IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 11, 2019, accepted February 24, 2019, date of publication March 5, 2019, date of current version March 26, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2902270

Korean-Vietnamese Neural Machine Translation System With Korean Morphological Analysis and Word Sense Disambiguation

QUANG-PHUOC NGUYEN[1], ANH-DUNG VO[1], JOON-CHOUL SHIN[1], PHUOC TRAN[2], AND CHEOL-YOUNG OCK[1]

[1]Department of IT Convergence, University of Ulsan, Ulsan 44610, South Korea
[2]NLP-KD Lab, Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Vietnam

Corresponding author: Cheol-Young Ock (okcy@ulsan.ac.kr)

This work was partly supported by the National Research Foundation of Korea Grant funded by the Korean Government (NRF-2017M3C4A7068187) and the ICT R&D Program of MSIP/IITP (2013-0-00179, Development of Core Technology for Context-aware Deep-Symbolic Hybrid Learning and Construction of Language Resources).

ABSTRACT Although deep neural networks have recently led to great achievements in machine translation (MT), various challenges are still encountered during the development of Korean-Vietnamese MT systems. Because Korean is a morphologically rich language and Vietnamese is an analytic language, neither have clear word boundaries. The high rate of homographs in Korean causes word ambiguities, which causes problems in neural MT (NMT). In addition, as a low-resource language pair, there is no freely available, adequate Korean-Vietnamese parallel corpus that can be used to train translation models. In this paper, we manually established a lexical semantic network for the special characteristics of Korean as a knowledge base that was used for developing our Korean morphological analysis and word-sense disambiguation system: UTagger. We also constructed a large Korean-Vietnamese parallel corpus, in which we applied the state-of-the-art Vietnamese word segmentation method RDRsegmenter to Vietnamese texts and UTagger to Korean texts. Finally, we built a bi-directional Korean-Vietnamese NMT system based on the attention-based encoder-decoder architecture. The experimental results indicated that UTagger and RDRsegmenter could significantly improve the performance of the Korean-Vietnamese NMT system, achieving remarkable results by 27.79 BLEU points and 58.77 TER points in Korean-to-Vietnamese direction and 25.44 BLEU points and 58.72 TER points in the reverse direction.

INDEX TERMS Korean-Vietnamese machine translation, Korean-Vietnamese parallel corpus, lexical semantic network, morphological analysis, neural machine translation, word sense disambiguation.

I. INTRODUCTION

Neural machine translation based on the attention-based encoder-decoder model [1], [2] has emerged as the dominant paradigm in MT. It has achieved state-of-the-art performance in the translation of language pairs that have large amounts of training parallel corpora, such as English-French [3] and English-German [4]. However, it has shown poor translation quality in low-resource language pairs where training parallel corpora are scarce [5], [6].

Korean-Vietnamese is a low-resource language pair, and Korean-Vietnamese MT systems need to be built to serve the growing communication between South Korean and Vietnamese people. Since 2014, South Korea has been Vietnam's biggest investor based on foreign direct investment, establishing thousands of companies and factories in Vietnam [7]. Additionally, Vietnam ranks among the top three in terms of exports and the direct investment destination of South Korea, following the United States and China [8]. The Vietnamese community living in South Korea consists of 159,512 people, which is the second largest foreign community living in the country, according to statistics provided by the South Korea Immigration Service (http://www.immigration.go.kr; June 2018). Therefore, having high-quality Korean-Vietnamese MT systems is necessary.

The associate editor coordinating the review of this manuscript and approving it for publication was Yang Zhen.

2169-3536 © 2019 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

TABLE 1. An example of one *Hanja* that Represents Two Different *Hán_Việt*.

Chinese	Korean	Vietnamese	Meaning
家庭 [jiā-tíng]	가정 [ga-jeong]	gia đình [jia-dìng]	family
假定 [jiǎ-dìng]	가정 [ga-jeong]	giả định [jiả-dịng]	assumption

In this research, we defined four challenging issues encountered when building a high-quality Korean-Vietnamese NMT system and proposed solutions to address these issues.

Firstly, Korean is a synthetic language that synthesizes multiple concepts into an *eojeol* (i.e., a token unit delimited by whitespaces). This means that Korean does not have clear word boundaries, which is a major problem in Korean MT systems [9]. Therefore, every *eojeol* needs to be morphologically analyzed before being input into the NMT system. In this research, we constructed a pre-analyzed partial *eojeol* dictionary and combined this with the sub-word conditional probability to train our morphological analysis system.

Secondly, Vietnamese is an analytic language where one word consists of one or more tokens. The whitespaces cannot be used to determine Vietnamese word boundaries. Thus, in the training parallel corpus, we segmented Vietnamese words using RDRsegmenter [10], which was developed using the ripple down rules methodology [11] and achieves state-of-the-art performance.

Thirdly, approximately 65% of the vocabularies of Korean and Vietnamese contain words that originated from Chinese [12], [13], namely *hanja* for Sino-Korean and *Hán_Việt* for Sino-Vietnamese. There are five and six categories of tone marks (diacritics) in Chinese and Vietnamese, respectively, whereas there is only the neutral tone in Korean. Consequently, one *hanja* word may represent several different Chinese or *Hán_Việt* words. For instance, the *hanja* word "*ga-jeong*" represents two different *Hán_Việt* words: "*gia đình*" (family) and "*giả định*" (assumption), as shown in Table 1. This leads to a high rate of homographs in Korean.

However, handling homographs is a weakness of NMT [14]. In the word-embedding step, multiple senses of a word are encoded into one continuous vector. The NMT model must learn how to select the correct word from a group of candidates, which are translated from different senses of one input word. As a result, NMT has failed to disambiguate the word sense [15], [16]. In this research, we propose a Korean word sense disambiguation (WSD) based on our lexical semantic network (LSN) UWordMap, which was manually established for the special characteristics of Korean. Currently, UWordMap is the largest Korean LSN, enabling the high accuracy of our Korean WSD system.

Lastly, a high-quality NMT system requires a parallel corpus with a tremendous number of sentence pairs to train the translation model. Nevertheless, there is no freely available, adequate Korean-Vietnamese parallel corpus. OPUS [17] provides a Korean-Vietnamese parallel corpus extracted from movie subtitles and technical documents (i.e., GNOME and Ubuntu), but it is very noisy and its sentences are short. Therefore, we built a Korean-Vietnamese parallel corpus with 454,751 sentence pairs.

We established Korean-Vietnamese NMT systems based on the attention-based sequence-to-sequence architecture [1], [2]. The experimental results show many potential benefits as compared to existing MT systems that use both statistical-based and neural-based methods.

The Korean LSN UWordMap, which is the knowledge base for WSD, is described below in Section II. Because Korean morphological analysis is an initial step of WSD, both Korean morphological analysis and WSD systems are reported in Section III. Section IV presents the Vietnamese word segmentation system, and Section V gives information about the Korean-Vietnamese parallel corpus. In Section VI, we implement the Korean-Vietnamese NMT systems and report the experimental results. Finally, we summarize the related work in Section VII and present our conclusions in Section VIII.

II. KOREAN LEXICAL SEMANTIC NETWORK

Because semantic processing systems rely on LSN, most popular languages have their own LSNs, such as English WordNet [18], Chinese HowNet [19], and European languages' EuroWordNet [20]. For Korean, KorLex [21] and CoreNet [22] were constructed by translating and mapping from English and Japanese LSNs; ETRI lexical concept networks (LCNs) [23] were built for nouns and verbs only.

We have been working to manually establish UWordMap with the special characteristics of Korean since 2002. It now stands as the largest Korean LSN. UWordMap consists of lexical networks of nouns, predicates, and adverbs, as shown in FIGURE 1. In each network, a node is connected to others through six types of semantic relations: hyponymy, synonymy, similarity, antonymy, part-whole, and association relations. The predicate network is connected to the noun and adverb networks through subcategorization information.

In UWordMap, each node is comprised of a word and a sense code to represent a certain sense. The vocabulary and sense codes were extracted from the Standard Korean Language Dictionary (SKLD), which is the best Korean monolingual dictionary. Each sense code is defined by numerals to represent the special sense of a word in the SKLD. The number of sense codes of a word is identical to the number of senses that the word has.

A. LEXICAL SEMANTIC NETWORK FOR NOUNS

In the lexical semantic network for nouns (LSNN), the hyponymy is the fundamental relation, forming a hierarchical structure in which an upper node is a hypernym of lower nodes. This is the "IS-A" relation, where a node is connected to only one upper node that means a node does not have multiple superordinates. To construct the LSNN, we first made the basic framework by determining the set

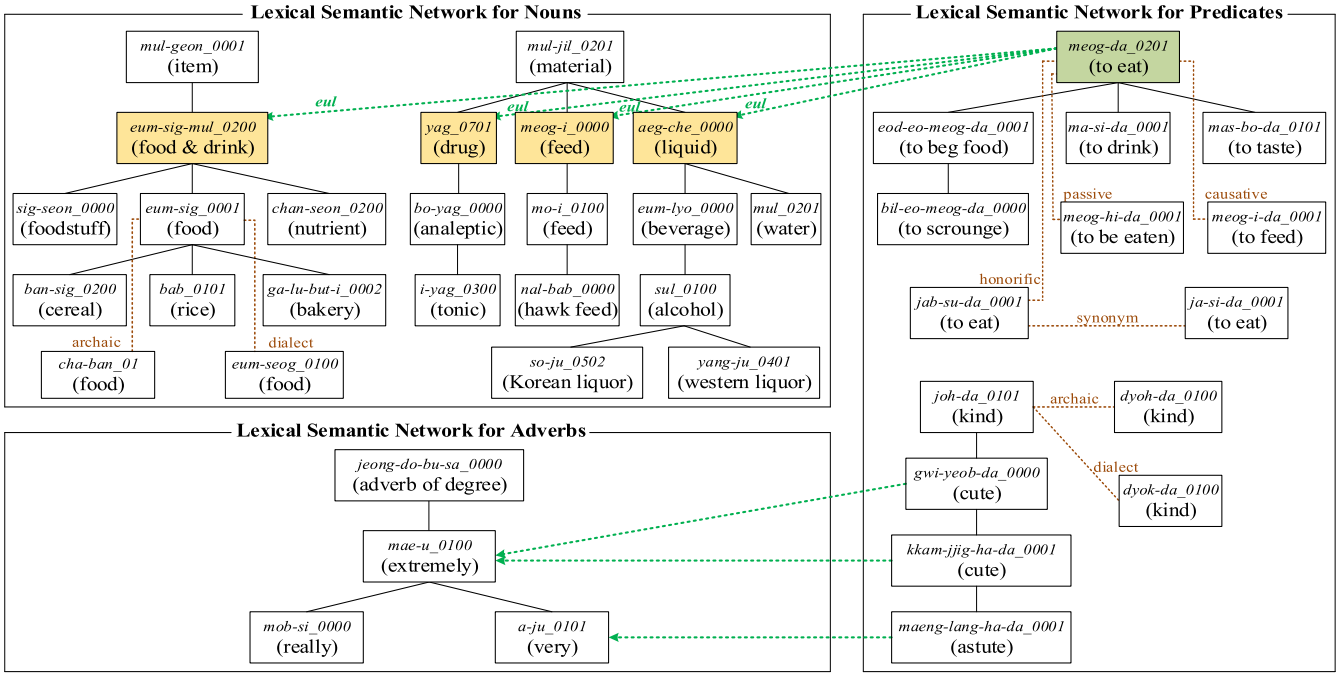

FIGURE 1. Overview of the Korean lexical semantic network UWordMap.

of top-level nodes that satisfy the following principles: have a clear meaning, are used frequently, do not have a duplicate concept with others, and composite meanings of lower nodes. As a result, we chose 23 top-level nodes: space, process, relation, symbol, unit, object, shape, item, method, scope, organism, characteristic, time, element, cognition, effect, material, degree, existence, kind or type, organization, action, and power.

Then, we considered both morphologic and semantic aspects to establish the hyponymy relation among nodes. The top-down and bottom-up methods were used to ensure the following principles.

- An upper node must contain the definition of its lower nodes.
- The information of a lower node must be derived from those of its upper node.
- For words borrowed from Chinese vocabulary, the semantic connections are made based on the core meaning of Chinese words.
- For words that have a suffix derived from Chinese characters, the relations are based on the suffixes. For instance, the words "*geon-chug-ga*" (architect) and "*gyo-yug-ga*" (educator) are connected to the upper node "*jeon-mun-ga*" (expert) based on the suf fix "*ga*" (specialist).
- For compound nouns, the relations are made based on the core meaning, which is usually stored in the left syllables.

Currently, we have constructed an LSNN with 377,961 words. The most common distributed depths are from 4 to 7 and the maximum depth is 17. FIGURE 2 shows the detailed distribution of nodes by depth.

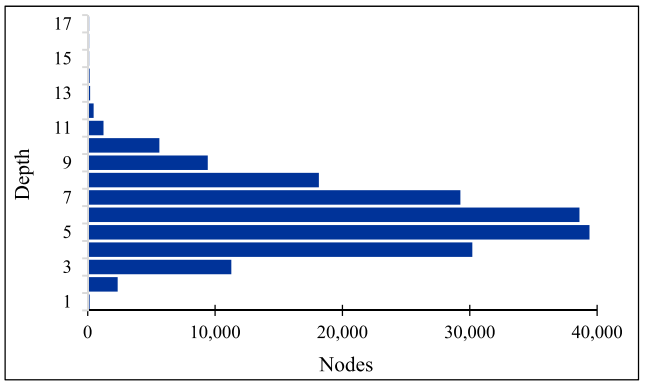

FIGURE 2. Distribution of nodes in the lexical semantic network for nouns.

B. LEXICAL SEMANTIC NETWORK FOR PREDICATES

In Korean, predicates embrace verbs and adjectives because of their similar grammatical constructions. In addition to the aforementioned semantic relations, the subcategorization is the most important element in the lexical semantic network for predicates (LSNP). It indicates the ability of predicates to allow the types of syntactic arguments (i.e., postpositional particles and nouns) with which they co-occur. For instance, Table 2 gives a part of the subcategorization of the verb "*geod-da*" (to collect or to walk). The particles are attached behind nouns to indicate their grammatical relation to the predicate. "*Eul*," "*e-ge-seo*," and "*e-seo*" indicate their attaching nouns are objects, peoples, and places, respectively.

Subcategorization information is used to connect nodes in the LSNP with those in the LSNN. However, each node in the LSNP can be connected to many nodes in the LSNN. To restrict the number of connections, we connect the

TABLE 2. A Part of the subcategorization of the verb "geod-da".

Predicate	Arguments	
	Postpositional particles	Nouns
geod-da_02 (to walk)	*eul*	*gil_0101* (street), *geoli_0101* (avenue), *gong-won_03* (park), ...
geod-da_04 (to collect / to gather)	*e-ge-seo*	*baeg-seong_0001* (subjects, the people)
	e-seo	*si-heom-jang_0001* (exam place, test site), *jib_0101* (house), …
	eul	*seong-geum_03* (donation), *hoe-bi_03* (fee, dues), *ssal_0003* (rice), ...

predicates with only the least common subsumes (LCS) in the LSNN. An LCS is the most specific common ancestor-node of two nodes in the hierarchical structure of the LSNN according to ontology theory. For instance, in FIGURE 1, instead of directly connecting the predicate "*meod-da_0201*" with all possible nodes in LSNN, we connected it with only LCSs (i.e., "*eum-sig-mul_0200*", "*yag_0701*", "*meog-i_0000*", and "*aeg-che_0000*").

We built the subcategorization for predicates based on example sentences in the SKLD. We manually extracted the predicate and its arguments (particles and noun) from the example sentence. The argument noun was checked to ensure that it exists and is the LCS in the LSNN. If it does not exist, we search for its hypernym and insert that it into the LSNN. If it is the LCS, we connect it with its predicate directly. Otherwise, we search its LCS to connect it with its predicate.

C. CURRENT STATUS OF UWORDMAP

We have built UWordMap, which contains 514,314 words, including nouns, verbs, adjectives, and adverbs. Its detailed statistics are shown in Table 3, and the number of words in UWordMap is compared with those in KorLex, CoreNet, and ETRI-LCN. The results show that UWordMap is the largest Korean LSN. UWordMap is available for online usage and its API libraries (C/C++/C#/JAVA/Python3 languages) can be downloaded at http://nlplab.ulsan.ac.kr/doku.php?id = uwordmap.

III. KOREAN MORPHOLOGICAL ANALYSIS AND WORD SENSE DISAMBIGUATION SYSTEM

A. OVERVIEW OF KOREAN MORPHOLOGICAL ANALYSIS

The problem with Korean morphological analysis is that several different morphemes and parts of speech (POS) may be encoded into the same *eojeol*. For instance, the four different sets of morphemes and POS shown in Table 4 can make up the same *eojeol*: "*ga-si-neun*". The phonemes in morphemes may be changed with many kinds of regularities and irregularities. Yet the same morphemes that are tagged with different POS have different meanings. The morphological analysis has to discover the correct set in a given context.

Most of the conventional methods [24]–[26] used in Korean morphological analysis have had to do the following tasks.

TABLE 3. Comparison of UWordMap and existing Korean word nets.

	Nouns	Verbs	Adjectives	Adverbs	Total
ETRI-LCN	49,000	30,000			79,000
CoreNet	51,607	5,290	2,801		58,985
KorLex	104,417	20,151	20,897	3,123	150,199
UWordMap	**293,547**	**78,563**	**18,539**	**105,450**	**496,099**

TABLE 4. The *Eojeol* "*ga-si-neun*" and its analyzable morphemes.

Eojeol	Morphemes	Meaning
ga-si-neun	*ga*/VV + *si*/EP + *neun* /ETM	to go (honorific form)
	gal/VV + *si*/EP + *neun*/ETM	to sharpen (honorific form)
	ga-si/VV + *neun*/ETM	to disappear, vanish
	ga-si/NNG + *neun*/JX	a prickle, thorn, or needle

VV, EP, ETM, NNG, and JX are the tagged POS indicating the intransitive verb, pre-final ending, suffix, noun, and auxiliary particle, respectively.

- Segment the input *eojeol* into morphemes
- Recover the changed phonemes to the original
- Assign or tag POS to each morpheme

Because these methods must perform various interim processes and transform character codes to recover the original form, they increase the frequency of dictionary accesses, which leads to overanalyzing problems. The longest match strategy [27] was proposed to reduce the frequency of dictionary accesses, and the syllable-based prediction model [28] was introduced to handle the overanalyzing problem. Recently, statistics-based [29]–[31] and deep learning-based approaches [32], [33] have been investigated to address these problems. However, the high computational complexity of these approaches causes low performance in the systems. Additionally, they cause maintenance problems when any neologism occurs in the language.

Using the pre-analysis *eojeol* dictionary (PED) [34]–[36] can overcome these problems. According to this method, a dictionary of analyzed *eojeol* is built in advance, and the problem turns into looking up morphologically analyzed *eojeol* in the PED. This method can be performed quickly because it does not need to identify the changed phonemes or recover the original form. It is also easily maintained by editing or inserting data into the PED. However, building a PED containing all *eojeols* of the Korean language is an impossible task.

Instead of using the PED, we constructed a pre-analysis partial *eojeol* dictionary (PPED). We then proposed a statistical-based method using a combination of the PPED and analyzed corpus to analyze the morphology. This method can take advantage of the fast performance and easy maintenance of the PED method, while also ensuring high accuracy.

B. CONSTRUCTION OF THE PRE-ANALYSIS PARTIAL EOJEOL DICTIONARY

As a dictionary, the PPED has a key and value for each entry. A key is a group of syllables separated from the surface form of an *eojeol* (the so-called surface form); the value may

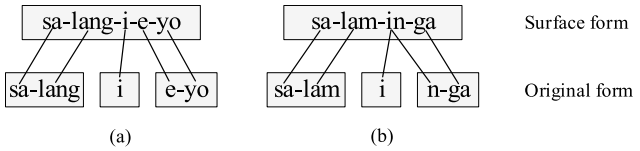

FIGURE 3. Examples of syllable connections between the surface and original form.

TABLE 5. Extracted PPED entries from the *Eojeol* "*sa-lam-in-ga*".

No.	Key	Value
1	sa-lam-in-ga	sa-lam_01/NNG + i /VCP + n-ga/EF
2	sa-lam-in*	sa-lam_01/NNG + i /VCP
3	sa-lam	sa-lam_01/NNG
4	in-ga	i /VCP + n-ga/EF
5	*in-ga	n-ga/EF

The number "01" is the sense code of "*sa-lam*"

consist of one or more analyzed morphemes (the so-called original form). The data that were used to build the PPED were extracted from the Sejong corpus [37]. To determine the key corresponding with a value, we make connections between syllables of the surface form and those of the original form. If the phonemic change makes the length of the original form longer than the surface form, one syllable of the surface form can be connected with two or more syllables of the original form; otherwise, we simply make connections syllable-by-syllable. For instance, there is no phonemic change in FIGURE 3 (a), whereas the syllable "*in*" is changed into "*i*" and "*n*" in FIGURE 3 (b). Hence, "*in*" is connected to both "*i*" and "*n*".

Table 5 shows five entries extracted from the *eojeol* "*sa-lam-in-ga*" (FIGURE 3 (b)). The values were made by enumerating all possible combinations of morphemes, and the keys were made by selecting the corresponding syllables according to the syllable connections. The asterisks "∗" indicate that the adjacent syllable has two connections with syllables in the original form but one of them is eliminated. The plus signs "+" are used to separate morphemes. The first entry of each *eojeol* is its whole morphological analysis. The PPED was constructed by examining all *eojeol* from the Sejong corpus.

In the PPED, the only key is not unique but the key and its values id unique. One *eojeol* can be morphologically analyzed into different morphemes and POS sets because of segmentation and POS ambiguity. Alternatively, it can be analyzed into the same morphemes and POS sets but different sense codes because of sense ambiguity. Hence, there are entries that have the same key but different values.

C. USING THE PRE-ANALYSIS PARTIAL EOJEOL DICTIONARY

The first step is to search the whole morphological analysis for the input *eojeol*. If it exists in the PPED, the sub-word conditional probability (SCP) method is used to select the correct one based on the adjacent *eojeols*. Otherwise, we split

TABLE 6. The order of the splitting process for the *Eojeol* "*sa-lam-in-ga*".

Order	Left	Right
1	sa-lam-in-ga*	*ga
2	sa-lam-in	ga
3	sa-lam-in*	*in-ga
4	sa-lam	in-ga
5	sa-lam*	*lam-in-ga
6	sa	lam-in-ga
7	sa*	*sa-lam-in-ga

the input *eojeol* into two parts: left and right. Then, we look up the analyzed morphemes for each part in the PPED.

The split process of an *eojeol* is started with the last syllable and executed from left to right. For instance, the *eojeol* "*sa-lam-in-ga*" is split into left and right parts, as shown in Table 6. With a left and right part pair, we may search more than one pair of analyzed morphemes, which we refer to as candidates. Since one *eojeol* is usually split into many pairs of left and right parts, there are many candidates for each *eojeol*.

To increase the performance of our WSD system, we need to reduce the number of candidates before inputting them into the SCP process. We select the top five candidates based on their scores. The score of a candidate is calculated by

$$score(C) = freq\,(L) \times freq\,(R) \times P\,(l_N, r_1) \times P_{first}\,(l_1) \times P_{last}(r_M), \quad (1)$$

where $L = l_1 + \cdots + l_N$ and $\mathrm{R} = r_1 + \cdots + r_M$ are the left and right partial *eojeol*, respectively, of the candidate C. l_i and r_i are the $i-th$ morphemes in the left and right partial *eojeol*, respectively. $freq\,(L)$ and $freq\,(R)$ are frequencies of the left and right partial *eojeol* occurring in the Sejong corpus

$$P\,(l_N, r_1) = freq\,(l_N, r_1)\,/freq(l_N) \quad (2)$$

Equation (2) is the probability that the last morpheme l_N of a left partial *eojeol* is adjacent to the fist morpheme r_1 of a right partial *eojeol* in the Sejong corpus. Additionally, $freq\,(l_N, r_1)$ is the frequency with which morphemes l_N and r_1 occur adjacently, and $freq(l_N)$ is the frequency that morpheme l_N occurs in the corpus

$$P_{First}\,(l_1) = freq_{First}\,(POS(l_1))\,/totalWord \quad (3)$$

Equation (3) is the probability that the morpheme l_1 occurs first in *eojeols*, where $freq_{First}\,(POS\,(l_1))$ is the number of *eojeols* that contain the tagging POS of the morpheme l_1 at the first morpheme. Additionally, *totalWord* is the total number of *eojeols* in the training corpus

$$P_{Last}\,(r_M) = freq_{Last}\,(r_M)\,/freq\,(r_M) \quad (4)$$

Equation (4) is the probability that the morpheme r_M occurs last in *eojeols*, and $freq_{Last}\,(r_M)$ is the number of *eojeols* that contain the morpheme r_M at the last morpheme. Additionally, $freq\,(r_M)$ is the frequency with which the morpheme r_M occurs in the training corpus.

D. SUB-WORD CONDITIONAL PROBABILITY

After creating a list of candidates, this step selects the only correct one based on the adjacent *eojeols*. In this paper, we propose a method using sub-word information of adjacent *eojeols* to identify the correct candidate; this is referred to as a WSD process. We assume that the correct candidate can be identified based on only one left and one right adjacent *eojeols*. We defined a formula to identify the correct candidate:

$$WSD\,(w_i) = argmax_j P(c_{i,j}|w_{i-1}, w_i, w_{i+1}), \tag{5}$$

where

$$\begin{aligned} P(c_{i,j}|w_{i-1}, w_i, w_{i+1}) &\simeq P(c_{i,j}|w_{i-1}, w_i) \times P(c_{i,j}|w_i, w_{i+1}) \\ P_{Left} &= P(c_{i,j}|w_{i-1}, w_i) \\ P_{Right} &= P(c_{i,j}|w_i, w_{i+1}) \end{aligned}$$

Then,

$$WSD\,(w_i) = argmax_j \left(P_{Left} \times P_{Right}\right) \tag{6}$$

where w_i is the $i-th$ *eojeol* of the sentence $w_1 w_2 \ldots w_n$. Additionally, $c_{i,j}$ is the $j-th$ candidate of the $i-th$ *eojeol* and w_{i-1} and w_{i+1} are the left-adjacent and right-adjacent *eojeols* of the current $(i-th)$ eojeol, respectively. Table 7 gives an example of two candidates selected by analyzing the *eojeol* "*sa-gwa-leul*" in the sentence "*mas-iss-neun sa-gwa-leul meog-eoss-da*" (i.e., I ate a delicious apple).

In Korean, the first syllables often express the core meaning of an *eojeol*, while the last syllables often express its grammatical relations. For instance, "*meog-eoss-da*", "*meog-eoss-eu-na*", and "*meog-eoss-gess-ji*" have the same core meaning (ate), which is indicated by the first two syllables "*meog-eoss*". Examining all syllables of an *eojeol* may lead to problems related to a lack of training data (i.e., there is not "*meog-eoss-da*" in the training data set, but we can identify its meaning by using the first syllables "*meog-eoss*"). Hence, we only consider the first two syllables of each *eojeol* to identify the correct candidate. In this step, as we examine the surface forms of the left and right adjacent *eojeols*; P_{Left} and P_{Right} are replaced by P_{Left_Surf} and P_{Right_Surf}, respectively. P_{Left_Surf} and P_{Right_Surf} are computed by

$$P_{Left\_Surf} = P\left(m_{i,j,1}|w_{i-1}, s_{i,1}, s_{i,2}\right)^U \times P(c_{i,j}|w_i) \tag{7}$$

$$P_{Right\_Surf} = P(c_{i,j}|w_i, s_{i+1,1}, s_{i+1,2}). \tag{8}$$

P_{Left_Surf} is computed based on the entire left-adjacent *eojeol* w_{i-1} and the first two syllables $s_{i,1}$ and $s_{i,2}$ of the current *eojeol*. Since only the two syllables $s_{i,1}$ and $s_{i,2}$ are used, only the first morpheme $m_{i,j,1}$ of the $j-th$ candidate of the $i-th$ *eojeol* is computed. In Table 7, for instance, $m_{2,1,1}$ = "*sa-gwa_05*/NNG" and $M_{2,2,1}$ = "*sa-gwa_08*/NNG". Because the remaining morphemes of the current *eojeol* are not involved in the computation of P_{Left_Surf}, we need to compute the probability of each candidate given its *eojeol* $P(c_{i,j}|w_i)$. In addition, to judge the relative importance of the probability of the entire candidate and the probability of the first morpheme, we use the weight U in Equation (7).

TABLE 7. An example of WSD candidates of the *Eojeol* "*sa-gwa-leul*".

w_1	w_2	w_3
mas-iss-neun (delicious)	*sa-gwa-leul* (apple)	*meog-eoss-da* (ate)
	$c_{2,1}$: *sa-gwa_05*/NNG + *leul*/JKO	
	$c_{2,2}$: *sa-gwa_08*/NNG + *leul*/JKO	

$c_{2,1}$ and $c_{2,2}$ are candidates of the *eojeol* "*sa-gwa-leul*." "*sa-gwa_05*" means an apple, whereas "*sa-gwa_08*" means an apology.

TABLE 8. An example of different word stems extracted from an *Eojeol*.

Eojeol	Word Stem	Function Word
gan-da	v_1: *ga_01*/VV (to go)	*n-da*/EF
	v_2: *ga_01*/VX (on going)	*n-da*/EF
	v_3: *gal_01*/VV (to replace)	*n-da*/EF
	v_4: *gal_02*/VV (to grind)	*n-da*/EF

P_{Right_Surf} is simply computed based on the entire current *eojeol* w_i and the first two syllables $s_{i+1,1}$ and $s_{i+2,2}$ of the right-adjacent *eojeol*.

Using the surface form of *eojeols* can lead to high-performing systems because it is not time-consuming to analyze *eojeols*. However, systems must be able to overcome a lack of training data. When surface forms cannot be used to identify the correct candidate (i.e., $P_{Left\_Surf} = 0$ or $P_{Right\_Surf} = 0$), we analyze *eojeols* and use their word stems. For instance, in the phrase "*sa-gwa-leul meog-eul-lae*," the meaning of "*sa-gwa*" cannot be identified by using the two syllables "*meog-eul*", which do not exist in the training dataset. However, by using "*meog_02*/VV," (i.e., the word stem of "*meog-eul-lae*", which does exist in the training dataset), we can identify the meaning of "*sa-gwa*".

When using the word stem, P_{Left} and P_{Right} are replaced by P_{Left_Stem} and P_{Right_Stem}, respectively. Since several different word stems may be extracted from the same *eojeol*, P_{Left_Stem} and P_{Right_Stem} are computed by picking word stems that maximize the following conditional probabilities

$$P_{Left\_Stem} = argmax_k (P\left(m_{i,j,1}|w_{i-1}, v_{i,k}\right)^U \times P(c_{i,j}|w_i)) \tag{9}$$

$$P_{Right\_Stem} = argmax_k P(c_{i,j}|w_i, v_{i+1,k}) \tag{10}$$

Here, $v_{i,k}$ is the $k-th$ word stem of the $i-th$ *eojeol*. For instance, Table 8 shows four word stems extracted from the same *eojeol* "*gan-da*", and k runs from 1 to 4. Since the word stem is always contained in the first morpheme, only the first morpheme $m_{i,j,1}$ is computed in Equation (9). This is equivalent to Equation (7), where we need to calculate the probability for each candidate $P(c_{i,j}|w_i)$ and use the weight U.

In summary, P_{Left} and P_{Right} are calculated by the proposed SCP method

$$P_{Left} = \begin{cases} P_{Left\_Surf} & if\ P_{Left\_Surf} > 0 \\ P_{Left\_Stem} & if\ P_{Left\_Surf} = 0 \end{cases} \tag{11}$$

$$P_{Right} = \begin{cases} P_{Right\_Surf} & if\ P_{Right\_Surf} > 0 \\ P_{Right\_Stem} & if\ P_{Right\_Surf} = 0 \end{cases} \tag{12}$$

E. KNOWLEDGE-BASED WORD SENSE DISAMBIGUATION

The SCP is a corpus-based approach to WSD that must overcome the problem of missing training data, even if using the word stem. Each noun can be connected with various verbs and adjectives; however, the training corpus cannot contain all nouns and their connectable verbs and adjectives. Even advanced methods, such as statistical-based [38], deep learning-based with recurrent neural networks [39], and embedded word space [40] methods, still encounter the missing data problem because of limited training corpora.

Knowledge-based approaches can overcome this problem, but they require an accurate and large lexical network [41], [42]. Korean WordNet KorLex [21], which was constructed by translating English WordNet to Korean, is either used as a knowledge base [43] or combined with the Korean monolingual dictionary [44]. However, because of the limited lexicons of KorLex (refer to Table 3) and the difference between characteristics of English and those of Korean (i.e., ''to take medicine'' in English but ''to eat medicine'' in Korean), the accuracy is insufficient.

Moreover, corpus-based approaches suffer from the neological problem, which requires approaches using WSD models to be re-trained when a neologism occurs. For instance, corpus-based approaches cannot identify the sense of ''*tta-leu-da*'' (follow, respect, or pour) in the sentence ''*le-deu-bul-eul tta-leu-da…*'' (I pour Red Bull...); this is the case because ''*le-deu-bul*'' is a neologism that does not exist in the training corpus. In this case, knowledge-based WSD systems are easily maintained by adding ''*le-deu-bul*'' to the hypernym (beverage), and the sense of ''*tta-leu-da*'' can be identified as ''pour'' based on the hypernym beverage.

In this paper, we use UWordMap as a knowledge base to disambiguate Korean word senses. UWordMap has been constructed with the special characteristics of Korean. UWordMap contains subcategorization information that defines the connections between each predicate with LCSs in the LSNN through postpositional particle arguments. Based on this subcategorization, we can generate more sentences for the training corpus. Using the subcategorization in Table 2, we generated the following sentences.

gil_0101/NNG *eul*/JKO *geod-da_02*/VV.
baeg-seong_0001/NNG *e-ge-seo*/SRC *geod-da_04*/VV.
si-heom-jang_0001/NNG *e-seo*/LOC *geod-da_04*/VV.
seong-geum_03/NNG *eul*/JKO *geod-da_04*/VV.
…

Moreover, we can also generate sentences by connecting the predicate with hyponyms of the LCS. For instance, using the hyponyms of ''*gil_0101*'' shown in FIGURE 4, we generated more sentences:

mi-lo_0101/NNG eul/JKO geod-da_02/VV.
san_gil_02/NNG *eul*/JKO *geod-da_02*/VV.
deung-san-lo_01/NNG *eul*/JKO *geod-da_02*/VV.
…

There are 421 direct hyponyms of the Korean noun ''*gil_0101*''. Each of these also has a large number of its own hyponyms. If Korean sentences are generated by this method, the training corpus will be expanded significantly to solve the missing data problem.

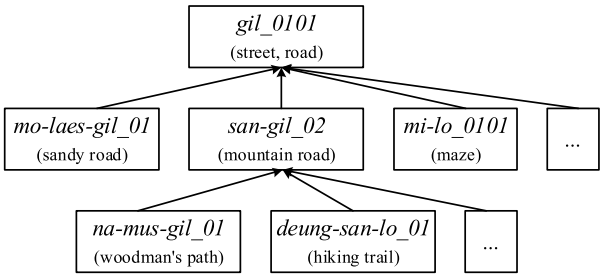

FIGURE 4. Hypernyms of the Korean noun "gil_0101" in the LSNN.

There is another way to exploit UWordMap. When calculating P_{Left} and P_{Right} in the SCP method, if $P_{Left\_Surf} = 0$ or $P_{Right\_Surf} = 0$, we can replace the examining noun by its hypernym and re-calculate P_{Left} and P_{Right}. If the sense still cannot be identified, even when examining the hypernym, we continue replacing the noun with hypernyms of the hypernym in a looping process. This process stops when the sense is identified or the hypernym is the top-level node. All processes of the morphological analysis and WSD system are shown in FIGURE 5.

F. EXPERIMENTAL RESULTS

Based on the method proposed above, we developed a Korean morphological analysis and WSD system: UTagger. UTagger's model was trained on UWordMap and the Sejong corpus. UWordMap is described in Section II, and the Sejong corpus consists of 11 million *eojeols* that have been morphologically analyzed and tagged with POS. The homographs in the Sejong corpus were tagged with sense codes, which are identical to those in the SKLD.

We extracted sentences that had orders divisible by 10 from the Sejong corpus; as a result, we obtained 1,108,204 *eojeols* (10%) as the evaluating dataset. In Equations (7) and (9), we chose the weight $U = 2.5$ to maximize the system accuracies after repetitive experiments with various values. We evaluated UTagger on a system with an i7 860 (2.8 GHz) CPU core and 16 GB of RAM. The accuracy of UTagger reached 98.2% for morphological analysis and 96.52% for WSD. It could process approximately 30,000 *eojeols* per second.

We compared the accuracies of UTagger with those of recent machine learning methods. These methods also used the same Sejong corpus to train and evaluate their systems. The morphological analysis accuracy of UTagger was compared with those of the conditional random fields (CRF) [30], phrase-based statistical model (PSM) [31], recurrent neural network-based with copying mechanism (RNN-CM) [32], and bi-long short-term memory (Bi-LSTM) [33] methods. The WSD of UTagger was compared with those of the statistical-based [38], bidirectional recurrent neural network (Bi-RNN) [39], and embedded word space (EWS) [40] methods. The results in Table 9 show that UTagger achieves

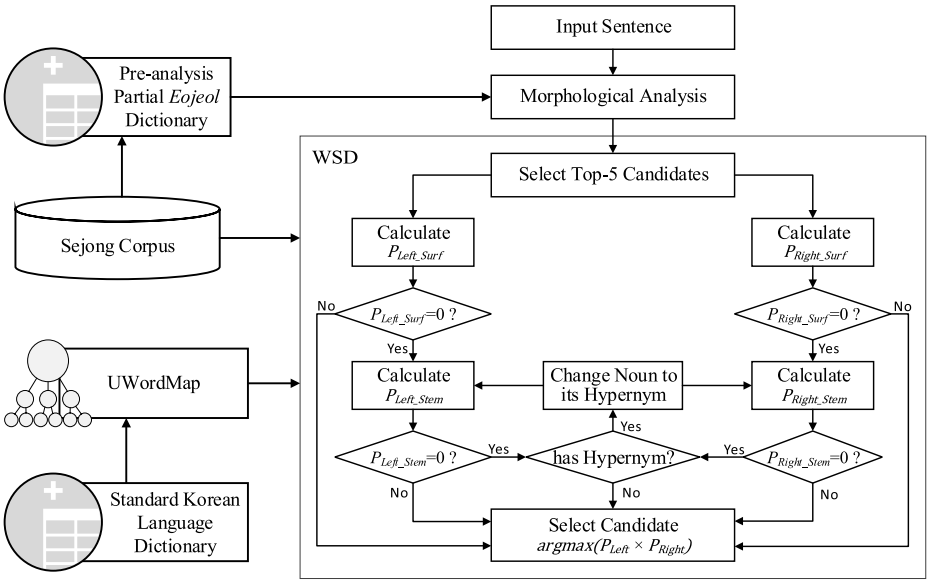

FIGURE 5. Korean morphological analysis and word sense disambiguation system architecture.

TABLE 9. Accuracies of morphological analysis and WSD.

Methods	Morph. Ana.	WSD
CRF (Na, 2015) [30]	95.22%	
PSM (Na et al., 2018) [31]	96.35%	
RNN-CM (Jung et al. 2018) [32]	97.08%	
Bi-LSTM (Matteson et al. 2018) [33]	96.20%	
Statistical-based (Shin and Ock, 2014)[38]		96.42%
Bi-RNN (Min et al., 2017) [39]		96.20%
EWS (Kang et al., 2017) [40]		85.50%
UTagger	**98.2%**	**96.52%**

state-of-the-art accuracies for both Korean morphological analysis and WSD.

UTagger is available for online use and its libraries (Linux and Windows systems with C/C++/C#/JAVA/Python3 languages) can be downloaded at http://nlplab.ulsan.ac.kr/doku.php.

IV. VIETNAMESE WORD SEGMENTATION

The writing system for Vietnamese includes Latin script and five kinds of diacritics (i.e., *á*, *à*, *ả*, *ã*, and *ạ*). In this writing system, blank space is not used to separate words; instead, it is only used to separate syllables. 84.31% of Vietnamese words are composed of at least two syllables [45]. Hence, we cannot determine Vietnamese word boundaries based on blank space.

Furthermore, each syllable in Vietnamese has meanings by itself. For instance, the word "*nhữ ng đất nư ớ c*" (countries) has three syllables separated by blank spaces. The first syllable "*nhữ ng*" indicates that this word in the plural, the second "*đất*" means the soil or land, and the last syllable "*nhữ ng*" means the water. The individual meanings of syllables are different from the meaning of a word composed by them. This causes rare and ambiguous word problems in NMT.

The function of a Vietnamese word segmentation system is to identify words (i.e., groups of syllables) in the input sentences and replace blank spaces inside words with underscores (i.e., "*nhữ ng_đất_nư ớ c*"). After the word segmentation process, the blank space is used to separate words and becomes the word boundary's indicator.

Many approaches have been investigated to develop word segmenters for Vietnamese. CLC_VN_WS was built based on the maximum entropy model [46], which was trained on the corpus for word sense disambiguation with 3M Vietnamese syllables [47]. Nguyen *et al.* [48] developed JVnSegmenter using the conditional random fields and support vector machine models and trained their models on a small corpus of 8K sentences that they built themselves. vnTokenizer [45] was developed by combining the maximum matching strategy and the finite-state automata technique, and its model was trained on a corpus of 507K words that were manually segmented by the Vietnam Lexicography Center. DongDu [49], UETsegmenter [50], and RDRsegmenter [10] were trained on the same corpus, i.e., Vietnamese treebank [51], which consists of over 1.7M words. Where DongDu and UETsegmenter are based on the pointwise method [52], RDRsegmenter is based on the ripple down rules method [11].

The comparison between these segmenters using the same system and test set demonstrates that RDRsegmenter outperforms the others and achieves a precision of 97.46%, recall of 98.35%, and F1 of 97.90% [10]. It is also the fastest segmenter, analyzing 62K words per second; the segmenter was run on a personal computer with a Core i7 2.2 GHz CPU. Hence, in this paper, we used RDRsegmenter to process the Vietnamese text in our parallel corpus.

V. KOREAN-VIETNAMESE PARALLEL CORPUS

In addition to the methodology, the most essential constituent of MT systems is the parallel corpus, which is used to train the translation models. A high-quality MT system requires a parallel corpus with a large number of qualified aligning sentence pairs. Manually compiled parallel corpora, which require great time and effort to produce, are usually used for commercial purposes. However, automatically collected parallel corpora have been produced and are available for research applications [53]–[55]. Unfortunately, these are limited to popular language pairs.

For the Korean-Vietnamese language pair, several groups have tried to build parallel corpora. The computational linguistics center (University of Science, HCM City) built a Korean-Vietnamese bilingual corpus [56] with 500K sentence pairs (14.5M words), but this is used for commercial purposes. Nanyang Technological University NTU-MC [57] and OPUS [17] provided multilingual corpora that included Korean and Vietnamese. However, NTU-MC is very small and contains only 15K sentences. Additionally, since OPUS was extracted from movie subtitles and technical documents (i.e., GNOME and Ubuntu), it is very noisy and its sentences are short. These corpora are insufficient to train a qualified MT system.

In this paper, we built a large-scale Korean-Vietnamese parallel corpus for training our MT systems by collecting bilingual aligned texts from many resources. We extracted definition statements and examples of Korean- Vietnamese pairs from the National Institute of Korean Language's Learner Dictionary.[1] We also downloaded and aligned Korean-Vietnamese texts from articles in multilingual magazines and books, such as "Watchtowers and Awake![2]", "Books & Brochures [2]", and "Rainbow[3]", which include many categories of text (economics, health, entertainment, science, social issues, politics, and technology). These resources are well-aligned and well-translated. We also crawled texts from online journals and websites that contain Korean and Vietnamese. Because these contain many mismatches, we had to carefully select and filter the texts to ensure good alignment.

Next, we removed noise from the collected Korean- Vietnamese bilingual aligned texts. Noise comes in the form of messy codes, HTML tags, and special symbols and characters used on websites. We also removed long sentences, which can crash MT systems. In this corpus, we define a long sentence as a sentence with over 80 words. We also removed duplicate sentences, which sometimes occurred because we collected texts from so many resources. The corpus was re-corrected by splitting sentences, and each sentence was stored in one line on a disk file. As a result, we obtained over 454K Korean-Vietnamese sentence pairs with 5M Korean and 8.5M Vietnamese tokens. This is large enough to train MT systems.

[1]https://krdict.korean.go.kr
[2]https://www.jw.org/en/publications/
[3]https://www.liveinkorea.kr

TABLE 10. Sentence transform after applying RDRsegmenter and UTagger.

Original form
Kr: *manh-eun sa-lam-eun wae se-sang sang-tae-ga ag-hwa-doe-go iss-da-neun geos-eul kkae-dad-ji mos-hab-ni-kka*?
Vn: *tại sao nhiều người không nhận thấy tình hình xã hội ngày càng tồi tệ*?
Form after applying UTagger and RDRsegmenter
Kr: *manh*/VA *eun*/ETM sa-lam/NNG eun/JX wae_02/MAG se-sang_01/NNG sang-tae_01/NNG ga/JKS ag-hwa-doe/VV go/EC iss_01/VX da-neun/ETM geos_01/NNB eul/JKO kkae-dad/VV ji/EC mos-ha/VX b-ni-kka/EC ?/SF
Vn: *tại_sao nhiều người không nhận thấy tình_hình xã_hội ngày_càng tồi_tệ* ?

TABLE 11. Statistics of the Korean-Vietnamese parallel corpus.

		#Sent.	#Avg. Len.	#Tokens	#Vocabulary
Vietnamese	Original	454,751	19.3	8,790,197	40,090
	Segmented		16.3	7,409,163	49,208
Korean	Original		12.0	5,435,686	397,130
	Morph. Ana.		21.4	9,728,801	63,735
	WSD				68,856

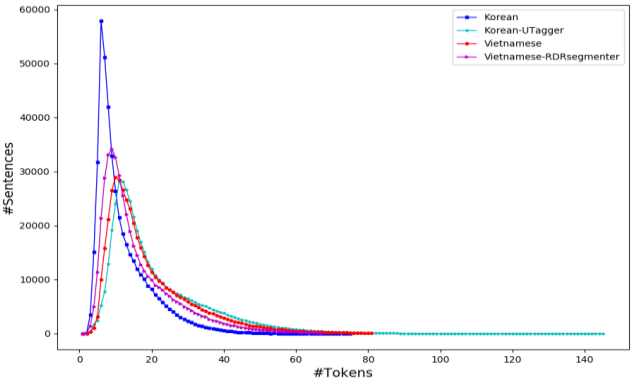

FIGURE 6. The distributions of sentence lengths in the Korean-Vietnamese parallel corpus.

The Korean texts in the corpus underwent morphological analysis and WSD with UTagger. Morphological analysis segmented each Korean word into morphemes and recovered the original forms. Morpheme segmentation increased the token size, and recovering the original forms reduced the vocabulary size. Because WSD tagged different sense codes into the same form of words, it increased the vocabulary size.

The Vietnamese texts in the corpus underwent word segmentation with RDRsegmenter. RDRsegmenter merged tokens into one word, consequently reducing the token size and increasing the vocabulary size. Table 10 shows a sample of how a sentence pair was transformed after applying RDRsegmenter and UTagger.

Table 11 gives a detailed statistical report regarding the number of sentences, tokens, and vocabularies, as well as the average sentence length of each language in the Korean-Vietnamese parallel corpus. FIGURE 6 presents the

distribution of sentence lengths in the corpus, where the sentence length is counted based on the number of tokens. Most sentences are 5 to 35 tokens in length. We limited the sentence length to 80 tokens; however, UTagger extended the Korean sentence length to 129 tokens and RDRsegmenter cut the Vietnamese sentence length down to 67 tokens.

VI. EXPERIMENTS AND RESULTS

A. NEURAL MACHINE TRANSLATION

Recent NMT systems usually employ an attention-based encoder-decoder architecture [1], [2]. The encoder and decoder are recurrent neural networks (RNNs), which are implemented as long short-term memory (LSTM) networks [58] or gated recurrent units [59] with a single or multiple hidden layers.

1) ENCODER

The encoder is composed of forward and backward RNNs. The forward RNN reads a source sentence $x = (x_1, x_2, \ldots, x_{T_x})$ from x_1 to x_n and computes forward states $(\overrightarrow{h}_1, \overrightarrow{h}_2, \ldots, \overrightarrow{h}_{T_x})$. Alternatively, the backward RNN processes the sentence from x_n to x_1 (i.e., the reverse direction) and calculates backward states $(\overleftarrow{h}_1, \overleftarrow{h}_2, \ldots, \overleftarrow{h}_{T_x})$. Here, T_x refers to the length of sentence x.

Each forward state is updated by

$$\overrightarrow{h}_t = \begin{cases} (1 - \overrightarrow{z}_1) \circ \overrightarrow{h}_{t-1} + \overrightarrow{z}_t \circ \underline{\overrightarrow{h}}_t & if\ t > 0 \\ 0 & if\ t = 0 \end{cases} \quad (13)$$

where

$$\underline{\overrightarrow{h}}_t = \tanh\left(\overrightarrow{W}\overline{E}x_t + \overrightarrow{U}\left[\overrightarrow{r} \circ \overrightarrow{h}_{t-1}\right]\right) \quad (14)$$

$$\overrightarrow{z}_t = \sigma\left(\overrightarrow{W}_z\overline{E}x_t + \overrightarrow{U}_z\overrightarrow{h}_{t-1}\right) \quad (15)$$

$$\overrightarrow{r}_t = \sigma\left(\overrightarrow{W}_r\overline{E}x_t + \overrightarrow{U}_r\overrightarrow{h}_{t-1}\right) \quad (16)$$

$\overline{E}x_t$ denotes the embedding of the source word x_t, $\overrightarrow{W}_*$ and $\overrightarrow{U}_*$ refer to weight matrices, and σ is used as a logistic function.

The backward states $(\overleftarrow{h}_1, \overleftarrow{h}_2, \ldots, \overleftarrow{h}_{T_x})$ are calculated in a similar manner, using the same word embedding E as forward states but different weights $\overleftarrow{W}_*$ and $\overleftarrow{U}_*$. Then, the source representations $(h_1, h_2, \ldots, h_{T_x})$ are acquired by concatenating the two sequences of states.

2) DECODER

The decoder is an RNN that generates the translated sentence $y = (y_1, y_2, \ldots, y_{T_y})$, where T_y is the sentence length. Each word y_i is generated through the conditional probability

$$p(y_i | \{y_1, \ldots, y_{i-1}\}, x) = g(y_{i-1}, s_i, c_i), \quad (17)$$

where g is used as a nonlinear function. s_i denotes the decoding state at the $i - th$ time calculated by

$$s_i = \begin{cases} (1 - z_i) \circ s_{i-1} + z_i \circ \tilde{s}_i & if\ i > 0 \\ \tanh(W_s h_1) & if\ i = 0 \end{cases} \quad (18)$$

where

$$\tilde{s}_i = \tanh\left(WEy_{i-1} + U\left[r_i \circ s_{i-1}\right] + Cc_i\right) \quad (19)$$

$$z_i = \sigma\left(W_zEy_{i-1} + U_zs_{i-1} + C_zc_i\right) \quad (20)$$

$$r_i = \sigma\left(W_rEy_{i-1} + U_rs_{i-1} + C_rc_i\right). \quad (21)$$

E is the embedding matrix of the word's language, and W_*, U_*, and C_* are weight matrices. c_i, which is the source context vector at the $i - th$ time, is calculated based on the source representation h_j:

$$c_i = \sum_{j=1}^{T_x} \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})} h_j. \quad (22)$$

Here, e_{ij} is an attention-based model calculated by

$$e_{ij} = v_a^T \tanh\left(W_a s_{i-1} + U_a h_j\right), \quad (23)$$

where v_a^T, W_a, and U_a are weight thresholds.

3) IMPLEMENTATION

In this work, we implemented our Korean-Vietnamese NMT systems using deep multi-layer LSTM networks for both the encoder and decoder. We used the TensorFlow-based sequence-to-sequence model [60] to train and test the systems. The testing dataset consists of 2,000 Korean-Vietnamese sentence pairs that were randomly extracted from the Korean-Vietnamese parallel corpus mentioned in section V. The remaining pairs were used to train the systems. The LSTM networks were set to be two layers x 512 units. We set the word-embedding dimension as 512 for both source and target languages (i.e., Korean and Vietnamese). We fed 13 epochs into the training processes.

We applied these settings for bi-directional translation of Korean-to-Vietnamese and Vietnamese-to-Korean. To measure the impact of the Korean morphological analysis, Korean WSD, and Vietnamese word segmentation on the translation qualities individually, we built five systems (i.e., Baseline, Morph. Anal., WSD, Word Seg., and Full) for each direction.

- *Baseline*: Uses the original Korean and Vietnamese texts in the Korean-Vietnamese parallel corpus (Table 11).
- *Morph. Anal.*: Uses the Korean texts after applying UTagger, but with sense code tags removed (i.e., Korean morphological analysis only), and the original Vietnamese texts.
- *UTagger*: Uses the Korean texts after applying UTagger (i.e., Korean morphological analysis and WSD) and the original Vietnamese texts.
- *RDRsegmenter*: Uses the original Korean texts and the Vietnamese texts after applying word segmentation with RDRsegmenter.
- *UTagger + RDRsegmenter*: Uses the Korean texts after applying UTagger and the Vietnamese texts after applying RDRsegmenter.

To compare the translation qualities of these systems with those of statistical machine translation (SMT) systems,

TABLE 12. Translation results.

Systems		BLEU	TER
SMT Korean-to-Vietnamese	Baseline	20.62	71.75
	Morph. Anal.	25.43	67.43
	UTagger	25.45	66.64
	RDRsegmenter	21.88	69.58
	UTagger + RDRsegmenter	26.42	65.82
NMT Korean-to-Vietnamese	Baseline	24.66	66.32
	Morph. Anal.	25.53	59.66
	UTagger	25.61	59.52
	RDRsegmenter	25.64	65.10
	UTagger + RDRsegmenter	**27.79**	**58.77**
SMT Vietnamese-to-Korean	Baseline	9.83	84.71
	Morph. Anal.	22.09	73.42
	UTagger	22.27	73.44
	RDRsegmenter	11.17	81.34
	UTagger + RDRsegmenter	24.22	70.38
NMT Vietnamese-to-Korean	Baseline	10.76	70.94
	Morph. Anal.	23.70	59.03
	UTagger	24.07	58.85
	RDRsegmenter	12.88	70.61
	UTagger + RDRsegmenter	**25.44**	**58.72**

as well as to evaluate the effectiveness of the proposed methods on SMT systems, we also built five translation systems (i.e., Baseline, Morph. Anal., UTagger, RDRsegmenter, and UTagger + RDRsegmenter) based on the SMT architecture using the Moses toolkit [61].

B. RESULTS

We used two metrics, i.e., BLEU [62] and TER [63], to automatically evaluate the translation qualities of our Korean-Vietnamese MT systems. BLEU (bilingual evaluation understudy) is the most common algorithm used to automatically evaluate the quality of MT systems. It computes the translated precision by counting the number of matches between n-grams of a machine-translated sentence and those of the corresponding reference. TER (translation error rate) determines the number of edits needed such that a machine-translated sentence exactly matches the corresponding reference. The translation results of the systems in terms of BLEU and TER scores are shown in Table 12.

1) COMPARISON BETWEEN SMT AND NMT

For both translation directions, NMT systems outperform SMT systems in terms of the BLEU and TER scores. In the baseline systems, NMT outperforms SMT by 4.06 and 0.93 BLEU points or 5.43 and 13.77 TER points for Korean-to-Vietnamese and Vietnamese-to-Korean directions, respectively. After using RDRsegmenter to segment Vietnamese words and UTagger to analyze Korean morphologies and disambiguate homographic senses, NMT still outperforms SMT by 1.37 and 1.22 BLEU points or 7.08 and 11.66 TER points for both translation directions. This is similar to the results of popular languages pairs, such as English, Chinese, French, German, Russian, and Spanish [64], [65], in which NMT is better than SMT.

2) IMPACT OF KOREAN MORPHOLOGICAL ANALYSIS

The Korean morphological analysis improved the translated results of both SMT and NMT baseline systems. Particularly, it improved the BLEU points by 0.87 and 12.94 for NMT systems in Korean-to-Vietnamese and Vietnamese-to-Korean translation directions, respectively. It also prevented translation errors in NMT systems (by 6.66 and 11.91 TER points) for both translation directions. The morphological complexity of Korean causes problems in MT related to word boundaries and rare words. Korean does not have clear word boundaries; one Korean token (*eojeol*) usually consists of one content word and one or more function words. Rare words are also challenging for NMT [3], leading to a large number of out-of-vocabulary (OOV) words. For instance, the Korean verb "*meog-da*" (to eat) has many forms, such as "*meog-eo-yo*", "*meog-seum-ni-da*", and "*meog-neun-da*", among others. Morphological analysis segments these into the stem word "*meog*" (eat) and ending words "*da*", "*eo-yo*", and "*seum-ni-da*" (grammatical expressions). This creates clear boundaries and reduces the number of OOV words. Hence, morphological analysis significantly improves the qualities of NMT systems.

3) IMPACT OF WORD SENSE DISAMBIGUATION

NMT systems are incapable of translating homographs [14]–[16] because multiple senses of a word are encoded into one continuous vector that forces NMT systems to select the correct word from a group of candidates that are translated from different senses of one input word. UTagger disambiguates the senses of homographs and tags them with the corresponding sense codes. For instance, the homograph "*nun*" has two senses in Sentence 1 in Table 13. UTagger tags this with the two different sense codes: "04" and "01". The tagged sense codes, which generate different words (i.e., "*nun_04*" and "*nun_01*") for one homograph, help NMT systems create more accurate word alignments and choose the correct translation candidates. The improvement in the translation qualities of NMT systems depends on the number of homographs that are found in the training and testing parallel corpora. In the experimental results, the WSD improved the outputs of NMT by an average of only 0.23 BLEU points and 0.16 TER points for both translation directions; this is the case because there are a small number of Korean homographs in our test set. It also improved the performance of SMT by the same amount of BLEU and TER points.

4) IMPACT OF VIETNAMESE WORD SEGMENTATION

Vietnamese word segmentation resolves the problem of word boundaries since the whitespaces are only used to separate syllables. It also resolves rare and ambiguous word problems caused by the different meanings between individual syllables and a word composed by them. For the example mentioned above, the word "*đất nu·ó·c*" (country) has two syllables; the first one "*đất*" means the soil or land, and the second one "*nuó·c*" means the water. These two

meanings are different from the meaning of the word "*đất nư·ớ·c*" (country). As a result, Vietnamese word segmentation improved the performance of both SMT and NMT. It improved the translation from Korean to Vietnamese by approximately one BLEU point for both SMT and NMT. In the reverse translation direction, it increased BLEU by about two points.

Overall, UTagger and RDRsegmenter significantly improved the translation qualities for both directions of Korean-Vietnamese NMT systems by an average of 8.9 BLEU points and 9.9 TER points. Based on our promising results, Korean morphological analysis, WSD, and Vietnamese word segmentation may be effective strategies for the development of Korean-Vietnamese NMT systems. Especially, UTagger leads to high-accuracy Korean WSD and helps the Korean-Vietnamese NMT system translate Korean homographs correctly.

In addition, we compared the translation of Korean homographs between widely used MT systems (i.e., Google Translate, Microsoft Bing, and Naver Papago) in Table 13. We examined the two homographs "*nun*" and "*bae*". "*Nun*" appears two times in Sentence 1 with two different meanings: "snow" and "eye". "*Bae*" occurs three times in Sentence 2 with three different meanings: "pear", "ship", and "stomach". In this case, Naver Papago seems to be the best MT system since it translated Sentence 1 into an acceptable output; however, it skipped the first meaning "snow". Google Translate and Microsoft Bing could not distinguish the different meanings of "*nun*" in Sentence 1. None of these methods could correctly translate Sentence 2.

TABLE 13. Korean-to-Vietnamese translation examples with Korean homographs.

Source	Sent 1: nun-e mi-kkeu-leo-jyeo-seo nun-eul da-chyeoss-da. Sent 2: bae-leul meo-ggo bae-leul tass-deo-ni bae-ga a-pass-da.
Reference	Sent 1: I slipped over the snow and my eyes are injured. Sent 2: After eating a pear and boarding the ship, I had a stomachache.
Naver Papago	Sent 1: Vì trơn quá nên mắt tôi bị thương. (*Because it was so slippery, my eyes hurt.*) Sent 2: Ăn no rồi đi thuyền nên bụng đau. (*Eat and then take the boat so the belly pain.*)
Google Translate	Sent 1: Tôi trượt mắt và làm đau mắt. (*I slipped on my eyes and hurt my eyes.*) Sent 2: Khi tôi ăn một chiếc thuyền và lên thuyền, dạ dày của tôi bị đau. (*When I ate a boat and got on the boat, my stomach hurt.*)
Microsoft Bing	Sent 1: Tôi trượt mắt tôi và bị thương mắt tôi. (*I slipped my eyes and injured my eyes.*) Sent 2: Tôi ăn một chiếc thuyền và lấy một chiếc thuyền và nó đau tôi. (*I ate a boat and take the boat and it make me hurt.*)
UTagger	Sent 1: nun_04+e/JBK mi-kkeu-leo-ji/VV+eo-seo/EC nun_01/NNG+eul/JKO da-chi_01/VV+eoss-da/EF ./SF Sent 2: bae_03/NNG+leul/JKO meog_02/VV+go/EC bae_02/NNG+leul/JKO ta_02/VV+ass/EP+deo-ni/EC bae_01/NNG+ga/JKS a-peu/VA+ass/EP+da/EF ./SF
Proposed system	Sent 1: Tôi trượt trên tuyết nên bị thương mắt (*I slipped on the snow and injured my eyes.*) Sent 2: Ăn lê và lái thuyền rồi thì bụng đau. (*Eat a pear and ride a boat then abdomen hurt.*)

Google Translate, Microsoft Bing, and Naver Papago were accessed on Nov. 7, 2018. In Sentence 1, the Korean homograph "*nun*" occurs two times, where the first meaning is "snow" and the second is "eye". In Sentence 2, the Korean homograph "*bae*" occurs three times, where the first meaning is "pear", the second is "ship", and the third is "stomach".

VII. RELATED WORK

Recently, to respond to the requirement of high-quality Korean-Vietnamese MT systems caused by the development of bilateral cooperation between South Korea and Vietnam, several research groups have tried to build Korean-Vietnamese MT systems. Our research is closely related to a previous study that sought to improve NMT by LNS [42]. In that study, Nguyen et al. proposed a method to improve the translation quality of Korean-Vietnamese NMT systems by adding sense codes to Korean words. They built bi-directional Korean-Vietnamese NMT systems using the OpenNMT toolkit [66]. However, their training parallel corpus was limited to 281K sentence pairs, and they did not apply any word segmenter to the Vietnamese texts in their training corpus. They also did not analyze the impact of the Korean morphological analysis on their translation qualities.

Additionally, Cho *et al.* [67] addressed the problems of the lexical gap and multiple word expression in Korean-Vietnamese SMT. To solve these problems, they created morpho-syntactic filters to group the component morphemes of Korean verbs and adjectives from a translation phrase table. Then they used the Moses toolkit to train their model in the Korean-to-Vietnamese translational direction only. Their experimental results showed that grouping component morphemes could improve the translation quality by approximately one BLEU point.

Nguyen *et al.* [9] analyzed morphologies for the Korean texts of their training corpus in the preprocessing step. They trained their bi-directional Korean-Vietnamese SMT based on the Moses toolkit. The translation quality was improved by over three BLEU points; however, the size of their training corpus was very small and had only 24K sentence pairs.

Eojeols usually contain one or more function words, such as postpositions or endings. The forms of these function words are changed by various regular transformations, depending on their final consonant. Lee *et al.* [68] standardized the forms of the endings and postpositions of *eojeols* in their training corpus before using these to train the SMT model. The experimental results showed that this method could improve the translation quality by approximately one BLEU point when translating from Vietnamese to Korean.

However, in the opposite direction, the performance was reduced.

In another research paper [69], Cho *et al.* proposed a simple method to extract words, phrases, or sentences inside brackets, parentheses, and quotes. Then, these words, phrases, or sentences were translated individually. Their experiments were carried out via SMT from Korean to Vietnamese based on the Moses toolkit. The results showed that this method is effective for translating sentences that have brackets, parentheses, and quotes insides.

Most of the proposed Korean-Vietnamese MT systems follow the SMT approach. However, the translation qualities of SMT are inferior to those of NMT for language pairs that have large amounts of training datasets [64], [65]. In this paper, we also proved that the translation qualities of NMT are better than those of SMT for the Korean-Vietnamese language pair. In contrast to previous research, we built our Korean-Vietnamese MT systems based on the latest NMT architecture. Moreover, we disambiguated the sense of Korean homographs and represented them by adding sense codes to the homographs.

VIII. CONCLUSION

In this paper, we defined various challenges encountered while building Korean-Vietnamese NMT systems and addressed these through four accomplishments, as follows.

- We built the Korean LSN UWordMap, which is currently the largest LSN for Korean. This is useful for various fields that deal with semantic problems in Korean language processing.
- We developed an open Korean morphological analysis and WSD tool, i.e., UTagger, based on UWordMap. This tool achieves state-of-the-art performance in terms of its speed and accuracy.
- We collected over 454K sentence pairs to make a Korean-Vietnamese parallel corpus.
- Based on this corpus, we built a bi-directional Korean-Vietnamese NMT system that can perform Korean morphological analysis, Korean WSD, and Vietnamese word segmentation. The experimental results show that these applications significantly improve the NMT results.

In the future, we plan to insert more words into UWordMap and our pre-analysis partial *eojeol* dictionary. This will make UTagger more accurate, enabling our Korean-Vietnamese translation system to translate Korean homographs more accurately. We also intend to collect more Korean-Vietnamese parallel corpora and apply syntactical dependency into our MT systems.

REFERENCES

[1] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. ICLR*, Sacramento, CA, USA, 2015, pp. 1–15.
[2] M. T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. EMNLP*, Lisbon, Portugal, 2015, pp. 1412–1421.
[3] M. T. Luong, I. Sutskever, Q. V. Le, O. Vinyals, and W. Zaremba, "Addressing the rare word problem in neural machine translation," in *Proc. ACL*, Beijing, China, 2015, pp. 11–19.
[4] S. Jean, K. Cho, R. Memisevic, and Y. Bengio, "On using very large target vocabulary for neural machine translation," in *Proc. ACL*, Beijing, China, 2015, pp. 1–10.
[5] P. Koehn and R. Knowles, "Six challenges for neural machine translation," in *Proc. NMT*, Vancouver, BC, Canada, 2017, pp. 28–39.
[6] B. Zoph, D. Yuret, J. May, and K. Knight, "Transfer learning for low-resource neural machine translation," in *Proc. EMNLP*, Austin, TX, USA, 2016, pp. 1568–1575.
[7] J. H. Oh and J. S. Mah, "The patterns of Korea's foreign direct investment in Vietnam," *Open J. Bus. Manage.*, vol. 5, no. 2, pp. 253–271, Apr. 2017.
[8] J. H. Lee, "Korea's recent export to Vietnam and implications," *KIEP World Economy Update*, vol. 6, no. 5, pp. 1–5, Feb. 2016.
[9] Q. P. Nguyen, J. C. Shin, and C. Y. Ock, "Korean morphological analysis for Korean–Vietnamese statistical machine translation," *Electron. Sci. Technol.*, vol. 15, no. 4, pp. 413–419, Dec. 2017.
[10] D. Q. Nguyen, D. Q. Nguyen, T. Vu, M. Dras, and M. Johnson, "A fast and accurate Vietnamese word segmenter," in *Proc. LREC*, Miyazaki, Japan, 2018, pp. 2582–2587.
[11] P. Compton and R. Jansen, "Knowledge in context: A strategy for expert system maintenance," in *Proc. Austral. Joint AI Conf., Adelaide, Aust.*, 1988, pp. 292–306.
[12] H.-M. Sohn, Ed., *Korean Language in Culture and Society*. Honolulu, HI, USA: Univ. Hawaii Press, 2006.
[13] M. J. Alves, "What's so Chinese about Vietnamese," in *Proc. SALS*, Phoenix, AZ, USA, 1999, pp. 221–242.
[14] F. Liu, H. Lu, and G. Neubig, "Handling homographs in neural machine translation," in *Proc. NAACL-HLT*, Los Angeles, CA, USA, 2018, pp. 1336–1345.
[15] R. Marvin and P. Koehn, "Exploring word sense disambiguation abilities of neural machine translation systems (non-archival extended abstract)," in *Proc. AMTA*, Boston, MA, USA, 2018, pp. 125–131.
[16] H. Choi, K. Cho, and Y. Bengio, "Context-dependent word representation for neural machine translation," *Comput. Speech Lang.*, vol. 45, pp. 149–160, Sep. 2017.
[17] J. Tiedemann, "OPUS—Parallel corpora for everyone," *Baltic J. Mod. Comput.*, vol. 4, no. 2, pp. 384–384, 2016.
[18] G. A. Miller, "WordNet: A lexical database for English," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.
[19] Z. Dong and Q. Dong, *Hownet And the Computation of Meaning*. River Edge, NJ, USA: World Scientific, 2006.
[20] P. Vossen, "Introduction to EuroWordNet," *Comput. Hum.*, vol. 32, nos. 2–3, pp. 73–89, Mar. 1998.
[21] A. S. Yoon, S. H. Hwang, E. R. Lee, and H. C. Kwon, "Construction of Korean WordNet," *J. KIISE, Softw. Appl.*, vol. 36, no. 1, pp. 92–108, 2009.
[22] K.-S. Choi, "CoreNet: Chinese-Japanese-Korean WordNet with shared semantic hierarchy," in *Proc. NLP-KE*, Beijing, China, 2003, pp. 767–770.
[23] M. Choi, J. Hur, and M.-G. Jang, "Constructing Korean lexical concept network for encyclopedia question-answering system," in *Proc. ECON*, Busan, South Korea, 2004, pp. 3115–3119.
[24] S.-S. Kang and Y. T. Kim, "Syllable-based model for the Korean morphology," in *Proc. COLING*, Kyoto, Japan, 1994, pp. 221–226.
[25] D.-B. Kim, S.-J. Lee, K.-S. Choi, and G.-C. Kim, "A two-level morphological analysis of Korean," in *Proc. COLING*, Kyoto, Japan, 1994, pp. 535–539.
[26] O. W. Kwon *et al.*, "Korean morphological analyzer and part-of-speech tagger based on CYK algorithm using syllable information," in *Proc. MATEC*, 1999, pp. 76–88.
[27] J. H. Choi and S. J. Lee, "A method for reducing dictionary access with bidirectional longest match strategy in Korean morphological analyzer," *J. Korean Inf. Sci. Soc. Soft. Appl.*, vol. 20, no. 10, pp. 1497–1507, Oct. 1993.
[28] G. G. Lee, J.-H. Lee, and J. W. Cha, "Syllable-pattern-based unknown-morpheme segmentation and estimation for hybrid part-of-speech tagging of Korean," *Comput. Linguistics*, vol. 28, no. 1, pp. 53–70, Mar. 2002.
[29] J.-S. Lee, "Three-step probabilistic model for Korean morphological analysis," *J. KIISE, Softw. Appl.*, vol. 38, no. 5, pp. 257–268, May 2011.
[30] S.-H. Na, "Conditional random fields for Korean morpheme segmentation and POS tagging," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 14, no. 3, Jun. 2015, Art. no. 10.

[31] S.-H. Na and Y.-K. Kim, "Phrase-based statistical model for Korean morpheme segmentation and POS tagging," *IEICE Trans. Inf. Syst.*, vol. E101-D, no. 2, pp. 512–522, Feb. 2018.
[32] S. Jung, C. Lee, and H. Hwang, "End-to-end Korean part-of-speech tagging using copying mechanism," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 17, no. 3, Jan. 2018, Art. no. 19.
[33] A. Matteson, C. Lee, Y.-B. Kim, and H. S. Lim, "Rich character-level information for Korean morphological analysis and part-of-speech tagging," in *Proc. COLING*, Santa Fe, NM, USA, 2018.
[34] J. C. Shin and C. Y. Ock, "A Korean morphological analyzer using a pre-analyzed partial word-phrase dictionary," *KIISE, Softw. Appl.*, vol. 39, no. 5, pp. 415–424, May 2012.
[35] K. S. Shim, "Cloning of Korean morphological analyzers using pre-analyzed Eojeol dictionary and syllable-based probabilistic model," *KIISE, Comput. Practices*, vol. 22, no. 3, pp. 119–126, Mar. 2016.
[36] C. H. Lee, J. H. Lim, S. Lim, and H. K. Kim, "Syllable-based Korean POS tagging based on combining a pre-analyzed dictionary with machine learning," *J. KIISE*, vol. 43, no. 3, pp. 362–369, Mar. 2016.
[37] H. Kim, "Korean national corpus in the 21st century Sejong project," in *Proc. NIJL*, Tokyo, Japan, 2006, pp. 49–54.
[38] J. C. Shin and C. Y. Ock, "Korean homograph tagging model based on sub-word conditional probability," *KIPS Trans. Softw. Data Eng.*, vol. 3, no. 10, pp. 407–420, 2014.
[39] J. Min, J. W. Jeon, K. H. Song, and Y. S. Kim, "A study on word sense disambiguation using bidirectional recurrent neural network for Korean language," *J. Korea Soc. Comput. Inf.*, vol. 22, no. 4, pp. 41–49, 2017.
[40] M. Y. Kang, B. Kim, and J. S. Lee, "Word sense disambiguation using embedded word space," *Comput. Sci. Eng.*, vol. 11, no. 1, pp. 32–38, 2017.
[41] Q.-P. Nguyen, A.-D. Vo, J. C. Shin, and C. Y. Ock, "Effect of word sense disambiguation on neural machine translation: A case study in Korean," *IEEE Access*, vol. 6, pp. 38512–38523, 2018.
[42] Q.-P. Nguyen, A.-D. Vo, J.-C. Shin, and C.-Y. Ock, "Neural machine translation enhancements through lexical semantic network," in *Proc. ACM ICCMS*, Sydney, NSW, Australia, 2018, pp. 105–109.
[43] M. H. Kim and H. C. Kwon, "Word sense disambiguation using semantic relations in Korean WordNet," *J. KIISE, Softw. Appl.*, vol. 38, no. 10, pp. 554–564, 2011.
[44] S. W. Kang, M. Kim, H. Kwon, S. K. Jeon, and J. Oh, "Word sense disambiguation of predicate using Sejong electronic dictionary and KorLex," *J. KIISE, Comput. Practices*, vol. 21, no. 7, pp. 500–505, 2015.
[45] L. H. Phuong, N. T. T. Huyên, A. Roussanaly, and H. T. Vinh, "A hybrid approach to word segmentation of Vietnamese texts," in *Proc. LATA*, Tarragona, Spain, 2008, pp. 240–249.
[46] D. Dinh and V. Thuy, "A Maximum Entropy approach for Vietnamese word segmentation," in *Proc. RIVF*, Hanoi, Vietnam, 2006, pp. 248–253.
[47] D. Dien and H. Kiem, "Building a training corpus for word sense disambiguation in the English-Vietnamese bilingual corpus," in *Proc. WMTA*, Taipei, Taiwan, 2002, pp. 26–32.
[48] C. T. Nguyen *et al.*, "Vietnamese word segmentation with CRFs and SVMs: An investigation," in *Proc. PACLIC*, Beijing, China, 2006, pp. 215–222.
[49] T. A. Luu and Y. Kazuhide, "Ung dung phuong phap pointwise vao bai toan tach tu cho tieng Viet," (Applying Pointwise Method to Vietnamese Word Segmentation Problem). 2012. [Online]. Available: https://github.com/rockkhuya/DongDu
[50] T.-P. Nguyen and A.-C. Le, "A hybrid approach to Vietnamese word segmentation," in *Proc. RIVF*, Hanoi, Vietnam, 2016, pp. 114–119.
[51] P. T. Nguyen, X. L. Vu, T. M. H. Nguyen, V. H. Nguyen, and H. P. Le, "Building a large syntactically-annotated corpus of Vietnamese," in *Proc. LAW*, Singapore, 2009, pp. 182–185.
[52] G. N. S. Mori, "Word-based partial annotation for efficient corpus construction," in *Proc. LREC*, Valletta, Malta, 2010, pp. 2723–2727.
[53] P. Koehn, "Europarl: A parallel corpus for statistical machine translation," in *Proc. MT Summit*, Phuket, Thailand, 2005, pp. 79–86.
[54] A. Rafalovitch and R. Dale, "United Nations general assembly resolutions: A six-language parallel corpus," in *Proc. MT Summit*, Ottawa, ON, Canada, 2009, pp. 292–299.
[55] R. Steinberger *et al.*, "The JRC-acquis: A multilingual aligned parallel corpus with 20+ languages," in *Proc. LREC*, Rome, Italy, 2006, pp. 2142–2147.
[56] D. Dinh, W. J. Kim, and N. Diep, "Exploiting the Korean-Vietnamese parallel corpus in teaching Vietnamese for Koreans," presented at the Interdiscipl. Study Lang. Commun. Multicultural Soc., Int. Conf. ISEAS/BUFS, Busan, South Korea, May 2017.
[57] L. Tan and F. Bond, "Building and annotating the linguistically diverse NTU-MC (NTU-multilingual corpus)," in *Proc. PACLIC*, Singapore, 2011, pp. 362–371.
[58] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[59] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proc. EMNLP*, Doha, Qatar, 2014, pp. 1724–1734.
[60] M.-T. Luong, E. Brevdo, and R. Zhao. (2017). *Neural Machine Translation (seq2seq) Tutorial*. [Online]. Available: https://github.com/tensorflow/nmt
[61] P. Koehn *et al.*, "Moses: Open source toolkit for statistical machine translation," in *Proc. ACL-Demonstration Session*, Prague, Czech Republic, 2007, pp. 177–180.
[62] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. ACL*, 2002, pp. 311–318.
[63] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A study of translation edit rate with targeted human annotation," in *Proc. Assoc. MT Americas*, Boston, MA, USA, 2006, pp. 223–231.
[64] L. Bentivogli, A. Bisazza, M. Cettolo, and M. Federico, "Neural versus phrase-based machine translation quality: A case study," in *Proc. EMNLP*, Austin, TX, USA, 2016, pp. 257–267.
[65] M. Junczys-Dowmunt, T. Dwojak, and H. Hoang, "Is neural machine translation ready for deployment? A case study on 30 translation directions," in *Proc. IWSLT*, Seattle, DC, USA, 2016.
[66] G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush, "OpenNMT: Open-source toolkit for neural machine translation," in *Proc. ACL*, Vancouver, BC, Canada, 2017, pp. 67–72.
[67] S. W. Cho, E. H. Lee, and J. H. Lee, "Phrase-level grouping for lexical gap resolution in Korean-Vietnamese SMT," in *Proc. PACLING*, Yangon, Myanmar, 2017, pp. 127–136.
[68] W. K. Lee *et al.*, "Improve performance of phrase-based statistical machine translation through standardizing Korean allomorph," in *Proc. HCLT*, Busan, South Korea, 2016, pp. 285–290.
[69] S. W. Cho *et al.*, "Embedded clause extraction and restoration for the performance enhancement in Korean-Vietnamese statistical machine translation," in *Proc. HCLT*, Busan, South Korea, 2016, pp. 280–284.

QUANG-PHUOC NGUYEN received the B.S. degree in information technology from the University of Sciences-Vietnam National University, Ho Chi Minh City, Vietnam, in 2005, and the M.S. degree in information technology from Konkuk University, Seoul, South Korea, in 2010. He is currently pursuing the Ph.D. degree with the University of Ulsan, Ulsan, South Korea. His research interests include natural language processing, machine learning, and machine translation.

ANH-DUNG VO received the B.S. degree in computer science from the Hanoi University of Science and Technology, Vietnam, in 2010, and the M.S. degree in information technology from the University of Ulsan, Ulsan, South Korea, in 2013, where he is currently pursuing the Ph.D. degree. His research interests include natural language processing, machine learning, and sentiment analysis.

JOON-CHOUL SHIN received the B.S., M.Sc., and Ph.D. degrees in information technology from the University of Ulsan, South Korea, in 2007, 2009, and 2014, respectively, where he is currently a Postdoctoral Researcher. His research interests include Korean language processing, document clustering, and software engineering.

PHUOC TRAN received the B.S. degree in information technology from the University of Pedagogy, Ho Chi Minh City, Vietnam, in 2006, and the M.Sc. and Ph.D. degrees in computer science from the VNU Ho Chi Minh City University of Science, in 2011 and 2018, respectively. He is currently a Researcher with NLP-KD Lab, Ton Duc Thang University, Vietnam. His research interests include natural language processing, machine translation, and text clustering.

CHEOL-YOUNG OCK received the B.S., M.S., and Ph.D. degrees in computer engineering from the National University of Seoul, South Korea, in 1982, 1984, and 1993, respectively. He was a Visiting Professor with the Russia Tomsk Institute, Russia, in 1994, and Glasgow University, U.K., in 1996. He was also a Chairman with sigHCLT, KIISE, South Korea, from 2007 to 2008. He was a Visiting Researcher with the National Institute of Korean Language, South Korea, in 2008. He is currently a Professor with the School of IT Convergence, University of Ulsan, South Korea. His research interests include natural language processing, machine learning, and text mining. He received an Honorary Doctorate from the School of IT, National University of Mongolia, in 2007, and received a medal for Korean development from the Korean Government, in 2016.