Natural Language Engineering (2024), **30**, pp. 625–649
doi:10.1017/S1351324923000311

ARTICLE

Korean named entity recognition based on language-specific features

Yige Chen[1,†], KyungTae Lim[2,†] and Jungyeul Park[3]

[1]The Chinese University of Hong Kong, Hong Kong, [2]Seoul National University of Science and Technology, Seoul, South Korea, and [3]The University of British Columbia, Vancouver, BC, Canada
Corresponding author: Jungyeul Park; Email: jungyeul@mail.ubc.ca

(Received 5 October 2021; revised 23 May 2023; accepted 3 June 2023; first published online 29 June 2023)

Abstract
In this paper, we propose a novel way of improving named entity recognition (NER) in the Korean language using its language-specific features. While the field of NER has been studied extensively in recent years, the mechanism of efficiently recognizing named entities (NEs) in Korean has hardly been explored. This is because the Korean language has distinct linguistic properties that present challenges for modeling. Therefore, an annotation scheme for Korean corpora by adopting the CoNLL-U format, which decomposes Korean words into morphemes and reduces the ambiguity of NEs in the original segmentation that may contain functional morphemes such as postpositions and particles, is proposed herein. We investigate how the NE tags are best represented in this morpheme-based scheme and implement an algorithm to convert word-based and syllable-based Korean corpora with NEs into the proposed morpheme-based format. Analyses of the results of traditional and neural models reveal that the proposed morpheme-based format is feasible, and the varied performances of the models under the influence of various additional language-specific features are demonstrated. Extrinsic conditions were also considered to observe the variance of the performances of the proposed models, given different types of data, including the original segmentation and different types of tagging formats.

Keywords: Named entity; Named entity recognition; Linguistic features; Morphology; Korean

1. Introduction

Named entity recognition (NER) is a subfield of information extraction that searches for and extracts named entities (NEs) such as locations, person names, organizations, and numbers from texts (Tjong Kim Sang and De Meulder 2003). With the rapid development in the field of machine learning, recent works on NER rely significantly on machine learning methods instead of grammar-based symbolic rules, with a particular focus on neural network algorithms. Numerous studies have been conducted on NER concerning several natural languages, including Carreras *et al.* (2003) for Catalan, Li and McCallum (2003) for Hindi, Nouvel *et al.* (2013) and Park (2018) for French, Benikova, Biemann, and Reznicek (2014); Benikova *et al.* (2015) for German, and Ahmadi and Moradi (2015) for Persian. There have also been language-independent multilingual models proposed (Tjong Kim Sang and De Meulder 2003; Nothman *et al.* 2013; Hahm *et al.* 2014). However, NER for the Korean language has not been explored thoroughly in previous research. This is partially because NE detection in Korean is difficult, as the language does not

[†]Yige Chen and KyungTae Lim contributed equally.
Article updated 9 November 2023.

utilize specific features existing in some other languages such as capitalization to emphasize NEs (Chung, Hwang, and Jang 2003). There have been studies and attempts on this topic, one of the examples being an hidden Markov model (HMM)-based co-trained model in which co-training boosts the performance of HMM-based NE recognition (Chung *et al.* 2003). Recently, neural methods have been applied to the NER tasks of Korean, which include Bi-long short-term memory (LSTM)-conditional random field (CRF) with masked self-attention (Jin and Yu 2021), as well as Bidirectional Encoder Representations from Transformers (BERT) (Kim and Lee 2020). Such approaches require annotated corpora of high quality, which are usually difficult to source for Korean.

Although gold standard corpora, such as CoNLL-03 (Tjong Kim Sang and De Meulder 2003), are still not available for the Korean language, it is probable that by constructing silver standard corpora with automatic approaches, both NE annotation and NER modeling can be handled decently. A silver standard NE corpus in which NE annotations are completed automatically has been constructed based on Wikipedia and DBpedia Ontology including Korean (Hahm *et al.* 2014). This high-quality corpus outperforms the manually annotated corpus. On the other hand, there have been new attempts that build a Korean NER open-source dataset based on the Korean Language Understanding Evaluation (KLUE) project (Park *et al.* 2021). KLUE's NER corpus is built mainly with the WIKITREE corpus,[a] which is a body of news articles containing several types of entities.

Due to the linguistic features of the NE in Korean, conventional *eojeol*-based segmentation, which makes use of whitespaces to separate phrases, does not produce ideal results in NER tasks. Most language processing systems and corpora developed for Korean use *eojeol* delimited by whitespaces in a sentence as the fundamental unit of text analysis in Korean. This is partially because the Sejong corpus, the most widely used corpus for Korean, employs *eojeol* as the basic unit. The rationale of *eojeol*-based processing is simply treating the words as they are in the surface form. It is necessary for a better format and a better annotation scheme for the Korean language to be adapted. In particular, a word should be split into its morphemes.

To capture the language-specific features in Korean and utilize them to boost the performances of NER models, we propose a new morpheme-based scheme for Korean NER corpora that handles NE tags on the morpheme level based on the CoNLL-U format designed for Korean, as in Park and Tyers (2019). We also present an algorithm that converts the conventional Korean NER corpora into the morpheme-based CoNLL-U format, which includes not only NEs but also the morpheme-level information based on the morphological segmentation. The contributions of this study for Korean NER are as follows: (1) An algorithm is implemented in this study to convert the Korean NER corpora to the proposed morpheme-based CoNLL-U format. We have investigated the best method to represent NE tags in the sentence along with their linguistic properties and therefore developed the conversion algorithm with sufficient rationales. (2) The proposed Korean NER models in the paper are distinct from other systems since our neural system is simultaneously trained for part-of-speech (POS) tagging and NER with a unified, continuous representation. This approach is beneficial as it captures complex syntactic information between NE and POS. This is only possible with a scheme that contains additional linguistic information such as POS. (3) The proposed morpheme-based scheme for NE tags provides a satisfying performance based on the automatically predicted linguistic features. Furthermore, we thoroughly investigate various POS types, including the language-specific XPOS and the universal UPOS (Petrov, Das, and McDonald 2012) and determine the type that has the most effect. We demonstrate and account for the fact that the proposed BERT-based system with linguistic features yields better results over those in which such linguistic features are not used.

In this paper, we summarize previous work on Korean NER in Section 3 and present a linguistic description of NEs in Korean in Section 2, focusing on the features that affect the distribution

[a] https://www.wikitree.co.kr/

of NE tags. We further provide a methodology for representing NEs in Korean in Section 4. Subsequently, we introduce morphologically enhanced Korean NER models which utilize the predicted UPOS and XPOS features in Section 5. To evaluate the effect of the proposed method, we test the proposed morpheme-based CoNLL-U data with NE tags using three different models, namely CRF, RNN-based, and BERT-based models. We provide detailed experimental results of NER for Korean, along with an error analysis (Sections 6 and Section 7). Finally, we present the conclusion (Section 9).

2. Linguistic description of NEs in Korean

2.1. Korean linguistics redux

Korean is considered to be an agglutinative language in which functional morphemes are attached to the stem of the word as suffixes. The language possesses a subject-object-verb (SOV) word order, which means the object precedes the verb in a sentence. This is different from English grammar where the object succeeds the verb in a sentence. Examples from Park and Kim (2023) are presented as follows.

(1) a. 존이 다리도 건넌다
jon-i dali-do geonneo-n-da
John-NOM bridge-ALSO cross-PRES-DCL
"John crosses the bridge, too"

b. 다리도 존이 건넌다
dali-do jon-i geonneo-n-da
bridge-ALSO John-NOM cross-PRES-DCL
"John crosses the bridge, too"

c. 존 다리를 건넌다
jon dali-leul geonneo-n-da
John bridge-ACC cross-PRES-DCL
"John crosses the bridge"

Two properties of the Korean language are observed. While Korean generally follows the SOV word order (1a), scrambling may occur from time to time which shifts the order of the subject and the object. Moreover, the postpositions in Korean, such as the nominative marker *-i* in (1a) and (1b), follow the stems to construct words, but there are also words in Korean that take no postposition, such as the subject *jon* ("John") as presented in (1c).

Another notable property of Korean is its natural segmentation, namely *eojeol*. An *eojeol* is a segmentation unit separated by space in Korean. Given that Korean is an agglutinative language, joining content and functional morphemes of words (*eojeols*) is very productive, and the number of their combinations is exponential. We can treat a given noun or verb as a stem (also content) followed by several functional morphemes in Korean. Some of these morphemes can, sometimes, be assigned its syntactic category. Let us consider the sentence in (2). *Unggaro* ("Ungaro") is a content morpheme (a proper noun) and a postposition *-ga* (nominative) is a functional morpheme. They form together a single *eojeol* (or word) *unggaro-ga* ("Ungaro+nom"). The nominative case markers *-ga* or *-i* may vary depending on the previous letter – vowel or consonant. A predicate *naseo-eoss-da* also consists of the content morpheme *naseo* ("become") and its functional morphemes, *-eoss* ("past") and *-da* ("decl"), respectively.

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

(2)

프랑스의	세계적인	의상	디자이너	엠마누엘	웅가로가	실내
peurangseu-ui	*segye-jeok-i-n*	*uisang*	*dijaineo*	*emmanuel*	*unggaro-ga*	*silnae*
France-GEN	world class-REL	fashion	designer	Emanuel	Ungaro-NOM	interior

장식용	직물	디자이너로	나섰다.
jangsik-yong	*jikmul*	*dijaineo-ro*	*naseo-eoss-da.*
decoration	textile	designer-AJT	become-PAST-DECL

”The world-class French fashion designer Emanuel Ungaro became an interior textile designer.”

2.2. NEs in Korean

Generally, the types of NE in Korean do not differ significantly from those in other languages. NE in Korean consists of person names (PER), locations (LOC), organizations (ORG), numbers (NUM), dates (DAT), and other miscellaneous entities. Some examples of NE in Korean are as follows:

- PER: 이창동 *ichangdong* (“Lee Chang-dong,” a South Korean film director)
- LOC: 제주 *jeju* (“Jeju Island”)
- ORG: 서울대학교 *seouldaehaggyo* (“Seoul National University”)

While NEs can appear in any part of the sentence, they typically occur with certain distributions with regard to the postpositions that follow the NEs. Figure 1 presents the overall distribution of the postpositions after NEs, whereas Table 2 lists the percentage of the postpositional markers with respect to three types of typical NEs, namely PER, ORG, and LOC, based on NAVER’s Korean NER dataset.[b] For instance, a PER or an ORG is often followed by a topic marker *eun/neun* (JX), a subject marker *-i/-ga* (JKS), or an object marker *-eul/-leul* (JKO). As detailed in Table 2, the percentages of JX, JKS, and JKO after ORG and PER are sufficiently high for us to make this conclusion, considering that in several cases, NEs are followed by other grammatical categories (e.g., noun phrases), which takes up high percentages (58.92% for LOC, 60.13% for ORG, and 66.72% for PER). Moreover, a PER or an ORG tends to occur at the front of a sentence, which serves as its subject. Higher percentages of occurrences of JX and JKS than JKO after ORG and PER prove this characteristic. Meanwhile, a LOC is typically followed by an adverbial marker (JKB) such as *-e* (locative “to”), *-eseo* (“from”), or -(*eu*)*ro* (“to”), because the percentage of JKB after LOC is considerably higher.

As mentioned in Section 1, NE in the Korean language is considered to be difficult to handle compared to Latin alphabet-based languages such as English and French. This is mainly because of the linguistic features of NE in Korean. The Korean language has a unique phonetic-based writing system known as *hangul*, which does not differentiate lowercase and uppercase. Nor does it possess any other special forms or emphasis regarding NEs. Consequently, an NE in Korean, through its surface form, makes no difference from a non-NE word. Because the Korean writing system is only based on its phonological form owing to the lack of marking on NE, it also creates type ambiguities when it comes to NER.

Another feature of NE in Korean is that the natural segmentation of words or morphemes does not always represent NE pieces. The segmentation of Korean is based on *eojeol*, and each *eojeol* contains not only a specific phrase as a content morpheme but also postpositions or particles that are attached to the phrase as functional morphemes, which is typical in agglutinative languages.

[b]The source of Figure 1, Table 1, and Table 2 is the morpheme-based CoNLL-U data with NE annotations we generate for this study using the NER data from NAVER, which will be introduced in detail in Section 6.

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Table 1. XPOS (Sejong tag set) introduced in the morpheme-based CoNLL-U data converted from NAVER's NER corpus and its type of named entities.

XPOS		Type of NE	
NONE	Non-postposition	AFW	Artificially made articles: *e.g.* 여객기 *yeogaeggi* ("airliner")
JX	Topic marker (은|는 *eun|neun*)	ANM	Animal: *e.g.* 플랑크톤 *peullangkeuton* ("plankton")
JKV	Vocative marker (아|야 *a|ya*)	CVL	Civilization or culture related: *e.g.* 홍보대사 *hongbodaesa* ("ambassador")
JKS	Nominative marker (이|가 *i|ga*)	DAT	Date: *e.g.* 주말 *jumal* ("weekend")
JKQ	Postposition for quotations (고 *go*)	EVT	Event: *e.g.* 챔피언스리그 *chaempieonseuligeu* ("Champions League")
JKO	Accusative marker (을|를 *eul|leul*)	FLD	Academic fields, theories, laws, and technologies: *e.g.* 음성 인식 *eumseong insig* ("Speech Recognition")
JKG	Genitive marker (의 *ui*)	LOC	Location: *e.g.* 아메리카 *amelika* ("America")
JKC	Postposition for complements (이|가 *i|ga*, same as of JKS)	MAT	Metals, rocks, and chemicals material: *e.g.* 칼슘 *kalsyum* ("calcium")
JKB	Postposition for adverbs (에|에서 *e|eseo*)	NUM	Number: *e.g.* 세번 *sebeon* ("third time")
JC	Postposition for conjunctions (와|과 *wa|gwa*)	ORG	Organization: *e.g.* 서울대학교 *seouldaehaggyo* ("Seoul National University")
		PER	Person: *e.g.* 세종대왕 *sejongdaewang* ("Sejong the Great")
		PLT	Plant: *e.g.* 포도나무 *podonamu* ("vine")
		TIM	Time: *e.g.* 한시간 *hansigan* ("one hour")
		TRM	Term: *e.g.* 납 중독 *nab jungdog* ("lead poisoning")

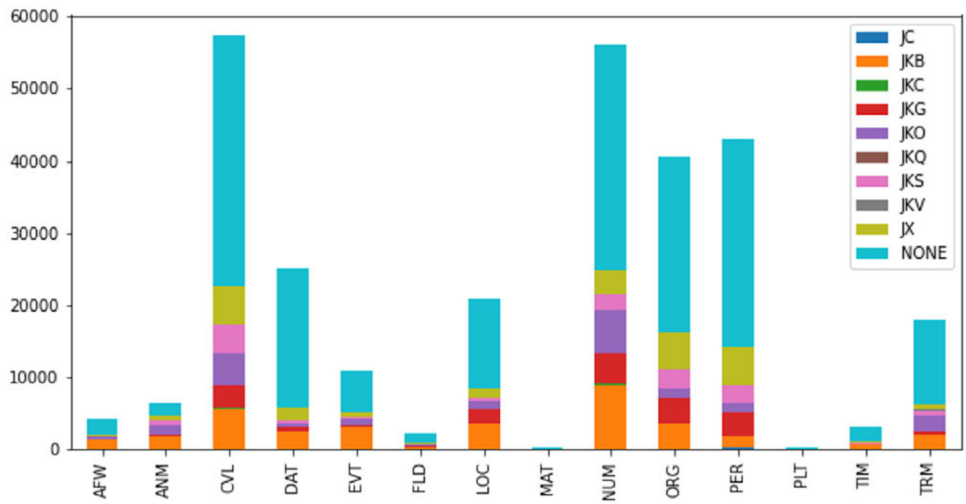


Figure 1. Distribution of each type of postposition/particle after NEs (NER data from NAVER). The terms and notations in the figure are described in Table 1.

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Table 2. Percentage (%) of occurrences of JKB/JKO/JKS/JX among all types of parts-of-speech after LOC/ORG/PER (NER data from NAVER).

	JKB	JKO	JKS	JX
LOC	16.72	4.92	2.59	6.51
ORG	8.61	3.38	6.30	12.49
PER	3.50	2.94	5.49	12.72

	프	랑	스	의	
	peu	*rang*	*seu*	*ui*	*peurangseu-ui* (‘France.GEN’)
Eojeol:	B-LOC				
Morpheme:	B-LOC			O	
Syllable:	B-LOC	I-LOC	I-LOC	O	

Figure 2. Various approaches of annotation for named entities (NEs): the *eojeol*-based approach annotates the entire word, the morpheme-based annotates only the morpheme and excludes the functional morphemes, and the syllable-based annotates syllable by syllable to exclude the functional morphemes.

Normally, Korean NEs only consist of noun phrases and numerical digits. But under special circumstances, a particle or a postposition could be part of the NE, such as the genitive case maker -*ui*. Moreover, an NE in Korean does not always consist of only one *eojeol*. In Korean, location and organization can be decomposed into several *eojeols* (Yun 2007), which implies that there is no one-to-one correspondence with respect to a Korean NE and its *eojeol* segment.

Therefore, the main difficulties in Korean NEs are as follows: (1) NEs in Korean are not marked or emphasized, which requires other features from adjacent words or the entire sentence to be used; (2) in a particular *eojeol* segment, one needs to specify the parts of this segmentation that belong to the NE, and within several segments, one needs to determine the segments that correspond to the NE.

3. Previous work

As a sequence-labeling task, Korean NER aims at assigning tags based on the boundary of the morphemes or the *eojeols* (words or phrases separated by whitespaces in Korean). Thus, two major approaches, namely morpheme- and *eojeol*-based NER, have been proposed depending on their corresponding target boundary. The morpheme-based approach separates the word into a sequence of morphemes for detection of the NE, and accordingly, the recognized NE is the sequence of morphemes itself. The *eojeol*-based approach, on the other hand, uses the word (or *eojeol*) as the basic unit for detection of the NE, and the recognized entity may contain extra morphemes such as functional morphemes (e.g. the case marker) which should be excluded from the NE. However, this post-processing task has been ignored in the *eojeol*-based system. Apart from the two major approaches described above, some Korean NER datasets also adopt the syllable-based annotation approach which splits Korean texts into syllables and assigns NE tags directly on the syllables. Although the target boundaries of NER tasks between these approaches differ considerably, the statistical and neural systems for performing NER remain identical for both approaches as they both rely on the sequence-labeling algorithm. Figure 2 illustrates various annotation approaches for NEs in Korean, including *eojeol*, morpheme, and syllable.

The morpheme-based approach has been adopted in the publicly available Korean Information Processing System (KIPS) corpora in 2016 and 2017. The two corpora consist of 3660 and 3555

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

sentences, respectively, for training.[c] Since the extraction targets of the morpheme-based NER are NEs on the morpheme level, it requires a morphological analyzer to be used in the pre-processing step based on the input sentence. Yu and Ko (2016) proposed a morpheme-based NER system using a bidirectional LSTM-CRFs, in which they applied the morpheme-level Korean word embeddings to capture features for morpheme-based NEs on news corpora. Previous works such as Lee *et al.* (2018) and Kim and Kim (2020) also used the same morpheme-based approach. In these studies, the morphological analysis model was integrated into NER. However, for the morpheme-based NER, considerable performance degradation occurs when converting *eojeols* into morphemes. To mitigate this performance gap between morpheme- and *eojeol*-based NER, Park *et al.* (2017a) proposed a hybrid NER model that combined *eojeol*- and morpheme-based NER systems. The proposed *eojeol*- and morpheme-based NER systems yield the best possible NER candidates based on an ensemble of both predicted results using bidirectional gated recurrent units combined with CRFs.

In the last few years, studies on Korean NER have been focusing on *eojeol*-based NER because such datasets have become more widely available, including NAVER's NER corpus which was originally prepared for a Korean NER competition in 2018.[d] In *eojeol*-based Korean NER tasks, the transformer-based model has been dominant because of its outstanding performance (Min *et al.* 2019; Kim, Oh, and Kim 2020; You *et al.* 2021). Various transformer-based approaches, such as adding new features including the NE dictionary in a pre-processing procedure (Kim and Kim 2019) and adapting byte-pair encoding (BPE) and morpheme-based features simultaneously (Min *et al.* 2019), have been proposed. Min *et al.* (2021) proposed a novel structure for the Korean NER system that adapts the architecture of Machine Reading Comprehension (MRC) by extracting the start and end positions of the NEs from an input sentence. Although the performance of the MRC-based NER is inferior to those using token classification methods, it shows a possibility that the NER task can be solved based on the span detection problem. Another notable approach for Korean NER is data argumentation. Cho and Kim (2021) achieved improvement in the performance of the Korean NER model using augmented data that replace tokens with the same NE category in the training corpus.

To avoid potential issues of the *eojeol*-based approach where additional morphemes that should be excluded from NEs may be included, syllable-based Korean NER corpus was proposed, such as in KLUE in 2021 (Park *et al.* 2021). The NE tags in the corpus are annotated on the syllable level. For instance, the NE tags of the word "'경찰'" (*gyeongchal*, "police") are marked on the two syllables, namely 경 *gyeong* bearing `B-ORG` and 찰 *chal* bearing `I-ORG`. Table 3 summarizes the currently available Korean NER datasets.

The morpheme-based approach in previous studies differs from the proposed method. In traditional morpheme-based approaches, only the sequences of morphemes are used, and there are no word boundaries displayed. This is because of the nature of the annotated KIPS corpus in which the tokens included are merely morphemes. Our proposed morpheme-based approach conforms to the current language annotation standard by using the CoNLL-U format and multiword annotation for the sequence of morphemes in a single eojeol, where morphemes and word boundaries are explicitly annotated. Limited studies have been conducted on the mechanism of efficiently annotating and recognizing NEs in Korean. To tackle this problem, an annotation scheme for the Korean corpus by adopting the CoNLL-U format, which decomposes Korean words into morphemes and reduces the ambiguity of NEs in the original segmentation, is proposed in this study. Morpheme-based segmentation for Korean has been proved beneficial. Many downstream applications for Korean language processing, such as POS tagging (Jung, Lee, and Hwang 2018; Park and Tyers 2019), phrase-structure parsing (Choi, Park, and Choi 2012; Park, Hong, and Cha 2016; Kim and Park 2022), and machine translation (Park, Hong, and Cha, 2016, 2017b), are based on

[c]https://github.com/KimByoungjae/klpNER2017
[d]https://github.com/naver/nlp-challenge/tree/master/missions/ner

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Table 3. Summary of publicly available Korean NER datasets.

Name	Type	Train	Dev	Test
2016 KIPS	morpheme	3,660	–	–
2017 KIPS	morpheme	3,555	501	2,569
NAVER NER	eojoel	90,000	–	–
MODU 2019	syllable	150,082	–	–
MODU 2021	syllable	68,400	1,085	8,685
KLUE	syllable	21,008	5,000	5,000*

KIPS = Korean Information Processing System (DATE, LOC, ORG, PER, TIME); NAVER (14 entity labels, which we will describe in the next section); MODU (150 entity labels); KLUE = Korean Language Understanding Evaluation (KIPS's 5 labels, and QUANTITY); *KLUE's test dataset is only available through https://klue-benchmark.com/.

the morpheme-based segmentation, in which all morphemes are separated from each other. In these studies, the morpheme-based segmentation is implemented to avoid data sparsity because the number of possible words in longer segmentation granularity (such as *eojeols*) can be exponential given the characteristics of Korean, an agglutinative language. Since a morpheme-based dependency parsing system for Universal Dependencies with better dependency parsing results has also been developed (Chen *et al.* 2022), we are trying to create a consortium to develop the morphologically enhanced Universal Dependencies for other morphologically rich languages such as Basque, Finnish, French, German, Hungarian, Polish, and Swedish.

4. Representation of NEs for the Korean language

The current Universal Dependencies (Nivre *et al.*, 2016, 2020) for Korean uses the tokenized word-based CoNLL-U format. Addressing the NER problem using the annotation scheme of the Sejong corpus or other Korean language corpora is difficult because of the agglutinative characteristics of words in Korean.[e] They adopt the *eojeol*-based annotation scheme which cannot handle sequence-level morpheme boundaries of NEs because of the characteristics of agglutinative languages. For example, an NE *emmanuel unggaro* (person) without a nominative case marker instead of *emmanuel unggaro-ga* ("Emanuel Ungaro-nom") should be extracted for the purpose of NER. However, this is not the case in previous work on NER for Korean.

We propose a novel approach for NEs in Korean by using morphologically separated words based on the morpheme-based CoNLL-U format of the Sejong POS tagged corpus proposed in Park and Tyers (2019), which has successfully obtained better results in POS tagging compared to using the word-based Sejong corpus. While Park and Tyers (2019) have proposed the morpheme-based annotation scheme for POS tags and conceived the idea of using their scheme on NER, they have not proposed any practical method to adopt the annotation scheme to NER tasks. As a result, existing works have not explored these aspects such as how to fit the NE tags to the morpheme-based CoNLL-U scheme and how the NEs are represented in this format. Our proposed format for NER corpora considers the linguistic characteristics of Korean NE, and it also allows the automatic conversion between the word-based format and the morpheme-based format.[f] Using the proposed

[e]Although the tokenized words in Korean as in the Penn Korean treebank (Han *et al.* 2002) and the *eojeols* as the word units in the Sejong corpus are different, they basically use the punctuation mark tokenized *eojeol* and the surface form *eojeol*, respectively. Therefore, we distinguish them as word (or *eojeol*)-based corpora from the proposed morpheme-based annotation.

[f]Details of the automatic conversion algorithm between the word-based format and the morpheme-based format are addressed in Section 6.

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

sent_id = BTAA0001-00000012

text = 프랑스의 세계적인 의상 디자이너 엠마누엘 웅가로가 실내 장식용 직물 디자이너로 나섰다.

1-2	프랑스의	_	_	_	_	_	
1	프랑스	프랑스	PROPN	NNP	B-LOC	_	*peurangseu* ('France')
2	의	의	ADP	JKG	_	_	*-ui* ('-GEN')
3-6	세계적인	_	_	_	_	_	
3	세계	세계	NOUN	NNG	_	_	*segye* ('world')
4	적	적	PART	XSN	_	_	*-jeok* ('-SUF')
5	이	이	VERB	VCP	_	_	*-i* ('-COP')
6	ㄴ	은	PART	ETM	_	_	*-n* ('-REL')
7	의상	의상	NOUN	NNG	_	_	*uisang* ('fashion')
8	디자이너	디자이너	NOUN	NNG	_	_	*dijaineo* ('designer')
9	엠마누엘	엠마누엘	PROPN	NNP	B-PER	_	*emmanuel* ('Emanuel')
10-11	웅가로가	_	_	_	_	_	
10	웅가로	웅가로	PROPN	NNP	I-PER	_	*unggaro* ('Ungaro')
11	가	가	ADP	JKS	_	_	*-ga* ('-NOM')
12	실내	실내	NOUN	NNG	_	_	*silnae* ('interior')
13-14	장식용	_	_	_	_	_	
13	장식	장식	NOUN	NNG	_	_	*jangsik* ('decoration')
14	용	용	PART	XSN	_	_	*-yong* ('usage')
15	직물	직물	NOUN	NNG	_	_	*jikmul* ('textile')
16-17	디자이너로	_	_	_	_	_	
16	디자이너	디자이너	NOUN	NNG	_	_	*dijaineo* ('designer')
17	로	로	ADP	JKB	_	_	*-ro* ('-AJT')
18-20	나섰다	_	_	_	_	SpaceAfter=No	
18	나서	나서	VERB	VV	_	_	*naseo* ('become')
19	었	었	PART	EP	_	_	*-eoss* ('PAST')
20	다	다	PART	EF	_	_	*-da* ('-DECL')
21	.	.	PUNCT	SF	_	_	

Figure 3. CoNLL-U style annotation with multiword tokens for morphological analysis and POS tagging. It can include BIO-based NER annotation where B-LOC is for a beginning word of location and I-PER for an inside word of person.

annotation scheme in our work as demonstrated in Figure 3, we can directly handle the word boundary problem between content and functional morphemes while using any sequence labeling algorithms. For example, only *peurangseu* ("France") is annotated as a NE (B-LOC) instead of *peurangseu-ui* ("France-gen"), and *emmanuel unggaro* ("Emanuel Ungaro-") instead of *emmanuel unggaro-ga* ("Emanuel Ungaro-nom") as B-PER and I-PER in Figure 3.

In the proposed annotation scheme, NEs are, therefore, no longer marked on the *eojeol*-based natural segmentation. Instead, each NE annotation corresponds to a specific set of morphemes that belong to the NE. Morphemes that do not belong to the NE are excluded from the set of the NE annotations and thus are not marked as part of the NE. As mentioned, this is achieved by adapting the CoNLL-U format as it provides morpheme-level segmentation of the Korean language. While those morphemes that do not belong to the NE are usually postpositions, determiners, and particles, this does not mean that all the postpositions, determiners, and particles are not able to be parts of the NE. An organization's name or the name of an artifact may include

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

postpositions or the name of an artifact may include postpositions or particles in the middle, in which case they will not be excluded. The following NE illustrates the aforementioned case:

1	...			
2	‘	_	SpaceAfter=No	
3-4	프로페셔널의	_	_	*peuropesyeoneol-ui*
3	프로페셔널	B-AFW	_	*peuropesyeoneol* (‘professional’)
4	의	I-AFW	_	*-ui* (‘-GEN’)
5	원칙	I-AFW	SpaceAfter=No	*wonchik* (‘principle’)
6	’	_	SpaceAfter=No	
7	은	_	_	*-eun* (‘-TOP’)
8	...			

The NE 프로페셔널의 원칙 *peuropesyeoneol-ui wonchik* (“the principle of professional”) is the title of a book belonging to `AFW` (artifacts/works). Inside this NE, the genitive case marker *-ui*, which is a particle, remains a part of the NE. Because these exceptions can also be captured by sequence labeling, such an annotation scheme of Korean NER can provide a more detailed approach to NER tasks in which the NE annotation on *eojeol*-based segments can now be decomposed to morpheme-based segments, and purposeless information can be excluded from NEs to improve the performance of the machine learning process.

5. NER learning models

In this section, we propose our baseline and neural models that consider the state-of-the-art feature representations. NER is considered to be a classification task that classifies possible NEs from a sentence, and the previously proposed NER models are mostly trained in a supervised manner. In conventional NER systems, the maximum entropy model (Chieu and Ng 2003) and CRFs (McCallum and Li 2003) are used as classifiers. Recently, neural network-based models with continuous word representations outperformed the conventional models in lots of studies (Lample *et al.* 2016; Dernoncourt, Lee, and Szolovits 2017; Strubell *et al.* 2017; Ghaddar and Langlais 2018; Jie and Lu 2019). In this study, we mainly present the neural network-based systems.

5.1. Baseline CRF-based model

The objective of the NER system is to predict the label sequence $Y = \{y_1, y_2, \ldots, y_n\}$, given the sequence of words $X = \{x_1, x_2, \ldots, x_n\}$, where n denotes the number of words. As we presented in Section 2, y is the gold label of the standard NEs (PER, LOC, and ORG). An advantage of CRFs compared to previous sequence labeling algorithms, such as HMMs, is that their features can be defined as per our definition. We used binary tests for feature functions by distinguishing between unigram ($f_{y,x}$) and bigram ($f_{y',y,x}$) features as follows:

$$f_{y,x}(y_i, x_i) = \mathbf{1}(y_i = y, x_i = x)$$
$$f_{y',y,x}(y_{i-1}, y_i, x_i) = \mathbf{1}(y_{i-1} = y', y_i = y, x_i = x)$$

where $\mathbf{1}(condition) = 1$ if the condition is satisfied and 0 otherwise. Here, $(condition)$ represents the input sequence x at the current position i with CRF label y. We used word and other linguistic information such as POS labels up to the ± 2-word context for each surrounding input sequence x and their unigram and bigram features.

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

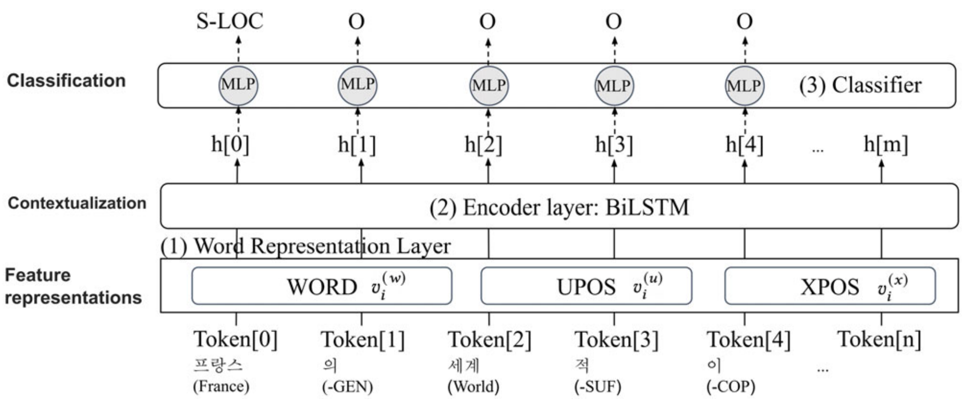


Figure 4. Overall structure of our RNN-based model.

5.2. RNN-based model

There have been several proposed neural systems that adapt the LSTM neural networks (Lample *et al.* 2016; Strubell *et al.* 2017; Jie and Lu 2019). These systems consist of three different parts. First, the word representation layer transforms the input sequence X as a sequence of word representations. Second, the encoder layer encodes the word representation into a contextualized representation using RNN-based architectures. Third, the classification layer classifies NEs from the contextualized representation of each word.

We construct our baseline neural model following Lample *et al.* (2016) by using pre-trained fastText word embedding (Bojanowski *et al.* 2017; Grave *et al.* 2018) as the word representation layer and LSTM as the encoder layer. The overall structure of our baseline neural model is displayed in Figure 4.

Word representation layer. To train an NER system, the first step is to transform the words of each input sequence to the word vector for the system to learn the meaning of the words. Word2vec (Mikolov *et al.* 2013) and GloVe (Pennington, Socher, and Manning 2014) are examples of word representations. Word2Vec is based on the distributional hypothesis and generates word vectors by utilizing co-occurrence frequencies within a corpus. fastText, on the other hand, is an extension of the Word2Vec that incorporates subword information. Rather than treating each word as a discrete entity, fastText breaks words down into smaller subword units, such as character n-grams, and learns embeddings. This approach allows fastText to capture information about word morphology and handle out-of-vocabulary words more effectively than other methods.

Given an input word x_i, we initialize word embedding $v_i^{(w)}$ using pre-trained fastText embeddings (Grave *et al.* 2018) provided by the 2018 CoNLL shared task organizer. Randomly initialized POS features, UPOS $v_i^{(u)}$ and XPOS $v_i^{(x)}$ are concatenated to $v_i^{(w)}$ if available.

Encoder layer. The word embedding proposed in this study utilizes context-aware word vectors in a static environment. Therefore, in the NER domain we are using, it can be considered context-independent. For example, each word representation $v_i^{(w)}$ does not consider contextual words such as $v_{i+1}^{(w)}$ and $v_{i-1}^{(w)}$. Furthermore, since the POS features are also randomly initialized, there is an issue of not being able to determine the context. LSTMs are typically used to address the context-independent problem. The sequence of word embeddings is fed to a two-layered LSTM to transform the representation to be contextualized as follows:

$$h_i^{(\mathrm{ner})} = \mathrm{BiLSTM}\left(r_0^{(\mathrm{ner})}, \left(v_1^{(w)}, \ldots, v_n^{(w)}\right)\right)_i \qquad (1)$$

where $r_0^{(\mathrm{ner})}$ represents the randomly initialized context vector, and n denotes the number of word representations.

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Classification layer. The system consumes the contextualized vector $h_i^{(\text{ner})}$ for predicting NEs using a multi-layer perceptron (MLP) classifier as follows:

$$z_i = Q^{(\text{ner})}\text{MLP}\left(h_i^{(\text{ner})}\right) + b^{(\text{ner})} \tag{2}$$

where MLP consists of a linear transformation with the ReLU function. The output z_i is of size $1 \times k$, where k denotes the number of distinct tags. We apply Softmax to convert the output, z_i, to be a distribution of probability for each NE.

During training, the system learns parameters θ that maximize the probability $P(y_i|x_i, \theta)$ from the training set T based on the conditional negative log-likelihood loss.

$$\text{Loss}(\theta) = \sum_{(x_i, y_i) \in T} -\log P(y_i|x_i, \theta) \tag{3}$$

$$\hat{y} = \arg\max_{y} P(y|x_i, \theta) \tag{4}$$

where θ represents all the aforementioned parameters, including $Q^{(\text{ner})}$, $\text{MLP}^{(\text{ner})}$, $b^{(\text{ner})}$, and BiLSTM. $(x_i, y_i) \in T$ denotes input data from training set T, and y_i represents the gold label for input x_i. $\hat{y} = \arg\max_y P(y|x_i, \theta)$ is a set of predicted labels.

It is common to replace the MLP-based classifiers with the CRF-based classifiers because the possible combinations of NEs are not considered in MLP. Following the baseline CRF model, we also implemented CRF models on top of LSTM output h as the input of CRF. The performance of the CRFs and MLP classifiers is investigated in Section 7.

5.3. BERT-based model

As mentioned in the previous section, the word vectors we used were trained in a static environment, which led to a lack of context information. To address this issue, a system should model the entire sentence as a source of contextual information. Consequently, the word representation method has been rapidly adjusted to adapt new embeddings trained by a masked language model (MLM). The MLM randomly masks some of the input tokens and predicts the masked word by restoring them to the original token. During the prediction stage, the MLM captures contextual information to make better predictions.

Recent studies show that BERT trained by bidirectional transformers with the MLM strategy achieved outperforming results in many NLP tasks proposed in the General Language Understanding Evaluation (GLUE) benchmark (Wang *et al.* 2018). Since then, several transformer-based models have been proposed for NLP, and they show outstanding performances over almost all NLP tasks. In this section, we propose an NER system that is based on the new annotation scheme and apply the multilingual BERT and XLM-RoBERTa models to the Korean NER task.

BERT representation layer. BERT's tokenizer uses a WordPiece-based approach, which breaks words into subword units. For example, the word "snowing" is divided into "snow" and "##ing". In contrast, the XLM-RoBERTa tokenizer uses a BPE-based approach (Sennrich, Haddow, and Birch 2016), which can produce more subwords than the WordPiece approach. XLM-RoBERTa uses bidirectional modeling to check context information and uses BPE when training MLM. The XLM-RoBERTa decomposes a token x_i into a set of subwords. For example, a token 실내 *silnae* ("interior") is decomposed to three subwords, namely "시 *si*", "ㄹ *l*", and "내 *nae*" by the XLM-RoBERTa tokenizer. Since Korean has only a finite set of such subwords, the problems of unknown words that do not appear in the training data T are addressed effectively. Once the BERT representation is trained by MLM using massive plain texts, the word representation and all parameters from training are stored as the pre-trained BERT. Recently, there have been several pre-trained transformer-based models in public use. Users are generally able to fine-tune the

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

pre-trained BERT representation for their downstream tasks. We implemented our BERT-based NER systems using the pre-trained BERT models provided by Hugging Face.[g] In the BERT-based system, a word is tokenized and transformed into a sequence of BERT representations as $V_i^{(b)} = \left\{v_{i,1}^{(b)}, v_{i,2}^{(b)}, \ldots, v_{i,j}^{(b)}\right\}$, where i and j j denote the i-th word from the number of n words and the number of BPE subwords of the i-th word, respectively. We only take the first BPE subword for each word as the result of the BERT representations proposed in Devlin *et al.* (2019). As an illustration, our system obtains embeddings for the subword "snow" in the word "snowing," which is segmented into "snow" and "##ing" by its tokenizer. This is because BERT is a bidirectional model, and even the first BPE subword for each token contains contextual information over the sentence. Finally, the resulting word representation becomes $V^{(b)} = \left\{v_{1,1}^{(b)}, v_{2,1}^{(b)}, \ldots, v_{n,1}^{(b)}\right\}$. Following what has been done to the RNN-based model, we concatenate the POS embedding to $v_{i,1}^{(b)}$ if available.

Classification layer. We apply the same classification layer of the RNN-based model with an identical negative log-likelihood loss for the BERT-based model.

6. Experiments

6.1. Data

The Korean NER data we introduce in this study are from NAVER, which was originally prepared for a Korean NER competition in 2018. NAVER's data include 90,000 sentences, and each sentence consists of indices, words/phrases, and NER annotation in the BIO-like format from the first column to the following columns, respectively. However, sentences in NAVER's data were segmented based on *eojeol*, which, as mentioned, is not the best way to represent Korean NEs.

We resegment all the sentences into the morpheme-based representation as described in Park and Tyers (2019), such that the newly segmented sentences follow the CoNLL-U format. An `eoj2morph` script is implemented to map the NER annotation from NAVER's *eojeol*-based data to the morpheme-based data in the CoNLL-U format by pairing each character in NAVER's data with the corresponding character in the CoNLL-U data, and removing particles and postpositions from the NEs mapped to the CoNLL-U data when necessary. The following presents an example of such conversions:

	NAVER			(proposed) CONLLU			
				1-2	프랑스의	_	*peurangseu-ui*
1	프랑스의	B-LOC	⇒	1	프랑스	B-LOC	*peurangseu* ('France')
				2	의	_	*-ui* ('-GEN')

In particular, the script includes several heuristics that determine whether the morphemes in an *eojeol* belong to the NE the *eojeol* refers to. When both NAVER's data that have *eojeol*-based segmentation and the corresponding NE annotation, and the data in the CoNLL-U format that only contain morphemes, UPOS (universal POS labels, Petrov *et al.* 2012), and XPOS (Sejong POS labels as language-specific POS labels) features are provided, the script first aligns each *eojeol* from NAVER's data to its morphemes in the CoNLL-U data and then determines the morphemes in this *eojeol* that should carry the NE tag(s) of this *eojeol*, if any. The criteria for deciding whether the morphemes are supposed to carry the NE tags are that these morphemes should not be adpositions (prepositions and postpositions), punctuation marks, particles, determiners, or verbs. However, cases exist in which some *eojeols* that carry NEs only contain morphemes of the aforementioned types. In these cases, when the script does not find any morpheme that can carry the NE tag, the size of the excluding POS set above will be reduced for the given *eojeol*. The script will first attempt

[g]https://github.com/huggingface/transformers

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Algorithm 1. Pseudo-code for converting data from NAVER's *eojeol*-based format into the morpheme-based CoNLL-U format

```
Result: CoNLL-U format for Korean NER
align words ws between S_NAVER and S_CoNLL-U where S_NAVER = w_1 : t_1 ... w_x : t_x and
 S_CoNLL-U = w'_1 ... w'_{l-n}, m_l ... m_n, w'_{n+1} ... w'_y;
while w_i S_NAVER and w'_j S_CoNLL-U where i and j are aligned indexes do
    if w_i : t_i and t_i == O then
        assign O to w'_j or Os to m_j, ..., m_k for w'_{j-k};
    else
        if w_i : t_i and t_i != O and w'_j then
            assign t_i to w'_j
        else
            # in case of w'_{j-k};
            assign t_i to m_j ;
            while m_h where m_{j+1} .. m_k do
                assign I-t_i until m_h is not a functional morpheme;
            end
        end
    end
end
```

to find a verb in the *eojeol* to bear the NE annotation and, subsequently, will attempt to find a particle or a determiner. The excluding set keeps shrinking until a morpheme is found to carry the NE annotation of the *eojeol*. Finally, the script marks the morphemes that are in between two NE tags representing the same NE as part of that NE (e.g., a morpheme that is between B-LOC and I-LOC is marked as I-LOC) and assigns all other morphemes an "O" notation, where B, I, and O denote beginning, inside and outside, respectively. Because the official evaluation set is not publicly available, the converted data in the CoNLL-U format from NAVER's training set are then divided into three subsets, namely the training, holdout, and evaluation sets, with portions of 80%, 10%, and 10%, respectively The corpus is randomly split with seed number 42 for the baseline. In addition, during evaluation of neural models, we use the seed values 41–45 and report the average and their standard deviation. Algorithm 1 describes the pseudo-code for converting data from NAVER's *eojeol*-based format into the morpheme-based CoNLL-U format. Here, w_i:t_i in S_{NAVER} represents a word w_i and its NE tag t_i (either an entity label or `O`). $w'_{i-k}m_i \ldots m_k$ in $S_{\text{CoNLL-U}}$ displays a word w'_{i-k} with its separated morphemes $m_i \ldots m_k$.

We also implement a `syl2morph` script that maps the NER annotation from syllable-based data (e.g., KLUE and MODU) to the data in the morpheme-based CoNLL-U format to further test our proposed scheme. While the `eoj2morph` script described above utilizes UPOS and XPOS tags to decide which morphemes in the *eojeol* should carry the NE tags, they are not used in `syl2morph` anymore, as the NE tags annotated on the syllable level already exclude the syllables that belong to the functional morphemes. Additionally, NEs are tokenized as separate *eojeols* at the first stage before being fed to the POS tagging model proposed by Park and Tyers (2019). This is because the canonical forms of Korean morphemes do not always have the same surface representations as the syllables do. Because the `syl2morph` script basically follows the similar principles of `eoj2morph`, except for the two properties (or the lack thereof) mentioned above, we simply present an example of the conversion described above:

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

	KLUE			(proposed) CONLLU			
1	프	B-LOC		1-2	프랑스의	_	*peurangseu-ui*
2	랑	I-LOC	⇒	1	프랑스	B-LOC	*peurangseu* ("France")
3	스	I-LOC		2	의	_	*-ui* ("-GEN")
4	의	_					

We further provide `morph2eoj` and `morph2syl` methods which allow back-conversion from the proposed morpheme-based format to either NAVER's *eojeol*-based format or the syllable-based format, respectively. The alignment algorithm for back-conversion is simpler and more straightforward given that our proposed format preserves the original *eojeol* segment at the top of the morphemes for each decomposed *eojeol*. As a result, it is not necessary to align morphemes with *eojeols* or syllables. Instead, only *eojeol*-to-*eojeol* or *eojeol*-to-syllable matching is required. The `morph2eoj` method assigns NE tags to the whole *eojeol* that contains the morphemes these NE tags belong to, given that in the original *eojeol*-based format, *eojeols* are the minimal units to bear NE tags. The `morph2syl` method first locates the *eojeol* that carries the NE tags in the same way as described above for `morph2eoj`. Based on the fact that NEs are tokenized as separate *eojeols* by `syl2morph`, the script assigns the NE tags to each of the syllables in the *eojeol*.

Back-conversions from the converted datasets in the proposed format to their original formats are performed. Both *eojeol*-based and syllable-based datasets turned out to be identical to the original ones after going through conversions and back-conversions using the aforementioned script, which shows the effectiveness of the proposed algorithms. Manual inspection is conducted on parts of the converted data, and no error is found. While the converted dataset may contain some errors that manual inspection fails to discover, back-conversion as a stable inspection method recovers the datasets with no discrepancy. Therefore, we consider our conversion and back-conversion algorithms to be reliable. On the other hand, no further evaluation of the conversion script is conducted, mainly because the algorithms are tailored in a way that unlike machine learning approaches, linguistic features and rules of the Korean language, which are regular and stable, are employed during the conversion process.

6.2. Experimental setup

Our feature set for baseline CRFs is described in Figure 5. We use `crf++`[h] as the implementation of CRFs, where `crf++` automatically generates a set of feature functions using the template.

The hyperparameter settings for the neural models are listed in Table 9, Appendix A. We apply the same hyperparameter settings as in Lim *et al.* (2018) for BiLSTM dimensions, MLP, optimizer including β, and learning rate to compare our results with those in a similar environment. We set 300 dimensions for the parameters including $LSTM$ in (1), Q, and MLP in (2). In the training phase, we train the models over the entire training dataset as an epoch with a batch size of 16. For each epoch, we evaluate the performance on the development set and save the best-performing model within 100 epochs, with early stopping applied.

The standard F_1 metric $(= 2 \cdot \frac{P \cdot R}{P+R})$ is used to evaluate NER systems, where precision (P) and recall (R) are as follows:

$$P = \frac{\text{retrieved named entities} \cap \text{relevant named entities}}{\text{retrieved named entities}}$$

$$R = \frac{\text{retrieved named entities} \cap \text{relevant named entities}}{\text{relevant named entities}}$$

We evaluate our NER outputs on the official evaluation metric script provided by the organizer.[i]

[h] https://taku910.github.io/crfpp

[i] https://github.com/naver/nlp-challenge

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Table 4. CRF/Neural results using different models using NAVER's data converted into the proposed format.

Baseline	LSTM	BERT-MULTI		XLM-ROBERTA		KLUE-ROBERTA	
CRF	+CRF	+MLP	+CRF	+MLP	+CRF	+MLP	+CRF
71.50	$84.76_{\pm 0.29}$	$87.04_{\pm 0.41}$	$87.28_{\pm 0.37}$	$87.93_{\pm 0.30}$	$88.16_{\pm 0.33}$	$88.77_{\pm 0.39}$	$88.84_{\pm 0.43}$

fastText (Joulin *et al.* 2016) for LSTM+CRF word embeddings.

`# Unigram`			
w_{-2}		p_{-2}	
w_{-1}		p_{-1}	
w_0	(current word)	p_0	(current pos)
w_1		p_1	
w_2		p_2	
`# Bigram`			
w_{-2}/w_{-1}		p_{-2}/p_{-1}	
w_{-1}/w_0		p_{-1}/p_0	
w_0/w_1		p_0/p_1	
w_1/w_2		p_1/p_2	

Figure 5. CRF feature template example for *word* and *pos*.

7. Results

We focus on the following aspects of our results: (1) whether the conversion of Korean NE corpora into the proposed morpheme-based CoNLL-U format is more beneficial compared to the previously proposed *eojeol*-based and syllable-based styles, (2) the effect of multilingual transformer-based models, and (3) the impact of the additional POS features on Korean NER. The outputs of the models trained using the proposed morpheme-based data are converted back to their original format, either *eojeol*-based or syllable-based, before evaluation. Subsequently, all reported results are calculated in their original format for fair comparisons, given the fact that the numbers of tokens in different formats vary for the same sentence. Nevertheless, it is worth noting that all experiments using the morpheme-based CoNLL-U data are both trained and predicted in this proposed format before conducting back-conversion.

7.1. Intrinsic results on various types of models

We compare the intrinsic results generated by different types of models. By saying "intrinsic," it implies that the results in this subsection differ owing to the ways and approaches of learning from the training data. We compare the performances of the baseline CRFs and our proposed neural models, and we also investigate the variations when our models use additional features in the data.

Table 4 summarizes the evaluation results on the test data based on the proposed machine learning models in Section 5. Comparing the transformer-based models with LSTM + crf, we found that both multilingual BERT-based models (bert-multi, xlm-roberta) outperformed LSTM + crf. The comparison reveals a clear trend: the word representation method is the most

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Table 5. CRF/Neural results using the various sets of features using NAVER's data converted into the proposed format.

Baseline CRF	LSTM+CRF		XLM-ROBERTA+CRF		
WORD	WORD	+UPOS	WORD	+UPOS	+XPOS
71.50	$84.76_{\pm 0.29}$	$84.94_{\pm 0.34}$	$88.16_{\pm 0.33}$	$88.41_{\pm 0.27}$	$88.37_{\pm 0.22}$

Table 6. CRF/Neural result comparison between the proposed CoNLL-U format versus NAVER's *eojeol*-based format using NAVER's data where POS features are not applied.

Baseline CRF		XLM-ROBERTA+CRF	
CONLL-U	NAVER	CONLL-U	NAVER
71.50	49.15	$88.16_{\pm 0.33}$	$86.72_{\pm 0.49}$

effective for the NER tasks adopting our proposed scheme. For the LSTM + crf model, we initialized its word representation, v_i^w, with a `fastText` word embedding (Joulin *et al.* 2016). However, once we initialized word representation using BERT, we observed performance improvements up to 3.4 points with the identical neural network structure as shown in Table 4. Meanwhile, there are two notable observations in our experiment. The first observation is that the CRF classifier exhibits slightly better performance than the MLP classifier, with an improvement of 0.23, for xlm-roberta. However, the improvement through the use of the CRF classifier is relatively marginal compared with the reported results of English NER (Ghaddar and Langlais 2018). Moreover, when comparing the multilingual models (bert-multilingual and xlm-roberta) to the monolingual model (klue-roberta), we found that klue-roberta outperforms both bert-multilingual and xlm-roberta. This is because klue-roberta is trained solely on Korean texts and utilizes better tokenizers for the Korean language (Park *et al.* 2021).

Table 5 details results using various sets of features. We use incremental words, UPOS, and XPOS, for the input sequence x and their unigram and bigram features for CRFs. Both LSTM and XLM-RoBERTa achieve their best F_1 score when only the +UPOS feature is attached.

7.2. Extrinsic results on different types of data

This subsection examines the extrinsic results given different types of data, whereas the previous subsection focuses on the differences in various models given only the morpheme-based CoNLL-U data. In this subsection, the performances of our models trained on the datasets either in the proposed format or in their original formats, and in either the BIO tagging format or the BIOES tagging format, are investigated.

As described in Table 6, both the baseline CRF-based model and the BERT-based model achieve higher F_1 scores when the proposed CoNLL-U data are used, in contrast with NAVER's *eojeol*-based data. The testing data are organized in a way that the total number of tokens for evaluations remains the same, implying that the F_1 scores generated by `conlleval` are fair for both groups. This is realized by converting the model output of the morpheme-based CoNLL-U data back into NAVER's original *eojeol*-based format such that they have the same number of tokens. The morpheme-based CoNLL-U format outperforms the *eojeol*-based format under the CRFs, whereas the CoNLL-U format still outperformed the *eojeol*-based format by over 1% using the BERT-based model.

Table 7 presents the comparative results on two types of tagging formats where the models do not use additional POS features. Previous studies show that the BIOES annotation scheme yields superior results for several datasets if the size of datasets is enough to disambiguate more number

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Table 7. CRF/Neural result comparison between BIO versus BIOES annotations using NAVER's data converted into the proposed format where POS features are not applied.

Baseline CRF		XLM-ROBERTA+CRF	
BIO	BIOES	BIO	BIOES
71.50	70.67	$88.16_{\pm 0.33}$	$85.70_{\pm 0.35}$

Table 8. Result comparison between the proposed CoNLL-U format and the syllable-based format using MODU (19 21), KLUE, and ETRI datasets where POS features are not applied.

MODU 19		MODU 21		KLUE		ETRI	
CONLL-U	SYLLABLE	CONLL-U	SYLLABLE	CONLL-U	SYLLABLE	CONLL-U	SYLLABLE
$88.03_{\pm 0.20}$	$84.91_{\pm 0.35}$	$81.72_{\pm 0.31}$	$78.10_{\pm 0.45}$	$91.72_{\pm 0.29}$	$88.15_{\pm 0.42}$	$97.59_{\pm 0.12}$	$93.28_{\pm 0.37}$

Model: XLM-ROBERTA+CRF.

of labels (Ratinov and Roth 2009). Both the baseline CRF and neural models achieve higher F_1 scores than that of the BIOES tagging format when the BIO tagging format is used, which has two more types of labels – E as endings and S as single entity elements. Our result reveals that adding more target labels to the training data degrades model prediction (14 labels × 2 for E and S). Accordingly, we observe that in this specific case, adding the two additional labels mentioned increases the difficulty of predictions.

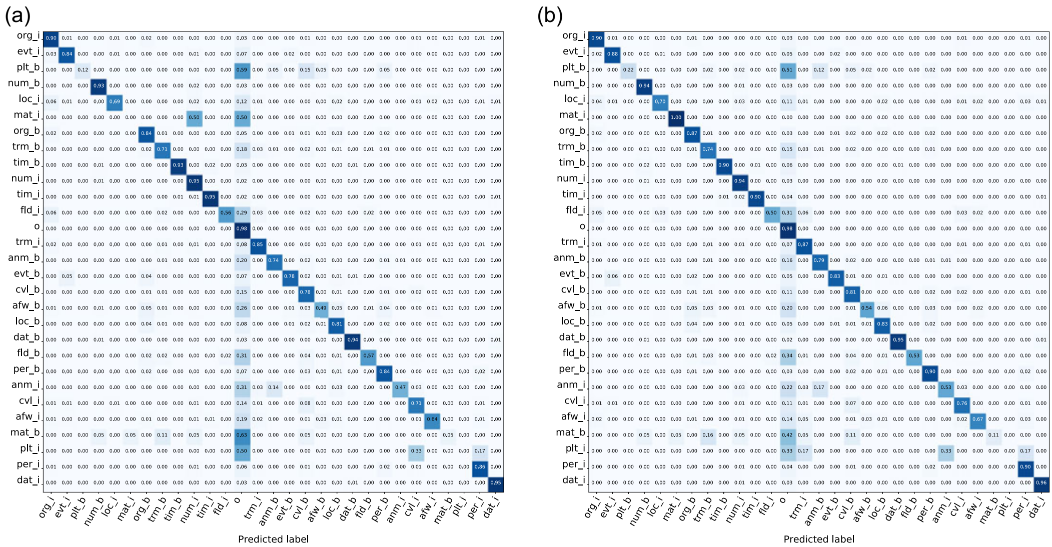
Table 8 compares the results of the proposed morpheme-based CoNLL-U data and the syllable-based data. We use the xlm-roberta+crf model and only use word features for the experiments. Similar to the previous experiments, we back-convert the model output of the morpheme-based data back to the syllable-based format for fair comparisons. The results are consistent that for all four datasets, we observe performance improvement ranging from 3.12 to 4.31 points when the CoNLL-U format is adopted.

The performance difference between syllable-based NER results and morpheme-based NER results is mainly due to the fact that the xlm-roberta+crf model we used employs a subword-based tokenizer with larger units, rather than a syllable-based tokenizer. Therefore, one needs to use a BERT model with a syllable-based tokenizer for fair comparisons, if the goal is to only compare the performance between the morpheme-based format and the syllable-based format. However, this makes it difficult to conduct a fair performance evaluation in our study, because the evaluation we intended is based on BERT models trained in different environments when morpheme-based, *eojeol*-based, or syllable-based data are given. Since subword-based tokenizers are widely employed in a lot of pre-trained models, our proposed format would benefit the syllable-based Korean NER corpora in a way that not only language models using syllable-based tokenizers can take them as the input, but those using BPE tokenizers or other types of subword-based tokenizers can also be trained on these syllable-based corpora once converted.

7.3. POS features and NER results

Additional POS features help detect better B-∗ tag sets. Comparing the XLM-RoBERTa models trained with and without the UPOS tags, we observe an average performance improvement of 0.25% when the UPOS feature is applied to the XLM-RoBERTa model. However, if we only focus on the entities named B-∗, the performance improvement increases by 0.38. We hypothesize that POS features in the NER system play a role of evidence in tagging decision-making for the start of its entities across tokens. For example, a sentence in (3a) exhibits unknown words such as the names of a location: 하르키우와 돈바스 *haleukiu-wa donbaseu* ("Kharkiv and Donbas").

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press



BERT confusion matrix XLM-RoBERTa confusion matrix

Figure 6. Comparison of the confusion matrix between the BERT (6a) and XLM-RoBERTa (6b). Model on the test set. XLM-RoBERTa tends to show better results in finding the "I-∗" typed entities, with +0.7% of average score.

(3) a.

하르키우와	돈바스	지역	방면을	향한	공세는	...
haleukiu-wa	*donbaseu*	*jiyeog*	*bangmyeon-eul*	*hyanghan*	*gongseneun*	...
Kharkiv-CONJ	Donbas	regions	around-ACC	toward	offensive-TOP	...

"(The intensity of Russian) fire toward the regions of Kharkiv and Donbas ..."

b.

1-2	하르키우와	_	_	_	*haleukiu-wa*
1	하르키우	하르키우	PROPN	NNP	*haleukiu* ("Kharkiv")
2	와	과	ADP	JC	*-wa* ("CONJ")
3	돈바스	돈바스	PROPN	NNP	*donbaseu* ("Donbas")
...					

The NER system suffers from inadequate information, but when the system has POS information (proper noun, `PROPN` in UPOS and `NNP` in XPOS) as in (3b), it can probably make better decisions.

The UPOS tag set benefits Korean NER. Because syntactic-relevant tasks such as dependency parsing for Korean have mainly applied XPOS tag sets, we assumed the XPOS tags are more important than UPOS tags. However, results in Table 5 demonstrate that applying the UPOS feature gives marginally better results than using the XPOS feature. We consider this to be due to the predicted POS tags. Since the UPOS set contains 17 labels whereas the XPOS set (Sejong POS tag set) has 46 labels for Korean, the current prediction model is more accurate in tagging UPOS labels than tagging XPOS labels. As a result, a more reliable UPOS feature provided by the tagging model also turns out to be more helpful in NER, compared with the XPOS feature.

7.4. Comparison between XLM-RoBERTa and BERT

We reported that XLM-RoBERTa outperforms the multilingual BERT model in Table 4. The differences between the two language models are discerned from Figure 6, which displays the

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Eojeol			Morpheme			Syllable			Proposed		
1	프랑스의	B-LOC	1	프랑스	B-LOC	1	프	B-LOC	1-2	프랑스의	_
			2	의	O	2	랑	I-LOC	1	프랑스	B-LOC
						3	스	I-LOC	2	의	O
						4	의	O			

Figure 7. Various approaches of annotation for NEs in Korean.

confusion matrix between the predicted and true labels for both BERT and XLM-RoBERTa. XLM-RoBERTa tends to exhibit superior results in finding the "I-∗" typed entities, with +0.7% of the average score. Moreover, we observed that the XLM-RoBERTa model exhibits fewer false predictions of `b-mat` and `i-plt` as `o` than the BERT model. It was difficult to explain the exact factors leading to the clear performance differences between BERT and XLM-RoBERTa. However, given the size of the training models where BERT and XLM are 641 MB and 1.2 GB, respectively, we assume that the model size might have contributed to the performance differences. However, it seemed that the method of tokenizing and word dictionaries had a larger impact than the model size because the size of the monolingual KLUE-RoBERTa model is 110 MB which is much smaller.

8. Discussion

In this section, we discuss how the proposed scheme differs from the previously adopted schemes that represent NEs in Korean corpora. As described previously in Figure 2, previous studies proposed other annotation formats for Korean NER, including *eojeol*-based, morpheme-based, and syllable-based approaches. We demonstrate how different annotation formats look like in Korean corpora in Figure 7, which also includes our proposed format.

8.1. Comparison with the eojeol-based format

The *eojeol*-based format, as adopted in NAVER's dataset, considers *eojeols* as the tokens to which NE tags are assigned. The format preserves the natural segmentation of the Korean language and is commonly used in other Korean processing tasks such as dependency parsing (Choi *et al.* 1994; McDonald *et al.* 2013; Chun *et al.* 2018). On the other hand, taking *eojeols* as the basic units for NER analyses has an inevitable defect. Korean is an agglutinative language in which functional morphemes, which are not parts of the NE, are attached to the *eojeol* as well. As a result, it is impossible to exclude what does not belong to the NE from the token that bears the NE. Our proposed format has two advantages compared to the *eojeol*-based format. While the functional morphemes cannot be excluded from the NE annotation, a morpheme-based format deals with this problem properly, as the tokens are no longer *eojeols*, but morphemes instead. Consequently, the NE tags are assigned to the morphemes that are not functional. Moreover, since our proposed format keeps the *eojeol* representation on the top of its morphemes, the information on the natural segmentation of the sentence is still preserved. The advantages of the proposed scheme are testified through experiments, and the results presented in Table 6, Section 7.2 show that the format in which functional morphemes are excluded from NE labels improves the performance of the NER models trained on the same set of sentences but in different formats.

8.2. Comparison with the previous morpheme-based format

Previous studies on Korean NER attempted to employ morpheme-based annotation formats to tackle the problem of Korean's natural segmentation which is not able to exclude functional

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

morphemes. As discussed in Section 3, the KIPS corpora consider morphemes to be the basic units for NER analyses, and therefore the NE tags are assigned to only those morphemes that belong to the NE. While the morpheme-based format adopted in the KIPS corpora seems to be able to solve the problem the *eojeol*-based format has, it also introduces new problems. The information on the natural segmentation (i.e., *eojeols*) has been completely lost, and it is no longer possible to tell which morphemes belong to which *eojeol*. This poses challenges both for human researchers to understand the corpora and for the conversion from the morpheme-based format back to the *eojeol*-based format. Not being able to observe the word boundaries and how the morphemes constitute *eojeols* makes it hard for linguists and computer scientists to understand the language documented in the corpora. Moreover, since there is no information that suggests which morphemes should form an *eojeol*, the script is not able to recover the corresponding *eojeol*-based representation with NE tags. Our proposed format preserves all the *eojeols* in the dataset and clearly states which morphemes belong to the *eojeols* through indices, as illustrated in Figure 7. Conversions back to the *eojeol*-based format have been made possible because of the *eojeol*-level representations, which the `morph2eoj` script can refer to. Meanwhile, the proposed format is much more human-readable and presents linguistically richer information, which benefits the language documentation of Korean as well.

8.3. Comparison with the syllable-based format

Another annotation format that deals with the defect of the *eojeol*-based format, in which functional morphemes cannot be excluded, is the syllable-based format. Given that any syllable in Korean is never separated or shared by two morphemes or *eojeols*, marking the NE tags on those syllables which do not belong to functional morphemes solves the problem as well. However, there are some potential issues when adopting this straightforward approach. The major problem here is that most, if not all, linguistic properties are lost when the sentences are decomposed into syllables, as a syllable is an even smaller unit than a morpheme which is the smallest constituent that can bear meaning. Moreover, it is impossible to assign POS tags to the syllable-based data, which have been proven helpful for Korean NER using morpheme-based data as presented in Table 5. It is also not ideal to train a model which does not employ a syllable-level tokenizer using syllable-based corpora as discussed in Section 7.2, whereas many transformer-based models adopt BPE tokenization. Our proposed format addresses the above issues by preserving linguistic information and properties in the smallest units possible, namely morphemes. POS tags can therefore be assigned to each token as long as it carries meaning. We also provide a script that performs conversion from the syllable-based dataset into a morpheme-based dataset, such that a model using a subword-based tokenizer can fully utilize syllable-based corpora as additional data for further training.

9. Conclusion

In this study, we leverage the morpheme-based CoNLL-U format based on Park and Tyers (2019) and propose a new scheme that is compatible with Korean NER corpora. Compared with the conventional *eojeol*-based format in which the real pieces of NEs are not distinguished from the postpositions or particles inside the *eojeol*, the proposed CoNLL-U format represents NEs on the morpheme-based level such that NE annotations are assigned to the exact segments representing the entities. The results of both the CRF models and the neural models reveal that our morpheme-based format outperforms both the *eojeol*-based format and the syllable-based format currently adopted by various Korean NER corpora. We also examine the performance of the above formats using various additional features and tagging formats (BIO and BIOES). Meanwhile, this paper also investigates the linguistic features of the Korean language that are related to NER. This includes the distribution of postpositions and particles after NEs, the writing system of Korean, and the *eojeol*-based segmentation of Korean. The linguistic features we stated further justify the annotation scheme that we propose. Finally, we confirm the functionality of our NER models, and the results obtained reveal the differences among all these models, where the XLM-RoBERTa model outperformed other types of models in our study. The results of the study validate the feasibility of the proposed approach to the recognition of NEs in Korean, which is justified to be appropriate linguistically. The proposed method of recognizing NEs can be used in several real-world applications such as text summarization, document search and retrieval, and machine translation.

Acknowledgements. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2021R1F1A1063474) to KyungTae Lim.

References

Ahmadi F. and Moradi H. (2015). *A hybrid method for Persian named entity recognition*. **In** *Proceedings of the 7th Conference on Information and Knowledge Technology (IKT)*. Urmia: Institute of Electrical and Electronics Engineers, pp. 1–7.

Benikova D., Biemann C. and Reznicek M. (2014). *NoSta-D named entity annotation for German: Guidelines and dataset*. **In** *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik: European Language Resources Association (ELRA), pp. 2524–2531.

Benikova D., Yimam S. M., Santhanam P. and Biemann C. (2015). *GermaNER: Free open German named entity recognition tool*. **In** *Proceedings of International Conference of the German Society for Computational Linguistics and Language Technology (GSCL-2015)*. Essen: University of Duisburg-Essen, pp. 31–38.

Bojanowski P., Grave E., Joulin A. and Mikolov T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* **5**, 135–146.

Carreras X., Màrquez L. and Padró L. (2003). *Named entity recognition for Catalan using only Spanish resources and unlabelled data*. **In** *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics*. Budapest: Association for Computational Linguistics, pp. 43–50.

Chen Y., Jo E. L., Yao Y., Lim K., Silfverberg M., Tyers F. M. and Park J. (2022). *Yet another format of universal dependencies for Korean*. **In** *Proceedings of the 29th International Conference on Computational Linguistics*. Gyeongju: International Committee on Computational Linguistics, pp. 5432–5437.

Chieu H. L. and Ng H. T. (2003). *Named entity recognition with a maximum entropy approach*. **In** *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*. Edmonton: SIGNLL, pp. 160–163.

Cho G. S. and Kim S. B. (2021). *Korean named entity recognition using data augmentation techniques*. **In** *Proceedings of 2021 Fall Conference by Korean Institute of Industrial Engineers*. Seoul: Korean Institute of Industrial Engineers, pp. 648–662.

Choi D., Park J. and Choi K.-S. (2012). *Korean treebank transformation for parser training*. **In** *Proceedings of the ACL 2012 Joint Workshop on Statistical Parsing and Semantic Processing of Morphologically Rich Languages*. Jeju: Association for Computational Linguistics, pp. 78–88.

Choi K.-S., Han Y. S., Han Y. G. and Kwon O. W. (1994). *KAIST tree bank project for Korean: Present and future development*. **In** *Proceedings of the International Workshop on Sharable Natural Language Resources*. Nara Institute of Science and Technology, pp. 7–14.

Chun J., Han N.-R., Hwang J. D. and Choi J. D. (2018). *Building universal dependency treebanks in Korean*. **In** *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki: European Language Resources Association (ELRA).

Chung E., Hwang Y.-G. and Jang M.-G. (2003). *Korean named entity recognition using HMM and CoTraining model*. **In** *Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages*. Sapporo: Association for Computational Linguistics, pp. 161–167.

Dernoncourt F., Lee J. Y. and Szolovits P. (2017). *NeuroNER: An easy-to-use program for named-entity recognition based on neural networks*. **In** *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Copenhagen: Association for Computational Linguistics, pp. 97–102.

Devlin J., Chang M.-W., Lee K. and Toutanova K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*. **In** *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, MN: Association for Computational Linguistics, pp. 4171–4186.

Ghaddar A. and Langlais P. (2018). *Robust lexical features for improved neural network named-entity recognition*. **In** *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, NM: Association for Computational Linguistics, pp. 1896–1907.

Grave E., Bojanowski P., Gupta P., Joulin A. and Mikolov T. (2018). *Learning word vectors for 157 languages*. **In** *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki: European Language Resources Association (ELRA), pp. 3483–3487.

Hahm Y., Park J., Lim K., Kim Y., Hwang D. and Choi K.-S. (2014). *Named entity corpus construction using Wikipedia and DBpedia ontology*. **In** *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. Reykjavik: European Language Resources Association, pp. 2565–2569.

Han C.-H., Han N.-R., Ko E.-S., Palmer M. and Yi H. (2002). *Penn Korean treebank: Development and evaluation*. **In** *Proceedings of the 16th Pacific Asia Conference on Language, Information and Computation*. Jeju: Pacific Asia Conference on Language, Information and Computation, pp. 69–78.

Jie Z. and Lu W. (2019). *Dependency-guided LSTM-CRF for named entity recognition*. **In** *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong: Association for Computational Linguistics, pp. 3862–3872.

Jin G. and Yu Z. (2021). A Korean named entity recognition method using Bi-LSTM-CRF and masked self-attention. *Computer Speech Language* **65**, 101134.

Joulin A., Grave E., Bojanowski P., Douze M., Jégou H. and Mikolov T. (2016). FastText.zip: Compressing text classification models. http://arxiv.org/abs/1612.03651

Jung S., Lee C. and Hwang H. (2018). End-to-end Korean part-of-speech tagging using copying mechanism. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)* **17**(3), 19:1–19:8.

Kim H. and Kim H. (2020). Integrated model for morphological analysis and named entity recognition based on label attention networks in Korean. *Applied Sciences* **10**(11), 3740.

Kim H.-J. and Kim H.-S. (2019). *How to use effective dictionary feature for deep learning based named entity recognition*. **In** *Proceedings of the 31st Annual Conference on Human and Cognitive Language Technology*. Daejeon: SIGHCLT: Special Interest Group of Human and Cognitive Language Technology, pp. 293–296.

Kim H.-J., Oh S.-H. and Kim H.-S. (2020). *Korean named entity recognition using ELECTRA and label attention network*. **In** *Proceedings of the 32nd Annual Conference on Human and Cognitive Language Technology*. Online: SIGHCLT: Special Interest Group of Human and Cognitive Language Technology, pp. 333–336.

Kim M. and Park J. (2022). A note on constituent parsing for Korean. *Natural Language Engineering* **28**(2), 199–222.

Kim Y.-M. and Lee T.-H. (2020). Korean clinical entity recognition from diagnosis text using BERT. *BMC Medical Informatics and Decision Making* **20**(7), 242.

Lample G., Ballesteros M., Subramanian S., Kawakami K. and Dyer C. (2016). *Neural architectures for named entity recognition*. **In** *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, CA: Association for Computational Linguistics, pp. 260–270.

Lee H.-g., Park G. and Kim H. (2018). Effective integration of morphological analysis and named entity recognition based on a recurrent neural network. *Pattern Recognition Letters* **112**, 361–365.

Li W. and McCallum A. (2003). Rapid development of Hindi named entity recognition using conditional random fields and feature induction. *ACM Transactions on Asian Language Information Processing* **2**(3), 290–294.

Lim K., Park C., Lee C. and Poibeau T. (2018). *SEx BiST: A multi-source trainable parser with deep contextualized lexical representations*. **In** *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Brussels: Association for Computational Linguistics, pp. 143–152.

McCallum A. and Li W. (2003). *Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons*. **In** *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*. Edmonton: SIGNLL, pp. 188–191.

McDonald R., Nivre J., Quirmbach-Brundage Y., Goldberg Y., Das D., Ganchev K., Hall K., Petrov S., Zhang H., Täckström O., Bedini C., Bertomeu Castelló N. and Lee J. (2013). *Universal dependency annotation for multilingual parsing*. **In** *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Sofia: Association for Computational Linguistics, pp. 92–97.

Mikolov T., Sutskever I., Chen K., Corrado G. S. and Dean J. (2013). Distributed representations of words and phrases and their compositionality. In **Burges C. J. C., Bottou L., Welling M., Ghahramani Z. and Weinberger K. Q.**, (eds), *Advances in Neural Information Processing Systems 26*. Lake Tahoe, NV: Curran Associates, Inc., Lake Tahoe, pp. 3111–3119.

Min J., Na S.-H., Shin J.-H. and Kim Y.-K. (2019). *RoBERTa for Korean natural language processing: Named entity recognition, sentiment analysis, dependency parsing*. **In** *Proceedings of Korea Software Congress 2019*. Pyeongchang: Korean Institute of Information Scientists and Engineers, pp. 407–409.

Min J., Na S.-H., Shin J.-H., Kim Y.-K. and Kim K. (2021). *Korean named entity recognition as machine reading comprehension*. **In** *Proceedings of Korea Software Congress 2021*. Pyeongchang: Korean Institute of Information Scientists and Engineers, pp. 344–346.

Nivre J., de Marneffe M.-C., Ginter F., Goldberg Y., Hajic J., Manning C. D., McDonald R., Petrov S., Pyysalo S., Silveira N., Tsarfaty R. and Zeman D. (2016). Universal dependencies v1: A multilingual treebank collection. In L. von Ahn (ed), *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. Portorož: European Language Resources Association (ELRA), pp. 1659–1666.

https://doi.org/10.1017/S1351324923000311 Published online by Cambridge University Press

Nivre J., de Marneffe M.-C., Ginter F., Hajič J., Manning C. D., Pyysalo S., Schuster S., Tyers F. and Zeman D. (2020). *Universal dependencies v2: An evergrowing multilingual treebank collection.* **In** *Proceedings of the 12th Language Resources and Evaluation Conference*. Marseille: European Language Resources Association, pp. 4034–4043.

Nothman J., Ringland N., Radford W., Murphy T. and Curran J. R. (2013). Learning multilingual named entity recognition from Wikipedia. *Artificial Intelligence* **194**(Supplement C), 151–175.

Nouvel D., Antoine J.-Y., Friburger N. and Soulet A. (2013). *Mining partial annotation rules for named entity recognition (Fouille de règles d'annotation partielles pour la reconnaissance des entités nommées) [in French]).* **In** *Proceedings of TALN 2013 (Volume 1: Long Papers)*. Les Sables d'Olonne: ATALA, pp. 421–434.

Park G., Park S., Jang Y., Choi K. and Kim H. (2017a). *KACTEIL-NER: Named entity recognizer using deep learning and ensemble technique.* **In** *Proceedings of the 29th Annual Conference on Human and Cognitive Language Technology*. Daegu: SIGHCLT: Special Interest Group of Human and Cognitive Language Technology, pp. 324–326.

Park J. (2018). Le benchmarking de la reconnaissance d'entités nommées pour le français. In *Actes de la Conférence TALN. Volume 1 - Articles longs, articles courts de TALN*. Rennes: ATALA, pp. 241–250.

Park J., Dugast L., Hong J.-P., Shin C.-U. and Cha J.-W. (2017b). *Building a better bitext for structurally different languages through self-training.* **In** *Proceedings of the First Workshop on Curation and Applications of Parallel and Comparable Corpora*. Taipei: Asian Federation of Natural Language Processing, pp. 1–10.

Park J., Hong J.-P. and Cha J.-W. (2016). *Korean language resources for everyone.* **In** *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation: Oral Papers (PACLIC 30)*. Seoul: Pacific Asia Conference on Language, Information and Computation, pp. 49–58.

Park J. and Kim M. (2023). A role of functional morphemes in Korean categorial grammars. *Korean Linguistics* **19**(1), 1–30.

Park J. and Tyers F. (2019). *A new annotation scheme for the Sejong part-of-speech tagged Corpus.* **In** *Proceedings of the 13th Linguistic Annotation Workshop*. Florence: Association for Computational Linguistics, pp. 195–202.

Park S., Moon J., Kim S., Cho W. I., Han J., Park J., Song C., Kim J., Song Y., Oh T., Lee J., Oh J., Lyu S., Jeong Y., Lee I., Seo S., Lee D., Kim H., Lee M., Jang S., Do S., Kim S., Lim K., Lee J., Park K., Shin J., Kim S., Park L., Oh A., Ha J.-W. and Cho K. (2021). KLUE: Korean language understanding evaluation. Technical report. https://klue-benchmark.com

Pennington J., Socher R. and Manning C. D. (2014). *GloVe: Global vectors for word representation.* **In** *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha: Association for Computational Linguistics, pp. 1532–1543.

Petrov S., Das D. and McDonald R. (2012). *A universal part-of-speech tagset.* **In** *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*. Istanbul: European Language Resources Association (ELRA), pp. 2089–2096.

Ratinov L. and Roth D. (2009). *Design challenges and misconceptions in named entity recognition.* **In** *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*. Boulder, CO: Association for Computational Linguistics, pp. 147–155.

Sennrich R., Haddow B. and Birch A. (2016). *Neural machine translation of rare words with subword units.* **In** *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin: Association for Computational Linguistics, pp. 1715–1725.

Strubell E., Verga P., Belanger D. and McCallum A. (2017). *Fast and accurate entity recognition with iterated dilated convolutions.* **In** *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen: Association for Computational Linguistics, pp. 2670–2680.

Tjong Kim Sang E. F. and De Meulder F. (2003). *Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition.* **In** *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*. Edmonton: Association for Computational Linguistics, pp. 142–147.

Wang A., Singh A., Michael J., Hill F., Levy O. and Bowman S. (2018). *GLUE: A multi-task benchmark and analysis platform for natural language understanding.* **In** *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Brussels: Association for Computational Linguistics, pp. 353–355.

You H.-J., Song Y., Kim M. S., Yun G. and Cheong Y. (2021). Error analysis and evaluation of deep-learning based Korean named entity recognition. *The Korean Journal of Linguistics* **46**(3), 803–828.

Yu H. and Ko Y. (2016). *Named entity recognition using bidirectional LSTM CRFs based on the POS tag embedding and the named entity distribution of syllables.* **In** *Proceedings of the 28th Annual Conference on Human and Cognitive Language Technology*. Busan: SIGHCLT: Special Interest Group of Human and Cognitive Language Technology, pp. 105–110.

Yun B.-H. (2007). HMM-based Korean named entity recognition for information extraction. In Zhang Z. and Siekmann J., (eds), *Knowledge Science, Engineering and Management*. Berlin, Heidelberg: Springer, pp. 526–531.

Appendix A. Hyperparameters of the Neural Models

Table 9. Hyperparameters.

Component	Value
Q (parameters) dim.	300
$v_i^{(w)}$ (fastText) dim.	300
$v_i^{(u)}$ (UPOS) dim.	50
$v_i^{(x)}$ (XPOS) dim.	50
$v_{i,j}^{(b)}$ (BERT) dim.	768
BiLSTM hidden dim.	512
No. BiLSTM layers	2
MLP output dim.	300
Dropout	0.3
Learning rate of RNN	0.002
Learning rate of BERT	0.00002
β_1, β_2	0.9, 0.99
Epoch	100
Batch size	16
Random seed	41, 42*, 43, 44, and 45
Gradient clipping	5.0

* For the baseline system.

Cite this article: Chen Y, Lim K and Park J (2024). Korean named entity recognition based on language-specific features. *Natural Language Engineering* **30**, 625–649. https://doi.org/10.1017/S1351324923000311