Data & Knowledge Engineering xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Data & Knowledge Engineering

journal homepage: www.elsevier.com/locate/datak

Constructing a paraphrase database for agglutinative languages

Hancheol Park[a], Kyo-Joong Oh[a], Ho-Jin Choi[a], Gahgene Gweon[b,*]

[a] School of Computing, KAIST, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea
[b] Department of Transdisciplinary Studies, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 08826, Republic of Korea

ARTICLE INFO

Keywords:
Affix modification-based bilingual pivoting method
Paraphrase database
Text mining
Paraphrase generation
Question-answering systems

ABSTRACT

Paraphrase databases (PPDBs) are valuable resources for applications that use natural language processing (NLP) technology. In order to construct a high-quality PPDB for agglutinative languages, we propose a phrasal paraphrase extraction method; namely, affix modification-based bilingual pivoting method (AMBPM). AMBPM is suitable for agglutinative languages because it addresses the problems of lexical data sparsity and of not considering morphological word structure. In addition, we propose "improved AMBPM", which is an improvement on AMBPM by addressing the problem of extracting incorrect stem paraphrase pairs caused by low semantic content stems (LSCSs) by using a rule-based filtering approach. In our experiments on AMBPM, we evaluate AMBPM and compare two state-of-the-art paraphrase extraction methods: the syntactic constraints-based bilingual pivoting method (SCBPM) and word embedding method. In the experiments on improved AMPBM, we evaluate our method and compare the resulting PPDB with four types of databases; PPDB constructed by using the original AMBPM, two PPDBs constructed by using two types of word-embedding-based methods (stem embedding and phrase embedding), and an existing thesaurus. The comparison is performed by using two NLP applications: sentential paraphrase generation and a question answering (QA) system. The experimental results demonstrate that, AMBPM outperforms the state-of-the-art paraphrase extraction methods. In addition, the improved AMBPM, which uses a rule-based filtering method, significantly improves AMBPM. Moreover, although a small amount of training data was used with no aid from linguistic resources, the PPDB constructed with the improved AMBPM is more useful than the four databases for the agglutinative language used in our study. We also publicized the Korean PPDB that was constructed using the improved AMBPM.

1. Introduction

A paraphrase database (PPDB) is a collection of paraphrase expression pairs [1–3]. Each paraphrase pair consists of a *source*, which is the expression to be paraphrased, and *target*, which is the paraphrased expression of the source. For example, < "return to my home", "come back to my home" > is a paraphrase pair in which the former expression in the angle bracket is the source and latter one is the target. PPDBs have been widely used to improve the performance of various natural language processing (NLP) applications. For example, the database could be used to mitigate word mismatch problems when a question answering (QA) system retrieves the answer patterns from a corpus [4–6], or solve out-of-vocabulary problems in the source language when a statistical machine translation (SMT) system looks over translated expressions for source phrases [7]. In addition, PPDB has been used to generate alternative sentences that convey the same meaning as input sentences in sentential paraphrase generation [8,9] and sentence summarization [8,10] systems.

* Corresponding author.
E-mail addresses: hancheol.park@kaist.ac.kr (H. Park), aomaru@kaist.ac.kr (K.-J. Oh), hojinc@kaist.ac.kr (H.-J. Choi), ggweon@snu.ac.kr (G. Gweon).

http://dx.doi.org/10.1016/j.datak.2017.07.007

One type of PPDB that is used in NLP applications is a thesaurus. The entities that compose the thesaurus are highly accurate because they are manually constructed by lexicographers based on linguistic knowledge. Nevertheless, a thesaurus has several limitations for use in NLP applications. First, because synonym sets in thesauruses are paraphrase pairs at the word-level, managing cases where phrases have a meaning that is not a simple composition of the meaning of its individual words, such as idiomatic expressions, is difficult. Second, measuring the semantic closeness between a source and target is difficult because a similarity measure is not specified in thesauruses. A third limitation is that constructing a thesaurus is labor-intensive. For those reasons, researchers have recently focused on constructing PPDBs using data-driven paraphrase extraction methods. Such extraction methods do not only extract phrasal paraphrase pairs, but also generate a similarity measure between a source and target based on the distribution of those phrases in the training corpus. Moreover, data-driven extraction methods help us to easily construct PPDBs by using automated computational algorithms on large-scale corpora.

To date, various data-driven paraphrase extraction methods have been proposed for constructing PPDBs [1–3,11,12]. Those methods are mostly centered on morphologically poor languages, such as isolating (e.g., Chinese) and fusional (e.g., German) languages. Although studies on methods for languages that have the traits of both isolating and fusional languages, such as English, have been dominant, methods that can be applied to morphologically complex languages, such as agglutinative languages (e.g., Korean and Japanese) are relatively rare. Compared to morphologically poor languages, agglutinative languages are more complex in terms of the composition of morphemes in each word. For example, agglutinative languages inflect a word by attaching inflectional affixes to the word-stem to represent grammatical categories (e.g., case, tense, person, and number) of the word. Although several studies [12–14] on Japanese, one of the agglutinative languages, have been proposed, these methods are limited to predicate phrasal paraphrase pairs. In addition, those methods are not generically applicable to other agglutinative languages because many of the features used are Japanese language-dependent. Because such Japanese paraphrase extraction methods are not generalized for other agglutinative languages, in this paper, we propose a generalizable method for agglutinative languages by using the common features present in agglutinative languages.

A common feature of agglutinative languages is that they naturally produce a larger amount of vocabulary compared with other languages because a stem can be represented in various word-forms, based on the grammatical categories. That is to say, a word in agglutinative languages is composed of a stem that represents the concrete meaning of the word, and inflectional affixes that represent the grammatical categories of the word. In comparison, morphologically poor languages contain a relatively smaller vocabulary size because they exploit the word order or make use of a few inflectional morphemes in order to represent grammatical categories. Because of the smaller vocabulary size in morphologically poor languages, existing methods that model each word-form as a separate word seem reasonable. However, such a modeling approach leads to two major issues when applied to agglutinative languages. First, the issue of lexical data sparsity occurs because the larger vocabulary size in agglutinative languages reduces the frequency of each word-form within a corpus. Considering that the paraphrase extraction methods are a special case of unsupervised learning, lexical data sparsity could lead to difficulties in inferring semantically equivalent phrases. Second, the issue of neglecting to consider the morphological word structure occurs because the existing methods do not decompose and analyze the internal structure of words. Instead, the existing methods model each word-form. This complicates the extraction of grammatically equivalent phrases in agglutinative languages because the inflectional affixes within each word are not considered.

To address the two aforementioned issues, we propose a phrasal paraphrase extraction method for agglutinative languages, namely, the affix modification-based bilingual pivoting method (AMBPM). AMBPM is an expansion of a well-developed paraphrase extraction method, the bilingual pivoting method (BPM) [15]. AMBPM models a stem as a separate word, instead of a word-form, to address the problem of lexical data sparsity. This modeling approach improves the meaning preservation between a source and target. This method also modifies the inflectional affixes of phrases in a paraphrase pair such that they represent the same grammatical categories. This modification process also improves the grammatical equivalence of each pair by addressing the problem of not considering the morphological word structure. AMBPM is designed to be a generic method for agglutinative languages in that it considers only the common features of agglutinative languages and does not rely on the individual traits of a specific language. In our experiments on AMBPM using the same size of training data, we demonstrate that PPDBs constructed by using AMBPM significantly outperforms PPDBs from two state-of-the-art paraphrase extraction methods, namely the syntactic constraints-based bilingual pivoting method (SCBPM) [1,3,16] and the word embedding model [17] with respect to meaning preservation and grammaticality. From the error analysis of the paraphrase pairs extracted by AMBPM, we observe two major error patterns, errors from word alignment and those from a low semantic content stem (LSCS). LSCS is a stem that contains weak semantic content (e.g., a word-stem that provides functional or grammatical meaning rather than semantic meaning, such as light and auxiliary verbs in English and bound nouns in Korean). Errors from word alignment are cases when two expressions that have different semantic meanings are aligned as a paraphrase pair. Errors from LSCSs occur when either the source or target has an additional LSCS. It also occurs when the source and target in a paraphrase pair have different types of LSCSs. These errors from LSCSs cause a source and target to either have meanings with subtle differences or different grammatical categories. Although the word alignment error has been addressed by increasing the size of the corpora [15] or using more informative features [3,11], to our best knowledge, the problem caused by LSCSs have not been addressed yet. Therefore, in this paper, we address the problem of extracting incorrect stem paraphrase pairs caused by LSCSs.

In addition to AMBPM, we propose “improved AMBPM”, which addresses the problem of LSCSs identified from the error analysis of AMBPM, by using a rule-based filtering approach. Because the purpose of this improvement is applying the PPDB constructed by using our method for NLP applications, we evaluate the usefulness of the PPDB by using NLP applications, namely *sentential paraphrase generation* and *QA systems*. We evaluate our method and compare the resulting PPDB with four types of databases; PPDB constructed by using the original AMBPM, two PPDBs constructed by using two types of word-embedding-based

methods (stem embedding and phrase embedding), and an existing thesaurus. The experimental results demonstrate that the improved AMBPM, which uses a rule-based filtering method, significantly improves AMBPM. Moreover, even though a small amount of training data is used without the aid of linguistic resources, the PPDB constructed with the improved AMBPM is more useful than the four databases for the agglutinative language used in our study.

In addition, we publicized the Korean PPDB constructed with the improved AMBPM[1] as a public resource. The remainder of this paper is organized as follows: in Section 2, we review the two major approaches for constructing PPDBs. Section 3 describes AMBPM and Section 4 describes the improved AMBPM, each with corresponding experiments and results. We conclude the paper in Section 5.

2. Related work

In this section, we review two major approaches used by state-of-the-art phrasal paraphrase extraction methods, namely the *bilingual pivoting approach* and *distributional hypothesis-based approach*.

2.1. Bilingual pivoting approach

In the bilingual pivoting approach, we can consider two phrases as a paraphrase pair if the phrases are translated to be the same pivot, which is the same foreign expression as shown in Fig. 1 [15]. The major techniques used in this approach are word [18] and phrase alignment [19] techniques that are used in phrase-based SMTs. These techniques infer the alignment relationship between source (e.g., "텔레비전을 시청하는" or "TV를 보는" in Fig. 1) and foreign expressions (e.g., "watching television" in Fig. 1) in each bilingual sentence pair based on the count of candidate alignment pair co-occurrence throughout the entire corpus. The alignment techniques allow us to discover semantically aligned word and phrase pairs from bilingual parallel sentences. Bannard and Callison-Burch [15] first proposed BPM and defined the paraphrase likelihood between two phrases as a conditional paraphrase probability, as follows:

$$P(k_2|k_1) = \sum_f P_{MLE}(k_2|f) P_{MLE}(f|k_1) \tag{1}$$

where k_1 and k_2 are the Korean phrases and f is the English phrase. Each conditional probability P_{MLE} is the phrase translation probability estimated by the word and phrase alignment techniques. Those conditional probabilities can be calculated by maximum likelihood estimation (MLE) based on the count of the alignment of two phrases. After BPM was initially proposed, researchers improved this method and proposed SCBPM [1,3,16] by constraining the syntactic type (e.g., part-of-speech and phrase type of words and phrases) between two phrases so that they are identical.

Although BMP and SCBPM are innovative, they are not sufficient for application to agglutinative languages for two reasons. First, the problem of lexical data sparsity occurs because BPM and SCBPM model each word-form as a separate word. This causes high alignment error rate between two languages when conducting word and phrase alignment if one of the languages is an agglutinative language. Second, the problem of not considering the morphological word structure occurs. This leads to difficulty in identification of grammatically equivalent phrases in the agglutinative languages because BPM does not consider the affixes. Although SCBPM attempts to consider internal word structures by managing syntactic types of words or phrases presented by the internal morphemes of each word, it is insufficient for several reasons. Depending on the standard part-of-speech tag set for an individual language, SCBPM performance can vary. For example, although the Korean standard tag set, Sejong, includes a label for an affix (as in the tense pre-final ending), the label does not explicitly tell us whether the ending indicates past or future tense. This makes it difficult to extract paraphrase pairs that contain the same tense when using SCBPM. From a linguistic perspective, we also find that syntactic types do not address all grammatical categories. Therefore, in agglutinative languages, modifying inflectional affixes such that two phrases can indicate exactly the same grammatical categories by containing the same inflectional affixes is more efficient. Moreover, the presence of syntactic constraints in SCBPM significantly reduces the paraphrase coverage by filtering out the paraphrase pairs that indicate syntactic types that are different from each other [16]. Because of the large number of inflected variants of phrases in agglutinative languages, NLP applications, such as SMT and sentential paraphrase generation on agglutinative languages, may suffer from a high out-of-vocabulary problem when using PPDBs constructed by using SCBPM. We address the two aforementioned issues, i.e., the problems of lexical data sparsity and of not considering the morphological word structure by introducing AMBPM in Section 3, which is an extension of BPM. In this paper, we also present an improved AMBPM as an extension to AMBPM in Section 4.

2.2. Distributional hypothesis-based approach

In the distributional hypothesis-based approach, we can consider two phrases as a paraphrase pair if they frequently share the same contextual words in a monolingual single corpus. These assumptions are derived from Harris' distributional hypothesis [20]. The distributional hypothesis-based approach has been conventionally used to model a textual form of words or phrases [4,21]. Recently developed methods adopt this approach to train words or phrases as a low-dimensional dense vector using neural language models [17,22,23]. Such models are neural network models that predict the next word given the previously occurring 2–10 words.

[1] The Korean PPDB is available at https://sites.google.com/site/tonyhanpark/resources.

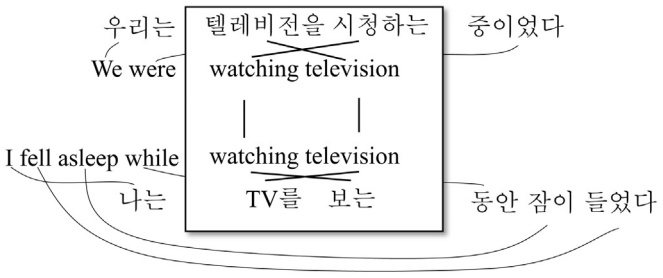

Fig. 1. BPM example.

That is to say, in neural network architectures, input nodes represent contextual words and output nodes are words or phrases that share the context. This is known as a "word embedding model". After training the model, we can easily calculate the paraphrase likelihood between two words or phrases using the cosine similarity of two embeddings.

One advantage of this approach is that one can easily obtain additional training data by using the distributional hypothesis-based approach, whereas using the bilingual pivoting method (BPM) requires bilingual parallel corpora, which is difficult to obtain in a large volume. Yet, we chose to use BPM since using the distributional hypothesis-based approach makes it difficult to distinguish real paraphrase pairs from other pairs that have different relations, such as antonyms [1]. Our approach also addresses the problems of lexical data sparsity and of not considering the morphological word structure, which are problems that exist in both the distributional hypothesis-based approach as well as BPM.

In the experiment in Section 3, we compare AMBPM with the word embedding method. We use a dataset of the same size for a fair comparison. Although the results show that AMBPM outperforms the word embedding model, one could argue that the setting is not realistic because it is relatively easy to obtain a large amount of data for use in the word embedding model, thus one would not use the same amount of data for both methods. Therefore, in our experiment in Section 4, we use a larger sized dataset to train word embedding-based paraphrase extraction methods compared to AMBPM. In addition, to provide further advantage for word embedding-based methods, we also remove paraphrase pairs that have antonym relations, and pairs with differently named entities and parts-of-speech between the source and target in the PPDBs.

3. Affix modification-based bilingual pivoting method

In this section, we introduce the affix modification-based bilingual pivoting method (AMBPM) to construct a high quality PPDB for agglutinative languages. For AMBPM, we assume that a phrase can be divided into several smaller meaning units (SMUs), which comprise the entire meaning of the phrase. Here, a SMU can be a single stem in a word or a sequence of stem groups in an idiomatic expression/ compound words. If all semantically aligned SMU pairs between a source and target have the same meaning and inflectional affixes, they can substitute each other, regardless of whether each is a well-formed phrase. As a result, the combination of those aligned SMU pairs can be said to be semantically and grammatically equivalent and are a substitutable paraphrase pair. In the following subsections, we detail AMBPM (3.1), the ranking model of the paraphrase pairs extracted with AMBPM (3.2), and evaluate AMBPM in an experiment (3.3).

3.1. AMBPM

The AMBPM proceeds through three phases to extract semantically and grammatically equivalent and directly interchangeable phrases in sentences of agglutinative languages, as shown in Fig. 2. In this study, we use a bilingual parallel corpora composed of the English and Korean languages, where the latter is an agglutinative language.

3.1.1. Phase 1: extracting stem paraphrase pairs using BPM

During the first phase, we extract stem paraphrase pairs composed of stem phrases. These two stem phrases are semantically equivalent and have the same semantic sequence. In this step, we solve the problem of the lexical data sparsity of an agglutinative language by modeling the word-stems instead of the word-forms. To extract Korean stem paraphrase pairs, (1) we remove all inflectional affixes of each word-form in the Korean language from the bilingual parallel corpora. Removal of the affixes mitigates the lexical data sparsity problem when conducting word and phrase alignment because this increases the frequency of each stem. Although there are several methods for dividing the stems and inflectional affixes, such as using stemmers, we identify the inflectional affixes of each word in the Korean language using part-of-speech tags obtained from the results of morphological analysis. (2) We align the stemmed Korean words with the corresponding English words using the standard word alignment technique [18]. We can also find phrase-level alignments using those word alignment results and phrase extraction heuristics [19]. The aligned Korean and English phrases are then stored in the form of a translation table. (3) We apply BPM to extract the Korean stem paraphrase pairs from the translation table. The extracted Korean stem paraphrase pairs are composed of the source (S − 1 + S − 2) and target (T − 1 + T − 2), as shown in Fig. 2.

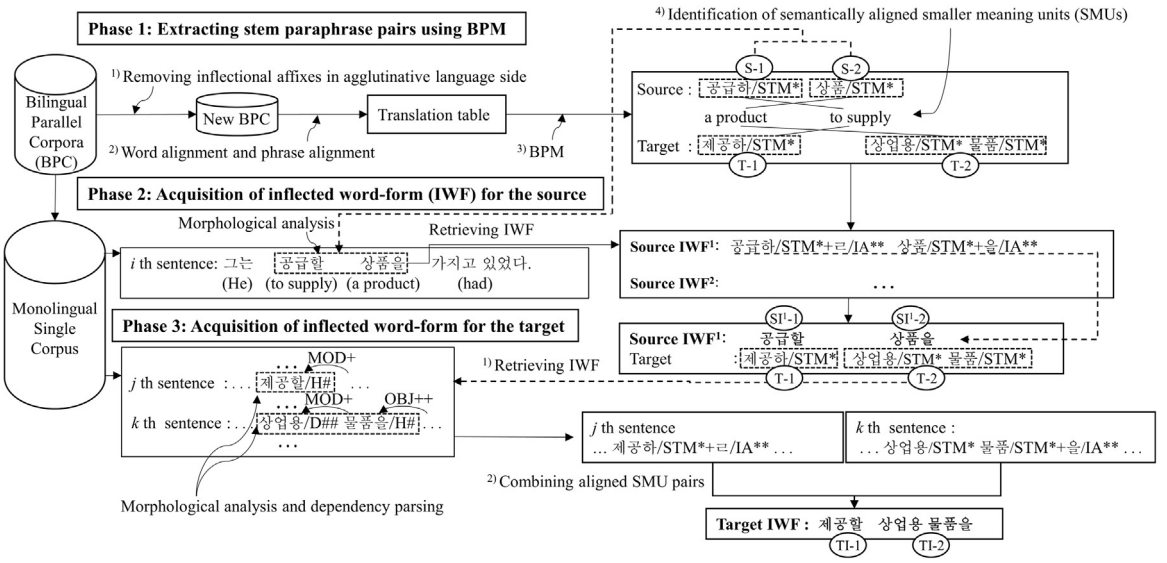

Fig. 2. Overview of AMBPM process. <S − 1, T − 1> and <S − 2, T − 2> are semantically aligned SMUs. STM* = stem of a word, IA** = inflectional affix of a word, MOD+ = modifier, OBJ++ = object, H# = head of a catena, and D## = dependent of a catena. The Korean inflectional affix "ㄹ" is an ending for converting a stem into a modifier and "을" is the object case marker.

In the final step, (4) we identify the semantically aligned SMU using pivot granularities. Each semantically aligned SMU pair does not share the translation of another pair, and has a partial meaning of the entire phrase. In Fig. 2, the semantically aligned SMUs pairs are <S − 1, T − 1> and <S − 2, T − 2> because they share the same pivot granularities, "to supply" and "a product", and those pivots do not overlap. Because various translations can exist between a stem paraphrase pair when we discover the semantically aligned SMU pairs, we simply use the translation for which the stem paraphrase probability is the highest, which is calculated by the paraphrase probability used in Bannard and Callison-Burch's research [15]. The stem paraphrase pairs obtained at the end of phase 1 are then examined in phase 2 and 3 if the stems of the source and target phrases have the same semantic sequence in addition to equivalent semantic meaning. This is because after each semantically aligned SMU pair obtains the same inflectional affixes in the affix modification process (phase 2 and 3), the difference in the order of SMUs results in semantic and grammatical inequivalence between two phrases. Therefore, such phrase pairs are no longer substitutable. For example, let us consider the sentence, "we have *a product to supply* to the department store". In this sentence, the phrase "a product to supply (공급할 상품을)" cannot be substituted with the phrase "to supply a product (상품을 공급할)". This example phrase is composed of two Korean words, " 상품을 (a product)" and " 공급할 (to supply)". The word " 상품 을(a product)", acts as an object since an inflectional affix "을" (object case maker) is attached to the Korean stem " 상품 (a product)", making the combined word "상품 + 을" to be an object. On the other hand, the word " 공급 할 (to supply)" would have a different role in a sentence depending on its position in a sentence. Despite the same word-form, when the word " 공급할 (to supply)" is placed after the word "product", "to supply" modifies the object in a sentence, whereas when the word "to supply" is placed before the "a product", "to supply" is modified by the object in a sentence. Therefore, the two phrases "a product to supply" and "to supply a product" are not substitutable because they have a different meaning from each other. On the other hand, phrases with the same semantic sequence as "a product to supply", or "a commercial item to provide", can be substituted.

3.1.2. Phase 2 and 3: acquisition of inflected word-form (IWF) for the source and target

Phase 2 and 3 solve the problem associated with refraining to consider the morphological word structure by modifying the inflectional affixes of each of the semantically aligned SMU pairs such that they contain the same inflectional affixes. The affix modification process improves the grammaticality of a sentence when substituting the sources in the original sentences with the corresponding targets because the source and target have the same inflectional affixes, which means that the source and target belong to the same grammatical category. Through the two phases, we transform the sources and targets in the stem paraphrase pairs into inflected word-forms (IWFs), which is the form that we actually use in the sentences we speak and write. A sequence of stems (i.e., a stem phrase) is not sufficient for becoming the perfect element of a sentence. Therefore, attaching inflectional affixes to stems is required. One could insist that it is more efficient to perform the affixation after selecting the best stem paraphrase in a sentential paraphrase generation system. However, affixation is a time-consuming process. If we paraphrase a sentence in the QA system, such post-affixation process would not be appropriate because this system is required to obtain an answer within a very short time. Therefore, conducting such an affixation process in advance is important for constructing PPDBs for NLP applications. In this study, we refer to affixation as the "affix modification process" because in the third phase, we modify the inflectional affixes of a target based on the IWF of the corresponding source.

In phase 2, to obtain such grammatical variants by attaching inflectional affixes into the source of a stem paraphrase pair, we retrieve the phrases that contain the stem sequence of the source in the stem paraphrase pairs from the Korean language of the

bilingual parallel corpora. In this phase, we use the original bilingual corpora that were not modified in the first phase. To identify a stem sequence of each sentence in the Korean corpus, we use a morphological analyzer. Although we can obtain various IWFs of a source, we assume that only one IWF of the source is retrieved from the corpus in our example described in Fig. 2, for convenience.

In phase 3, our method (1) modifies the inflectional affixes of the target side of each aligned SMU pair so that a source and target in each aligned SMU can contain the same inflectional affixes; (2) combines those IWFs of the target side in the SMU pairs in order to obtain the final phrasal paraphrase pair. If the length of the stems in both sides of an aligned SMU pair is 1, we retrieve word-forms that contain the same stem with the target side of each SMU, as well as the same affixes with the source side of the aligned SMU from the Korean language of the bilingual parallel corpora. In this case, we can easily determine the stem in the target side of each aligned SMU that has to be modified. In Fig. 2, SI[1]-1 and T-1 comprise a single word-form and stem, respectively, and thus the affix of SI[1]-1, “ㄹ (an ending for converting a stem into a modifier)”, is attached to the stem of T-1, “제공하 (provide)”, in order to create equivalent grammatical categories. On the other hand, when the length of the aligned SMU pairs spans more than two stems or is different, the location of inflectional affixes from the source SMU and the corresponding target SMU is unclear. As a working example, Fig. 2 shows that the alignment of words “상품 (a product)” and “상업용 물품 (a commercial item)”, does not follow a one-to-one relationship. In this case, we retrieve a catena of which the head contains the same affixes as the head of the source side of its aligned SMU, regardless of the dependents in the corresponding source SMU. A catena is a combination of elements that are continuous in dependency grammar. For example, in Fig. 2, “상업용 물품을 (a commercial item)” is a catena, where the Korean “상업용 (commercial)” is dependent on “물품을 (item)” in dependency grammar. In such grammar, the head is a word (or stem in this context) that is modified by dependents. The head is the central word of a catena of which the grammatical category determines the syntactic type (i.e., phrase type). It also represents the primary meaning of the catena. Therefore, the inflectional affixes of the heads in each aligned SMU pair should be the same so that two head words that represent the primary meaning of the aligned SMU pair can represent the same syntactic type. We can identify the head and dependents in each catena using a dependency parser on the sentences that contain the retrieved catenae, as shown on the left lower side of Fig. 2. In our example described in Fig. 2, SI[1]-2 and T-2 comprise single and multiple stems, respectively, and thus the affix for SI[1]-2, “을 (an object case marker)”, is attached to the head of T-2, “물품 (an item)”, in order to ensure equivalent grammatical categories. As a result, the two words, “상품 (a product)” and “물품 (an item)” that indicate the same primary meaning (“an item”) of the two catena, obtain the same inflectional affix. After retrieving the IWF for the target, the inflectional affixes of the dependents in each aligned SMU can be different from each other. We simply penalize those pairs that are not similar in the sequence of affixes of dependents between catenae in each aligned SMU pair by using the paraphrase ranking model described in the next section. As a final step, we combine the IWFs of the target side in SMU pairs in each paraphrase pair in order to obtain the final phrasal paraphrase pair.

3.2. Ranking paraphrase pairs

We need to rank the targets of each source in order to clearly determine how each target is semantically and grammatically similar to the source. To do this, we adopt the *log-linear model* to calculate the conditional probability of a target, given a source. This model can easily and flexibly include associated feature functions, and thus we can separately consider the semantic and grammatical features by independently constructing feature functions for sequences of stems and inflectional affixes, respectively. This is a distinctive point with existing methods that model the word-form itself and do not manage the semantic and grammatical features separately in words. We define the paraphrase probability of target k_2 given source k_1 as a log-linear framework as follows:

$$P(k_2|k_1) = \frac{\exp\{\sum_{i=1}^{n} \lambda_i f_i(k_1, k_2)\}}{\sum_{k_2'} \exp\{\sum_{i=1}^{n} \lambda_i f_i(k_1, k'_2)\}} \tag{2}$$

where $f_i(k_1, k_2)$ is a feature function, λ_i is a parameter that indicates the importance of each feature function, and the denominator is a normalization term. In our method, we use three feature functions to estimate the paraphrase probability of each paraphrase pair. One is the semantic feature (stem paraphrase probability), and the other two functions are grammatical feature functions (similarity and fluency of sequences of inflectional affixes). The details are as follows:

- **Stem paraphrase probability**: The first feature function measures the stem paraphrase probability of the pairs extracted in the first phase of our method. The value of this function indicates how the paraphrase pairs that consist of stems are semantically equivalent. This can be classified into a semantic feature function such that a stem represents the concrete meaning of a word:

$$f_1(k_1, k_2) = \log P(stm(k_2)|stm(k_1)) \tag{3}$$

  where $stm(k_1)$ and $stem(k_2)$ are the sequences of the stems of phrases k_1 and k_2 respectively. The conditional probability $P(stm(k_2)|stm(k_1))$ can be calculated using the paraphrase probability used in Bannard and Callison-Burch's research [15].
- **Similarity of sequences of inflectional affixes**: The second feature function measures the similarity of the sequences of the inflectional affixes between two phrases. In our method, the source and target have common semantic sequences. Therefore, semantically paired stems between the source and target should also have the same grammatical sequences. This feature function can penalize pairs that are not similar in the sequence of affixes of the dependents between catenae in each aligned SMU, as mentioned in the third phase of our method. To measure this similarity, we use the edit distance-based similarity [24]. The edit distance in this feature function is defined as the word-level Levenshtein distance between two word sequences. More specifically, the distance is the minimum number of edits (deletions, insertions, or substitutions at the word level) needed to transform one

word sequence into another. This measures the sequential similarity of the inflectional affixes between source and target, assuming that each inflectional affix is a word. For instance, in our example, the sequences of the affixes in the source and target can be represented as "ㄹ 을 (an ending for converting a stem into a modifier and an object case maker)" and "ㄹ X 을 (an ending for converting a stem into a modifier, X, and an object case maker)", respectively, where X represents the absence of affixes. In our example above, the edit distance is 1 because "X" is additionally inserted in T-2. Then the distance is normalized by the length of the longer sequence. Therefore, the normalized edit distance is 1/3. The second feature function is as follows:

$$f_2(k_1, k_2) = -\log\{ED(aff(k_1), aff(k_2))\} \quad (4)$$

where ED is the normalized edit distance and $aff(k_1)$ and $aff(k_2)$ are the sequence of the inflectional affixes in source k_1 and target k_2, respectively. Given that the edit distance measure indicates the lexical dissimilarity of two sequences, the function -log() changes the distance into a similarity score [24].

- **Fluency of sequence of inflectional affixes**: The third feature function measures the extent to which the target combined with modified catenae is grammatically fluent. In our method, the sources are grammatically correct because these exist in a given corpus. However, the combinations of modified catenae could be ungrammatical phrases because those are newly generated by affix modification. Therefore, this feature function penalizes the paraphrase pairs that have an ungrammatical target. To measure the third feature, we adopt the concept of a statistical language model, which is a probability distribution over linear sequences of words. This model is useful in measuring the grammaticality of the sentences. For example, if a preceding word represents an adjunct that functions as an adverb, the probability is higher when the next word represents a modifier that functions as an adjective, rather than a subject, or an object that functions as a noun, because the latter cases rarely appear in the training corpus. Given that an agglutinative language represents grammatical categories using the inflectional affixes of a word, we measure the grammaticality of each phrase using the sequences of inflectional affixes. We define the feature function as follows:

$$f_3(k_1, k_2) = \log P_{LM}(aff(k_2)) \quad (5)$$

where $aff(k_2)$ is the sequence of the inflectional affixes in target k_2. This can be calculated using the chain rule on the joint probability distribution with a second-order Markov chain used to calculate the trigram language model.

In this model, the parameters $\lambda_i (i = 1, 2, 3)$ need to be estimated. As a first step toward estimating the parameters, we construct a development set using 30 randomly retrieved sources from the extracted paraphrase pairs and 125 sentences that contain the retrieved source phrases.[2] We first generate paraphrased sentences by substituting a source in each sentence with all possible corresponding targets. Three human judges then manually label all the paraphrased sentences as "correct" if both the meaning preservation and grammaticality between the original and paraphrased sentences are correct according to the following metrics [3,16]:

- **Meaning preservation**: Does the extracted target phrase preserve the meaning of the source phrase? (1: worst, 5: best), if a target is awarded a score exceeding 3 points in meaning preservation, we consider it the correct target).
- **Grammaticality**: Is the paraphrased sentence grammatical? (1: worst, 5: best, if a target receives a score of more than 4 points in grammaticality, we consider it to be the correct target).

We consider a target to be correct in meaning preservation if it is assigned a score of 3 or greater and the target to be grammatically sound if it is assigned a score of 4 or 5 as in [16]. To estimate the parameters using the development set, we specify a cost function, namely, the phrase substitution error rate (PSER). The values of the function are the proportion of incorrectly substituted phrases:

$$PSER = \sum_{i=1}^{n} \frac{|PS_0(TP_i)|}{|PS(S_i)|} \quad (6)$$

where $PS(S_i)$ set of substituted sentences in an original input sentence S_i, and $PS_0(TP_i)$ is a set of incorrectly substituted target phrases among the substituted target phrases TP_i for the original input sentence. We estimate the parameters using the gradient descent algorithm [25] by minimizing the value of the function *PSER* in the development set.

3.3. Experiments on AMBPM

In this section, we compare the quality of the PPDB constructed by using ABMPM and the PPDBs constructed from two state-of-the-art paraphrase extraction methods, SCBPM and the word embedding method. We also examine the coverage of each PPDB and analyze errors contained in the PPDBs constructed by using AMBPM. These experiments build on a previously published study of AMBPM [26] in that we increased the size of the test dataset and additionally conducted the error analysis.

3.3.1. Resources

In order to construct PPDBs using AMBPM and SCBPM, we used the English-Korean bilingual parallel corpora that contain 153,778 sentence pairs as described in Table 1. Although bilingual parallel corpora in widely studied languages, such as European

[2] We collected the sentences from a Korean major news portal site (http://news.naver.com) for research purpose.

Table 1
English-Korean bilingual parallel corpora used in experiments.

Data source	Data size
KAIST language resources [29]	51,710 sentence pairs
Sejong parallel corpora [30]	18,984 sentence pairs
Text REtrievial Conference (TREC) question-answering track [31] data and their translations [9]	998 sentence pairs
Sample sentences in Daum open dictionary[a]	82,086 sentence pairs

[a] This was collected from http://endic.daum.net on 18th November, 2014 for research purpose.

languages or Chinese, consist of more than 1 M pairs, the amount of available data in agglutinative languages, with the exception of Japanese, is much more limited in general. We train the word embedding model by using the Korean side of the bilingual parallel corpora.

For implementing AMBPM, we used the Giza++ [18] and Moses toolkit [27] to conduct word alignment and phrase extraction heuristics, respectively, and the Electronics and Telecommunication Research Institute (ETRI) linguistic analyzer [28] for the phrase structure and dependency parsing and morphological analysis. For SCBPM, we used Callison-Burch's implementation.[3] Of the three types of methods for paraphrase likelihood calculation for the paraphrase pairs extracted from SCBPM [1,3,16], we used the probabilistic model of [16]. This is because the actual calculation methods for each feature function have not been reported in [1,3], and language-dependent features could be included in the estimation method of [1,3]. Moreover, although a gold standard is required in order to use the paraphrase likelihood calculation methods of [1,3], only English versions of gold standards are available [3]. For the word embedding model, we used word2vec implementation[4] to use the skip-gram model [17]. We trained this model with 300-dimensional word vectors because it showed the best performance in our preliminary experiments using Wikipedia (version on 26, Dec. 2015) data without any pre-processing of the data as shown in Fig. 3. In this experiment, we used 840 paraphrase pairs, 84 sources with 10 targets per source. If the target of a source is assigned a score of 3 or greater in meaning preservation, the paraphrase pair is labeled as “correct”. Then, we calculate the average precision at 10 in models that are trained with different dimensions. We also set the minimum count of the word as 5 when the word embedding model is trained, which means that words with a total frequency lower than 5 in the training corpus are ignored. This minimum count is based on the average frequency of words in the training corpus. If we selected the count to be more than the average, it would not have been possible to train most of the words. Moreover, if we used a number that is less than the average, the accuracy of the model would be decreased. Therefore, we set the count to be approximately at the level of average frequency.

3.3.2. Experimental setting

The purpose of the experiment is to evaluate the quality of paraphrase pairs in PPDBs, which are generated by using AMBPM, SCBPM, and the word embedding model. The evaluations were conducted at two levels; single-word and multi-word phrasal paraphrase pairs. Since the word embedding model was only devised to model single-word paraphrase pairs, it was excluded from the multi-word comparison. It should be noted that the size of the test dataset was expanded compared with the previously published study on AMBPM [26]. In the previous study [26], 43 multi-word and 17 single-word paraphrase pairs were used. However, this size could be too small to accurately compare the three methods. Therefore, in this paper, we expanded the size of the test dataset to contain 200 multi-word and 100 single-word paraphrase pairs. Each test paraphrase pair comprises one source and the best target phrase selected by the ranking model of each method. All sources in the 300 paraphrase pairs are commonly contained in all PPDBs.

In our experiments, two human judges evaluated the test paraphrase pairs using the metrics for meaning preservation and grammaticality of paraphrase pairs extracted when we estimated the parameter of our paraphrase ranking model as described in Section 3.2. For each source, we also provided five sentences, which contain the source, to assist judges to consider the context in which each source is used. In total, we obtained 2,800 responses from the two human judges. We also measured the inter-annotator variability among them using Cohen's Kappa. In the five-point scale, the value of Kappa was 0.53, which can be interpreted as a “moderate agreement”. This value was also higher than the Kappa in the previous work using the same evaluation metrics (i.e., 0.33) [16]. We also examine the coverage of each of the PPDBs in order to investigate how many sources each PPDB can accommodate. To this end, we investigated the number of unique sources in each PPDB. We also analyzed the errors that are contained in each PPDB, which is a newly introduced development in this paper, compared with our preliminary study of AMBPM [26].

3.4. Experimental results on performance of AMBPM

In the evaluation using the single-word paraphrase pairs, the PPDB constructed by using ABMPM outperformed the PPDBs constructed by SCBPM and the word embedding model in terms of meaning preservation and grammaticality, as shown in Table 2. To investigate whether there are statistical differences among the three PPDBs, we performed a one-way analysis of variance (ANOVA) with post hoc comparison using the Tukey HSD and Duncan test [32]. The statistical analysis indicates that there are significant differences between the PPDB constructed by AMBPM and the two types of PPDBs in terms of meaning preservation

[3] This software is available at http://www.cis.upenn.edu/ccb/howto-extract-paraphrases.html.
[4] This software is available at https://code.google.com/p/word2vec/.

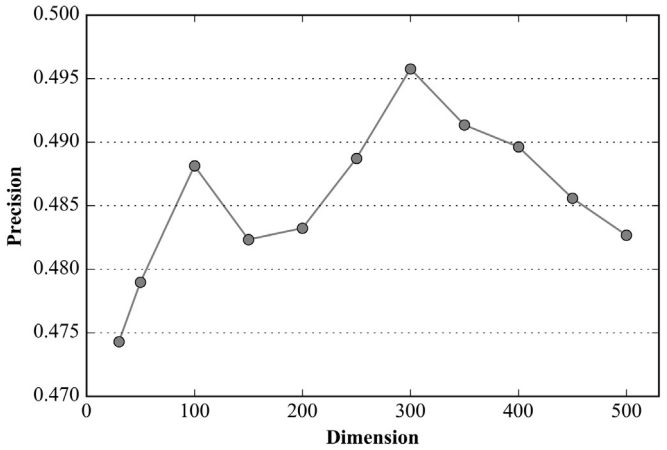

Fig. 3. Average precision at 10 of the word embedding models trained with different dimensions.

Table 2
The performance of PPDBs in single-word paraphrase pairs (mean score (standard deviation)).

PPDB	Meaning preservation	Grammaticality
AMBPM	3.71 (1.38)	4.84 (0.57)
SCBPM	3.48 (1.58)	3.97 (1.28)
Word embedding model	2.06 (1.56)	3.35 (1.52)

Table 3
The performance of PPDBs in multi-word paraphrase pairs (mean score (standard deviation)).

PPDB	Meaning preservation	Grammaticality
AMBPM	4.11 (1.11)	4.91 (0.39)
SCBPM	3.77 (1.37)	4.32 (1.07)

(F(2, 597) = 70.61, p = 0.00) as well as grammaticality (F(2, 597) = 79.20, p = 0.00). The F value is the test statistic used in a statistical hypothesis test using ANOVA. The numbers in parentheses are the degree of freedom between groups (i.e., PPDBs) and within groups (i.e., judges' responses). The p-value is a significance probability of the F value. In our experiment, the significance threshold was set at 0.05; therefore, if the p-value is less than 0.05, this indicates that there are significant differences among the three PPDBs. However, the p-value does not provide information as to where the differences occur. Therefore, we also conducted a post-hoc test to identify two pairs of PPDBs that show significant differences. In our comparison, there are significant differences between PPDBs constructed by using AMBPM and SCBPM and between SCBPM and the word embedding model. In the case of multi-word paraphrase pairs, the PPDB constructed by using ABMPM outperformed the PPDBs from SCBPM as shown in Table 3. In this case, we used the independent-samples t-test because two PPDBs are used in this evaluation. The t-test results indicates that there are significant differences between the two PPDBs in terms of meaning preservation (t(798) = 3.80, p = 0.00), grammaticality (t(798) = 10.43, p = 0.00).

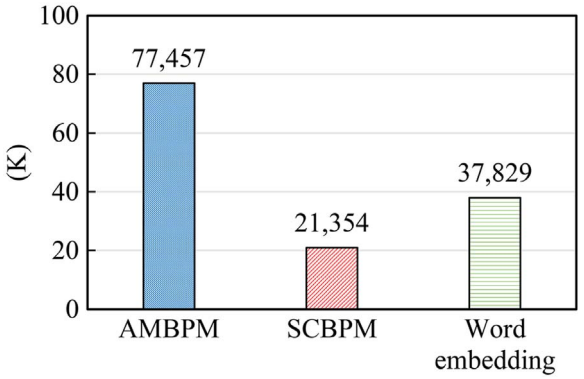

Fig. 4. Number of unique sources in PPDBs.

Table 4
Examples of paraphrase pairs extracted by each method.

Source	AMBPM	SCBPM	Word embedding model
심각한 (serious)	진지한 (serious)	큰 (large)	중대한 (important)
아내에게 (to wife)	부인에게 (to wife)	아내와 (with wife)	어젯밤에 (in yesterday)
관광객들이 (tourists)	여행객들이 (travelers)	화장실이 (a restroom)	방문객들이 (visitors)
낯선 사람에게 (to a stranger)	모르는 사람에게 (to a stranger)	낯선 사람을 주시하고 (and watch a stranger)	-
WTO의 보고서에 (in a report of WTO)	세계무역기구의 보고서에 (in a report of WTO)	WTO (WTO)	-
즐길 예정입니다 (be scheduled to enjoy)	즐길 겁니다 (will enjoy)	즐길 수 있을 것입니다 (could enjoy)	-
필요로 한다 (need)	요구한다 (require)	있다 (be)	-

From the coverage test, we observed that the PPDB constructed by AMBPM has the best coverage among all three PPDBs when the same training data are used, as shown in Fig. 4. Although the number of unique sources in the PPDB constructed by the word embedding model can be increased by selecting a smaller word count, as we mentioned in Section 3.3.1, it leads to lower performance. Therefore, this fact suggests that, in agglutinative languages, AMBPM is the most effective method that can obtain a PPDB that is both high in quality and coverage, compared with the existing paraphrase extraction methods.

3.5. Error analysis and discussion

The experimental results relating to the quality of extracted paraphrase pairs demonstrated that AMBPM outperformed the two state-of-the-art paraphrase extraction methods in terms of meaning preservation and grammaticality. The meaning preservation is strongly associated with the extent to which stems are correctly aligned with each other. Since AMBPM model stems instead of word-forms were used, our approach considerably reduces the word alignment errors by addressing the problem of lexical data sparsity compared with SCBPM. As a result, this modeling approach realized a more accurate estimation of the co-occurrence of a word from bilingual parallel corpora when conducting word and phrase alignment. Moreover, our affix modification process was more effective than constraining the syntactic type of SCBPM in terms of grammaticality. In particular, given that point 5 is the best score, the affix modifications are almost perfect with respect to the grammaticality. The results strongly support our claims that, in agglutinative languages, modifying inflectional affixes such that two phrases indicate exactly the same grammatical categories is more efficient. With respect to the word embedding method, our experimental results are also in agreement with the experiments performed in previous studies that used other types of word embedding models [3]. We present examples of paraphrase pairs extracted from each method in Table 4.

In terms of coverage, we found that PPDB constructed by using AMBPM has the best coverage among the compared PPDBs when trained with the same data. As we described in Section 2.1, we observed that constraining syntactic types in SCBPM to consider the morphological word structure has considerably reduced the number of paraphrase pairs extracted, compared with AMBPM. On the other hand, our method was able to increase the size of unique sources because it newly generates grammatical variants of phrases by affix modification. Therefore, in NLP applications using agglutinative languages, AMBPM can more broadly accommodate grammatical variants of phrases compared to the other databases.

We also analyzed the errors of paraphrase pairs extracted by AMPBM from the manually evaluated test data. In summary, there were 72 paraphrase pairs that were incorrect in terms of either meaning preservation or grammaticality. As we described in Section 3.2, we considered a target to be correct in meaning preservation if it is assigned a score of 3 or greater and grammatically sound if it was assigned a score of 4 or 5. From the incorrect paraphrase pairs, we found the following categories of errors:

- **Word alignment errors (58.33%)**: These errors occur when words in bilingual sentences are incorrectly aligned. These errors lead to the alignment of two stems that have different meanings.
- **Errors by LSCS (22.22%)**: An LSCS is a stem with weak semantic content. These errors occur when either the source or target has an additional LSCS. It also occurs when the source and target in a paraphrase pair have different types of LSCSs. Therefore, these errors cause the source and target to have subtle differences in meaning and grammatical functions.
- **Errors by translation (13.89%)**: These errors occur when translators omit the translation of certain expressions or rephrase the translation in a given context. In this case, although the source and target aligned with the correct pivot, they have different meanings as a result of rephrasing and omission. This is not a difference in the sense of a word. In some sentences, contexts support the meaning of an expression. For example, in the expression “reach out with his hand” in a sentence, the words “with his hand” could not be translated if “with his hand” is implied by the given situation that is described in the sentence. In this case, the phrase “reach out his hand” are aligned with “reach out” because they share the same pivot.
- **Others (5.56%)**: These errors include spelling mistakes and incorrect Korean phrases (Table 5).

Table 5
Examples of errors from AMBPM.

Error type	Source	Target
Word alignment	그 은행은 (the bank) 선거 운동에 (in election campaign)	한국 은행은 (the bank of Korea) 대선에 (in presidential election)
LSCS	예상된다 (be expected) 안 되지 (must not)	것으로 기대된다 (be expected that) 할 수는 없지 (cannot)
Trnslation	여동생의 (younger sister's) (pivot: sister's) 아저씨 (familiar middle-aged man) (pivot: uncle)	언니의 (older sister's) (pivot: sister's) 삼촌 (uncle) (pivot: uncle)
Others	나의 말이 (my words) 기다리게 한 것은 아닌지 모르겠습니다 (I have kept you waiting very long, I'm afraid)	*내의 한 말이 (*Incorrect Korean) *기다린 않았는지 모르겠습니다 (*Incorrect Korean)

To date, many researchers have reported that the word alignment errors are most prevalent in the paraphrase methods that employ the bilingual pivoting approach [11,15]. Our results are in agreement with the results of those studies in that this error type occupied the largest portion. In order to solve this problem, researchers have introduced various methods, e.g., increasing the training corpora such as multilingual parallel corpora [15] and using more informative features such as lexical weighting as used in phrase-based SMT [3,11]. Those methods were very effective in penalizing the paraphrase pairs that contained the word alignment errors.

The second category of error, errors by LSCSs, has not been frequently mentioned to our best knowledge. An LSCS is a stem with weak semantic content, and thus should be modified, or should modify other words to be specified. LSCSs do not contain specific meanings. For example, the English relative pronoun "that" also does not have a specific meaning, but it is specified by the clause that follows it. In addition, the English auxiliary verb "can" does not have a specific meaning, instead it specifies the meaning of a main verb by assigning a grammatical category. Therefore, LSCSs are closer to the grammatical meaning than the semantical meaning, and thus semantically aligned SMU pairs should contain the same LSCS in common with inflectional affixes. We can also find the reason for the LSCS to be the same in both the source and target from the classification of parts-of-speech. Parts-of-speech can be classified into open and closed classes. Open classes accept the addition of new words, such as nouns (with the exception of pronouns) and verbs (with the exception of auxiliary and light verbs), whereas closed classes infrequently add new words, such as prepositions and determiners. Open classes contain words with greater semantic content, whereas closed classes essentially perform grammatical functions [33]. That is to say, word classes that represent a grammatical function have a limited vocabulary size, and thus there are a few substitutable expressions for a word in these classes. Therefore, the LSCS and inflectional affixes should be equivalent between a source and target.

Although the LSCSs in SMU pairs should be equivalent as describe above, our error analysis revealed two classes of errors when the LSCS in SMU pairs differed. The first error caused by the LSCS is an inequivalence caused by additional LSCSs in either the source or target of each aligned SMU pair (e.g., "can supply" ⇔ "provide"). The first type of error, which is caused by the additional LSCS, usually leads to additional inflectional affixes from the affix modification process (Fig 2, phase 2 & 3), because the LSCS is also a stem to which inflectional affixes are attached. This additional affixation on either side of a source or target could naturally decrease the grammatical equivalence between the source and target. The second error caused by the LSCS is an inequivalence caused by differing the LSCS in either the source or target of each aligned SMU pair (e.g., "can supply" ⇔ "must provide"). The second type of error, which is caused by a different LSCS, also leads to grammatical inequivalence since a different LSCS between the source and target has a high probability that they represent different grammatical meaning. Moreover, because an LSCS has weak semantic meaning, it causes the meaning of the source and target to have subtle differences in both the first and second type of errors.

We observed that the inequivalence between LSCS pairs are attributable to the following two reasons. First, the LSCS in the source and foreign languages have a weak discriminative power for the word and phrase alignment process because they appear to play grammatical roles more frequently throughout entire training sentences than words in the open classes. That is to say, LSCSs have the probability of being aligned with most foreign words, whereas words in open classes have the probability of being aligned with a smaller amount of words. Second, in human translation, we have observed that LSCSs tend not to be directly translated from foreign LSCSs, thus transforming into other types of grammatical expressions. For example, an English sentence stated in the passive voice can be translated into the active voice. In this case, the translation of most semantic content words is preserved, but the translation of an LSCS could be transformed into different types of LSCSs. This could lead to a different LSCS between a source and target after paraphrase extraction. The second reason worsens the discriminative power of an LSCS, and in turn, leads to additional LSCS in either the source or target.

In the next section, we address the errors stemming from the LSCS to improve AMBPM, so that the PPDB constructed by using the improved AMBPM could be usefully utilized in NLP applications. We also compare the improved AMBPM with the word embedding models. In our experiments, we used a dataset of the same size for a fair comparison. Although the results showed that AMBPM outperforms the word embedding model, one could argue that the setting is not realistic since a large amount of data is relatively easy to obtain for use in the word embedding model, thus one would not use the same amount of data for both methods.

Therefore, in our experiment in Section 4, the size of the dataset we use to train word embedding-based paraphrase extraction methods is larger compared to that of AMBPM.

4. Improved affix modification-based bilingual pivoting method

We introduce improved AMBPM, a rule-based filtering approach, to address the problems that occur due to LSCS.

4.1. Improved AMBPM

In this section, we describe how we improve AMBPM by addressing the problems caused by the LSCS. More specifically, we introduce a rule-based filtering approach that removes the stem paraphrase pairs in which either the source or target has an additional LSCS or the source and target have different types of LSCSs. This approach is simple, but effective in improving the quality of paraphrase pairs extracted by AMBPM as shown by our experimental results (Section 4.3).

The use of a rule-based filtering approach first requires the construction of rules that specify which stems are the LSCS in an agglutinative language. For the improved AMBPM, the rule is to select an English noun or verbs that are functional words rather than content words, and translating them into Korean. In order to generally apply our rules for agglutinative languages, we avoid using language-dependent LSCSs that exist in a certain agglutinative language. To this end, we are only concerned with LSCSs in nouns and verbs because most languages contain nouns and verbs as word classes. We also use the translations of English LSCSs, because these LSCSs are also translated into LSCSs in the target language. Although the translations are different depending on the agglutinative languages, the method to identify the LSCS in a certain agglutinative language is the same across all agglutinative languages in that all the languages use the same criteria (i.e., an English LSCS). In this paper, we consider the following stems as an LSCS.

- **LSCS in noun**: Noun stems that are translations of English pronouns, including relative pronouns.
- **LSCS in verb**: (1) Verb stems that are translations of English light verbs (e.g., do, get, have, give, make, take, etc.) that form a predicate with some additional expression, which is usually a noun. (2) Verb stems that are translations of English auxiliary verbs (e.g., can, may, will, should, etc.); that is, a verb that adds functional or grammatical meaning to the main verb.

In this paper, in order to find the translation of the English LSCS, we used the online open dictionary by Daum, which was also used for the experiments on AMBPM as stated in Section 3.3.1. We also used example sentences in the Daum dictionary that contain those LSCSs. Using the translation examples, we conducted morphological analysis to extract the stems from the IWFs of the LSCSs and their part-of-speech. The reason why we also examine the part-of-speech is because some stems of LSCS are homonyms with other stems that are not LSCS. With this information, we remove the stem paraphrase pairs if either the source or target in at least one aligned SMU has additional LSCS or they have different types of LSCS.

After constructing the list of LSCS according to the rule for improved AMBPM, the filtering approach is applied after phase 1 of AMBPM (Fig. 2). Fig. 5 depicts the three steps involved in the filtering process; (1) conduct morphological analysis with part-of-speech tagging to the Korean language of the bilingual parallel corpora. (2) assign each part-of-speech tag to the stems in a source and its targets using the labeled monolingual single corpus. From the corpus, we can easily assign the part-of-speech by looking over the phrases that have the same sequence of stems with the source or targets. In this process, there could be various combinations of part-of-speech tags that are assigned to the sequence of the stems in the source and target. Therefore, we assigned all possible tags to the sequences. (3) filter out pairs, if there is an additional LSCS in either side of each aligned SMU pair or different LSCS in the aligned SMU pair. For example, in Fig. 5, we filter out Target^2 because T-1 in Target^2 has the additional LSCS "것", which is the Korean translation of the relative pronoun "what", although S-1 in the source does not have the same LSCS.

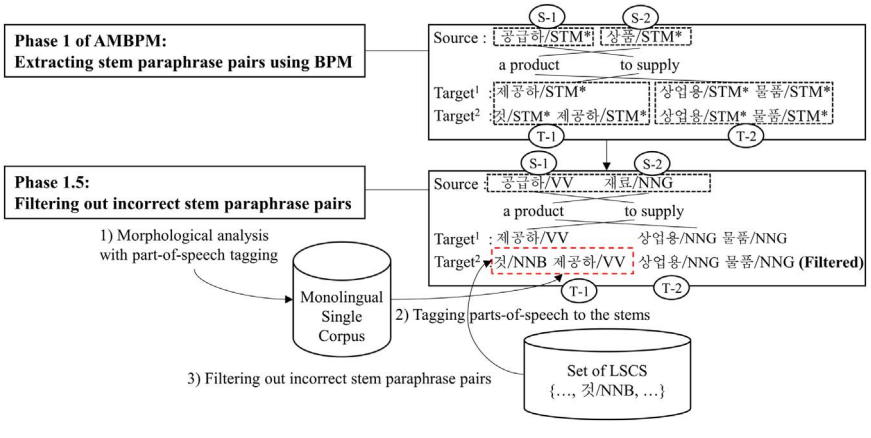

Fig. 5. Overview of the rule-based filtering process. NNB = bound noun, NNG = common noun, and VV = verb.

4.2. Experiments on improved AMBPM

In this section, we examine the effectiveness of the rule-based filtering approach by comparing the PPDB constructed by using the improved AMBPM with the PPDB from the original AMBPM. We also evaluate the usefulness of PPDB constructed by using improved AMBPM compared with four types of databases constructed from the original AMBPM, two PPDBs constructed from two different types of word embedding-based paraphrase extraction methods and a thesaurus. Because the purpose of the improvement of AMBPM is to apply it to NLP applications, we tested on two types of NLP applications: *sentential paraphrase generation* and *QA system*. This also enables us to evaluate the quality of a larger sized set of paraphrase pairs than the pairs used in the previous experiments as described in Section 3.3.

4.2.1. Training and test data

In order to construct PPDBs using the improved AMBPM, we used the same English-Korean bilingual parallel corpora used in Section 3.3.1. In order to enlarge the size of training data for word embedding-based paraphrase extraction methods, we used 3,393,022 monolingual sentences from Wikipedia (version on 26, Dec. 2015). Note that for the word embedding-based paraphrase extraction methods, the size of the dataset used was approximately 22 times greater than the training data for the two types of AMBPM. For the test sentences to be paraphrased in our paraphrase generation system and QA system, we collected 586 quiz sentences from a Korean television quiz show (i.e., Golden Bell Challenge!).

4.2.2. Word embedding-based paraphrase extraction methods and thesaurus

In our experiments, we evaluate the usefulness of the PPDB constructed by using the improved AMBPM compared with the PPDB constructed from the original AMBPM, two PPDBs constructed from two different types of word embedding-based paraphrase extraction methods and a thesaurus. All word embedding-based methods are commonly trained using the skip-gram model [17] and the methods generate 300-dimensional word vectors as described in Section 3.3.1. We also set the minimum word count as 10, which is based on the average frequency of each word occurrence in the training data. Note that in both the improved and original AMBPM, the corresponding targets to a source are determined by their pivot. In comparison, in word embedding-based paraphrase extraction methods, every target in the vector space can be paired with a source. Therefore, in this experiment, we create pairs using each source along with its top 10 targets based on the cosine similarity. Here we detail the word embedding-based methods and the thesaurus as follows:

- **Phrase embedding**: The word embedding method was originally devised to represent words as low-dimensional dense vectors. To train the method for phrases, we adopted a data-driven approach that combines group of words that frequently occur together, and infrequently occur in another context [17]. For example, the three words “New”, “York”, and “Times” can be combined into New_York_Times because they frequently occur in a certain context as a continuous sequence across sentences, but occur infrequently in other contexts. Those word sequences that are connected with an underline can be considered one word, and thus it is possible to train them using the word embedding method. We determined the canonical phrases by using the following score function proposed by [17], which is based on the unigram and bigram counts:

$$score(w_i|w_j) = \frac{count(w_i w_j) - \delta}{count(w_i)\,count(w_j)} \tag{7}$$

  where w_i and w_j are words in a continuous sequence and δ is a discounting coefficient to prevent too many phrases consisting of infrequent words from being counted as phrases. We can consider the sequence of words $w_i w_j$ as a canonical phrase if the score is higher than a certain threshold. After two words are grouped as one word by using the approach, we can also group three, four, and more words in a similar way. In our experiments, we used the bigram model because longer sequences tend to produce more errors, similar to the previous work [26], and empirically, the bigram model shows the best performance in our training data. For the threshold and discounting coefficient, we used the values provided by the word2vec implementation.
- **Stem embedding**: In order to demonstrate that the proposed processes for stem paraphrase pair extraction based on BPM and affix modification are more accurate than other types of processes, we modeled each stem using the word embedding method, as in AMBPMs, and conducted rule-based affix modification. For brevity, we refer to this as “stem embedding”. Because it is difficult to identify aligned SMU pairs between two phrases when applying the phrase embedding model (i.e., there is no pivot), we applied stem embedding to single unit paraphrase pairs only. To improve the results of stem paraphrase extraction using the word embedding method, we conducted three types of filtering approaches. We annotated the part-of-speech for each stem in each sentence in order to filter the pairs with different parts-of-speech after training. We also filtered the pairs with differently named entity (NE) types. To this end, we constructed an NE dictionary using Wikipedia text and an NE recognizer [34] that provides 184 fine-grained NE types. If two stems had different NE types according to the NE dictionary, the pair was removed. Finally, we filtered the antonym sets using two different types of thesauruses [35,36] that provide antonym information as well. As a result, we removed the fundamental problem concerning antonym pairs in the distributional hypothesis-based paraphrase extraction approach [1]. We then retrieved all possible IWFs for the sources that existed in the test sentences and modified the inflectional affixes of the corresponding targets such that they were the same as in AMBPM. In this modification process, we used the rules that indicate how to modify inflectional affixes depending on the phoneme of each stem so that the source and target could have the same type of inflectional affixes, instead of our proposed affix modification process. More specifically, those rules describe

cases such as the following: if the source and target "probability" and "chance" cause these two words to be plural, we remove "y" and attach the affix "-ies" to the word that ends in "y", and attach "s" to the others. This rule can be very useful in cases that consider such "allomorphs", which are morphemes that act as the same function, but take different forms, such as "-ies", "-s", and "-es". We constructed the rule based on the National Institute of Korean Language webpages [30].

- **Thesaurus**: For the thesaurus, we also used two thesauruses [35,36] as PPDB by combining them. In this condition, we assigned the paraphrase likelihood of all entities in the thesauruses using stem embedding results because most thesauruses do not explicitly specify semantic or grammatical similarity between words, as described in Section 1. Similar to stem embedding, we modify the inflectional affixes of each lexeme using the same affixation method with stem embedding.

4.2.3. Applications for experiments

In this section, we describe the two NLP applications used to evaluate the usefulness of our proposed method: sentential paraphrase generation and QA systems. We also describe the evaluation metrics in the experiments that use each application.

4.2.3.1. Sentential paraphrase generation system. In the experiment that uses the sentential paraphrase generation system, we used SMT-based discriminative paraphrase models, defined as follows:

$$\widehat{k_2} = \arg\max_{\widehat{k_2}} \{\lambda_1 \sum_i \log P(k_{2,i}|k_{1,i}) + \lambda_2 \log P_{LM}(K_2) + \lambda_3 \log(ED(K_1, K_2))\} \tag{8}$$

where $\widehat{k_2}$ is the best paraphrased sentence for source sentence K_1, $k_{1,i}$ and $k_{2,i}$ are the ith source and corresponding target phrases, respectively, and $P(k_{2,i}|k_{1,i})$ is the paraphrase likelihood between the ith source and its corresponding target, which is derived from each paraphrase extraction method (i.e., a paraphrase ranking model or cosine similarity). $\log P_{LM}(K_2)$ is the trigram language model for paraphrased sentence K_2, and $ED(K_1, K_2)$ is the word-level edit distance between two sentences that represent how many expressions are substituted. For our paraphrase model, the language model trained by the Wikipedia articles is described in Section 4.2.1. Those three features were widely used to generate paraphrased sentences in previous research [8,37], and represent the score of meaning preservation, grammaticality, and lexical dissimilarity in the order enumerated in the equation. Note that in our paraphrase model, there is no reordering model that allows the word order to be changed after generating a paraphrased sentence. This is because agglutinative languages do not rely on word order to represent the grammatical categories of each word. Moreover, under this condition, we want to examine the semantic and grammatical substitutability of each PPDB. If reordering were to occur, it would be difficult to be examined. Each parameter λ_i is estimated based on human judgment. Regardless of the paraphrase extraction methods, this model should have the same parameter values for fair comparison, thus avoiding biases by different turning processes that depend on each paraphrase extraction method. That is to say, by fixing the value of each parameter to be equal, we can examine whether the paraphrase likelihood of each PPDB is assigned appropriately. To estimate each parameter λ_i, we constructed a development set. For the first step in this construction, we generated paraphrased sentences using the paraphrase generator proposed in [9]. We asked three human judges to assign scores for meaning preservation, grammaticality, lexical dissimilarity, and overall score for the paraphrased sentences. Those metrics are discussed below. In order to estimate the importance that each feature contributed to determining the overall score, we used SVM regression. The values of the parameters calculated by the regression method were used as parameter values in our sentential paraphrase generation model.

In the experiment that used the sentential paraphrase generation system, we automatically and manually evaluated the paraphrased sentences generated from each of the five PPDBs; PPDBs from improved AMBPM, AMBPM, stem embedding, paraphrase embedding, and thesaurus. For automatic evaluation, we adopted the three automatic evaluation methods: bilingual evaluation understudy (BLEU) [38], US National Institute of Standards and Technology (NIST) [39], and translation edit rate (TER) [40], used widely in SMT. In the field of SMT, in order to evaluate a translation system, various references for a translation result are used, where both the references and result are written in the same language. Because a foreign sentence can be translated in various ways, the evaluation methods compare the system results with diverse candidate translations. Because the paraphrasing results can also vary, we can evaluate the quality of the paraphrased sentences using those evaluation methods with various paraphrase references. Each metric is described as follows:

- **BLEU**: Evaluates a translation system using n-gram precision (1–4) and brevity penalty between the system results and corresponding references. N-gram precision is the proportion of overlaps between the translation results and references. The brevity penalty is for penalizing results that are shorter than the references. A high BLEU score indicates that the system performance is high.
- **NIST**: This is a variation of BLEU, and it allocates more weight to more informative n-grams. An informative n-gram is an n-gram that does not occur frequently in references. The higher the NIST score, the higher is the quality of the system.
- **TER**: This is the minimum word-level edit distance between the translation results and the most similar reference normalized by the average word length of the references. In TER, the edit distance also considers an edit that moves the sequences of contiguous words. The lower the TER score, the higher is the quality of the system because TER represents the difference between the answer references and generated sentences.

To conduct an automatic evaluation, we randomly selected 100 test sentences from 586 test sentences, and manually generated seven reference sentences for each sentence. We used three different methods to generate various types of paraphrase references.

The first is to substitute words in the test sentences using a Korean online thesaurus.[5] The second is phrase substitutions, in which case we collected phrasal paraphrases from the results of the Google search engine. The final type is to use Google translation, where the original test sentence is first translated into Japanese and English sentences, then translated again to Korean sentences. Because much noise exists in the references generated in the last case, we manually revised those references so that they could be corrected. We publicized the paraphrase references on the web[6] for the benefit of researchers in this field. In this evaluation, we divided each word into a stem and inflectional affixes in order to reflect the result of the stem paraphrase extraction and affix modification processes. The automatic evaluation methods are efficient in that they have reproducibility and are not labor-intensive. However, the references cannot manage all possible paraphrased sentences. Moreover, the number of expressions in the references is limited, and thus it is difficult to determine the correctness of all paraphrased expressions. As a complement, we conducted a manual evaluation using 586 other test sentences.

For manual evaluation, three human judges manually evaluated the 586 paraphrased sentences generated using each paraphrase extraction method with the following criteria:

- **Meaning preservation**: Do the substituted target phrases preserve the meaning of the original source phrases? (1: worst, 6: best)
- **Grammaticality**: Is the paraphrased sentence grammatically correct? (1: worst, 6: best)
- **Overall score**: Is the paraphrased sentence synthetically correct? (1: worst, 6: best).

Note that we used a six-point Likert scale. When the initial pilot test was conducted using a five-point Likert scale for the evaluation of a paraphrase pair and other paraphrase generation research [37], we observed that all human judges showed a strong tendency for using a rating value of 3 for those paraphrased sentences where 40% (or 60%) of the phrases are correct, 60% (or 40%) of phrases are incorrect, or 50% of phrases are correct or incorrect. To differentiate between the scores of those three types of paraphrased sentences, we extended the Likert scale to range from 1 to 6. We obtained a total of 26,370 responses from three human judges in the manual evaluation. We measured the inter-annotator variability by using an intraclass correlation coefficient (ICC). On the six-point scale, the ICC is 0.654 ($p = 0.00$), which is a “moderate level of agreement”.

4.2.3.2. Question answering system. In the experiment that uses a QA system, we used the Information Retrieval (IR)-based QA system developed by ETRI.[7] This system follows the conventional architecture of IR-based QA systems [5]. The system first analyzes a given question to identify the question type, keywords, and semantic and syntactic relationship between the keywords. Then, the system retrieves relevant passages (or documents) similar to IR. Finally, the system finds the best answer from the passage based on the analysis of the questions.

In this application, we automatically evaluated the usefulness of five types of PPDBs; improved AMBPM, AMBPM, phrase embedding, stem embedding, and thesaurus. As input to the QA system, we used 586 paraphrased questions without their original question sentences. Many researchers have proposed methods that employ paraphrased expressions (or sentences) along with their original questions in QA systems [5,6]. Those methods can vary depending on the QA system. Our study focuses on the usefulness of our proposed method, but does not examine an effective method for flexibly using paraphrased expressions in QA systems. This is why we used only paraphrased sentences in the QA system as input. In this application, we examined the usefulness of PPDBs in two respects: how many core keywords in the original sentences preserve their meaning and the grammatical relationship among keywords in paraphrased sentences. In QA systems, those two factors are strong evidence for finding the answer to a question. Therefore, if the paraphrased sentences generated using PPDB preserve those factors, the paraphrased sentences are more likely to show a high answer rate. Therefore, we first examined the answer rate when using paraphrased sentences generated by different types of PPDBs.

We also investigated the potential for answer detection of paraphrased sentences in terms of improvement and error. When using only paraphrased sentences, the performance of the QA system could decrease because of an error in the loss of the popularity of expressions [41] and paraphrase generation methods. The original questions could be composed of expressions widely used (i.e., high popularity) by humans to delivery exact meaning to them. On the other hand, automatically paraphrased sentences could include expressions that are used less frequently than the expressions in the original sentences because the system transforms each expression in order to paraphrase the sentence. Those expressions that are less popular than the original have a lower possibility of existing in the corpus from which the QA system retrieves the answers [41]. Moreover, the original questions could include expressions that are contextually appropriate. On the other hand, the paraphrased sentences could include contextually inappropriate expressions led by errors in the paraphrase generation methods. In such cases one can use paraphrased expressions as alternative inputs for improving the performance of the QA system. If so, we need to investigate which of the PPDBs can be good alternative complements for the QA system. If the paraphrased sentences generated by a PPDB provide correct answers to the questions where the original question results in incorrect answers, the PPDB is said to be a good resource. We refer to this as *improvement*. On the contrary, if the paraphrased sentences generated using a PPDB provide incorrect answers to the questions

[5] We used the Naver Korean dictionary, available at http://krdic.naver.com/.
[6] The Korean paraphrase references are available at https://sites.google.com/site/tonyhanpark/resources.
[7] http://exobrain.kr/onedintro.

Table 6
Results of manual evaluation of five types of PPDBs on 586 test sentences (mean score (standard deviation)).

PPDB	Meaning preservation	Grammaticality	Overall
Improved AMBPM	4.22 (1.08)	5.09 (0.85)	4.39 (0.77)
AMBPM	4.10 (1.08)	4.93 (0.91)	4.29 (0.77)
Phrase embedding	2.71 (1.05)	3.36 (1.28)	3.15 (1.08)
Stem embedding	2.96 (1.15)	4.52 (1.12)	3.54 (1.00)
Thesaurus	3.23 (1.15)	4.67 (1.07)	3.75 (1.00)

Table 7
Results of automatic evaluation of five types of PPDBs on 100 test sentences.

PPDB	BLEU[a]	NIST[b]	TER[c]
Improved AMBPM	0.5204	8.6158	0.3064
AMBPM	0.4772	8.2476	0.3330
Phrase embedding	0.1039	4.6241	0.7434
Stem embedding	0.4039	6.9829	0.3399
Thesaurus	0.4253	7.3211	0.3324

[a] BLEU [38]: Algorithm for evaluating the overall quality of generated text by comparing the system results with corresponding references. BLEU simply calculates the n-gram precision and brevity penalty between the system results and corresponding references.
[b] NIST [39]: A variation of the BLEU method. NIST allocates more weight to more informative n-grams.
[c] TER [40]: Error metric for machine translation that measures the number of edits required to change the system output into one of the references.

where the original question results in correct answers, the PPDB is said to be a futile resource. This is certainly an *error*. In this respect, we can posit that a good resource would be able to provide more improvements and smaller errors. Therefore, when we employ a PPDB that shows high improvement and few errors, if we use the original questions and paraphrased sentences simultaneously, the QA system leads to excellent performance. Naturally, researchers should attempt to reduce errors by allocating more weight to the original questions than the paraphrased sentences, or by filtering paraphrased sentences that exhibit low confidence. We examine the improvement and error led by each PPDB in these experiments using the QA system.

4.3. Experimental results

4.3.1. Performance on sentential paraphrase generation

For manual and automatic evaluation, the paraphrase generation system that uses the PPDB constructed with the improved AMBPM showed the best performance in all evaluation criteria, as seen in Tables 6 and 7. We examined whether there are statistical differences among PPDBs by running a one-way ANOVA with post hoc comparisons based on the manual evaluation data. The statistical analysis indicates that there are significant differences between the PPDB constructed by using the improved AMBPM and the other four types of PPDBs in terms of meaning preservation ($F(4, 8785) = 671.172$, $p = 0.00$), grammaticality ($F(4, 8785) = 730.992$, $p = 0.00$), and overall score ($F(4, 8785) = 545.501$, $p = 0.00$).

4.3.2. Performance on question answering system

In terms of the answer rate, the QA system shows the best performance when using sentences paraphrased with the improved AMBPM, as indicated in Table 8. Given that the QA system considers keywords and the grammatical relationship between keywords in order to retrieve answers from given passages, these results indicate that those paraphrased sentences that used the improved AMBPM preserved more of the original meaning and grammatical structure of the given questions. Moreover, although the improved AMPBM does not produce the best results in respect of the improvement proportion, this method has the lowest error among all methods, as described in Table 8.

Table 8
Results for answer rate, and proportion of improvement and error on various PPDBs.

PPDB	Answer	Improvement	Error
Original sentence	74.74%	–	–
Improved AMBPM	62.46%	11.49%	20.31%
AMBPM	61.43%	12.83%	22.15%
Phrase embedding	33.79%	14.19%	59.59%
Stem embedding	41.30%	14.19%	49.54%
Thesaurus	40.27%	12.16%	50.23%

4.4. Error analysis and discussion

The results of experiments using the sentential paraphrase generation system demonstrated that the improved AMBPM that used the rule-based filtering approach is an effective solution for significantly reducing incorrect stem paraphrase pairs compared with the original AMBPM. Our experimental results suggest that considering the LSCSs between sources and targets is effective for improving the grammaticality and the meaning preservation of the paraphrase pairs. Our experimental results also demonstrated that the PPDBs constructed by using the improved AMBPM are more useful than the PPDBs constructed by using other word embedding-based methods and a thesaurus in sentential paraphrase generation systems. These results suggest that, despite the small amount of training data and the absence of aid from linguistic resources, for an agglutinative language the PPDB constructed with the improved AMBPM is more useful than the four databases. Moreover, extracting stem paraphrase pairs based on BPM is more accurate than stem embedding in that AMBPM shows improved meaning preservation compared to the other methods. In addition to this, all AMBPM-based PPDB outperformed stem embedding and thesaurus-based PPDBs demonstrate that our affix modification method is more effective than rule-based affixation in terms of grammaticality. We noticed that there are numerous affixation rules that have exceptions based on the stems, and those generate word-forms that cannot exist. On the other hand, because we modify the affixes based on existing word-forms, the results extracted with AMBPM are more accurate in terms of grammaticality than for the other methods.

The PPDB constructed by using improved AMBPM is the most helpful resource in that the database has the lowest error rate among all PPDBs. However, in terms of improvement measure, we did not observe any difference between all PPDBs. We analyzed the paraphrased questions to identify why the PPDB constructed by using the improved AMBPM did not show the best improvement. To this end, we used 28 original questions along with paraphrased questions from all types of PPDBs. From the improved AMBPM, we selected those paraphrased questions that resulted in an incorrect answer from the QA system. From the other PPDBs, we selected those paraphrased questions that resulted in a correct answer from the QA system. The five categories of our analysis are as follows:

- **Incorrect paraphrase of keywords (47.62%)**: In this case, because of the failure to correctly paraphrase keywords, the input to the QA system is incorrect, resulting in incorrectly answered questions. For example, although the question type is a key for answering questions, the failure of paraphrasing in this case, such as < which stage ?, when ?>, leads to a different answer type. Spatial information is also an important keyword to narrow down the search space. However, an error such as < the Earth, earth > could change the focus of the system for searching.
- **Lack of entailment of keywords (23.81%)**: The paraphrased sentences that resulted in incorrect answers did not contain keywords that specified the original phrases. For example, consider the case in which an original question contains the keyword “body”, and the word is paraphrased as “organ”. Since the word “organ” entails “body”, the paraphrased sentence from the other PPDB resulted in the correct answers, because entailment helped to narrow down the search space for answer candidates. Since the improved AMBPM does not consider textual entailment, such cases did not perform well.
- **Incorrect word sense of keywords (14.29%)**: In some paraphrased sentences generated by the improved AMBPM, we observed failure of word sense disambiguation. For example, in the original sentence, the word “star” denoted “fixed star in universe”, but this was incorrectly paraphrased into “celebrity”.
- **Incorrect paraphrase of named entity (9.52%)**: The improved AMBPM did not constrain the paraphrase pairs that have named entities that differ from each other. This leads to paraphrasing named entities in our experiments. For example, “The Brothers Karamazovy” with the entity type “novel”, was paraphrased into “The younger brother Karamazovy”, with a different entity type “none”.
- **Incorrect negation (4.76%)**: Some pairs that contain “not” on either side of the source or target were incorrectly negated. For example, an incorrect paraphrase pair < “can”, “can't” > paraphrases an affirmative sentence to be a negative sentence.

In our experiment, the improved AMBPM showed the best performance. Nevertheless, much additional work remains in that it is necessary to improve the quality of the PPDB using our proposed method. First, we notice that there are still numerous paraphrase pairs with different LSCSs between the sources and targets in the PPDB constructed with the improved AMPBM. An examination of the error analysis data revealed that our filtering approach decreases the error rate from LSCS by 75%. Most of these (25%) that were not filtered are errors caused by LSCS that exist in a few languages. In this regard, we observed classifiers that measure words, especially in East Asian languages and the bound noun in Korean. Table 9 presents examples of remaining errors due to LSCS.

Although our filtering approach shows the potential for improving the quality of PPDBs, it appears to be insufficient. Therefore, developing a language independent automatic detection method for LSCS is required in order to increase rule coverage. Second, our modification approach is fragile to allomorphs, unlike rule-based affixation. Therefore, devising an affix modification method is required for managing allomorphs. Third, although improved AMBPM was trained with a small dataset, it extracted a higher quality of paraphrase pairs than the other methods. However, the coverage of paraphrase expressions remains a challenge in that improved AMBPMs have a lower number of unique sources, as depicted in Fig. 6, and lower lexical dissimilarity (i.e., edit distance) in the sentential paraphrase generation system, as indicated in Table 10, compared with the other methods. In order to extend the coverage of PPDB using the improved AMBPM, we would have to collect additional bilingual parallel corpora from easily accessible data, such as the parallel web and subtitles. If we could collect such additional data, the PPDB quality would also improve by further addressing the lexical data sparsity problem.

Table 9
Examples of errors remaining due to LSCS.

Source	Target	
수십 개의 (dozens of "things")	수십 명의 (dozens of "people")	Although the pivot is "dozens of," the source is used for objects and the latter is used for people (i.e., classifier).
소음 때문에 (because of noise)	소음에 (by noise)	The pivot is "by noise." "때문" is a bound noun that indicates cause, but "에" is used to almost represent what has an influence on something or spatial information. However, this is infrequently used for causes.
터인 (would be that)	바란 (the way that)	"터" and "바" is a bound noun that is modified by the verb. The former is to add the meaning of "willingness" or "suggestion," whereas the latter is to add the meaning of "manner" or "emphasis."

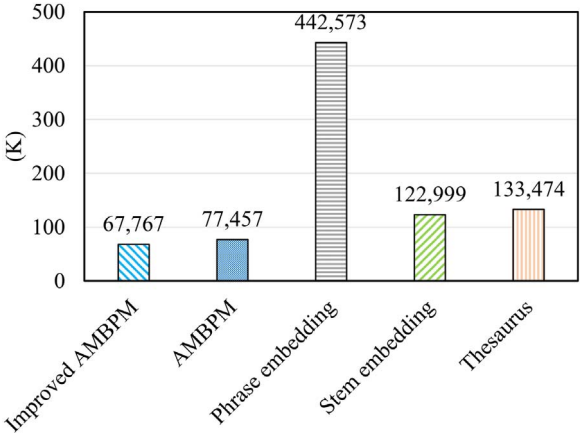

Fig. 6. Number of unique sources in PPDBs.

Table 10
Mean and standard deviation of word-level edit distance (lexical dissimilarity) between original and paraphrased sentences of entire 586 test sentences (average word length: 25.71(12.08)).

PPDB	Lexical dissimilarity
Improved AMBPM	9.10 (4.92)
AMBPM	9.75 (5.20)
Phrase embedding	19.67 (9.86)
Stem embedding	10.61 (5.58)
Thesaurus	10.62 (5.58)

5. Conclusion

This paper proposed an affix modification-based bilingual pivoting method (AMBPM) to address the problems of lexical data sparsity without considering the morphological word structure, which occurs when existing methods are applied to agglutinative languages. In our experiments on AMBPM, we demonstrated that AMBPM outperforms two state-of-the-art paraphrase extraction methods. We also proposed an improved AMBPM, which uses a filtering approach to filter out incorrect stem paraphrase pairs caused by LSCS. The experimental results on improved AMBPM demonstrate that this rule-based filtering method significantly outperforms AMBPM. Moreover, despite the small amount of training data and the use of no linguistic resources, the PPDB constructed with the improved AMBPM is more useful than the PPDBs constructed by two types of word-embedding-based methods (stem embedding and phrase embedding), and an existing thesaurus for NLP applications, such as sentential paraphrase generation and a QA system using an agglutinative language. We also contributed by publicizing our PPDB constructed with the improved AMBPM.

Nevertheless, future work would involve improving the quality of the PPDB using our proposed method and demonstrating the usefulness of our method. In terms of improvements to our method, developing a language-independent automatic detection method for LSCS is recommended in order to increase the coverage of filtering rules. Additionally, devising an affix modification method in order to manage allomorphs is also recommended. Finally, collecting additional bilingual parallel corpora and considering additional feature functions to reduce the word alignment error would improve our suggested method. In addition, although we used the traits commonly applied to agglutinative languages, the generality of our method is not demonstrated yet. Therefore, for future study, to investigate the generality of our proposed method for various agglutinative languages, we plan to examine our method with other agglutinative languages such as Japanese and Turkish.

Acknowledgement

This work was supported by the Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No. R0101-15-0062, Development of Knowledge Evolutionary WiseQA Platform Technology for Human Knowledge Augmented Services).

References

[1] J. Ganitkevitch, B. Van Durme, C. Callison-Burch, PPDB: the paraphrase database, in: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), 2013, pp. 758–764.
[2] J. Ganitkevitch, C. Callison-Burch, The multilingual paraphrase database, in: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC), 2014, pp. 4276–4283.
[3] E. Pavlick, P. Rastogi, J. Ganitkevitch, B. Van Durme, C. Callison-Burch, PPDB 2.0: better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP), 2015, pp. 425–430.
[4] D. Lin, P. Pantel, DIRT – discovery of inference rules from text, in: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2001, pp. 323–328.
[5] M.M. Soubbotin, S.M. Soubbotin, Patterns of potential answer expressions as clues to the right answers, in: Proceedings of the Tenth Text REtrieval Conference (TREC), 2001, pp. 175–182.
[6] S. Riezler, A. Vasserman, I. Tsochantaridis, V. Mittal, Y. Liu, Statistical machine translation for query expansion in answer retrieval, in: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL), 2007, pp. 464–471.
[7] C. Callison-Burch, P. Koehn, M. Osborne, Improved statistical machine translation using paraphrases, in: Proceedings of the Human Language Technology Conference of the NAACL (HLT-NAACL), 2006, pp. 17–24.
[8] S. Zhao, X. Lan, T. Liu, S. Li, Application-driven statistical paraphrase generation, in: Proceedings of the Joint Conference 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (ACL-IJCNLP), 2009, pp. 834–842.
[9] H. Park, G. Gweon, H.-J. Choi, J. Heo, P.-M. Ryu, Sentential paraphrase generation for agglutinative languages using SVM with a string kernel, in: Proceedings of the 28th Pacific Asia Conference on Language, Information, and Computation (PACLIC), 2014, pp. 650–657.
[10] K. R. McKeown, R. Barzilay, D. Evans, V. Hatzivassiloglou, J. L. Klavans, A. Nenkova, C. Sable, B. Schiffman, S. Sigelman, Tracking and summarizing news on a daily basis with columbia's newsblaster, in: Proceedings of the Second International Conference on Human Language Technology Research (HLT), 2002, pp. 280–285.
[11] S. Zhao, H. Wang, T. Liu, S. Li, Extracting paraphrase patterns from bilingual parallel corpora, Nat. Lang. Eng. 15 (2009) 503–526.
[12] M. Mizukami, G. Neubig, S. Sakti, T. Toda, S. Nakamura, Building a free, general-domain paraphrase database for japanese, in: Co-ordination and Standardization of Speech Databases and Assessment Techniques (COCOSDA), 2014 17th Oriental Chapter of the International Committee for the, 2014, pp. 1–4.
[13] A. Fujita, S. Sato, Measuring the appropriateness of automatically generated phrasal paraphrases, J. Nat. Lang. Process. 17 (2010) 183–219.
[14] C. Hashimoto, K. Torisawa, S. De Saeger, J. Kazama, S. Kurohashi, Extracting paraphrases from definition sentences on the web, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL), 2011, pp. 1087–1097.
[15] C. Bannard, C. Callison-Burch, Paraphrasing with bilingual parallel corpora, in: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL), 2005, pp. 597–604.
[16] C. Callison-Burch, Syntactic constraints on paraphrases extracted from parallel corpora, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2008, pp. 196–205.
[17] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Proceedings of the Twenty-seventh Annual Conference on Neural Information Processing Systems (NIPS), 2013, pp. 3111–3119.
[18] F.J. Och, H. Ney, A systematic comparison of various statistical alignment models, Comput. Linguist. 29 (2003) 19–51.
[19] P. Koehn, A. Axelrod, A. Birch Mayne, C. Callison-Burch, M. Osborne, D. Talbot, Edinburgh system description for the 2005 IWSLT speech translation evaluation, in: Proceedings of IWSLT, 2005, pp. 68–75.
[20] Z.S. Harris, Distributional structure, Word 10 (1954) 146–162.
[21] D. Gupta, J. Carbonell, A. Gershman, S. Klein, D. Miller, Unsupervised phrasal near-synonym generation from text corpora, in: Proceedings of AAAI Conference on Artificial Intelligence (AAAI), 2015, pp. 2253–2259.
[22] Y. Bengio, R. Ducharme, P. Vincent, C. Jauvin, A neural probabilistic language model, J. Mach. Learn. Res. 3 (2003) 1137–1155.
[23] A. Mnih, G. Hinton, Three new graphical models for statistical language modelling, in: Proceedings of the 24th International Conference on Machine Learning (ICML), 2007, pp. 641–648.
[24] G. Erkan, A. Özgür, D. R. Radev, Semi-supervised classification for extracting protein interaction sentences using dependency parsing, in: Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), 2007, pp. 228–237.
[25] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes in C: The Art Of Scientific Computing, 2nd Edition, Cambridge Univ. Press, UK, 1992.
[26] H. Park, G. Gweon, J. Heo, Affix modification-based bilingual pivoting method for paraphrase extraction in agglutinative languages, in: Proceedings of the 3rd International Conference on Big Data and Smart Computing (BigComp), 2016, pp. 199–206.
[27] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, et al., Moses: open source toolkit for statistical machine translation, in: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion (ACL), 2007, pp. 177–180.
[28] C. Lee, S. Lim, M.-G. Jang, Large-margin training of dependency parsers using pegasos algorithm, ETRI J. 32 (2010) 486–489.
[29] K.-S. Choi, Kaist language resources ver. 2001. The result of core software project from ministry of science and technology, 2001. ⟨http://semanticweb.kaist.ac.kr/home/index.php/KAIST_Corpus⟩.
[30] National Institute of Korean Language, Sejong Parallel Corpora. ⟨https://ithub.korean.go.kr/⟩.
[31] Text Retrieval Conference (TREC), Question Answering Collections, 2012. ⟨http://trec.nist.gov/data/qa.html⟩.

[32] G. Keppel, T.D. Wickens, Design and Analysis: A Researcher's Handbook, 4th Edition, Pearson Education Inc, USA, 2004.
[33] A. Carnie, Syntax: A Generative Introduction, 3rd Edition, Wiley-Blackwell, USA, 2012.
[34] C. Lee, P.-M. Ryu, H. Kim, Named entity recognition using a modified pegasos algorithm, in: Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM), 2011, pp. 2337–2340.
[35] A. Yoon, S.-H. Hwang, E. Lee, H.-C. Kwon, Construction of Korean wordnet KorLex 1.5, J. KIISE: Softw. Appl. 36 (2009) 92–108.
[36] A. Chagnaa, H.-S. Choe, C.-Y. Ock, H.-M. Yoon, On the evaluation of Korean wordnet, in: International Conference on Text, Speech and Dialogue, 2007, pp. 123–130.
[37] C. Liu, D. Dahlmeier, H. T. Ng, PEM: a paraphrase evaluation metric exploiting parallel texts, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2010, pp. 923–932.
[38] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, BLEU: a method for automatic evaluation of machine translation, in: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), 2002, pp. 311–318.
[39] G. Doddington, Automatic evaluation of machine translation quality using n-gram co-occurrence statistics, in: Proceedings of the Second International Conference on Human Language Technology Research (HLT), 2002, pp. 138–145.
[40] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, J. Makhoul, A study of translation edit rate with targeted human annotation, in: Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, 2006, pp. 223–231.
[41] J. M. Kim, H. Park, G. Gweon, J. Hur, The correlation between search quality and query popularity, in: Proceedings of the 3rd International Conference on Big Data and Smart Computing (BigComp), 2016, pp. 353–356.

Hancheol Park received his B.S. degree in industrial and information system engineering from Ajou University in 2012. He is now participating in the integrated Master's and Ph.D. program at the School of Computing, Korea Advanced Institute of Science and Technology (KAIST). His research interests are paraphrase extraction, generation, statistical machine translation, and sign language translation.

Kyo-Joong Oh received a Bachelor's degree in computer science in 2011 from the Korea Advanced Institute of Science and Technology (KAIST). He is currently an M.S./Ph.D. candidate in the Dept. of Computer Science at KAIST. His current research interests include artificial intelligence, machine learning, data mining, recommender systems, and natural language generation.

Ho-Jin Choi is currently an associate professor in the School of Computing at KAIST. In 1982, he received a B.S. in Computer Engineering from Seoul National University, Rep. of Korea. In 1985, he earned an MSc in Computing Software and Systems Design from Newcastle University, UK. In 1995, he obtained a Ph.D. in Artificial Intelligence from Imperial College, London, UK. Currently, he serves as a member of the board of directors for the Software Engineering Society of Korea, the Computational Intelligence Society of Korea, and the Korean Society of Medical Informatics. His current research interests include artificial intelligence, data mining, software engineering, and biomedical informatics.

Gahgene Gweon is an assistant professor at Seoul National University, at the Graduate School of Convergence Science and Technology. She received her B.A. in computer science and economics from the University of California, Berkeley. She also holds an M.S. and a Ph.D. in human-computer interaction from Carnegie Mellon University. Her research interests include natural language processing, human-computer interaction, learning science, and multimedia educational technology.