information

MDPI

Article

Towards Language Service Creation and Customization for Low-Resource Languages

Donghui Lin [1,*], Yohei Murakami [2] and Toru Ishida [3]

[1] Department of Social Informatics, Kyoto University, Kyoto 606-8501, Japan
[2] Faculty of Information Science and Engineering, Ritsumeikan University, Shiga 525-8577, Japan; yohei@fc.ritsumei.ac.jp
[3] School of Creative Science and Engineering, Waseda University, Tokyo 169-8555, Japan; toru.ishida@aoni.waseda.jp
[*] Correspondence: lindh@i.kyoto-u.ac.jp; Tel.: +81-75-753-4959

Received: 1 December 2019; Accepted: 21 January 2020; Published: 27 January 2020

Abstract: The most challenging issue with low-resource languages is the difficulty of obtaining enough language resources. In this paper, we propose a language service framework for low-resource languages that enables the automatic creation and customization of new resources from existing ones. To achieve this goal, we first introduce a service-oriented language infrastructure, the Language Grid; it realizes new language services by supporting the sharing and combining of language resources. We then show the applicability of the Language Grid to low-resource languages. Furthermore, we describe how we can now realize the automation and customization of language services. Finally, we illustrate our design concept by detailing a case study of automating and customizing bilingual dictionary induction for low-resource Turkic languages and Indonesian ethnic languages.

Keywords: low-resource languages; the Language Grid; language resources; language service infrastructure; bilingual dictionary

1. Introduction

There are over six thousand languages spoken in the world [1], but the availability of resources for each is extremely imbalanced. As of November 2019, Wikipedia was hosting 51,551,016 articles in 305 languages [2], among which there are only 16 languages that have more than 1,000,000 articles. The LRE map [3,4], initiated by the European Language Resources Association (ELRA) for language resource creation and sharing, represents information of 6143 resources covering 100 languages, among which only 12 languages have more than 50 resources registered. Google Translate [5], a major machine translation tool on the Internet, supports around 100 languages. Research on language resources and evaluation continues to focus on just a small number of languages, such as English, Chinese, French, German, Japanese, Spanish, etc. The majority of the world's languages are low-resourced, which makes it difficult to develop various tools for Natural Language Processing (NLP).

In recent years, the community of language resources and evaluation has been putting a lot of effort into supporting low-resource languages [6]. Previous studies have proposed computational linguistics methodologies for various types of NLP tools for low-resource languages, including machine translation [7–11], part-of-speech tagging and dependency parsing [12–15], speech recognition and keyword spotting [16,17], word embedding [18], lexicon creation [19], etc. Some studies use crowdsourcing approaches to annotate lexicons [20] and collect speech recognition data [21] for low-resource languages. Some other research proposes a collaborative model for different types of volunteers and contributors to take part in the knowledge organization process for sustainable preservation of language diversity and succeeds in building multilingual parallel corpora for low-resource languages [22,23]. However, it is always time consuming and costly to create resources and NLP tools for low-resource languages when starting from scratch. Therefore, it is necessary to deal with the issue of efficient resource creation. In this paper, we aim at providing a general framework that enables and enhances language resource creation for low-resource languages.

Our proposed framework is designed based on two aspects of language resource creation: *Automation* and *customization*. *Automation* means that the framework enables the automatic creation of new language resources for low-resource languages from existing language resources based on service-based workflows. For example, a pivot-based dictionary induction service workflow can be deployed in the framework, which can automatically generate a bilingual dictionary between two minor languages *A* and *C* based on two existing dictionaries *A*-*B* and *B*-*C*, where *B* is a major language. In the above service workflow, the pivot-based algorithm is one of the components. On the other hand, *customization* means that the framework enables customized automation processes of language resource creation based on the features of different tasks (e.g., different language pairs, different sizes of the data). Here, we consider a new example of generating a bilingual dictionary between two minor languages *D* and *F* via a major pivot language *E*. Since the situation is similar to the previous example, we can use the same automated service workflow for dictionary induction. However, the component of the pivot-based algorithm needs to be customized because different algorithms might be suitable for different language pairs.

Based on the above design concept, we need an infrastructure that can easily share existing language resources, compose various language resources for the automated creation of new resources, and customize the automated composition processes. To satisfy these infrastructure requirements, we developed the Language Grid, a service-oriented language infrastructure on the Web [24,25]. The Language Grid is built on service grid server software [26] and is based on the concepts of *fragmentation*, which provides various language services, and *recombination*, which realizes customizable language environments [27]. In fragmentation, existing language resources are wrapped using standardized interfaces to create atomic language services. In recombination, atomic language services are combined to create new services that offer automated service workflow processes. Users can customize atomic language services in such service workflow processes based on their own requirements. Since the Language Grid enables the automation and customization of language services, it can be used as a basic infrastructure for language resource creation to better support low-resource languages. In this sense, we are not aiming at building a unique framework just for low-resource languages. Rather, we focus on how to utilize and extend current language service infrastructures to support language resource creation for low-resource languages.

The contributions of this paper are as follows:

- We analyze the applicability of the Language Grid to low-resource languages. To enhance the sharing of low-resource language services, we established the federated operation of the Language Grid with three organizations in Bangkok, Jakarta, and Urumqi. As a result, the three federated operation centers have shared 49 language services, most of which are for low-resource languages, including southeast Asian languages, Indonesian languages, and Turkic languages. We confirm the potential of the Language Grid for low-resource language service sharing and show the necessity of enhancing language service creation by providing a general framework.
- We propose how to realize the framework based on the design concepts of automation and customization of language resource creation for low-resource languages. We then detail the requirements for the four service layers (service grid, atomic services, composite services, and application systems) in the proposed framework.
- We illustrate our proposed language service framework using a real-world case study of automating and customizing pivot-based bilingual dictionary induction services for low-resource Turkic languages and ethnic Indonesian languages.

2. Language Service Infrastructure: The Language Grid

The Language Grid [27] was born from a long-term research project of intercultural collaboration in Kyoto University. The motivation was to provide the end users with an infrastructure that enables unhindered access to language resources and creation of customized multilingual environments. We started the research and development of the Language Grid in 2006, aiming at building a service-oriented language service infrastructure on the Internet. The key idea of the Language Grid is to shift from language resources to language services. The objective continues to be to move beyond simply collecting language resources to sharing and interconnecting them as Web services. The stakeholders include the service providers, the service users, and the grid operators [24].

There are four main service layers in the Language Grid [28]. The bottom layer, called the service grid, manages all of the requests to the Language Grid and invokes language services. The service grid server software consists of five parts that are necessary for service-oriented architectures: The service manager, service supervisor, grid composer, service database, and composite service container [26]. The second layer is the atomic service layer, where users can create and register language services by wrapping language resources and tools based on the service interface types defined by the Language Grid. The third layer is the composite service layer, where atomic language services can be composed by Web service workflows for realizing complicated functions. The top layer is the application system layer, where different types of multilingual applications and intercultural collaboration tools are developed and provided to end users. To bridge the gap between language service infrastructures and application systems, we have also extended the architecture by introducing service invocation components, which transform Web service interfaces into libraries of various programming languages for easy service invocation and management [29].

To support the interconnection and customization of language services, we have put effort into improving service interoperability and the standardization of language services by constructing a Language Grid Ontology [30]. All language services use language resources wrapped in standardized Web service interfaces defined by the Language Grid Ontology. In the Language Grid, language service interfaces are organized in a hierarchical manner. The LanguageService class in the upper level can be further classified into four classes: The SpeechService class for processing speech data, DataService class for dealing with linguistic data resources, TransformationService class for transforming input texts to output texts, and AnalysisService class for analyzing input data and outputting analysis results [31]. Some of the available low-level service interface classes and examples of corresponding service types defined by the Language Grid Ontology are shown below.

- `<translate>` interface class: Translation, TranslationWithTemporalDictionary, BackTranslation, MultihopTranslation
- `<search>` interface class: BilingualDictionary, BilingualDictionaryWithLongestMatchSearch, ConceptDictionary, DialogCorpus, ParallelText, PictogramDictionary
- `<parse>` interface class: DependencyParse
- `<identify>` interface class: LanguageIdentification
- `<analyze>` interface class: MorphologicalAnalysis
- `<tag>` interface class: NamedEntityTagging
- `<recognize>` interface class: SpeechRecognition
- `<speak>` interface class: TextToSpeech
- `<paraphrase>` interface class: Paraphrase
- `<calculate>` interface class: SimilarityCalculation

Table 1 shows an example of the details of the `<translate>` interface class. Since `<translate>` has corresponding types of atomic services and composite services, including BackTranslation, MultihopTranslation, Translation, and TranslationWithTemporalDictionary, it can be used to invoke either an atomic translation service or a composite translation service, depending on the service endpoint information specified by the users. In the Language Grid, we developed a series of composite services that consist of a group of atomic services and that inherit from standardized service interfaces. For example, a composite machine translation service is composed of a morphological analysis service, a dictionary service, and a machine translation service.

Table 1. An example of the `<translate>` interface class in the Language Grid.

Interface Method	**String Translate (Language SourceLang, Language TargetLang, String Source)**
Parameters	**sourceLang**: The source language **targetLang**: The target language **source**: The string to be translated
Return value	The translation result will be returned.
Description	The `<translate>` interface class is standardized for invoking a translation service in the Language Grid, following the translation setting identified by three parameters: The source language, the target language, and the string to be translated. The `<translate>` interface class can be used to invoke an atomic translation service when specified with the service endpoint URL for atomic translation (e.g., Translation), or a composite translation service when specified with the service endpoint URL for composite translation (e.g., TranslationWithTemporalDictionary).
Service endpoint examples	Examples of the service endpoint of Translation: GoogleTranslate and KyotoUJserver. https://langrid.org/service_manager/wsdl/kyoto1.langrid:GoogleTranslate https://langrid.org/service_manager/wsdl/kyoto1.langrid:KyotoUJServer An example of the service endpoint of TranslationWithTemporalDictionary. https://langrid.org/service_manager/wsdl/kyoto1.langrid:TranslationCombinedWithBilingualDictionary

3. Applicability of the Language Grid to Low-Resource Languages

The Language Grid has been operating since 2007 from Kyoto University and, as of November 2019, has 183 participating groups from 24 countries sharing 226 language services. The Language Grid has been used for supporting various multilingual activities in hospital reception desks, local schools, shopping districts, international symposiums, and so on [27]. Research and development of the Language Grid have covered multiple areas, including artificial intelligence, NLP, services computing, and human–computer interaction [24,25].

The Language Grid is a general infrastructure for language services, and so its design and implementation did not focus on low-resource languages. In the early stages of the Language Grid operation, most of the registered language services involved only English, Japanese, Korean, and Chinese. This meant that we experienced many difficulties when the Language Grid was used to support real-world multilingual activities involving low-resource languages. To enhance the sharing of low-resource language services, we established the federated operation of the Language Grid with three organizations: NECTEC in Bangkok, Thailand, University of Indonesia in Jakarta, Indonesia, and Xinjiang University in Urumqi, China after 2010 (see Figure 1).

The Language Grids operated by different organizations are connected with each other for language service sharing. The intention is to develop low-resource language services through the Bangkok, Jakarta, and Urumqi operation centers, who provide language services for southeast Asian languages and the Indonesian and Turkic language families, respectively [27]. Among the 226 language services shared in the Language Grid, 49 are from the Bangkok, Jakarta, and Urumqi operation centers, most of which are low-resource language services. Table 2 shows a selected list of language services registered in these three operation centers.

As a first step, the Language Grid has already shown its potential for sharing low-resource language services by connecting the federated grids operated by organizations in different countries. However, we have also observed that low-resource language services were mostly registered at the very beginning of the federated grid operation and that the number of such services has basically remained static. Therefore, we need to consider how to stimulate the creation of low-resource language services based on existing language services.

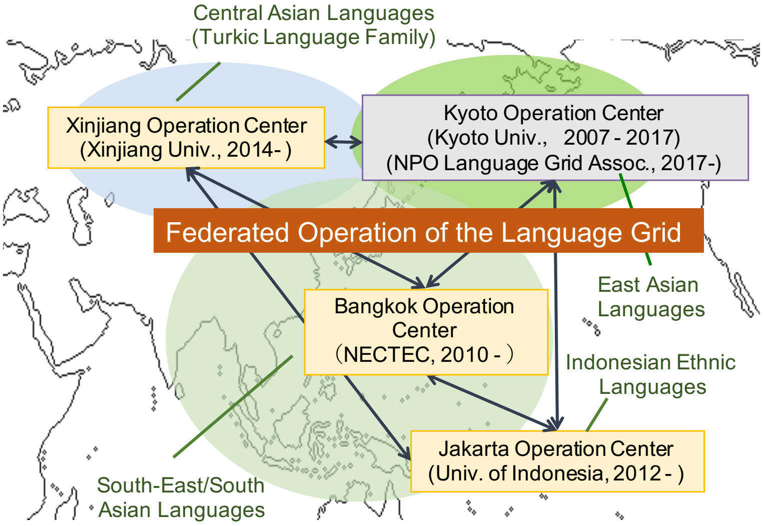

Figure 1. Federated operation of the Language Grid for low-resource languages.

Table 2. A selected list of language services registered by the Bangkok, Jakarta, and Urumqi operation centers with the Federated Language Grid. The services are categorized based on the service interfaces defined in the Language Grid.

Operation Center	Registered Language Services (Selected List)
Bangkok Language Grid Operation Center	**Translation**: ASEANMT (Indonesian–English), English–Tagalog Translation, ASEAN Machine Translation (English–Chinese, Brunei–English, English–Khmer, English–Laotian, Malaysia–English), Thai–Laotian Machine Translation, Parsit (English–Thai Machine Translation) **ConceptDictionary**: Asian WordNet (Bengali, Hindi, Indonesian, Japanese, Korean, Laotian, Mongolian, Burmese, Nepali, Singhalese, Sudanese, Thai, Vietnamese) **LanguageIdentification**: Data Extraction (Thai) **PictogramDictionary**: Thai Weaving Pattern with Impression **TextToSpeech**: Vaja6 API TTS (Thai, English) **MorphologicalAnalysis**: LexTo Word Segmentation (Thai) **BilingualDictionary**: LEXiTRON Bilingual Dictionary Service (English–Thai)
Jakarta Language Grid Operation Center	**Translation**: Indonesian–English Translation **MorphologicalAnalysis**: Indonesian Morphological Analysis, Indonesian POS Tagger **SpeechRecognition**: Indonesian Speech Recognition
Urumqi Language Grid Operation Center	**Translation**: Uyghur–Chinese Translator **ParallelText**: Kazakh–Chinese, Kyrgyz–Chinese, Uyghur–Chinese Parallel Text **BilingualDictionary**: Uyghur–Chinese Technological Terms Dictionary, Bilingual Dictionary (Kazakh–Chinese, Kyrgyz–Chinese, Uyghur–Chinese), Turkic Multilingual Dictionary (English, Turkmen, Uyghur, Kyrgyz, Kazakh, Turkish, Chinese, Azerbaijani, Uzbek, Tatar), Uyghur–Turkish–Chinese Dictionary

4. Language Service Creation and Customization for Low-Resource Languages

4.1. Design Concept

As discussed in previous sections, we need to consider the efficiency of resource creation and sharing for low-resource languages. Therefore, we aim at designing a framework that enables and enhances resource creation for low-resource languages. To achieve this goal, we focus on two considerations: *Automation* and *customization* of language resource creation. Figure 2 provides an example of a pivot-based bilingual dictionary induction service that illustrates our design concepts.

- **Automation of low-resource language service creation**
  The framework must enable the automatic creation of new language resources for low-resource languages from available language resources. For example, Uyghur and Kazakh are two closely-related languages belonging to the Turkic language family, but no comprehensive bilingual dictionary between the two low-resource languages exists. However, the Uyghur–Kazakh bilingual dictionary can be automatically induced from the Chinese–Uyghur and Chinese–Kazakh dictionaries if we develop a pivot-based algorithm that considers language similarity [32,33]. The whole process can be developed as an automated pivot-based dictionary induction service workflow and employed in other comparable situations.
- **Customization of low-resource language services**
  The framework must enable the customization of automation processes for language resource creation based on the features of different low-resource languages. Here, we consider an Indonesian ethnic language example which aims at inducing a new Malay–Minangkabau bilingual dictionary from existing Malay–Indonesian and Minangkabau–Indonesian dictionaries [34]. Since the situation is quite similar to the previous example, we can use the same automation process used in the Uyghur–Kazakh dictionary induction. However, the pivot-based algorithm must be customizable, since the second example has a different degree of language similarity.

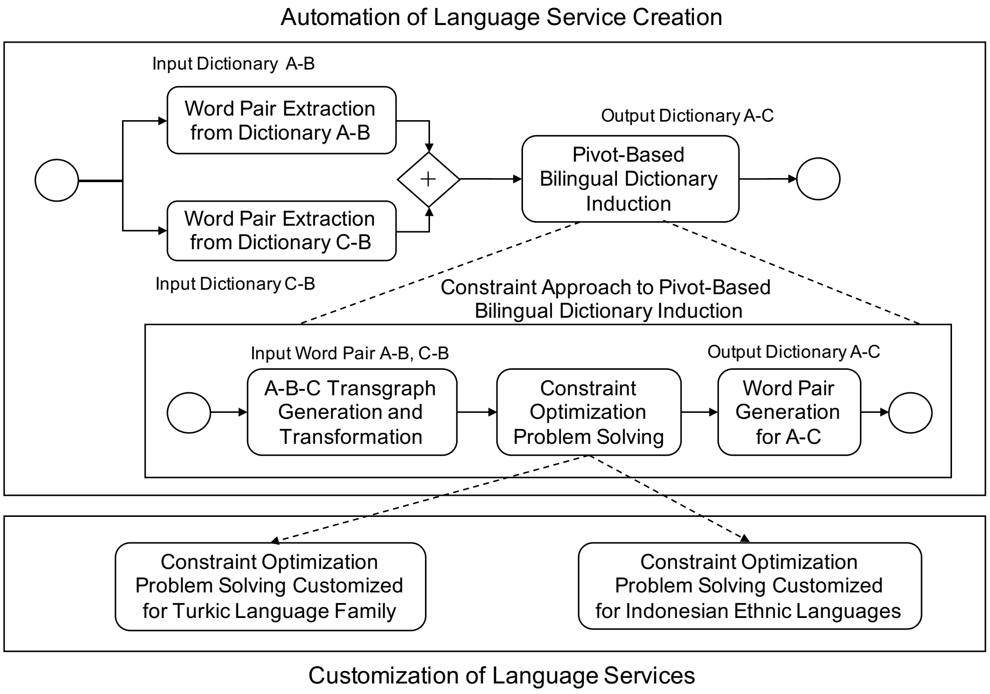
Automation of Language Service Creation
Input Dictionary A-B
Word Pair Extraction from Dictionary A-B
Word Pair Extraction from Dictionary C-B
Input Dictionary C-B
+
Output Dictionary A-C
Pivot-Based Bilingual Dictionary Induction
Constraint Approach to Pivot-Based Bilingual Dictionary Induction
Input Word Pair A-B, C-B
A-B-C Transgraph Generation and Transformation
Constraint Optimization Problem Solving
Output Dictionary A-C
Word Pair Generation for A-C
Constraint Optimization Problem Solving Customized for Turkic Language Family
Constraint Optimization Problem Solving Customized for Indonesian Ethnic Languages
Customization of Language Services

Figure 2. An example of the automation and customization of pivot-based bilingual dictionary induction for low-resource languages.

Automation of language service creation is realized by workflows that combine several atomic or composite services. In the example in Figure 2, we have two dictionaries *A-B* and *C-B* as input, where *A* and *C* are low-resource languages (e.g., Uyghur and Kazakh), while *B* is a major language (e.g., Chinese). We need to output a new dictionary *A-C*. At an abstract level, we need two services: A word pair extraction service, which extracts word pairs from dictionaries, and a pivot-based bilingual dictionary induction service, which executes the induction algorithm and outputs the target dictionary. The pivot-based bilingual dictionary induction service is an abstract composite service; it can be implemented by any algorithm that realizes the specified function. Here we suppose that the user chooses a constraint approach to the pivot-based bilingual dictionary induction service, which consists of three atomic services: A transgraph generation and transformation service, constraint optimization problem solving service, and word pair generation service. The transgraph generation and transformation service generates a graph whose edges represent connections among any three words in A, B, and C; the transgraph is transformed to include new edges that might be missing in original dictionaries A-B and C-B. The constraint optimization problem solving service executes an optimization algorithm to induce the word pairs of A and C. The word pair generation service generates the target dictionary A-C based on the word pairs. Among the three atomic services, the constraint optimization problem solving service can be further customized based on the features of language pairs. For example, the constraints created to model language pairs within the Turkic language family may not be useful for language pairs within the Indonesian ethnic languages. Therefore, users can customize the constraint optimization problem solving service while keeping all other services as they are.

4.2. Service Layers for Language Service Creation and Customization

Section 2 noted that the Language Grid is a language service infrastructure consisting of four service layers that include the service grid, atomic services, composite services, and application systems. We can also use these four service layers for realizing the design concepts of automation and customization for supporting low-resource languages. However, we need to define more specified roles for each service layer, as summarized in Table 3.

Table 3. Layers of language service infrastructure and their roles for low-resource language service creation and customization.

Service Layer	Description
Bottom Layer (Service Grid)	The service grid manages the requests to the infrastructure and invokes language services. In the context of low-resource language services, since it is difficult for one single organization or a small group of organizations to provide enough resources and services, the service grid must be realized in a distributed manner and must enable the coordination of interconnections of grids operated by different organizations.
Second Layer (Atomic Services)	In this layer, users create and register language services by wrapping language resources based on the service interface types. In the Language Grid, we have already defined over 20 types of standardized service interfaces. However, we need to deal with service interoperability if we consider a distributed service grid for low-resource languages, since different operators may have their own policies of providing services and the service ontology definitions.
Third Layer (Composite Services)	In the composite service layer, users create and use service workflows for realizing complicated functions like the example described in Figure 2. Due to the variety of the features exhibited by low-resource languages, the infrastructure must provide composite services with various granularities so that the users can make decisions on how to best balance the automation and customization for language service creation.
Top Layer (Application Systems)	In the application system layer, multilingual applications and intercultural collaboration tools for low-resource languages are developed by utilizing the composite services and provided to end-users. The usage of the composite services can be regarded as an evaluation method for the services provided through the infrastructure. The feedback from the real world can be used to improve existing composite services and design new services for low-resource languages.

5. Case Study: Bilingual Dictionary Induction for Low-Resource Languages

This section uses a real-world case study of bilingual dictionary induction for low-resource languages to illustrate our design concepts. We implemented the ideas described in Figure 2 by realizing the bilingual dictionary induction for Turkic languages and Indonesian ethnic languages (Austronesian low-resource languages).

Bilingual dictionaries are important for low-resource languages since they can assist in the creation or improvement of other language resources, such as systems of machine translation and cross-lingual information retrieval. A common effective approach to bilingual dictionary induction for resource-rich languages is to utilize additional resources such as parallel corpora, comparable corpora [35], parts of speech [36], WordNet [37], and so on. Other studies have proposed the pivot-based approach [38] and cognate recognition [39] for inducing bilingual dictionaries. However, the basic pivot-based approach may generate inaccurate dictionaries due to inconsistency, asymmetry, and intransitivity [40]. To address this issue, context-aware pivot-based approaches [41] have been proposed; they focus on selecting correct word translations by utilizing the context in sentences.

Bilingual dictionary induction is a difficult task for low-resource languages due to the shortage in additional resources like parallel corpora. However, the pivot-based approach becomes a natural candidate for low-resource languages if there are bilingual dictionaries for the low-resource languages and a resource-rich language. To overcome the problems of the existing pivot-based approach, we have proposed a new approach by modeling the pivot-based bilingual dictionary induction as a constraint optimization problem [42]. Since no additional resources can be assumed to be available, all of the constraints are defined based on the structures of the dictionaries and language similarity assumptions. The general bilingual dictionary induction process was described in Figure 2.

5.1. Automation of Bilingual Dictionary Induction for Turkic Languages

We started by creating an Uyghur–Kazakh bilingual dictionary from two available dictionaries: An Uyghur–Chinese dictionary (52,478 Chinese words, 70,989 Uyghur words, and 118,805 word pairs) and a Kazakh–Chinese dictionary (52,478 Chinese words, 102,426 Kazakh words, and 232,589 word pairs). Kazakh and Uyghur are Turkic languages and the lexicons of the two languages overlap by 81.9%, based on a classical lexicostatistical study [43]. From this fact, we made the assumption that *lexicons of intra-family languages offer one-to-one relationships*. We first generated the transgraph that connects all word pairs in the two dictionaries. Then, we defined the following constraints in the constraint optimization problem and implemented the solution, which is described as the atomic service of *Constraint Optimization Problem Solving Customized for the Turkic Language Family* in Figure 2.

- **Constraint 1**: A pair of words, w_i^A (a word i in language A) and w_j^C (a word j in language C), in a transgraph can be a one-to-one pair candidate if they are connected via at least one pivot word.
- **Constraint 2**: Given a pair of words, w_i^A and w_j^C, in a transgraph, if they are a one-to-one pair, then they should be symmetrically connected through pivot words.
- **Constraint 3**: Given a pair of words, w_i^A and w_j^C, in a transgraph, if they are a one-to-one pair, then they should be unique, such that no other candidates involving w_i^A and w_j^C are one-to-one pairs.
- **Constraint 4**: In a transgraph, at least one one-to-one pair should be extracted.

We finally generated an Uyghur–Kazakh bilingual dictionary with 43,615 word pairs, which is 84.2% of the theoretical maximum of word pairs that can be obtained. Moreover, the generated dictionary attained 83.7% precision, approximately 10% higher than baseline methods. Details of the constraint optimization problem formalization, solutions, and experiment results can be found in our previous paper [43]. We then applied and extended the automated dictionary induction service to generate a Kazakh–Kyrgyz dictionary and an Uyghur–Kyrgyz dictionary. Note that the three machine-generated dictionaries, Kazakh–Kyrgyz (https://langrid.org/service_manager/language-services/profile/kyoto1.langrid/KazakhKyrgyzDictionary), Uyghur–Kazakh (https://langrid.org/service_manager/language-services/profile/kyoto1.langrid/UyghurKazakhDictionary), and Uyghur–Kyrgyz (https://langrid.org/service_manager/language-services/profile/kyoto1.langrid/UyghurKyrgyzDictionary), have been registered and shared in the Language Grid.

5.2. Customization of Bilingual Dictionary Induction for Indonesian Ethnic Languages

We tried to apply the automated process developed for Turkic languages to Indonesian ethnic languages (Austronesian low-resource languages). Our target was to generate a Minangkabau–Riau Mainland Malay dictionary from a Minangkabau–Indonesian dictionary and a Riau Mainland Malay–Indonesian dictionary. However, we found that the assumption of *one-to-one mapping* greatly reduced the number of word pairs that could offset resource paucity [34]. This meant that we should customize the existing atomic service of *Constraint Optimization Problem Solving* to suit other low-resource languages.

Therefore, we proposed another constraint-based bilingual dictionary induction service by defining nine constraints from the recent pivot-based induction technique while also enabling the multiple symmetry assumption, which is described as the atomic service of *Constraint Optimization Problem Solving Customized for Indonesian Ethnic Languages* in Figure 2. This approach was also tested on three Indo-European high-resource languages to illustrate its generality. Experiments showed that the proposed approach offered a statistically significant improvement in precision and F-score over existing constraint-based methods, including the *Constraint Optimization Problem Solving Customized for the Turkic Language Family*. Details of the defined constraints, the optimization approach, and the experimental results can be found in our previous paper [34].

Currently, the work of bilingual dictionary induction of Indonesian ethnic languages is continuing as a project called Indonesia Language Sphere [44]. We also extended our work by including humans in the loop for the collaborative creation of bilingual dictionaries for Indonesian ethnic languages [45].

6. Conclusions

We proposed a language service framework for low-resource languages that enables the automatic creation and customization of new resources from existing ones. We first described the Language Grid, a service-oriented language infrastructure for sharing and combining language resources as language services. Since the Language Grid has already commenced federated operation with Asian partners to provide low-resource language services, it can be a basic infrastructure for low-resource languages. We used detailed examples to describe how we could realize the automation and customization of low-resource language services. Finally, we detailed a case study of automating and customizing bilingual dictionary induction for low-resource Turkic languages and Indonesian ethnic languages to illustrate our design concepts.

Author Contributions: Conceptualization, D.L., Y.M., and T.I.; methodology, D.L., Y.M., and T.I.; software, D.L. and Y.M.; validation, D.L. and Y.M.; formal analysis, D.L.; investigation, D.L.; resources, D.L. and Y.M.; data curation, D.L.; writing—original draft preparation, D.L.; writing—review and editing, D.L., Y.M., and T.I.; visualization, D.L.; supervision, T.I.; project administration, D.L.; funding acquisition, D.L., Y.M., and T.I. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially supported by a Grant-in-Aid for Scientific Research (A) (17H00759, 2017-2020), a Grant-in-Aid for Scientific Research (B) (18H03341, 2018-2020), and a Grant-in-Aid for Young Scientists (A) (17H04706, 2017–2020) from the Japan Society for the Promotion of Science (JSPS).

Acknowledgments: The authors would like to thank Mairidan Wushouer and Arbi Haza Nasution for their empirical studies of bilingual dictionary induction of Turkic languages [43] and Indonesian ethnic languages [34] based on the concept described in this research.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Nettle, D. Explaining global patterns of language diversity. *J. Anthropol. Archaeol.* **1998**, *17*, 354–374. [CrossRef]
2. List of Wikipedias. Available online: https://meta.wikimedia.org/wiki/List_of_Wikipedias (accessed on 28 November 2019).

3. LRE Map. Available online: http://lremap.elra.info (accessed on 28 November 2019).
4. Calzolari, N.; Del Gratta, R.; Francopoulo, G.; Mariani, J.; Rubino, F.; Russo, I.; Soria, C. The LRE Map. harmonising community descriptions of resources. In Proceedings of the Eighth International Conference on Language Resources and Evaluation, Istanbul, Turkey, 23–25 May 2012; pp. 1084–1089.
5. Google Translate. Available online: http://translate.google.com/ (accessed on 28 November 2019).
6. Del Gratta, R.; Frontini, F.; Khan, A.F.; Mariani, J.; Soria, C. The LREMap for under-resourced languages. In Proceedings of the Workshop on Collaboration and Computing for Under-Resourced Languages in the Linked Open Data Era, Reykjavik, Iceland, 26 May 2014; p. 78.
7. Zoph, B.; Yuret, D.; May, J.; Knight, K. Transfer learning for low-resource neural machine translation. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 1568–1575.
8. Gu, J.; Hassan, H.; Devlin, J.; Li, V.O. Universal neural machine translation for extremely low resource languages. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 344–354.
9. Tiedemann, J. Character-based pivot translation for under-resourced languages and domains. In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, 23–27 April 2012; Association for Computational Linguistics: Stroudsburg, PA, USA, 2012; pp. 141–151.
10. Farhath, F.; Theivendiram, P.; Ranathunga, S.; Jayasena, S.; Dias, G. Improving domain-specific SMT for low-resourced languages using data from different domains. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation, Miyazaki, Japan, 7–12 May 2018; pp. 3789–3794.
11. Honnet, P.E.; Popescu-Belis, A.; Musat, C.; Baeriswyl, M. Machine translation of low-resource spoken dialects: Strategies for normalizing Swiss German. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation, Miyazaki, Japan, 7–12 May 2018, pp. 3781–3788.
12. Alonso, H.M.; Schluter, N.; Søgaard, A. Multilingual projection for parsing truly low-resource languages. *Trans. Assoc. Comput. Linguist.* **2016**, *4*, 301–312.
13. Garrette, D.; Mielens, J.; Baldridge, J. Real-world semi-supervised learning of POS-taggers for low-resource languages. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, Volume 1 (Long Papers), Sofia, Bulgaria, 4–9 August 2013; pp. 583–592.
14. Duong, L.; Cohn, T.; Bird, S.; Cook, P. A neural network model for low-resource universal dependency parsing. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 339–348.
15. Lim, K.; Partanen, N.; Poibeau, T. Multilingual dependency parsing for low-resource languages: Case studies on North Saami and Komi-Zyrian. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation, Miyazaki, Japan, 7–12 May 2018, pp. 2230–2235.
16. Gales, M.J.; Knill, K.M.; Ragni, A.; Rath, S.P. Speech recognition and keyword spotting for low-resource languages: BABEL project research at CUED. In Proceedings of the Workshop on Spoken Language Technologies for Under-Resourced Languages, St. Petersburg, Russia, 14–16 May 2014.
17. Wang, H.; Ragni, A.; Gales, M.; Knill, K.; Woodland, P.; Zhang, C. Joint decoding of tandem and hybrid systems for improved keyword spotting on low resource languages. In Proceedings of the 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, 6–10 September 2015; pp. 3660–3664.
18. Adams, O.; Makarucha, A.; Neubig, G.; Bird, S.; Cohn, T. Cross-lingual word embeddings for low-resource language modeling. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1 (Long Papers), Valencia, Spain, 3–7 April 2017; pp. 937–947.
19. Andrews, N.; Dredze, M.; Van Durme, B.; Eisner, J. Bayesian modeling of lexical resources for low-resource settings. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics: Volume 1 (Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 1029–1039.
20. Irvine, A.; Klementiev, A. Using Mechanical Turk to annotate lexicons for less commonly used languages. In Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, Los Angeles, CA, USA, 6 June 2010; Association for Computational Linguistics: Stroudsburg, PA, USA, 2010, pp. 108–113.

21. De Vries, N.J.; Davel, M.H.; Badenhorst, J.; Basson, W.D.; De Wet, F.; Barnard, E.; De Waal, A. A smartphone-based ASR data collection tool for under-resourced languages. *Speech Commun.* **2014**, *56*, 119–131. [CrossRef]
22. Fraisse, A.; Jenn, R.; Fishkin, S.F. Building multilingual parallel corpora for under-resourced languages using translated fictional texts. In Proceedings of the 3rd Workshop on Collaboration and Computing for Under-Resourced Languages: Sustaining Knowledge Diversity in the Digital Age, Miyazaki, Japan, 12 May 2018; pp. 39–43.
23. Fraisse, A.; Zhang, Z.; Zhai, A.; Jenn, R.; Fisher Fishkin, S.; Zweigenbaum, P.; Favier, L.; Mustafa El Hadi, W. A sustainable and open access knowledge organization model to preserve cultural heritage and language diversity. *Information* **2019**, *10*, 303. [CrossRef]
24. Ishida, T. *The Language Grid: Service-Oriented Collective Intelligence for Language Resource Interoperability*; Springer: Berlin, Germany, 2011.
25. Murakami, Y.; Lin, D.; Ishida, T. *Services Computing for Language Resources*; Springer: Singapore, 2018.
26. Murakami, Y.; Lin, D.; Tanaka, M.; Nakaguchi, T.; Ishida, T. Service Grid architecture. In *The Language Grid: Service-oriented Collective Intelligence for Language Resource Interoperability*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 19–34.
27. Ishida, T.; Murakami, Y.; Lin, D.; Nakaguchi, T.; Otani, M. Language service infrastructure on the Web: The Language Grid. *Computer* **2018**, *51*, 72–81. [CrossRef]
28. Ishida, T.; Murakami, Y.; Lin, D. The Language Grid: Service-oriented approach to sharing language resources. In *The Language Grid: Service-Oriented Collective Intelligence for Language Resource Interoperability*; Springer: Berlin, Germany, 2011; pp. 3–17.
29. Lin, D.; Murakami, Y.; Ishida, T. A framework for multi-language service design with the Language Grid. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation, Miyazaki, Japan, 7–12 May 2018, pp. 3276–3281.
30. Murakami, Y.; Lin, D.; Ishida, T. Service-oriented architecture for interoperability of multilanguage services. In *Towards the Multilingual Semantic Web*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 313–328.
31. Murakami, Y.; Nakaguchi, T.; Lin, D.; Ishida, T. Federated grid architecture for language services. In *Services Computing for Language Resources*; Springer: Singapore, 2018; pp. 3–20.
32. Wushouer, M.; Ishida, T.; Lin, D. A heuristic framework for pivot-based bilingual dictionary induction. In Proceedings of the 2013 International Conference on Culture and Computing, Kyoto, Japan, 16–18 September 2013; pp. 111–116.
33. Wushouer, M.; Ishida, T.; Lin, D.; Hirayama, K. Bilingual dictionary induction as an optimization problem. In Proceedings of the Ninth International Conference on Language Resources and Evaluation, Reykjavik, Iceland, 26–31 May 2014; pp. 2122–2129.
34. Nasution, A.H.; Murakami, Y.; Ishida, T. A generalized constraint approach to bilingual dictionary induction for low-resource language families. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2018**, *17*, 9. [CrossRef]
35. Kaji, H.; Tamamura, S.; Erdenebat, D. Automatic construction of a Japanese-Chinese dictionary via English. In Proceedings of the Sixth International Conference on Language Resources and Evaluation, Marrakech, Morocco, 28–30 May 2008; pp. 699–706.
36. Bond, F.; Ogura, K. Combining linguistic resources to create a machine-tractable Japanese-Malay dictionary. *Lang. Resour. Eval.* **2008**, *42*, 127–136. [CrossRef]
37. István, V.; Shoichi, Y. Bilingual dictionary generation for low-resourced language pairs. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Singapore, 6–7 August 2009; Association for Computational Linguistics: Stroudsburg, PA, USA, 2009; Volume 2, pp. 862–870.
38. Tanaka, K.; Umemura, K. Construction of a bilingual dictionary intermediated by a third language. In Proceedings of the 15th Conference on Computational Linguistics, Kyoto, Japan, 5–9 August 1994; Association for Computational Linguistics: Stroudsburg, PA, USA, 1994; Volume 1, pp. 297–303.
39. Mann, G.S.; Yarowsky, D. Multipath translation lexicon induction via bridge languages. In Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies, Pittsburgh, PA, USA, 2–7 June 2001; Association for Computational Linguistics: Stroudsburg, PA, USA, 2001; pp. 1–8.
40. Tanaka, R.; Murakami, Y.; Ishida, T. Context-based approach for pivot translation services. In Proceedings of the 21st International Joint Conference on Artificial intelligence, Pasadena, CA, USA, 11–17 July 2009; pp. 1555–1561.

41. Matsuno, J.; Ishida, T. Constraint optimization approach to context based word selection. In Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Spain, 16–22 July 2011; pp. 1846–1851.
42. Wushouer, M.; Lin, D.; Ishida, T.; Hirayama, K. Pivot-based bilingual dictionary extraction from multiple dictionary resources. In *2014 Pacific Rim International Conference on Artificial Intelligence*; Springer: Cham, Switzerland, 2014; pp. 221–234.
43. Wushouer, M.; Lin, D.; Ishida, T.; Hirayama, K. A constraint approach to pivot-based bilingual dictionary induction. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2016**, *15*, 4. [CrossRef]
44. Murakami, Y. Indonesia Language Sphere: An ecosystem for dictionary development for low-resource languages. *J. Phys. Conf. Ser.* **2019**, *1192*, 012001. [CrossRef]
45. Nasution, A.H.; Murakami, Y.; Ishida, T. Designing a collaborative process to create bilingual dictionaries of Indonesian ethnic languages. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation, Miyazaki, Japan, 7–12 May 2018; pp. 3397–3404.