain-related material.

The *average human adequacy* for all the p requests are calculated in both experiments, and the *correlation to human evaluation* is also calculated in the second experiment.

3.4.2 Experiment I: Translation Requests in the Same Language Pair and Domain

For a simple situation, the user sends translation requests of the same language pair and domain. Thus, for each domain corpus, we have to train our two-phase selection, then to select MT for each request.

Machine Translation Results in Different Corpora

All the sampled 100 pairs of each domain are translated. The Japanese-English parallel texts are translated by *Google*, *J-Server*, *Web-transer*, and *YakushiteNet* services, while Japanese-Chinese parallel texts are translated by *Google*, *J-Server*, *Web-transer* services. These machine translation services are from the Language Grid platform[11]. With all the translation results, 6 people (3 for Japanese-English, 3 for Japanese-Chinese) evaluated the adequacy in a five-level scores (5:All, 4:Most, 3:Much, 2:Little, 1:None).

Then we can see the human evaluation (adequacy) of the machine translation requests. Firstly, it shows that, for one user's different translation requests, highest adequacy MT system are not always the same (see Figure 3.6). In the same domain, each machine translation gains the highest adequacy, but different machine translation shows different percentage. For example, for the request of NTT corpus, Web-transfer got the largest percentage as the highest adequacy machine translation, while *Google* gets the lowest percentage. In the different domains, the percentages are not consistent. Secondly, it shows that, for different domains, the machine translation quality can be very different (see Figure 3.7). For example, for the domain *NTT*, , the average adequacy of *Web-Transer* or *J-Server* is larger than 4 (see Figure 3.7). But, for both the domain *Tanaka* and domain *Disaster*, the highest average adequacy is lower than any machine translations from *NTT* corpus. Thus, selection of machine translation is important. Our design aims to help the user face such situation, so as to select best machine translation in a row.

Training for Two-Phase Selection

We randomly divided 100 pairs from each domain into 2 groups, with 50 pairs in each group. For the two-phase selection, one group is used for training and the other group is used for testing. We are able to generate all

[11] http://langrid.org/playground/translation.html

the results by two-phase MT selection through exchanging the two groups, which is similar to cross-validation. For the deployment, we trained the J48 decision tree for decision of evaluation methods. The train sets are selected evaluation methods correlated to human evaluation. As for each corpus, they are in the same language pair and domain, the feature used for training includes only the *length of translation request* (number of words).

Results

We compare our *two-phase selection* in two situations:

- Using one MT system without selection: the results of *Google*, *J-Server*, *Web-transer*, and *YakushiteNet*.
- Selecting MT systems with one evaluation method: the results of *BLEU selection*, *NIST selection*, *WER selection*, and *METEOR selection*.

The adequacy by human evaluation has been provided by one evaluation method selection. Firstly, within the same domain, different evaluation methods will show very different results. For example, for NTT corpus (Japanese-to-English), the results of four evaluation methods are almost the same (around 3.90). But, for Tanaka corpus, the NIST evaluation method gets the highest adequacy (3.40), while WER gets much lower adequacy (2.30). Still, mostly, each evaluation selection gains higher adequacy than using one machine translation without selection, for BLEU selection gets higher score (3.90) than any machine translation (3.65) (see Table 3.4). Secondly, the promotion by one evaluation method selection can not always be explicit. For example, in the Japanese-to-Chinese translation of School corpus, the adequacy promotion by one evaluation method selection, such as BLEU (3.60), is not obvious in comparison to the highest machine translation J-Server (3.75) (see Table 3.5). Lastly, the two-phase strategy shows the highest adequacy in each domain. For example, for Tanaka corpus requests, two-phase strategy shows higher adequacy (3.50) than NIST selection (3.40). Even though sometimes, certain evaluation method selection like WER, does not produce as high adequacy as the best machine transla-

tion like J-Server, it produces better results than the worst machine translation. Thus, after the selection of evaluation methods, we not only prevent that poor situation of certain evaluation method, but also get a chance to promote translation adequacy. But, it has to train the decision tree for each domain in the first, which costs a lot. We will test on training only once for requests from mixed domains in the next experiment.

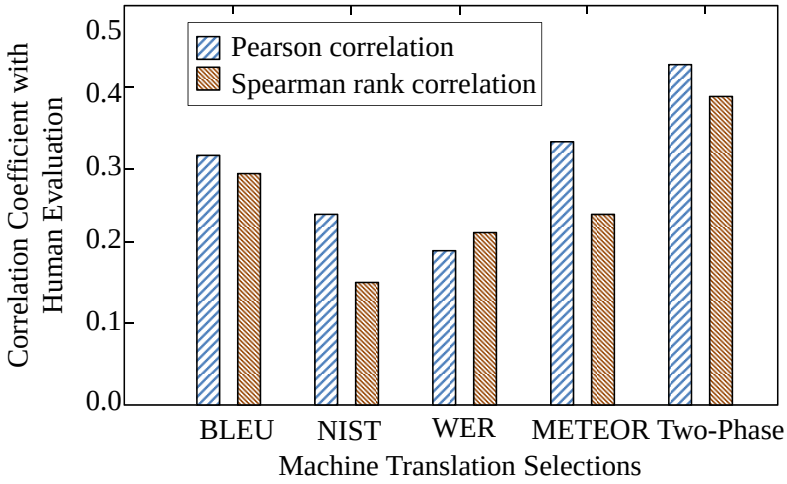

Figure 3.8: Correlation coefficient of machine translation selections

3.4.3 Experiment II: Dynamic Translation Requests

For a more complex situation, the user will dynamically send translation requests of different domains, and the training of decision tree is needed only for once.

Training for Two-Phase Selection

We randomly divide all the 500 pairs into 2 groups, with 250 pairs in each group. We only use the one group for this training two-phase MT selection, and leave one group for testing. As they are from different domains and language pairs, the feature set includes *language pair*, *domain*, and *length of translation request* (number of words).

Table 3.4: Selection for translation requests in separate domain corpus

(Japanese to English) (comparing "using one machine translation without selection", "selecting machine translation with one evaluation method", and "the proposed two-phase machine translation selection". 100 parallel text pairs are sampled for each domain.)

Domain of Requests	Service	Average Evaluation Score				Average Adequacy (Human)
		BLEU	NIST	WER	METEROR	
NTT	Google	0.238	1.284	-0.915	0.336	3.40
	J-Server	0.254	1.550	-0.727	0.372	3.65
	Web-Transer	0.308	1.656	-0.706	0.433	3.50
	Yakushitenet	0.196	1.161	-0.860	0.298	3.35
	BLEU Selection	**0.350**	1.908	-0.623	0.464	3.85
	NIST Selection	0.346	**1.909**	-0.621	0.461	3.90
	WER Selection	0.346	1.909	**-0.576**	0.461	3.90
	METEOR Selection	0.342	1.894	-0.603	**0.467**	3.90
	Two-phase Strategy	0.277	1.459	-0.727	0.413	**4.10**
Medical	Google	0.127	0.879	-1.039	0.239	2.95
	J-Server	0.263	1.393	-0.759	0.374	3.40
	Web-Transer	0.161	1.224	-0.937	0.303	3.55
	Yakushitenet	0.185	1.106	-0.915	0.261	3.30
	BLEU Selection	**0.282**	1.457	-0.783	0.390	3.55
	NIST Selection	0.282	**1.457**	-0.783	0.390	3.55
	WER Selection	0.261	1.468	**-0.721**	0.354	3.75
	METEOR Selection	0.278	1.521	-0.784	**0.402**	3.80
	Two-phase Strategy	0.239	1.328	-0.795	0.372	**4.15**
Tanaka	Google	0.291	1.472	-0.791	0.379	2.40
	J-Server	0.164	0.969	-1.070	0.299	3.00
	Web-Transer	0.155	0.894	-1.108	0.272	2.80
	Yakushitenet	0.184	1.042	-1.008	0.282	2.50
	BLEU Selection	**0.330**	1.575	-0.776	0.412	3.10
	NIST Selection	0.323	**1.584**	-0.813	0.415	3.40
	WER Selection	0.311	1.460	**-0.744**	0.399	2.30
	METEOR Selection	0.312	1.557	-0.843	**0.431**	3.00
	Two-phase Strategy	0.225	1.223	-1.065	0.351	**3.50**

Results

The two-phase MT selection shows better adequacy even for this dynamic requests, which is simulated by mixed corpora of different domains (here are NTT corpus's *everyday life*, *medical*, Tanaka corpus's *English textbook*,

Table 3.5: Selection for translation requests in separate domain corpus

(Japanese to Chinese) (comparing "using one machine translation without selection", "selecting machine translation with one evaluation method", and "the proposed two-phase machine translation selection". 100 parallel text pairs are sampled for each domain.)

Domain of Requests	Service	Average Evaluation Score				Average Adequacy (Human)
		BLEU	NIST	WER	METEROR	
School	Google	0.125	0.957	-1.086	0.209	3.15
	J-Server	0.181	1.370	-0.860	0.266	3.75
	Web-Transer	0.162	1.175	-0.959	0.250	3.30
	BLEU Selection	**0.173**	1.173	-0.897	0.245	3.60
	NIST Selection	0.191	**1.501**	-0.868	0.298	3.90
	WER Selection	0.199	1.516	**-0.835**	0.302	4.05
	METEOR Selection	0.206	1.488	-0.869	**0.309**	4.15
	Two-phase Strategy	0.185	1.363	-0.860	0.269	**4.20**
Disaster	Google	0.128	0.936	-0.982	0.186	2.50
	J-Server	0.130	1.159	-0.889	0.231	3.25
	Web-Transer	0.149	1.164	-0.897	0.219	3.15
	BLEU Selection	**0.149**	1.122	-0.903	0.216	3.20
	NIST Selection	0.174	**1.364**	-0.897	0.269	3.15
	WER Selection	0.137	1.205	**-0.859**	0.227	3.00
	METEOR Selection	0.176	1.357	-0.892	**0.270**	3.25
	Two-phase Strategy	0.145	1.124	-0.901	0.222	**3.35**

school guidance, and *disaster handbook*). We use mixed corpus as dynamic translation requests, to test how this two-phase MT selection works. From the results (see Table 3.6), our way still got better average adequacy, compared to one evaluation method's selection. Firstly, for the dynamic translation request, the adequacy of mixed corpus by each machine translation, is not as high as the adequacy of easy translation domain, but it is indeed better than the difficult translation domain like Disaster corpus. For example, for the BLEU selection, average adequacy of mixed corpora (3.38) is not as good as NTT corpus (3.85), but is better than Disaster corpus (3.20). Secondly, the proposed two-phase strategy gets better adequacy (3.62) than the maximum of single evaluation method selection (3.45).

We also calculate the coefficient, both Pearson correlation coefficient and Spearman rank correlation coefficient, which represents correlation of

Table 3.6: Selection for dynamic translation requests in five domain corpora

(comparing "using one machine translation without selection", "selecting machine translation with one evaluation method", and "the proposed two-phase machine translation selection". 250 parallel text pairs are sampled from five domains.)

Domain of Requests	Service	Average Evaluation Score				Average Adequacy (Human)
		BLEU	NIST	WER	METEROR	
Dynamic	Google	0.158	0.980	-1.036	0.231	2.85
	J-Server	0.133	1.012	-1.037	0.227	3.36
	Web-Transer	0.149	0.939	-0.946	0.260	3.33
	BLEU Selection	**0.219**	1.335	-0.845	0.332	3.43
	NIST Selection	0.217	**1.415**	-0.821	0.327	3.38
	WER Selection	0.203	1.349	**-0.814**	0.314	3.41
	METEOR Selection	0.218	1.315	-0.861	**0.335**	3.45
	Two-phase strategy	0.183	1.187	-1.017	0.259	**3.62**

these evaluation method selection with human evaluation (see Figure 3.8). For each single evaluaiton method selection, we only count in the selected translation results and its evaluation score and its human adequacy score. For the two-phase strategy, first we process the human adequacy score with equation (3.5), then we calculate the correlation of our assessment score and this processed score. Compared with the single evaluation method selection, the proposed way got better Pearson correlation coefficient (0.42), and better Spearman rank correlation coefficient (0.39).

3.5 Discussion

3.5.1 Scalability of the Proposed Architecture

Current deployment of the proposed architecture is on small scale, and the experiment results in the last section have only four machine translation results, two language pairs, and four evaluation methods. We would like to make a larger scale of deployment, without the limitation of available human evaluation, and that's why we choose Japanese related translation

requests. Actually, because of the unpredictable characteristics of machine translation, even with a large scale of data, it is hard to make perfect prediction of translation quality of the new translation requests. That is the same reason for the users to prefer to treat the machine translation as an imperfect black box. Based on this consideration, the controllable and small scale data will also show the problems of helping the users take advantage of multiple evaluation methods to pick out a machine translation. From this small data, the unpredictable feature of machine translations is obvious. The imperfect parts of current evaluation methods are showed, such as with the Tanaka corpus as requests, WER evaluation method based selection produces lower adequacy than single J-Server machine translation (see Table 3.4). But, our design is loosely based on the available evaluation methods or machine translation systems. It is designed to automatically take advantage of available resources. Then, try to adapt to applications by selecting the evaluation method in the first place, so as to use a better evaluation method to bring better translation quality. Thus, the proposed architecture is not limited to this small scale.

Currently, with the development of the federation of Language Grid [Ishida, 2011], and the development of evaluation packages like Stanford Phrasal [Cer et al., 2010b], the large scale of machine translations will be available through Language Grid service register. Then, our design would like to be an interface for the users to access to the world's machine translations.

3.5.2 Challenging Issues

There are limitations of current evaluation methods not only on efficiency, but also on automation. The preparation of reference is a tough issue for automatic evaluation. When there are not many parallel texts, there is no other choice but the round-trip translation way (see Figure 3.5), which is controversial in terms of efficiency. Firstly, we provide the parallel text service, which allows the user to provide their own parallel text service from scratch. Secondly, our design is still meaningful in that we do not bind it to certain

evaluation methods. When there are breakthroughs of new evaluation methods, it can be registered to the proposed architecture. Lastly, in view of current usage of machine translations, either human-aided machine translation, or machine-aided human translation [Hutchins, 2005], human interaction is often the choice for higher quality. We can add human interaction to round-trip translation way, which is often in use in certain machine-assisted human translation, for example, the application of BLEU to reduce manual post-editing in machine assisted translation domains [Sankaran et al., 2012]. In such application, our proposal will be a good choice, because of the availability, quality promotion, and selection assessment goal of our proposal.

There are no standardized and generally accepted interfaces of all the machine translations and evaluation methods. Though we use unique interfaces for the wrapped machine translation services and evaluation method services, an international standard of such interfaces will indeed help. Linguistic service ontologies have been proposed for Language service [Klein, 2004, Ishida, 2011]. When the standards of language service ontology description are created and widely accepted, the users will benefit from these online services.

3.6 Conclusion

We examine current machine translations and evaluation methods from the users' view. We proposed a two-phase MT service selection architecture for the machine translation users. Because of the convenient availability and flexible applicability, many more MT translation systems pop out. Based on Language Grid platform, the machine translations and evaluation methods are wrapped as services. We proposed to automatically select a proper evaluation methods for better machine translations in this two-phase architecture. In the first phase, we import multiple evaluation methods, analyze features of translation requests, and find a proper evaluation method using decision tree. This data-driven method helps the users dynamically adapt multiple evaluation methods for application, other than make use of single

evaluation. In the second phase, the MT services and the selected evaluation method are invoked through Language Grid platform, then the evaluation results are calculated, the best translation is selected, and the assessment of selection is informed.

We deployed the architecture with four machine translations and four evaluation methods. Dynamic translation requests are simulated from five domains, and translated them into two languages. Two experiments were finished. When trained for each domain, the evaluation methods were selected according to the length of request by decision tree. When trained for mixed domains, two more features include languages and domains. Both experiments showed that the proposed architecture would increase the sum of translation quality of all the requests, in comparison to the use of single evaluation method.

Above all, we took advantage of multiple evaluation methods, designed and implemented the proposed MT service selection architecture, and calculated the assessment of MT service selection. Our experience showed that our proposed strategy had gained better translation quality than just using single evaluation method.

Chapter 4

Scenario Description for Domain Resources Integration

Users have to customize machine translation for integrating local domain resources for higher accurate machine translation. From the perspective of non-computing professional users in multilingual communication, flexible interface for integrating domain resources for different topics is needed. This chapter designs an interactive interface for users to flexibly compose domain resource services and machine translation services for customizing machine translations for different topics [Shi et al., 2012a].

4.1 Introduction

When a multilingual communication has been planned between two monolinguals, the communication designer, who want to monitor the communication, has to consider providing a certain translation system for this multilingual communication. Nowadays, machine translators become increasingly popular, because of cheaper cost, higher speed, and better availability. The inaccurate translation will be the barrier for machine translation mediated multilingual communication. Thus, the communication designer has to pay attention to how to provide accurate translation.

Generally, a multilingual communication falls into its task related domain [Bangalore et al., 2006]. Without integrating the domain resource, general machine translators cannot provide acceptable translation accuracy. In the traditional view, promotion of translation accuracy is transparent to the translation users. Here, taking the perspective of the designer, we focus on how to help accuracy promotion. Through a pyramid view of the translation environment, we check the translation systems, which are proper for the task-oriented multilingual communication . The translation environment of task-oriented translation involves *tasks*, *human* and *translation functions*. Accordingly, a translation system involves the machine translation function, domain relationship (task domain), and human effort. From down to top, the provided translation accuracy is increasing, while the automation is decreasing. The base is general machine translation, mainly the rule-based, statistical, example-based, and hybrid. The top is human experts translation. For the upper of this pyramid, human effort is more user-oriented and it is more domain related. Above the general machine translation, there are the domain related machine translations. Below the human expert translation, the computer-assisted translation and human-assisted machine translation are two main types. Both types need human-machine interaction. According to the role of the designer, who has the information of task related domain resources, we propose a human-assisted machine translation using the scenario as interaction.

Considering the ways to integrate domain resources, there are two existing directions to realize integration. The first direction is the domain adaptation for machine translation system [Bertoldi and Federico, 2009, Wu et al., 2008, Koehn and Schroeder, 2007, Sankaran et al., 2012]. The starting point is exploiting domain resources (bilingual dictionary or text) to adapt existent machine translators, which needs special training of domain resources. The other direction is the domain act based interlingual machine translation [Levin et al., 1998, Levin et al., 2002, Schultz et al., 2006]. Its key point is creating an expressive but simple interlingua, which is based on speech act analysis in this specific domain. It will bridge the source language message to the target language through extracted rules. However,

from the perspective of a designer, both directions of accuracy promotion are *heavy weight* and *costly*. The former needs a large amount of training domain resources. We cannot a communication designer to finish the training, which is allowed only under the instructions technical developers. The latter needs many manual notations of domain acts, such as speech acts types, parameters, and exceptions. Obviously, it is not possible for a designer to finish either of them on his or her own. A simple way is needed that would allow a designer to integrate the domain resources. Based on the *interaction* between the designer and the machine translator, we propose a lightweight task-oriented translation for the multilingual communication.

Thus, from the designer's perspective, we proposed a light-weight translation system, and it is based on service composition scenario [Shi et al., 2012a]. Because scenario is a synoptical sketch of further possible actions, it is a proper light-weight description of the overall information for interaction. Here, a language service composition scenario is designed for a light-weight description of interaction between the designer and the target translation system integrating domain resources.

4.2 Interaction for Accuracy Promotion

Language service composition techniques provide an alternative way to take advantage of domain resources, such as the bilingual dictionary or parallel text.

4.2.1 Language Services for In-Domain Resources Integration

Language Grid allows wrapping domain resources into atomic language services [Ishida, 2011], such as the dictionary service, and the parallel text service. The main categories of language services include *machine translator*, *dictionary*, *parallel text*, *morphological analyzer*, and *dependency parser*. Not only Each category has multiple existing services, and

it allows end-user to create atomic services from domain resources using a Web-based interface[1]. On the other hand, it allows the composition of language services as integration of domain resources. For example, a dictionary-translator composition service combines dictionary service, machine translator, and morphological analyzer to provide better translation accuracy [Bramantoro et al., 2010]. Also, the selection among multiple machine translation results helps accuracy promotion [Shi et al., 2012c]. Thus, with light-weight interaction, language service composition techniques will integrate domain resources to promote translation accuracy.

However, this language service composition technique has limitation on the polysemy and execution time. When the same word has different meanings in different domain, then different domain dictionaries will have conflict. Moreover, it is not very fast when combining dictionary service and machine translator service. Thus, it is necessary to *choose* proper domain dictionary services, parallel text services, and machine translator services as candidates rather than composition of all available language services.

4.2.2 Designer's Contribution to In-Domain Resources Integration

For a multilingual *campus orientation* example, a teacher from a university's student office wants to help foreign parents build an image of the university, and the teacher wants to plan multilingual communication allowing native volunteers to help those foreign parents eagerly. In this multilingual communication, this teacher can be viewed as the designer. According to the teacher's previous experience, the important information is divided into two topics: legal procedures and student life. When the general translator Google Translate was used, the communication history of Japanese-to-English *campus orientation* showed that, the untranslated or mistranslated messages because of lacking domain resources were counted (see Table. 4.1). The domain resources include location address name, or

[1] http://langrid.org/

educational organization names, etc.

Table 4.1: Due to lacking domain resources, inaccurate translation exists in Google Translate mediated campus-orientation multilingual communication

Topics (number of messages)	Inaccurate Translation of General Translator Google Translate
campus orientation (51)	all 13 inaccurate words.
1 legal procedure (22)	7 inaccurate words of location address.
1) *office (14)*	
2) *warning (8)*	
2 student life (29)	8 inaccurate words of location address, medical glossary, and office name.
1) *tuition (9)*	
2) *class schedule (11)*	
3) *health check (9)*	

Meanwhile, this teacher has collected the domain resources, such as bilingual dictionary and bilingual text. Assuming the designer owns the knowledge of the domain resources, with the online tools from Language Grid, the designer, who is non-computing professional, can wrap domain resources into the language services. Then, a proper interaction is necessary for taking advantage of those language services. Here, this teacher needs to interact to make sure mapping dictionary services of location address name, or office names to proper topics. We design a scenario description to realize this interaction.

4.2.3 Scenario as Designer's Interaction

A scenario is a proper way for a designer to tell the topics that are likely to be raised in the designed task and the language service candidates integrated domain resources (see Figure 4.1). The content of task-oriented communication can be partitioned into sub-topics [Bangalore et al., 2006]. To succeed in task-oriented translation, higher accurate translations of each

topic are preferred. Meanwhile, language service composition techniques are applied to support scenario description. Thus, the designer will describe the scenario as the interaction for integrating the domain resources.

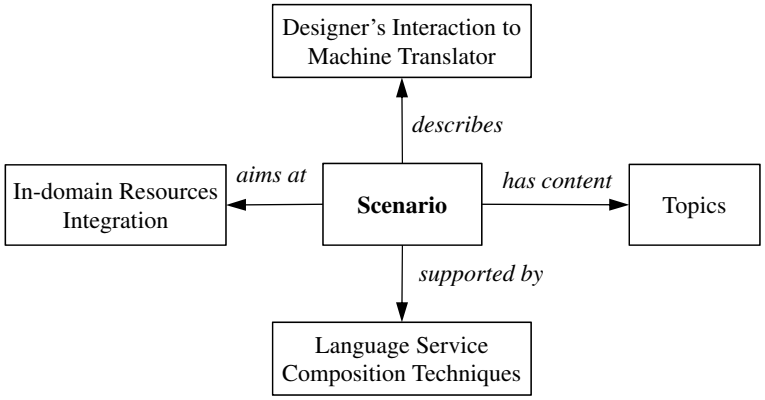

Figure 4.1: Role of a scenario in the machine translation mediated communication

Then, we design a scenario from the designer as a description of mapping of the proper language services to the planned target topics (see Figure 4.2). On the one hand, the topic structure, which is the sequence of subtopics, obviously affects the scenario description. On the other hand, with each detected topic, the goal is to map each topic with the proper resource wrapped language services according to the designer's knowledge and experience. Based on the existing research on the selection and composition of language services, it is able to select among several functionally-equivalent language services according to the accuracy or other quality of service (QoS) properties, such as the response time or the cost [Lin et al., 2010].

For a *campus orientation* communication example, there are two fixed sequence topics: legal procedure and student life. The legal procedure topic has two sub-topics: office (t_{x1}) and warning (t_{x2}) (see Figure 4.2). For the first sub-topic, Google Translate can be used (as S_{y11}). Foreign life parallel text can also be used (as S_{y12}). For simplicity, we can track certain keyword "notice" to detect the second sub-topic. For the second sub-topic, Google can still be one of the choices (as S_{y21}). Furthermore, a city location dictio-

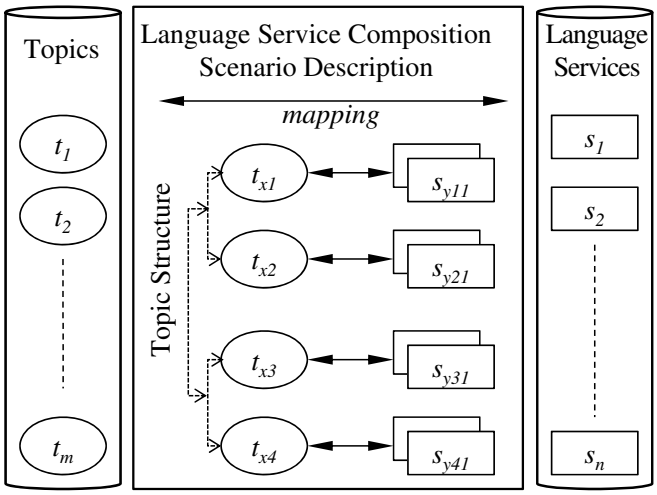

Figure 4.2: Scenario description aims at mapping proper language services to each topic

nary (as S_{y22}) can be helpful. Then, we need to select the translation result among multiple translations or combine dictionary service and translator service for each sub-topic (S_{y12} and S_{y12} for t_{x1}, S_{y21} and S_{y22} for t_{x2}).

From the reuse angle, different communication tasks might share topics or available language services. A topic that has already been configured and categorized, can be reused by the designer. Moreover, the mapped language services can also be reused, especially when they have been mapped by former designers. Thus, the potential topics and available language services can be maintained and reused. For example, given configured topics $\{t_1, t_2, \ldots, t_m\}$ and language services $\{s_1, s_2, \ldots, s_n\}$, the duty of a designer will be to choose the proper topics $\{t_{x1}, t_{x2}, t_{x3}, t_{x4}\}$ and proper language services $\{s_{y11}, s_{y12}, \ldots\}$).

The role of a scenario was explained above. In the following, we show how communication designers describe a scenario in detail, and how to realize this scenario-based machine translation.

4.3 Scenario Description for Interaction

We propose a scenario description language for the designer to describe the interaction of mapping the language services to the potential topics.

4.3.1 Scenario Description Language for Interaction

It is probable that designers are non-computer professional, who will not handle programming concept or programming syntax. Thus, this scenario description language has to be declarative, which is simple to interpret, and easy to write. Thus, we propose a prolog-like declarative script language, Scenario Description (SDL) Language. Its Backus-Naur Form (BNF) definition is described later in this chapter. There are three parts: topic structure, language services, and property requirements.

Topic Structure Description

We design *<topic-forest>* syntax for the designer to describe a topic structure. Generally, there are two types of topic sequences: fixed and dynamic. For example, the medical reception for foreigners is a typical fixed sequence. With regards to dynamic sequences, remote fault diagnosis has unfixed topics, because different faults are encountered. For the fixed sequence topics, it is easy to detect topics through techniques such as tracking and segmenting boundary. For the dynamic, it requires classification or searching according to the features, such as the comparable texts or keywords to detect the current topic [Shen et al., 2006]. Here, a topic forest will describe either fixed or dynamic topic. Its BNF description of *<topic-forest>* is:

<topic-forest> **::=** *<topic-forest><topic-tree>* | *<topic-tree>*.
<topic-tree> **::=** *<topic>':='<topic-list>*.
<topic-list> **::=** *<topic>* | *<topic-list>, <topic>*.
<topic> **::=** *<topic-name>(<service-variable>, <requirement-variable>).*

The topics of different grain levels and their sequence are depicted by a topic forest <*topic-forest*>. Firstly, the precedence of fixed sequence topics is depicted by sequence list. The fixed topic sequence can be well depicted by topic tree <*topic-tree*>. The dynamic topic sequence will be depicted as a set of fixed topics, or a topic forest. Secondly, the granularity of sub-topics is described by the designer according to his/her knowledge of available resources. Here, the *parent-child* link, depicted by ':=' mark, will be used for sub-topics description. Finally, each sub-topic will be combined with a service variable and a requirement variable.

Language Service Composition Description

To map each topic with language services, the designer not only needs to prepare the candidate language services, but also point out the usage of language services. Currently, there are mainly two types of language service composition techniques: service *selection* [Shi et al., 2012c] and *composition* [Bramantoro et al., 2010]. Atomic language services, wrapping domain resources, include dictionary (*dict*), parallel text (*para*). When several candidate services are chosen to be mapped to a certain topic <*topic-name*>, a service variable <*service-variable*> will represent those candidates <*candidates*>. Then, the language service selection is depicted by the mark *'|'*, while the composition of the dictionaries and translator, are depicted by mark *'+'*. Moreover, the types of dictionary services and parallel text services are marked with *'-dict'* and *'-para'* respectively

<*services*>**::=** <*service*> |<*services*><*service*>.
<*service*>**::=** <*service-variable*>*':='* <*candidates*>.
<*candidates*>**::=** <*service-name*> | <*candidates*>*'|'* <*service-name*>.
<*service-name*>**::=** <*atomic-name*> | <*service-name*>*'+'* <*atomic-name*>.

For example, "*foreign-life-para* | *google+city-dict*" represents selecting between parallel text (*foreign-life* is its service identification) and the com-

position of translator (*google*) and dictionary (*city*).

Language Service Property Requirements

Property requirements are references for selecting language service according to not only the *accuracy* but also other properties, such as the *response time* and the *price cost*. The multiple properties way is quality of service (QoS) based service selection [Yu et al., 2007]. The default property of language services is translation accuracy. The BLEU score, an automatic machine translation evaluation score, is often used as the metric of accuracy property [Shi et al., 2012c]. With more selection preferences on response time or price cost, the designer has to provide this <*requirement-variable*>, which is mapped to a topic <*topic-name*>.

<*requirements*>**::=** <*requirements*><*requirement*> | <*requirement*>.
<*requirement*>**::=** <*requirement-variable*>*':='* <*constraint-list*>.
<*constraint-list*>**::=** <*constraint*>|<*constraint-list*>, <*constraint*>.
<*constraint*>**::=** <*property-name*><*operator*><*value*>.

Then, each <*requirement-variable*> is depicted as a list of constraints <*constraint-list*> on multiple language service properties, and each constraint is a value limitation on one property.

Campus Orientation Example

The student office teacher wants to plan the task of campus orientation communication, and the example script of scenario description is provided (see Figure 4.3). It includes two fixed sequenced topics, *legal_procedure* and *student_life* . They are noted as two top-grained topics, each of which has low-grained sub-topics. Here, *office* and *warning* are the low-grained subtopics of *legal_procedure*. Each sub-topic is mapped with a variable of language services. For example, the topic *office* is mapped with *Serv1*, which is *selection* among parallel text service *foreign-life-para* and two composition *google+city-dict* and *j-server+city-dict*. Besides the default *accuracy*

```
1 campus_orientation(_, QosCo):=
2       legal_procedure(_, _), student_life(_, _).
3 legal_procedure(_, _):=
4       office(Serv1, _), warning(Serv2, _).
5 student_life(_, _):=
6       tuition(Serv3, _), class_schedule(Serv4, _),
7       health_check(Serv5, _).
8 Serv1:= foreign-life-para   | google + city-dict |
9           j-server + city-dict .
10 Serv2:= crime-disaster-para | google + city-dict |
11          j-server + city-dict .
12 Serv3:= school-life-para    | google + edu-dict  |
13          j-server + edu-dict .
14 Serv4:= google + edu-dict   | j-server + edu-dict.
15 Serv5:= medic-para  | google + city-dict + medic-dict  |
16          j-server + city-dict + medic-dict .
17 QosCo:= cost = 0 .
```

Figure 4.3: Script of scenario description for the campus orientation task ("-para": parallel text; "-dict": dictionary; "+": language service composition of dictionary and translator; "|": language service selection;"_": empty combination)

property, a requirement of price cost is also noted that the language services should be free. Here, the root topic *campus_orientation* is mapped *QosCo*, which is depicted as zero cost. Finally, all the variables, *Serv[1-5]* and *QosCo*, are concreted by the designer (see Figure 4.3).

4.3.2 Architecture

We propose an architecture to realize such scenario-based mechanism for task-oriented communication (see Figure 4.4). We start with the participants in this communication task: designer and communication subjects.

- Designer: with a clear image of planned topics within this task and information of domain resources. The designer has the duty of inte-

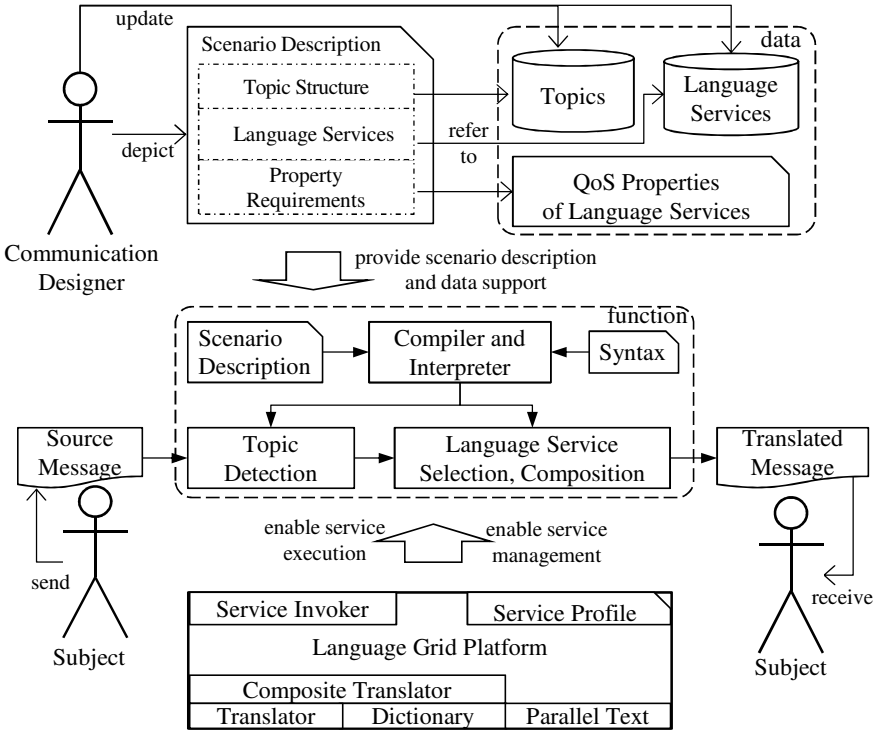

Figure 4.4: Architecture of scenario based language service composition

grating domain resources to raise translation accuracy, which is essential for fluent multilingual communication. Note that the designer is likely to be a non-computing-professional, so a simple interaction is preferred.

- Communication subjects: as the subjects in the communication task, the sender need to finish the planned topics to transfer task information to the receiver. They have the freedom to react with each other and elaborate on the topic content. Otherwise, a bilingual question-and-answer (QA) system will be chosen, rather than the task-oriented translation. In particular, the receiver will provide feedback based on his or her own understanding. Then, the receiver will be informed of the status of understanding, and provide further information as needed. However, they will face the problem of poor translation,

which breaks the communication circle, wears down subjects' effort, and hurts the subjects' enthusiasm. On the other handle, the higher accuracy will promote the communication fluency.

The inner function model includes three main components: topic detection, compiler and interpreter, language service selection and composition (see Figure 4.4).

1) Topic detection: it locates topics in the source messages from the sender. The categories of topics are the output of the Compiler and Interpreter component. The content of the detected topic will be translated by mapped language services. We implement this function by simply tracking appointed keywords for fixed topics, or classifying keywords for dynamic topics. For complex situations, existing research can be used [Allan, 2002].
2) Compiler and interpreter: based on the syntax (see Section 4.3.1), the scenario description script, which is depicted by the designer, will be compiled and interpreted into the topics, language services, and property requirements. We use SWI-Prolog[2] for compiling and interpreting the declarative interface language, which is easy.
3) Language service selection and composition: based on service selection and composition techniques, the most appropriate translation of the detected topic content will be deduced. After interpretation, the requirements on language service are interpreted to yield quality of service (QoS) constraints. For each detected topic, the source message is the input, the property requirements and the candidate language services are constraints and translation candidates. It accesses the language grid platform and returns the translation results. With the default accuracy based selection and quality requirements (time or cost) based selection, the selected translation result will be the output, and sent to the receiver. We make use of a Grid client, which invokes language services according to the input of service name and parameters. Moreover, various language services can be managed through the Language Grid platform.

[2]http://www.swi-prolog.org/

4.3.3 Interaction Process of Designer

The two-step process of interaction will be made by the designer to make both ends of language services and topics meet.

- First, the designer wraps available domain resources into language services, and registers potential topics by keywords and sequences. If in a reuse condition, the designer can locate the most related services and topics, and update them for the current task.
- Second, the designer describes the scenario script to map planned topics with language services wrapped domain resources. The scenario described for this multilingual communication task includes the topic structure, mapped language services, and property requirements (see Section 4.3.1).

Afterwards, the subjects will benefit from the scenario-based task-oriented translation.

4.4 Case Study

We provide a case study of Japanese-English *campus orientation* communication, we check the interaction process of designer and domain resource integration.

4.4.1 Interaction Process for Designer

According to the two-step interaction process of designer, domain resources are wrapped into services. In this case, with the language grid platform, the designer can manually wrap the domain dictionary service and parallel text service. For example, the designer can manually create and edit a Japanese-English city dictionary of location address names. After the language services are wrapped, the scenario script will be provided by the designer. In this case, the scenarios script is provided with reuse of the wrapped city dictionary (see Figure 4.3).

4.4.2 Domain Resource Integration

To determine the usage of domain resources, we counted the number of sentences translated by Parallel Text, Dictionary and Translator in each leaf topic. The ratios of the number of sentences were determined (see Figure 4.5). The *Parallel Text* and *Dictionary* services, wrapped versions of domain resources, improved the translation accuracy. For example, in the topic *office*, the contribution of Parallel Text is obvious (see Figure 4.6), and the scenario-based composite service has much higher adequacy than Google or J-Server. The default QoS property is the BLUE score based on the back-translation [Miyabe and Yoshino, 2009], and it will be used for selecting the best translation result.

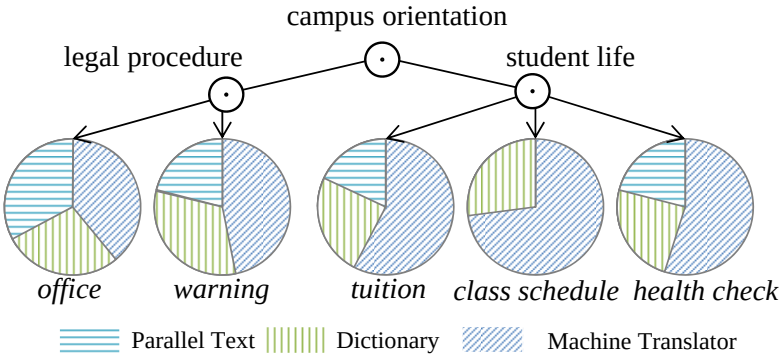

Figure 4.5: Ratio of the number of messages translation in each leaf topic (*Parallel Text*, *Dictionary* and *Translator*)

Two concrete messages are described here, see Figure 4.6, Figure 4.7 for *warning* topic and *health_check* topic respectively (see Figure 4.3). The former requires the use of *parallel text*. Here *crime-disaster-para* parallel text service provides the Japanese-English sentence pairs. The Japanese sender in the orientation task can use it for communication, and the English sentence will be sent to the English receiver. Obviously, parallel texts have higher *adequacy* than *Google* or *J-Server* outputs (see Figure 4.6). Here, the *adequacy* is a five-level (5:, 4:, 3:, 2:, 1:) human evaluation score of the translation accuracy. The latter shows how the dictionary can raise the

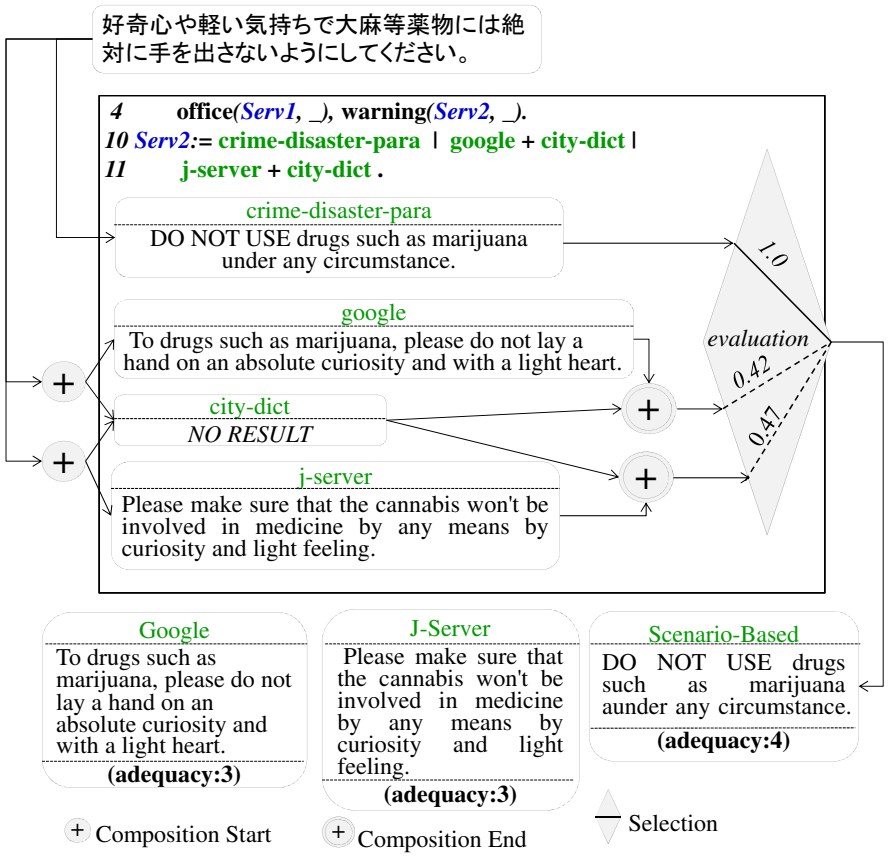

Figure 4.6: Integrating parallel text through selection (*warning* topic)

adequacy of translation. This is because the street name *Higashioji street* name is not one word for either Google or J-Server translation. If the *city-dict* is used, it will not be improperly cut, and the rest sentence parts are not mistranslated. The results of the multi-hop composition service demonstrate its superior adequacy compared to either Google composition or J-Server composition (see Figure 4.7).

To compare the accuracy promotion after the integration of the domain resources, we calculate the average adequacy of the messages by Google, J-Server, and the proposed scenario (see Table 4.2). It is obvious that the

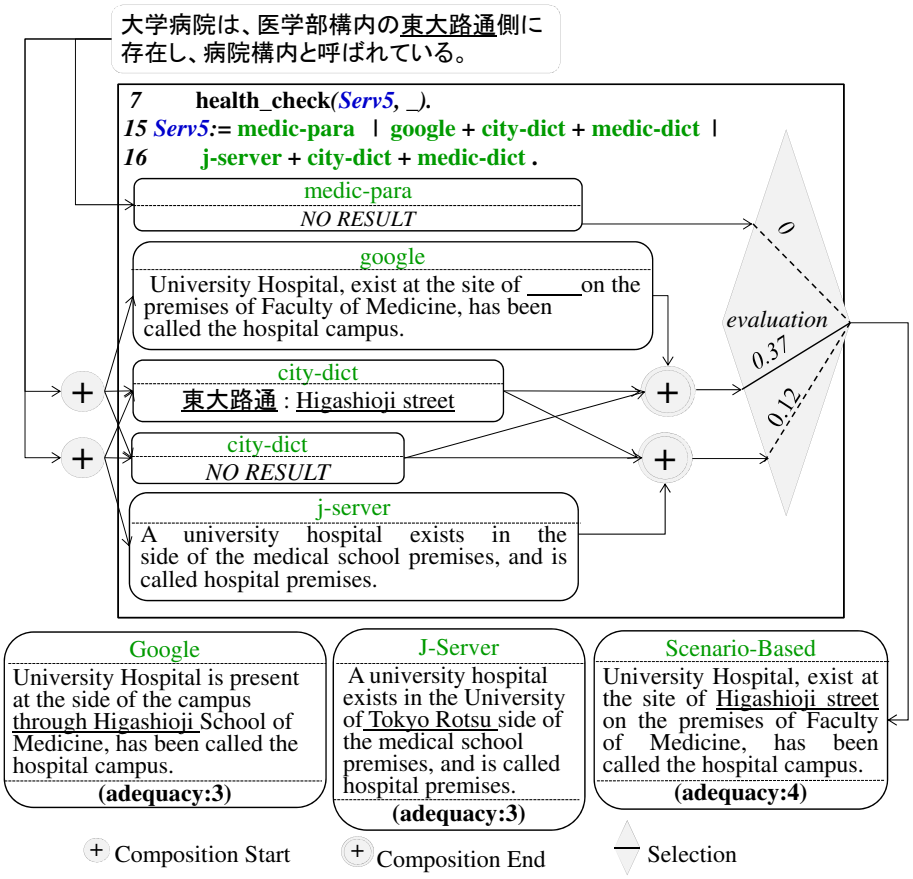

Figure 4.7: Integrating dictionary through composition (*health check* topic)

proposed scenario gained higher adequacy score (3.8) than either Google or J-Server machine translator. Thus, the accuracy is promoted by the proposed scenario strategy of integration of the domain resources.

Finally, note that the topics are stored in the topic database. When planning another related communication task, we can easily find and reuse useful topics; for example, the topic *office* in the "campus orientation" task can be used in other similar tasks like "school history introduction".

Table 4.2: Average adequacy of translated messages by Google, J-Server and scenario description based language service composition

Google	J-Server	Scenario Description based Language Service Composition
2.8	2.9	3.8

4.5 Discussion

Before concluding, we want to consider the scalability and limitation of the proposed scenario. Only one case study has been provided here, but the proposed scenario way is applicable to other multilingual communication, especially in some low-cost short-term international cooperations. On the one hand, the language service composition has been used for multilingual communication in multilingual games, student interactions, and eduction [Ishida, 2011]. On the other hand, the role of the designer is to describe the language service composition scenario in order to actively improve fluency of the multilingual communication. Through globalization, an increasing number of volunteers will experience internal cooperation. Thus, cooperation planners are welcomed. For example, a non-profit group organized volunteer Japanese agriculture experts to help Vietnamese farmers and agriculture students to collect bilingual dictionaries through Japanese-English-Vietnamese mapping. With the task requirement and available domain resources, the proposed scenario would be welcomed by those designers to plan short-term multilingual communication tasks.

Meanwhile, some limitations of our scenario should also be noted. Firstly, language service composition takes a longer online time to finish several language services. That is also one of the reasons why the scenario description is needed, because it is too costly to invoke free combination of available language services. Secondly, the scenario description needs designers' experience or understanding of both communication topics and available language services. Still, considering integrating domain resources, the proposed scenario will be convenient for designers to provide

a lightweight task-oriented translation.

4.6 Conclusion

We proposed a light-weight approach for designers to integrate domain resources so as to raise translation accuracy in task-oriented machine translation. Based on existing techniques (topic detection and QoS-based language service selection and composition), we conducted scenario-based interaction design, and provided a simple declarative scenario description language for the communication designer.

By using the simple scenario language, a designer can easily combine the topics of communication tasks with domain resources. Language Grid provides tools to conveniently wrap domain resources into dictionary service and parallel text service. With the services of domain resources available, and the SDL program script of composition scenario from the designer, QoS based language service selection and composition will be executed yielding more accurate translation results.

By using our architecture, it is easy to take advantage of existing language services on the Language Grid platform, refer to and reuse already configured topics and language services, and automatically select proper language services.

Finally, our case study of campus orientation task showed the translation accuracy can be raised through integrating domain resources.

Chapter 5

Interactivity Solution for Repair Translation Errors

Users have to improve translation quality for complementing low quality machine translation. From the perspective of non-experts machine translation users, the notification of translation errors and motivation of repairs are needed. The chapter proposes interactions between translation system and users to make users aware of machine translation quality and suggest users to conduct repairs, which becomes possible with the ability of evaluating translation quality by translation system [Shi et al., 2013].

5.1 Introduction

In view of the fact that machine translation errors cannot be ignored in machine translation mediated communication. We propose to shift from the transparent-channel metaphor to the human-interpreter (agent) metaphor. The agent metaphor was originally introduced by [Ishida, 2006a], in which *interactivity* is suggested as a new goal of the machine translator. Interactivity is the machine initiated interaction among the communication participants; it represents the ability to take positive actions to improve grounding and to negotiate meaning [Ishida, 2006a, Ishida, 2010]. Different from the

traditional metaphor of machine translation as a transparent channel, interactivity makes it clear that translation errors are to be treated as channel noise.

In this chapter, we propose an implementation of the agent metaphor for better interactivity. Interactivity is influenced strongly by the translation environment. Most translation environments involve the translation function and the user [Carl et al., 2002]. First, we have to mention the two characteristics of complex machine translation. One is the variable quality of machine translator output. The other is that, two messages expressing the same information can have widely different translation quality by the same machine translator. Second, in the transparent channel metaphor, the *activeness* of the user is ignored. Activeness plays an important role in interactivity. For example, certain people get better translation results than others because they are able to modify expressions to suit the characteristic of that machine translator. Thus, we need careful designs to promote interactivity.

We start by examining the machine translation of task-oriented dialogs. We list the typical translation errors leading to miscommunication. By analyzing the interactivity that can eliminate those errors, we formalize the requirements of an agent for encouraging interactivity. On one hand, the agent needs to know the translation quality. On the other, the agent needs to help the dialog participants adapt to the machine translator. Furthermore, we provide details of the design of the agent metaphor, including its architecture, interaction, and functions. In order to evaluate our prototype, we conduct an experiment on the multilingual tangram arrangement task. Next, we summarize what has been learned before discussing the limitations and implications of the current design.

5.2 Problems of Current Machine Translation Mediated Communication

Generally speaking, a communication dialog can be tagged as task-oriented, emotion-oriented, or both [Lemerise and Arsenio, 2000]. According to so-

cial information process theory, emotion-oriented dialog involves not only the cognitive process but also the emotion transfer process. Task-oriented dialog mainly focuses on the acquisition of information in the task domain [Bangalore et al., 2006]. In machine translation of task-oriented dialogs, the accurate translation of concepts is the basis of successful information transfer [Yamashita and Ishida, 2006a].

5.2.1 Multilingual Communication Task

As an example, we established several sessions of a concrete English-Chinese communication task. The goal was the *tangram arrangement* task in which an English participant instructs a Chinese participant to construct a tangram object from seven shapes. Because of the geometric shapes, the words and phrases mainly fall into the geometry domain. Google Translate[1], one of the most popular online machine translation services, was used as the machine translator.

5.2.2 Communication Break Due to Translation Errors

Based on the observations made during these sessions, we analyzed the communication breaks occasioned by translation errors. In one observation, due to phrase concept mistranslation, the word "square" in the geometry domain was translated into "plaza" of another domain, because the word "square" is polysemous. The machine translator just provides the everyday meaning of the word, but its true meaning depends on the task domain. In the next observation, the mistranslated sentence is an imperative sentence that requests the receiver to conduct an act ("put something someplace"). The dialog participants often describe actions in imperative sentences, such as requests and commands. Machine translators often fail to translate imperative sentences as well as declarative sentences. Another observation is the mistranslation of inconsistent phrases (see Figure 5.1); the abbreviated reference ("the light one") is not translated accurately, and it is unnatural

[1] http://translate.google.com/

to stick to exactly the same expression globally. Such inconsistency easily leads to translation errors.

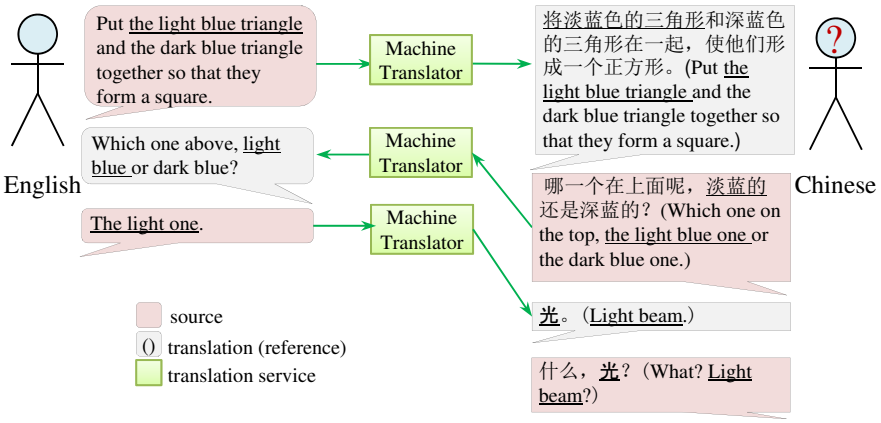

Figure 5.1: English-Chinese tangram arrangement communication (the Chinese receives an inconsistent translated phrase and the communication breaks.)

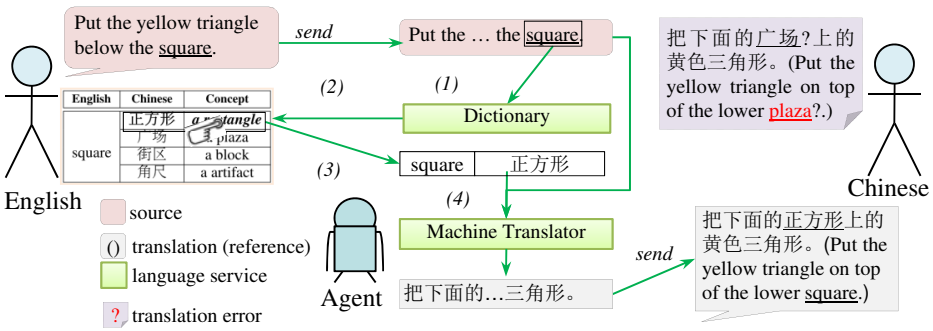

Figure 5.2: Interaction to handle inadequately translated phrase (*(1)* check the feature that the word, “square”, has one-to-many dictionary results. *(2)* suggest the sender select the correct concept. *(3)* the sender chooses the target concept. *(4)* translate by the dictionary translator composite machine translator.)

Analyzing miscommunication at the phrase, sentence, and dialog level is popular in machine-mediated communication research [Kiesler et al., 1985,

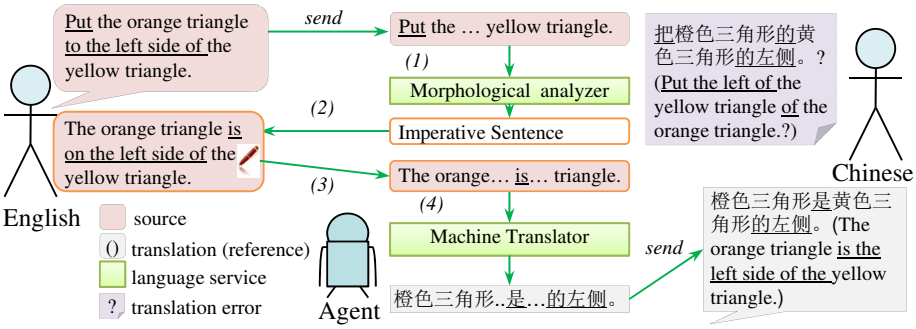

Figure 5.3: Interaction to handle mistranslated sentence
(*(1)* check the feature that it is an imperative sentence starting with a verb. *(2)* suggest the sender rewrite the sentence into declarative version. *(3)* rewrite the sentence. *(4)* translate by the machine translator.)

Table 5.1: Existing work on three levels and their corresponding mistranslation problems

Level	**Existing Work**	**Mistranslation**
Phrase level	Extract and highlight inaccurate words [Miyabe et al., 2008], picture icons as precise translation of basic concepts [Song et al., 2011].	Inadequate
Sentence level	Round-trip monolingual collaborative translation of sentence [Hu, 2009, Morita and Ishida, 2009a], Examine back-translation for sentence level accuracy check [Miyabe and Yoshino, 2009].	Influent and inadequate
Dialog level	Examine asymmetries in machine translations [Yamashita and Ishida, 2006b], Predict misconception due to unrecognized translation errors [Yamashita and Ishida, 2006a].	Inconsistent

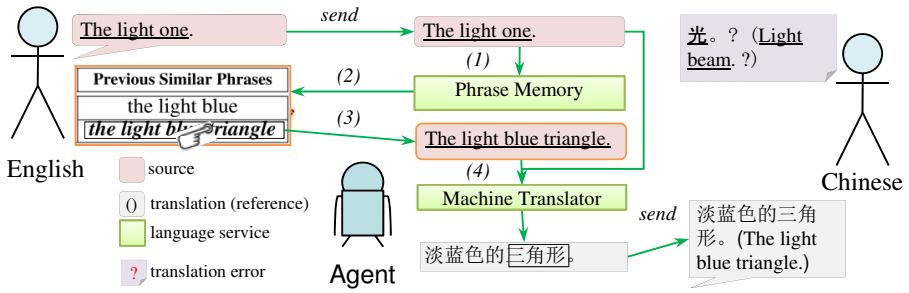

Figure 5.4: Interaction to handle inconsistently translated dialog (*(1)* check the feature of similar phrases existing in previous dialog. *(2)* suggest selection of appropriate previous phrase. *(3)* choose a replacement of the previous phrase. *(4)* translate by the machine translator.)

Yamashita and Ishida, 2006a]. These three observations of machine translation errors are picked up according to these levels: *phrase-level*, *sentence-level*, and *dialog-level*. Table 5.1 shows several existing work on examining mistranslation problems, providing suggestions and strategies for improving accuracy at each level. We summarize the mistranslation found in existing works. It shows that mistranslation often happens and can lead to communication breaks.

5.3 Interactivity and Agent Metaphor

5.3.1 Accuracy and Interactivity

When translation errors cannot be ignored in MT-mediated communication, the dialog participants can do nothing according to the transparent channel metaphor of machine translation (see Figure 5.1). The responses open to the machine translator fail to guarantee accuracy. If the dialog participants are encouraged to collaborate to eliminate such translation errors, the goal of the machine translator becomes to encourage interactivity. We studied what forms of interactivity could eliminate the translation errors expected. We replace the transparent channel model by introducing three interactions

to eliminate translation errors (see Figures 5.2, 5.3, 5.4).

When a translation failure is detected, the interaction process (see Figure 5.5) consists of: *(1)* Agent's effort to determine the *feature* of current dialog, *(2)* Agent's effort to *suggest* repair tips to the sender. Here, the human effort is referred to as "repair" as per machine translation mediated communication [Ishida, 2010, Miyabe et al., 2008]. *(3)* Sender's effort to repair the failure. *(4)* Agent's effort to translate the repaired message, and output an acceptable translation result. Given that there are multiple repair strategies, the agent has to decide the cause of the failure, send the appropriate repair suggestion to the sender. Other types of repair strategies, such as selecting phrases based on the prediction of available information [Carl et al., 2002], rephrasing based on back-translation results and sentence rewriting [Miyabe et al., 2008], can also be used.

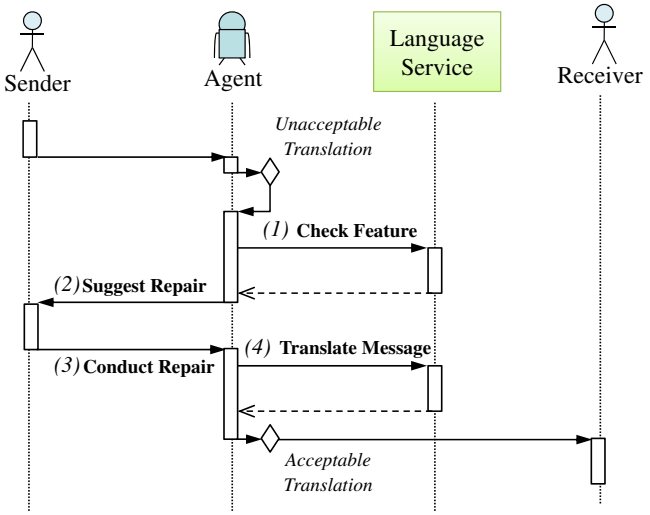

Figure 5.5: Four steps of the interaction process for one repair strategy (*(1)* Check Feature, *(2)* Suggest Repair, *(3)* Conduct Repair (Sender), and *(4)* Translate Message.)

Obviously, if the agent can initiate a proper interactivity with dialog participants, most translation errors will not be sent to the receiver. Still, we have to mention that the sensibility of dialog participants does not neces-

sarily lead to the elimination of translation errors, because of the unpredictability of the machine translation function, and the uncertainty of the human repair action. Thus, the interactivity between the agent and dialog participants must be carefully designed to motivate participants by making their actions easy, even for monolingual neophytes.

5.3.2 Agent Metaphor for Interactivity

Our case study showed that interactivity can eliminate most translation errors. Here, we discuss why the agent metaphor is needed to establish such interactivity. Basically, there are two reasons for applying the agent metaphor: agent sophistication, and the role of the agent [Jennings and Wooldridge, 1998]. In this study, the agent metaphor offers *flexible autonomous behavior* and a *decision support functionality*.

Flexible Autonomous Behavior: Because MT-mediated communication requires online translation and interactivity, a proactive agent has the ability to avoid unnecessary operations. For example, process protocol based collaborative translation [Hu et al., 2011, Morita and Ishida, 2009a] will go through the complete preset process flow, which is potentially inefficient. An agent enables flexible autonomous behavior, which is much more efficient.

Decision Support Functionality: Interaction will be triggered when translation errors are detected. After that, many decisions, such as translation error candidates, repair suggestions or extra translation improvement actions, need decision support functionality. A simple premise of this decision can be drawn from current translation quality. Through further design enhancement, the agent metaphor will gather additional quality estimates or information from the participants. Thus, the agent metaphor has to sense the quality of current translations, build common consensus among dialog participants, and pass proper repair suggestions to participants.

5.4 Design of Agent

5.4.1 Architecture

Our translation agent is designed around three agent phases: observation, decision, and action (see Figure 5.6).

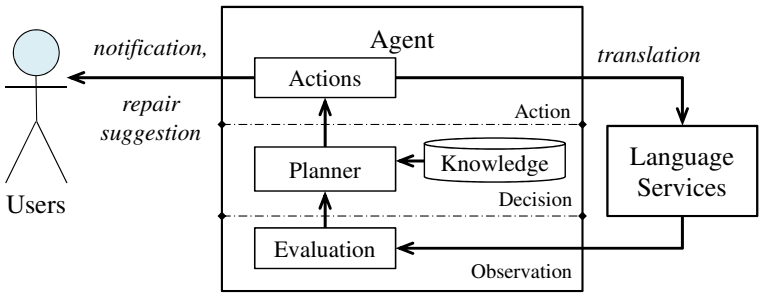

Figure 5.6: Architecture design of translation agent

Observation Phase

The goal is to discern the translation quality of each message. An *evaluation* module will fill this role. Popular evaluation methods such as BLEU [Papineni et al., 2002], METEOR [Banerjee and Lavie, 2005], compare the lexical similarity between the translation result and a standard reference to calculate an evaluation score. Other quality estimation approaches, such as the set of quality features [Specia et al., 2010] can be considered. Previous studies use back-translation to predict potential translation errors [Hu et al., 2011, Miyabe and Yoshino, 2009, Miyabe et al., 2008, Morita and Ishida, 2009a]. In this chapter, back-translation and the BLEU method (maximum 3-gram, smoothed) are used as a simple way to trigger interaction.

Decision Phase

This phrase decides the actions to be taken. Here, a real time *planner* is necessary, and a knowledge base is needed to keep experience and/or policy. The planner is critical to establishing autonomous behavior and decision support. Two important facts should be mentioned here. One is that the agent needs the ability to process the dialog in real time. The other is that the activities of the dialog participants will provide uncertain results. This is because the participants might have limited ability to generate correct repair actions or the machine translation quality of each message is unpredictable. Accordingly, the planner should provide online planning and decision support to counter this uncertainty. The knowledge base will save and allow access to experience and policy.

Action Phase

Three types of actions are needed. First, to help the dialog participants get an idea of current quality, a *notification* action is needed. Second, the detection of an unacceptable translation triggers the *repair suggestion* action. The repair suggestion is the key to interactivity. Last, *translation* actions are needed to implement the different repair strategies.

For the actions of notification and repair suggestion, the demand is that the agent and dialog participants talk. We use a simple meta-dialog for this purpose. For the translation actions, the repair strategies in the observations of the last section require the dictionary service result, and the dictionary translator composition service (see Figure 5.2). These services will be provided through the Language Grid [Ishida, 2011]. Through Language Grid, several categories of atomic language services are available, including dictionary, parallel text, morphological analyzer, dependency parser, machine translators, etc. Meanwhile, several composite services are available, including dictionary composite translation, multi-hop machine translation, and back-translation. Language Grid also provides a client that supports the invocation of both atomic and composite language services. People can develop their own version of services based on this client using Java programs.

Language Grid platform support allows translation actions to be realized and invoked flexibly.

5.4.2 Autonomous Behavior and Decision Support

Sharing the status of translation quality between participants, and helping participants adapt to machine translation, are two goals of interactivity. Each communication dialog consists of many rounds of message transferred from one participant to the other. Through this transfer, the agent triggers interactivity. There are two message transfer states: *Acceptable* accuracy, and *Unacceptable* accuracy. If the former, after the message is translated into the other language, and the accuracy is accepted, the translated message is sent to the receiver. If the latter, the agent will notify the participants and pass repair tips to the sender who then repairs the message. The message will be sent to the agent again, and the message transfer process repeats.

Two interactivity goals should be met. Satisfying the first goal, sharing the status of translation quality between participants, is obvious. In the above *Unacceptable* accuracy situation, an informational meta-dialog will be triggered and a notification meta-dialog message will be sent to the sender. A decision on whether it is acceptable or unacceptable is needed. For the second goal, helping participants adapt to machine translation, achieving the goal is essential. Based on the previous case study of interactivity, we learned three points. The first point is that there is more than one repair strategies. This means that the agent has to decide which strategy should be taken. The second point is that repair is a four-step process $\{$*feature*, *suggest*, *wait*, *translate*$\}$. The third point is that the effect of any repair action is uncertain. The decision, deciding which repair strategy is to be selected under uncertainty, is especially important.

The agent has to decide whether to pass the message to the receiver, and if not, which repair strategy is to be taken. When the message is received, translated, and evaluated, the evaluation score is calculated via back-translation. The evaluation score determines whether the message is passed on or a repair strategy is needed. About the next decision, which repair strat-

egy to adopt, the features of multiple repair strategies are checked and one is selected. These decision requirements can be met through a utility decision model [Bohnenberger and Jameson, 2001]. The agent's autonomous behavior and decision support allow it to issue the appropriate repair strategy even under uncertainty.

5.4.3 Repair Strategy Example

An example of issuing the repair strategy "split", is explained. Here, we picked one rule from the AECMA Simplified English Standard [AECMA, 1995], which is for technical manual preparation, and tried using it as the basis of a repair strategy, because simplified writing is effective in enhancing machine translation quality according to Pym's study [Pym, 1990].

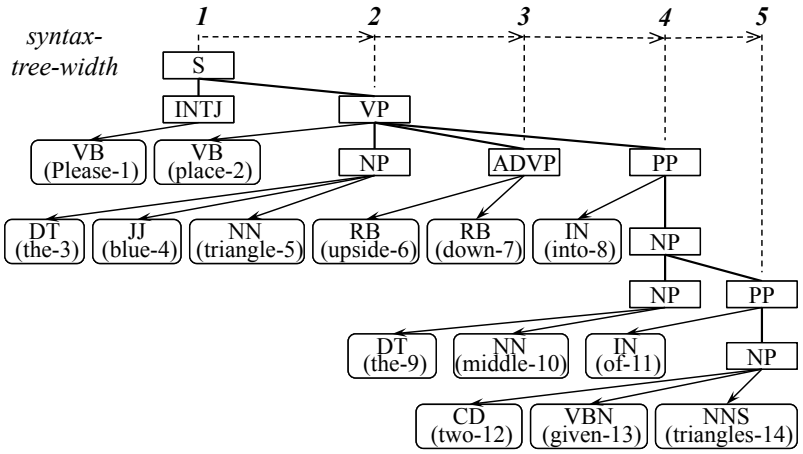

Figure 5.7: The syntax-tree-width feature of the repair strategy split (a width of non-leaf part of its constituency structure tree.)

Simplified Writing Rule: use short sentences. Restrict sentence length to no more than 20 words (procedural sentences) or 25 words (descriptive sentences). Inspired by this rule, we developed the repair strategy "split". *Repair Strategy Split*: when an unacceptable translation is detected, if the message is a long and complex sentence, the repair tip is to split the source

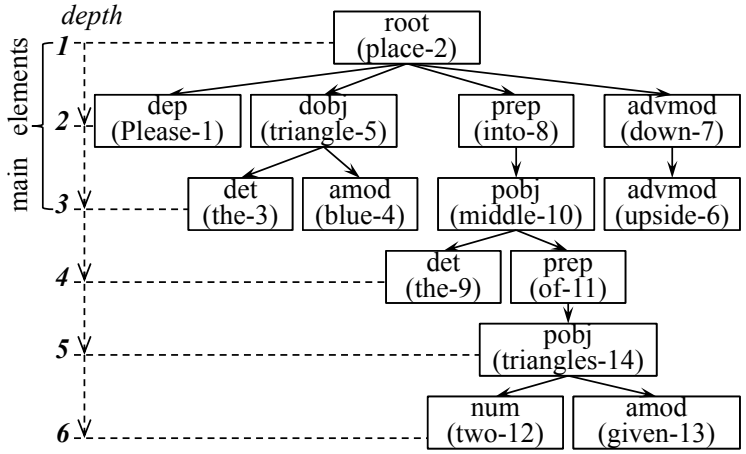

Figure 5.8: The tips for the repair strategy split
(the core of the message, which is the main elements of the sentence with low depth (less than 4) in the dependency structure tree.)

sentence into two sentences. *Feature of Split Strategy*: the literal length of the sentence is not directly used here. Instead, we choose the syntax-tree-width of its non-leaf syntax tree (see Figure 5.7). For example, the English message from the tangram arrange task, "Please place the blue triangle upside down into the middle of two given triangles.", is parsed into a constituency structure tree. The non-leaf part nodes form a non-leaf syntax, and its width is 5. Compared to the literal message length, this syntax-tree-width better represents the complexity of sentence structure.

Repair Suggestion: the tips are provided to help the sender undertake the repair. In this repair strategy, the core of the message, which is the main elements of the sentence with low depth (less than 3) in the dependency structure tree, is picked out for the sender (see Figure 5.8). This meta-dialog shows that, if the repair strategy is "split", then the suggestion and repair tips are passed to the sender (see Figure 5.9). The priority value is 0.5. It means that this will be the first message shown to the sender, if there is no higher priority meta-dialog defined for the IF premise. Both the constituency parse tree and dependency parse tree are from Stanford Parser [Klein and Manning, 2003], which is an open source Java im-

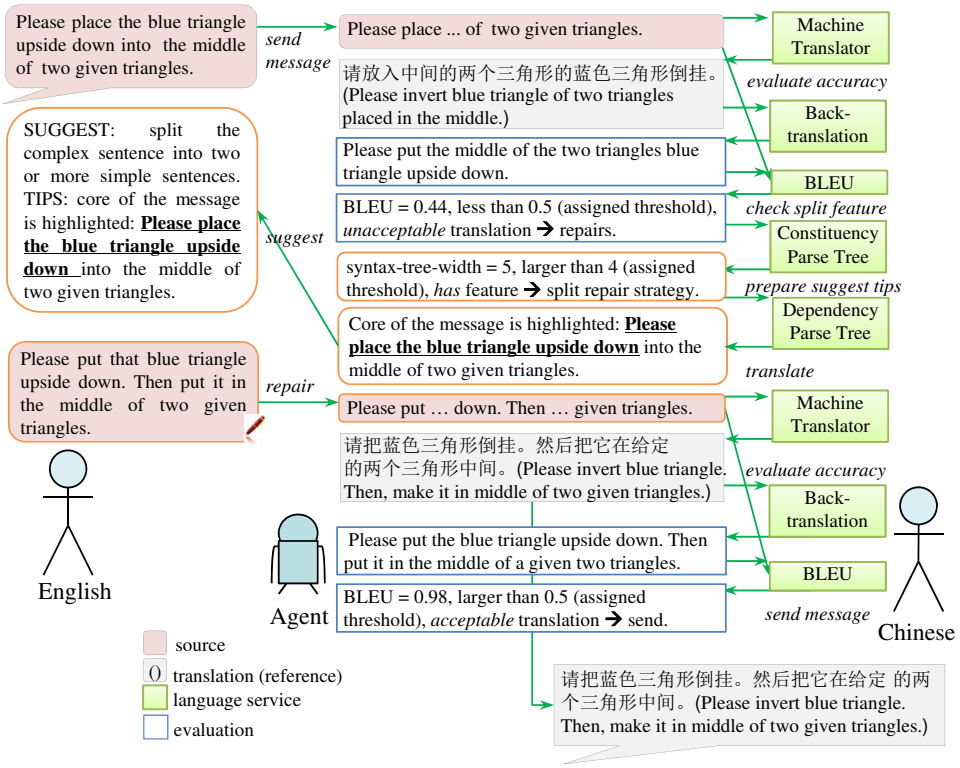

Figure 5.9: Example of agent's split strategy

plementation of natural language parsers. It provides a consistent interface for dependent parsers for English, Chinese, Arab, and German.

Here we describe the process of preparing our split repair strategy. According to our observation of the English-Chinese tangram arrangement sessions, we found instances in which this repair strategy was needed (see Figure 5.9). Obviously, the translated message is initially evaluated as unacceptable. We use the back-translation, the BLEU score, and the threshold for a simple decision. The usage of back-translation has been discussed a lot [Hu et al., 2011, Miyabe and Yoshino, 2009, Miyabe et al., 2008, Morita and Ishida, 2009a]. The interaction process of split strategy is given and its result is shown (see Figure 5.9). The agent checks the feature of split strategy, prepares the suggestion tips, and feeds

back the split suggestion. After the sender splits the message following the tips, the translation is evaluated again and it becomes accepted.

5.5 Evaluation

5.5.1 Evaluation Methods

In order to evaluate the impact of the agent on interactivity, we conducted a controlled experiment, which compared the machine translator mediated transparent channel approach to the proposed agent mediated interactivity approach.

We considered how the elimination of translation errors raised the efficiency of communication. Higher efficiency means that the information is transferred with fewer messages. According to conversation analysis [Goodwin and Heritage, 1990], the *turn* is the basic unit interaction in the communication. Here, the tangram arrangement task can be divided into seven subtasks; there are seven pieces to be arranged. For each arrangement, the information transferred per turn unit, includes piece, rotation type, and position. The *number of human messages per turn unit* is defined as the number of messages sent by the human participants during one turn unit of the multilingual communication. It reflects the participants' effort to transfer the task information. For better data collection, after one message is sent, the participants were asked to wait for feedback before issuing the next message.

Normally, a turn unit consists of 2 messages: 1 information message from the sender and 1 feedback message from the receiver. Here, to transfer the square's position information, 4 messages are needed (the number of human messages is 4) because the translation error misleads the message receiver, and the receiver has a query. It should be noted that, in the agent metaphor, the repaired message from the sender is counted, for example, the number of human messages in the turn unit is 2 (two messages from the English sender) in the split strategy example (see Figure 5.9).

An English-Chinese tangram arrangement communication task was conducted: an English user instructs a Chinese user how to arrange a tangram (see Figure 5.10). When the tangram is complex, this task is generally difficult to finish through text based messages, even for two native speakers. We set two limitations to make this task easier to finish. *Only use convex figures*[2]: there are only 13 convex figures. It is much easier to construct a convex figure. *Share initial state of tangram pieces*: both participants start with the same piece arrangement. With these two limitations, tangram arrangement focuses on communication.

For each tangram, we conducted the task using a single machine translator, a translation agent prototype, and bilingual translators. We randomly selected 5 tangram figures from the 13 convex figures. Two English and 2 Chinese, and 1 English-Chinese bilingual joined this experiment.

Repair Strategies for Agent Prototype

In this experiment the agent prototype knew three repair strategies; the *split* strategy of the last section, and the two repair strategies of Figure 5.2 and Figure 5.3: *phrase* and *rewrite*.

5.5.2 Result and Analysis

Each group was asked to finish 5 figures. The number of human messages and the average number of human messages in each turn were collected (see Table 5.2). The average number of human messages in each turn in human-mediated communication is 2.2. This shows that human-mediated communication is pretty efficient. The average number of human messages in each turn in machine translator mediated communication was 3.7. This shows that using machine translation needs much more the participants' effort. Our prototype agent held the average number of human messages in each turn to 2.9, a 21.6% improvement in communication efficiency.

[2]http://en.wikipedia.org/wiki/Tangram

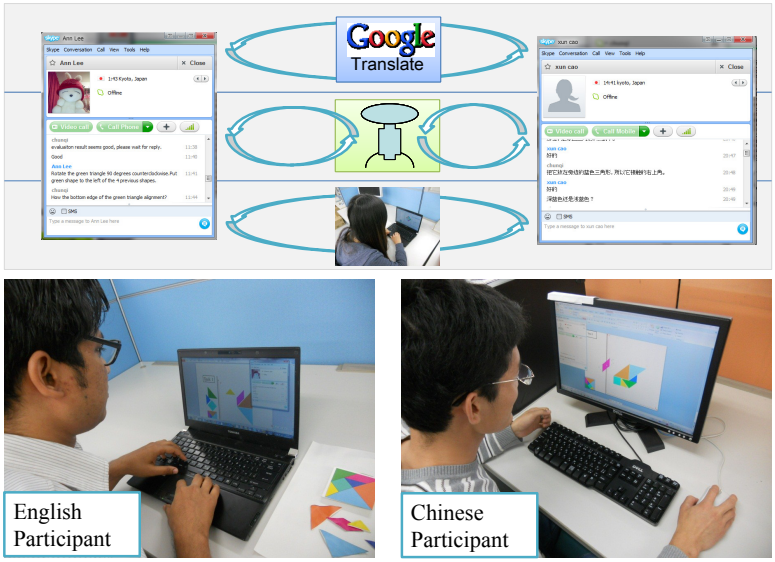

Figure 5.10: Experiment of English-Chinese tangram arrangement (through machine translator (MT), agent prototype using the wizard of OZ (Agent), and human bilingual (Human). There are two groups participants, E_1-and-C_1 and E_2-and-C_2 for this tangram arrangement experiment.)

Next, the total number of repair strategies in the English-Chinese dialogs was determined (see Table 5.3). First, the two different message senders had different repair strategies. Sender E_2's messages triggered more repair suggestions. The phrase and split strategies were used to almost the same extent. Second, different repair strategies took different amounts of time to complete. Here, the phrase strategy and split strategy were not activated as frequently as rewrite. This might be because there were few polysemous words, and the sentence structures were not too complex. We note that the senders tend to user many imperative messages.

5.6 Conclusion

Implementing the agent metaphor proposed herein represents a paradigm shift to using interactivity to eliminate translation errors in machine-

Table 5.2: Average number of human messages

(the average number of human messages in each turn unit for the 5 English-Chinese communication tasks)

Medium	**Average Number of Human Messages**		**Average Number of Human Messages / Turn**
	E_1**-and-**C_1	E_2**-and-**C_2	
MT	26.0	25.2	3.7
Agent	20.2	19.8	2.9
Human	15.6	14.8	2.2

Table 5.3: Total times of the repair strategies

Sender	**Total Times of the Repair Strategies**		
	Phrase	**Rewrite**	**Split**
E_1	6	21	9
E_2	5	17	10

translation-mediated communication. We examined the translation errors found in the dialogs of multilingual communication, and showed that interactivity could support the dialog participants in eliminating translation errors efficiently. Thus, our goals were to create a consensus as to the current translation state and provide repair suggestions to the sender. Both are realized by our agent metaphor. Evaluation of translation accuracy is critical for the agent to determine the current translation state. Back-translation and automatic evaluation methods, such as BLEU, are used to evaluate accuracy. To realize the autonomous agent mechanism, the process of repair suggestion was analyzed, the situations of message transfer were described, and decision dependency was analyzed for autonomous behavior and decision support. Our agent uses decision-theoretic planning to make online decisions under uncertainty. Finally, we described our experiments on a tangram arrangement task with English-Chinese task-oriented communication. The results showed that our agent prototype improved communication efficiency in the face of translation errors. The agent does help dialog participants raise the accuracy of translated messages.

Chapter 6

Conclusions

6.1 Summary of Original Contributions

The thesis presents three contributions toward user-centered design of translation systems for supporting multilingual communication. The first is the technique to automatic evaluate translation quality, two-phase evaluation architecture. The second is a machine translation customization interface, scenario description of service composition to integrate language services. The last is an agent metaphor, which motivates interactions to repair translation errors. Moreover, we have demonstrated that two-phase evaluation has application in a Japanese-Vietnamese communication for agriculture cooperation. We will review these contributions. After that, we will describe several areas for future research.

1) We design an adaptive architecture, two-phase evaluation, to help the user to pick out the best translation and calculate its accuracy for each translation message. It accesses to multiple machine translation services and multiple evaluation methods, such ad BLEU, NIST, and WER. Our strategy is to select a proper evaluation method in the first place, then to select the best machine translation using the evaluation results of the selected evaluation method in the next. Firstly, it raises *service availability* through the service-oriented language service platform, Language Grid,

making it easy to access both machine translation systems and evaluation methods as services. Secondly, it selects a proper evaluation method for each translation request. A data-driven way, decision tree is taken for this purpose, and its features include the translation languages, domains, and the length of translation request. Thus it gains *improved selection* as the proper evaluation method selects better machine translation. Thirdly, it offers a *selection assessment* to the user, informing the contribution of machine translation selection.

2) We propose a scenario as overall information for the communication designer to integrate in-domain resources for higher accuracy, which enable communication designer to prepare proper machine translation for multilingual communication. On the one hand, traditional way of integrating domain resources is too costly for communication designer to handle. We suggest to wrap in-domain resources as language services, and take advantage of language service composition technique for integration. On the other hand, we propose the scenario for designer to realize task-oriented machine translation, including the description language, and implementation architecture. Firstly, the task-oriented communication context is analyzed through scenario examination. Secondly, we introduce the interaction language allowing the task-designer to supervise the task-oriented translation in a convenient way. We design a light-weight architecture for task-oriented machine translation, based on the service composition mechanism. Finally, we do case study of Japanese-English school orientation communication task, the results show that our proposal makes good use of domain-resources in multilingual communication task, and the translation accuracy is improved from those in-domain resources.
3) We present agent metaphor as a novel interactivity solution to promote the efficiency in machine translation mediated communication. Machine translation is increasingly used to support multilingual communication. In the traditional, transparent-channel way of using machine translation for the multilingual communication, translation errors are not ignorable, due to the quality limitations of current machine translators. Those trans-

lation errors will break the communication and lead to miscommunication. We propose to shift from the transparent-channel metaphor to the human-interpreter metaphor, which motives the interactions between the users and the machine translator. Following this paradigm shifting from the transparent-channel metaphor to the human-interpreter metaphor, the interpreter (agent) encourages the dialog participants to collaborate, as their interactivity will be helpful in reducing the number of translation errors. We examine the translation issues raised by multilingual communication, and analyze the impact of interactivity on the elimination of translation errors. We propose an implementation of the agent metaphor, which promotes interactivity between dialog participants and the machine translator. We design the architecture of our agent, analyze the interaction process, describe decision support and autonomous behavior, and provide an example of repair strategy preparation.

Above all, we contribute in two aspects of machine translation mediated communication. In the first aspect, to gain better machine translation, we help users to deal with two types of changes of language services. On the one handle, we proposed two-phase evaluation to help user face the increasing number of machine translation services. On the other handle, we proposed scenario description to help users face the needs of different composition of language services. In the second aspect, to gain better communication in facing of low quality translation, we proposed the interactivity solution, agent metaphor, to help users to adapt to machine translators.

6.2 Future Direction

This work naturally leads to a number of future directions that may lead to further advances:

1) *User modeling for agent metaphor*

   Users of multilingual communication supporting tool varies in languages, foreign language skills, and experience in repairing translation errors. Translation agent interacts exactly same to all the users with-

out user modeling. However, different users will react to the translation agent very differently. For instance, certain users can repair the translation errors very well, based on his experience. Very simple quality notification from the agent will be helpful enough that the user will provide a fast repair. We need to build up user modeling for agent metaphor that are effective and scalable.

2) *Speech-act based negotiation protocol.*
Protocol design ranges from negotiation schemes to simple requests for a task [Mazouzi et al., 2002]. The negotiation between users and agent encourages each to be cooperative in solving translation errors. The speech act theory treats the uttering as actions, which change the state of the world. The protocol design is, therefore, not to simulate human negotiation, but to enhance agent's ability to participant in negotiation. The protocol for agent metaphor can be either facilitator or adapter (see Figure 6.1). The design and implementation of the negotiation protocol will be another challenge.

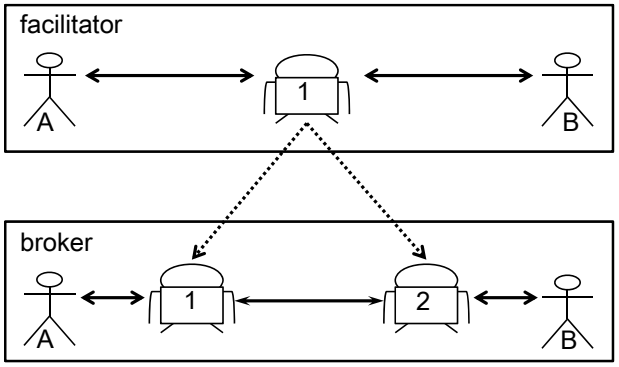

Figure 6.1: Two types of protocols: facilitator and adapter

3) *Mixed initiative planner for agent metaphor.*
Mixed Initiative interaction allows effective human-computer interaction [Allen et al., 1999]. In Section 5.4, the architecture of agent metaphor has been presented without a detailed planner description. The design of mixed initiative planner will ultimately promote flexibility in

the integration of repair strategies. In most case, agent's interactivity level is not determined in advance, but negotiated between the user and agent as the translation error is being repaired. The agent is reactive at one time, only feeding back the translation quality. At other time, the agent is mutual, motivating both users to repair translation errors.

4) *Automatic domain adaptation of translation agent.*
Domain adaptation aims for an integration of in-domain resources, such as dictionary [Wu et al., 2008]. In Section 4.1, scenario based language service composition has been proposed, which enables online integration of in-domain resource. It is still not automatic adaptation. Although this is an limitations of our work, we believe that it is an appropriate way to break down the problem. To further optimize our approach, automatic domain adaption improves the usability.

Bibliography

[AECMA, 1995] AECMA (1995). *A Guide for the Preparation of Aircraft Maintenance Documentation in the Aerospace Maintenance Language. AECMA Simplified English*. Brussels.

[Akiba et al., 2002] Akiba, Y., Watanabe, T., and Sumita, E. (2002). Using language and translation models to select the best among outputs from multiple mt systems. In *Proceedings of the 19th international conference on Computational linguistics - Volume 1*, COLING '02, pages 1–7.

[Allan, 2002] Allan, J., editor (2002). *Topic detection and tracking: event-based information organization*. Kluwer Academic Publishers, Norwell, MA, USA.

[Allen et al., 1999] Allen, J., Guinn, C., and Horvtz, E. (1999). Mixed-initiative interaction. *Intelligent Systems and their Applications, IEEE*, 14(5):14–23.

[Amigó et al., 2009] Amigó, E., Giménez, J., Gonzalo, J., and Verdejo, F. (2009). The contribution of linguistic features to automatic machine translation evaluation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, ACL '09, pages 306–314.

[Amigó et al., 2011] Amigó, E., Gonzalo, J., Gimenez, J., and Verdejo, F. (2011). Corroborating text evaluation results with heterogeneous mea-

sures. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 455–466.

[Arnold, 2007] Arnold, N. (2007). Reducing foreign language communication apprehension with computer-mediated communication: A preliminary study. *System*, 35(4):469 – 486.

[Banerjee and Lavie, 2005] Banerjee, S. and Lavie, A. (2005). Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the 43rd Annual Meeting of the Association of Computational Linguistics (ACL-2005)*, pages 65–72.

[Bangalore et al., 2006] Bangalore, S., Di Fabbrizio, G., and Stent, A. (2006). Learning the structure of task-driven human-human dialogs. In *Proceedings of ACL2006*, pages 201–208.

[Bangalore and Riccardi, 2000] Bangalore, S. and Riccardi, G. (2000). Stochastic finite-state models for spoken language machine translation. In *ANLP-NAACL 2000 Workshop: Embedded Machine Translation Systems*, EmbedMT '00, pages 52–59.

[Berger et al., 1994] Berger, A. L., Brown, P. F., Pietra, S. A. D., Pietra, V. J. D., Giuett, J. R., Lafferty, J. D., Mercer, R. L., Printz, H., and Urei, L. (1994). The candide system for machine translation. In *In Proceedings of the ARPA Conference on Human Language Technology*, pages 157–162.

[Bertoldi and Federico, 2009] Bertoldi, N. and Federico, M. (2009). Domain adaptation for statistical machine translation with monolingual resources. In *WMT*, pages 182–189.

[Bohnenberger and Jameson, 2001] Bohnenberger, T. and Jameson, A. (2001). When policies are better than plans: decision-theoretic planning of recommendation sequences. In *Proceedings of IUI2001*, pages 21–24.

[Bosca et al., 2012] Bosca, A., Dini, L., Kouylekov, M., and Trevisan, M. (2012). Linguagrid: a network of linguistic and semantic services for the

italian language. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 23–25.

[Bramantoro et al., 2010] Bramantoro, A., Schäfer, U., and Ishida, T. (2010). Towards an integrated architecture for composite language servicesand multiple linguistic processing components. In *LREC 10*, pages 3506–3511.

[Bramantoro et al., 2008] Bramantoro, A., Tanaka, M., Murakami, Y., Schäfer, U., and Ishida, T. (2008). A hybrid integrated architecture for language service composition. pages 345–352.

[Callison-Burch et al., 2008] Callison-Burch, C., Fordyce, C., Koehn, P., Monz, C., and Schroeder, J. (2008). Further meta-evaluation of machine translation. In *Proceedings of the Third Workshop on Statistical Machine Translation*, StatMT '08, pages 70–106.

[Callison-Burch et al., 2006] Callison-Burch, C., Koehn, P., and Osborne, M. (2006). Improved statistical machine translation using paraphrases. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 17–24.

[Carl et al., 2002] Carl, M., Way, A., and Schäler, R. (2002). Toward a hybrid integrated translation environment. In *Machine Translation: From Research to Real Users*, volume 2499 of *Lecture Notes in Computer Science*, pages 11–20.

[Cer et al., 2010a] Cer, D., Christopher, D. M., and Daniel, J. (2010a). The best lexical metric for phrase-based statistic mt system optimization. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 555–563.

[Cer et al., 2010b] Cer, D., Galley, M., Jurafsky, D., and Manning, C. D. (2010b). Phrasal: a toolkit for statistical machine translation with fa-

cilities for extraction and incorporation of arbitrary model features. In *Proceedings of the NAACL HLT*, pages 9–12.

[David Vilar, 2006] David Vilar, David Vilar, J. X. L. F. D. H. N. (2006). Error analysis of statistical machine translation output. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, pages 697–702.

[Doddington, 2002] Doddington, G. (2002). Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Human Language Technology conference (HLT-2002)*, page 128–132.

[Eduardo et al., 2007] Eduardo, Goncalves da, S., Luís, Ferreira, P., and Marten, van, S. (2007). An algorithm for automatic service composition. In *Proceedings of ACT4SOC*, pages 65–74. INSTICC Press.

[Estrella, 2008] Estrella, P., P.-B. A. K. M. (2008). Improving quality models for mt evaluation based on evaluators' feedback. In *Proc. LREC'08*, pages 933–937.

[Flickinger et al., 2005] Flickinger, D., Lønning, J. T., Dyvik, H., Oepen, S., and Bergen, U. I. (2005). Sem-i rational mt. enriching deep grammars with a semantic interface for scalable machine translation. In *In Proceedings of the 10th Machine Translation Summit*, pages 165–172.

[Giménez and Màrquez, 2010] Giménez, J. and Màrquez, L. (2010). Asiya: An Open Toolkit for Automatic Machine Translation (Meta-)Evaluation. *The Prague Bulletin of Mathematical Linguistics*, (94):77–86.

[Giménez and Amigó, 2006] Giménez, J. and Amigó, E. (2006). Iqmt: A framework for automatic machine translation evaluation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'06)*, pages 77–86.

[Goodwin and Heritage, 1990] Goodwin, C. and Heritage, J. (1990). Conversation analysis. *Annual Review of Anthropology*, 19:pp. 283–307.

[Goto et al., 2011] Goto, S., Murakami, Y., and Ishida, T. (2011). Reputation-based selection of language services. In *IEEE International Conference on Services Computing (SCC 2011)*, pages 330–337.

[Heyn, 1996] Heyn, M. (1996). Integrating machine translation into translation memory systems. In *In Proceedings of European Association for Machine Translation*, pages 32–38.

[Hu, 2009] Hu, C. (2009). Collaborative translation by monolingual users. In *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, CHI EA '09, pages 3105–3108.

[Hu et al., 2011] Hu, C., Bederson, B. B., Resnik, P., and Kronrod, Y. (2011). Monotrans2: a new human computation system to support monolingual translation. In *Proceedings of CHI2011*, pages 1133–1136.

[Hutchins, 2005] Hutchins, J. (2005). Current commercial machine translation systems and computer-based translation tools: system types and their uses. In *International Journal of Translation*, pages 5–38.

[Inaba, 2007] Inaba, R. (2007). Usability of multilingual communication tools. In *Usability and Internationalization. Global and Local User Interfaces*, volume 4560 of *Lecture Notes in Computer Science*, pages 91–97.

[Ishida, 2006a] Ishida, T. (2006a). Communicating culture. *IEEE Intelligent Systems*, 21:62–63.

[Ishida, 2006b] Ishida, T. (2006b). Language grid: An infrastructure for intercultural collaboration. In *SAINT*, pages 96–100.

[Ishida, 2010] Ishida, T. (2010). Intercultural collaboration using machine translation. *IEEE Internet Computing*, pages 26–28.

[Ishida, 2011] Ishida, T. (2011). *The Language Grid: Service-Oriented Collective Intelligence for Language Resource Interoperability*. Springer.

[Jennings and Wooldridge, 1998] Jennings, N. R. and Wooldridge, M. (1998). Agent technology. chapter Applications of intelligent agents, pages 3–28. Springer-Verlag New York, Inc.

[Josyula et al., 2003] Josyula, D. P., Anderson, M. L., and Perlis, D. (2003). Towards domain-independent, task-oriented, conversational adequacy. In *Proceedings of the 18th international joint conference on Artificial intelligence*, IJCAI'03, pages 1637–1638. Morgan Kaufmann Publishers Inc.

[Karakos et al., 2008] Karakos, D., Eisner, J., Khudanpur, S., and Dreyer, M. (2008). Machine translation system combination using itg-based alignments. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, HLT-Short '08, pages 81–84.

[Kay, 1998] Kay, M. (1998). The proper place of men and machines inlanguage translation. *Machine Translation*, 12(1/2):3–23.

[Kiesler et al., 1985] Kiesler, S., Zubrow, D., Moses, A. M., and Geller, V. (1985). Affect in computer-mediated communication: an experiment in synchronous terminal-to-terminal discussion. *Hum.-Comput. Interact.*, 1(1):77–104.

[Kim, 2002] Kim, K.-J. (2002). Cross-cultural comparisons of online collaboration. *Journal of Computer-Mediated Communication*, 8.

[Klein and Manning, 2003] Klein, D. and Manning, C. D. (2003). Fast exact inference with a factored model for natural language parsing. In *In Advances in Neural Information Processing Systems 15 (NIPS*, pages 3–10. MIT Press.

[Klein, 2004] Klein, E., P.-S. (2004). An ontology for nlp services. pages 177–180.

[Koehn and Schroeder, 2007] Koehn, P. and Schroeder, J. (2007). Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on StatMT*, pages 224–227.

[Lehmann et al., 2012] Lehmann, S., Gottesman, B., Grabowski, R., Kudo, M., Lo, S., Siegel, M., and Fouvry, F. (2012). Applying cnl authoring support to improve machine translation of forum data. In *Controlled Natural Language*, volume 7427 of *Lecture Notes in Computer Science*, pages 1–10.

[Lemerise and Arsenio, 2000] Lemerise, E. A. and Arsenio, W. F. (2000). An integrated model of emotion processes and cognition in social information processing. *Child Development*, 71(1):107–118.

[Levin et al., 1998] Levin, L., Gates, D., Lavie, A., and Waibel, A. (1998). An interlingua based on domain actions for machine translation of task-oriented dialogues. pages 129 –136.

[Levin et al., 2002] Levin, L., Gates, D., Wallace, D., Peterson, K., Lavie, A., Pianesi, F., Pianta, E., Cattoni, R., and Mana, N. (2002). Balancing expressiveness and simplicity in an interlingua for task based dialogue. In *Proceedings of the ACL-02 workshop on S2S '02*, pages 53–60.

[Lewis et al., 2009] Lewis, D., Curran, S., Feeney, K., Etzioni, Z., Keeney, J., Way, A., and Schäler, R. (2009). Web service integration for next generation localisation. In *Proceedings of the Workshop on Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, SETQA-NLP '09, pages 47–55.

[Lin, 2004] Lin, C.-Y. (2004). Rouge: a package for automatic evaluation of summaries. In *the Workshop on Text Summarization Branches Out (WAS 2004)*, pages 25–26.

[Lin et al., 2010] Lin, D., Murakami, Y., Ishida, T., Murakami, Y., and Tanaka, M. (2010). Composing human and machine translation services: Language grid for improving localization processes. In *LREC 10*, pages 500–506.

[Liu et al., 2011] Liu, C., Dahlmeier, D., and Ng, H. T. (2011). Better evaluation metrics lead to better machine translation. In *Proceedings of the*

2011 Annual Meeting on Empirical Methods in Natural Language Processing, pages 27–31, Edinburgh, Scotland, UK.

[Macherey and Inc, 2007] Macherey, W. and Inc, G. (2007). An empirical study on computing consensus translations from multiple machine translation systems. In *In EMNLP*, pages 129 –136.

[Matthias Eck and Waibel, 2006] Matthias Eck, S. V. and Waibel, A. (2006). A flexible online server for machine translation evaluation. In *Proceedings of EAMT 2006*, pages 223–231, Oslo, Norway.

[Matusov et al., 2005] Matusov, E., Kanthak, S., and Ney, H. (2005). On the integration of speech recognition and statistical machine translation. In *Proc. European Conf. on Speech Communication and Technology*, pages 467–474.

[Matusov et al., 2006] Matusov, E., Ueffing, N., and Ney, H. (2006). Computing consensus translation from multiple machine translation systems using enhanced hypotheses alignment. In *Cambridge University Engineering Department*, pages 33–40.

[Mazouzi et al., 2002] Mazouzi, H., Seghrouchni, A. E. F., and Haddad, S. (2002). Open protocol design for complex interactions in multi-agent systems. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 2*, AAMAS '02, pages 517–526, New York, NY, USA.

[Miyabe and Yoshino, 2009] Miyabe, M. and Yoshino, T. (2009). Accuracy evaluation of sentences translated to intermediate language in back translation. In *Proceedings of IUCS2009*, pages 30–35.

[Miyabe et al., 2008] Miyabe, M., Yoshino, T., and Shigenobu, T. (2008). Effects of repair support agent for accurate multilingual communication. In *PRICAI 2008: Trends in Artificial Intelligence*, volume 5351 of *Lecture Notes in Computer Science*, pages 1022–1027.

[Miyabe et al., 2009] Miyabe, M., Yoshino, T., and Shigenobu, T. (2009). Effects of undertaking translation repair using back translation. In *Proceedings of the 2009 international workshop on Intercultural collaboration*, IWIC '09, pages 33–40.

[Morita and Ishida, 2009a] Morita, D. and Ishida, T. (2009a). Collaborative translation by monolinguals with machine translators. In *Proceedings of the 14th IUI2009*, pages 361–366.

[Morita and Ishida, 2009b] Morita, D. and Ishida, T. (2009b). Designing protocols for collaborative translation. In *PRIMA '09*, PRIMA '09, pages 17–32, Berlin, Heidelberg.

[Nagao, 1984] Nagao, M. (1984). A framework of a mechanical translation between japanese and english by analogy principle. In *Artificial and Human Intelligence*, pages 173–180, North- Holland.

[Narayanan et al., 2006] Narayanan, S., Georgiou, P., Sethy, A., Wang, D., Bulut, M., Sundaram, S., Ettelaie, E., Ananthakrishnan, S., Franco, H., Precoda, K., Vergyri, D., Zheng, J., Wang, W., Gadde, R., Graciarena, M., Abrash, V., Frandsen, M., and Richey, C. (2006). Speech recognition engineering issues in speech to speech translation system design for low resource languages and domains. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 5, page V.

[Naruedomkul and Cercone, 2002] Naruedomkul, K. and Cercone, N. (2002). Generate and repair machine translation. *Computational Intelligence*, 18(3):254–269.

[Nießen et al., 2000] Nießen, S., Och, F. J., Leusch, G., and Ney, H. (2000). An evaluation tool for machine translation: Fast evaluation for mt research. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC)*, pages 39–45.

[Nirenburg et al., 1991] Nirenburg, S., Carbonell, J., Tomita, M., and Goodman, K. (1991). The kbmt project: A case study in knowledge-based machine translation. pages 297–303.

[Och, 2003] Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st ACL*, pages 160–167.

[Pado et al., 2009] Pado, S., Galley, M., Jurafsky, D., and Manning., C. (2009). Robust machine translation evaluation with entailment features. In *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJC-NLP of the AFNLP*, pages 297–305, suntec, Singapore.

[Padó et al., 2009] Padó, S., Galley, M., Jurafsky, D., and Manning, C. (2009). Robust machine translation evaluation with entailment features. In *Proceedings of ACL-IJCNLP*, pages 297–305.

[Papineni et al., 2002] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu:a method for automatic evaluation of machine translations. In *40th Annual Meeting of the Association for Computational Linguistics(ACL-2002)*, pages 311–318.

[Paul et al., 2007] Paul, M., Finch, A., and Sumita, E. (2007). Reducing human assessment of machine translation quality to binary classifiers. In *Proceedings of the 11th TMI*, pages 154–162.

[Plitt and Masselot, 2010] Plitt, M. and Masselot, F. (2010). A productivity test of statistical machine translation post-editing in a typical localisation context. In *The Prague Bulletin of Mathematical Linguistics*, pages 7–16.

[Popović and Ney, 2011] Popović, M. and Ney, H. (2011). Towards automatic error analysis of machine translation output. *Comput. Linguist.*, 37(4):657–688.

[Pym, 1990] Pym, P. J. (1990). Pre-editing and the use of simplified writing for mt: an engineer's experience of operating an mt system. In *Translating and the computer*, pages 80–96, London.

[Quinlan, 1993] Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc.

[Rafaeli, 1988] Rafaeli, S. (1988). Interactivity: From new media to communication. *Sage Annual Review of Communication Research: Advancing Communication Science*, 16:110–134.

[Resnik et al., 2010] Resnik, P., Buzek, O., Hu, C., Kronrod, Y., Quinn, A., and Bederson, B. B. (2010). Improving translation via targeted paraphrasing. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 127–137.

[Riva and Galimberti, 1998] Riva, G. and Galimberti, C. (1998). Computer-mediated communication: Identity and social interaction in an electronic environment. *Genetic, Social, and General Psychology Monographs*, 124:434–464.

[Sankaran et al., 2012] Sankaran, B., Razmara, M., Farzindar, A., Khreich, W., Popowich, F., and Sarkar, A. (2012). Domain adaptation techniques for machine translation and their evaluation in a real-world setting. In Kosseim, L. and Inkpen, D., editors, *Advances in Artificial Intelligence*, volume 7310 of *Lecture Notes in Computer Science*, pages 158–169.

[Schultz et al., 2006] Schultz, T., Black, A., Vogel, S., and Woszczyna, M. (2006). Flexible speech translation systems. *Audio, Speech, and Language Processing, IEEE Transactions on*, 14(2):403–411.

[Serhani et al., 2005] Serhani, M. A., Dssouli, R., Hafid, A., and Sahraoui, H. (2005). A qos broker based architecture for efficient web services selection. In *Proc. 2005 IEEE International Conference on Web Services (ICWS05*, pages 113–120. IEEE Computer Society.

[Shahaf and Horvitz, 2010] Shahaf, D. and Horvitz, E. (2010). Generalized task markets for human and machine computation. In *AAAI*, pages 113–120.

[Shen et al., 2006] Shen, D., Yang, Q., Sun, J.-T., and Chen, Z. (2006). Thread detection in dynamic text message streams. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 35–42.

[Shi et al., 2012a] Shi, C., Lin, D., and Ishida, T. (2012a). Service composition scenarios for task-oriented translation. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 2951–2958, Istanbul, Turkey.

[Shi et al., 2012b] Shi, C., Lin, D., and Ishida, T. (2012b). User-centered qos computation for web service selection. In *Proceedings of the 2012 IEEE 19th International Conference on Web Services*, ICWS '12, pages 456–463.

[Shi et al., 2013] Shi, C., Lin, D., and Ishida, T. (2013). Agent metaphor for machine translation mediated communication. In *Proceedings of the 2013 international conference on Intelligent user interfaces*, IUI '13, pages 67–74.

[Shi et al., 2012c] Shi, C., Lin, D., Shimada, M., and Ishida, T. (2012c). Two phase evaluation for selecting machine translation services. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 1771–1778.

[Snover et al., 2006] Snover, M., Dorr, B., Schwartz, R., Makhoul, J., and Micciula, L. (2006). A study of translation error rate with targeted human annotation. In *Proceedings of the Association for Machine Translation in the Americas Conference 2006*, pages 223–231.

[Somers and Jones, 1992] Somers, H. L. and Jones, D. (1992). Machine translation seen as interactive multilingual text generation. In *Proc. Translating and the Computer 13: The Theory and Practice of Machine Translation*, pages 153–165.

[Song et al., 2011] Song, W., Finch, A. M., Tanaka-Ishii, K., and Sumita, E. (2011). picotrans: an icon-driven user interface for machine translation on mobile devices. In *Proceedings of the 16th IUI2011*, pages 23–32.

[Specia et al., 2010] Specia, L., Raj, D., and Turchi, M. (2010). Machine translation evaluation versus quality estimation. *Machine Translation*, 24:39–50.

[Swift, 1991] Swift, J. S. (1991). Foreign language ability and international marketing. *European Journal of Marketing*, 25(12):36–49.

[Sánchez-Cartagena and Pérez-Ortiz, 2010] Sánchez-Cartagena, V. M. and Pérez-Ortiz, J. A. (2010). Scalemt: a free/open-source framework for building scalable machine translation web services. *The Prague Bulletin of Mathematical Linguistics*, 93:97–106.

[Tanaka et al., 2009] Tanaka, R., Murakami, Y., and Ishida, T. (2009). Context-based approach for pivot translation services. In *IJCAI*, pages 1555–1561.

[Tian et al., 2004] Tian, M., Gramm, A., Ritter, H., and Schiller, J. (2004). Efficient selection and monitoring of qos-aware web services with the ws-qos framework. In *in Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, pages 152–158.

[Toma, 1977] Toma (1977). Systran as a multi-lingual machine translation system. In *Commission of European Communities: Overcoming the Language Barrier*, page 129–160.

[Tsunoda and Hishiyama, 2010] Tsunoda, K. and Hishiyama, R. (2010). Design of multilingual participatory gaming simulations with a communication support agent. In *Proceedings of the 28th SIGDOC2010*, pages 17–25.

[veikko I. Rosti et al., 2007] veikko I. Rosti, A., Ayan, N. F., Xiang, B., Matsoukas, S., Schwartz, R., and Dorr, B. J. (2007). Combining outputs from multiple machine translation systems. In *In Proceedings of the*

North American Chapter of the Association for Computational Linguistics Human Language Technologies, pages 228–235.

[Wilks, 2009] Wilks, Y. (2009). *Machine Translation: Its Scope and Limits*. Springer.

[Wu et al., 2008] Wu, H., Wang, H., and Zong, C. (2008). Domain adaptation for statistical machine translation with domain dictionary and monolingual corpora. In *COLING*, pages 993–1000.

[Yamashita and Ishida, 2006a] Yamashita, N. and Ishida, T. (2006a). Automatic prediction of misconceptions in multilingual computer-mediated communication. In *IUI06*, pages 62–69.

[Yamashita and Ishida, 2006b] Yamashita, N. and Ishida, T. (2006b). Effects of machine translation on collaborative work. In *Proceedings of the 20th CSCW2006*, pages 515–524.

[Yu et al., 2007] Yu, T., Zhang, Y., and Lin, K.-J. (2007). Efficient algorithms for web services selection with end-to-end qos constraints. *ACM Trans. Web*, 1:159–166.

[Zhang and Vogel, 2010] Zhang, Y. and Vogel, S. (2010). Significance tests of automatic machine translation evaluation metrics. *Machine Translation*, 24:51–65.

Publications

Major Publications

Journals

1. **Chunqi Shi**, Toru Ishida, and Donghui Lin. "Translation Agent: A New Metaphor for Machine Translation." To *New Generation Computing*. (Conditional Accepted).

International Conference

1. **Chunqi Shi**, Donghui Lin, and Toru Ishida. "Agent metaphor for machine translation mediated communication." In *Proceedings of the 2013 international conference on Intelligent user interfaces* (IUI '13). ACM, New York, NY, USA, pp. 67-74, 2013.
2. **Chunqi Shi**, Donghui Lin, and Toru Ishida. "User-Centered QoS Computation for Web Service Selection." In *Proceedings of the 2012 IEEE 19th International Conference on Web Services* (ICWS '12). IEEE Computer Society, Washington, DC, USA, pp. 456-463. 2012.
3. Donghui Lin, **Chunqi Shi**, and Toru Ishida. "Dynamic Service Selection Based on Context-Aware QoS." In *Proceedings of the 2012 IEEE Ninth International Conference on Services Computing* (SCC '12). IEEE Computer Society, Washington, DC, USA, pp. 641-648. 2012.
4. **Chunqi Shi**, Donghui Lin, and Toru Ishida. "Two Phase Evalua-

tion for Selecting Machine Translation Services." In *Proceedings of the Eight International Conference on Language Resources and Evaluation* (LREC'12). European Language Resources Association (ELRA), Istanbul, Turkey, pp. 1771-1778. 2012.

5. **Chunqi Shi**, Donghui Lin, and Toru Ishida. "Service Composition Scenarios for Task-Oriented Translation." In *Proceedings of the Eight International Conference on Language Resources and Evaluation* (LREC'12). European Language Resources Association (ELRA), Istanbul, Turkey, pp. 2951-2958. 2012.

Workshops

1. **Chunqi Shi**, Toru Ishida, and Donghui Lin. " Interactivity Modeling for Machine Translation Mediated Communication." In *Proceedings of the IEICE Technical Report. Artificial Intelligence and Knowledge Processing* (IEICE-AI'13). Osaka, Japan, pp. 33-38. 2013.
2. **Chunqi Shi**. "Interactivity for Machine Translation Mediated Communication." In *Proceedings of the Workshop on the 75th National Convention of IPSJ* (IPSJ'13). Sendai, Japan, 5D-1. 2013.

Other Publications

Journal

1. **Chunqi Shi**, Zhiping Shi, Xi Liu, and Zhongzhi Shi. "Image Segmentation Based on Self-Organizing Dynamic Neural Network." in *Journal of Computer Research and Development*,Vol. 46, No. 01, pp. 23-30, 2009. (in Chinese).
2. **Chunqi Shi**,Fen Lin, and Zhongzhi Shi. "An Agent-Based Distributed Clustering System." in *Journal of Harbin Engineering University.*, Vol. 27, No. z1, pp. 346-350, 2006. (in Chinese).

International Conference

1. **Chunqi Shi**, Sulan Zhang, Zheng Zheng, and Zhongzhi Shi. "Geodesic Distance Based SOM for Image Clustering." in *Proceedings of the International Conference on Sensing, Computing and Automation* (ICSCA '06). DCSIS series B: Applications and Algorithms, Watam Press, Canada, pp. 2483-2488, 2006.
2. Sulan Zhang, **Chunqi Shi**, and Zhongzhi Shi. "Geometric Structure Based Image Clustering and Image Matching." in *Proceedings of the 5th IEEE International Conference on Cognitive Informatics* (ICCI '06). pp. 380-385, 2006.
3. Sulan Zhang, **Chunqi Shi**, and Zhongzhi Shi. "An Agent-Based Distributed Clustering System." in *Proceedings of the 18th International Conference on Pattern Recognition* (ICPR '06). pp. 1244-1247, 2006.