American Journal of Neural Networks and Applications
2026, Vol. 12, No. 1, pp. 11–24
https://doi.org/10.11648/j.ajnna.20261201.12

SciencePG
Science Publishing Group

Research Article

Word Embeddings Network and Transformer Based Part of Speech Tagging for Korean

Pong-Gol You, Chun-Sik So, Song-Min Choe, Yong-Hak Lee, Chol-Jun O*

Institute of Mathematics, State Academy of Sciences, Pyongyang, Democratic People's Republic of Korea

Abstract

Korean Part-of-Speech (POS) tagging is different from and more difficult than other languages such as English, Russian and Chinese due to raising issues of Korean word segmentation and analysis of sound-changed morphemes. In this paper we propose a transformer-based Korean POS tagging model, which combines the output vector of a encoder of the transformer with a representational vector of the input word obtained from character-level word embeddings network unlike existing deep learning-based POS tagging models based on BiLSTM. First, in order to perform segmentation of words and changed sound analysis at once, we have designed a model to make a new output sequence of the POS tagging model as a sequence of pairs of strings of morphemes and its POS tags. Second, in order to obtain character-level word representations, word embedding network employing convolution network and highway network are trained. Finally, to make more efficient use of the semantic information of the input word in generating of sequences of POS tagging, we combined the word representation vector obtained from the word-embedding generation network with the output of a encoder of the transformer. According to the experimental results, the proposed model achieves 1.4% performance improvement over the model without incorporating the word representation vector obtained from the word embeddings network, and as a result, the POS tagging accuracy is 96.1%, which is superior to all other compared models including the BiLSTM+CRF model.

Keywords

Korean POS Tagging, Word Embedding, Character-level Transformer, Natural Language Processing

1. Introduction

Korean Part-of-Speech (POS) tagging is different from the one of other languages such as English, Russian and more difficult and more important than them.

In a Korean sentence, each word is segmented by space like English or Russian. We can segment Korean words to morphemes using the space, but it is more difficult to extract construct of Korean words than English or Russian words due to complexity and variety of combining of morphemes within Korean words. Korean belonging to agglutinative language is very elegant morphologically and Korean *"To* (particle)", the grammatical particle which represents a grammatical morphology of Korean word, is very rich and combining forms of them are diverse. These attributes of Korean are able to raise language's power of expression and able to ensure correctness of intent carried, but make linguistic processing, particularly morphological processing such as POS tagging awkward.

Korean words can be formalized as the following.

$$W = (P^* + R^* + S^* + T^*)^*$$

*Correspondence: Chol-Jun O (ocj1989@star-co.net.kp)

Received: 16 November 2025; **Accepted:** 9 December 2025; **Published:** 7 April 2026

where W is word, P is prefix, R is the root of a word, S is suffix, T is Korean *"To"* and '*' represents that its preceding factor are repeated zero or more times than zero.

For example, Korean word "평교원들속에는" consists of prefix "평", the root of a word "교원", Korean *"To"* "들", the root of a word "속", Korean *"To"* "에", "는". Also "감자밭비배관리에서도" consists of the root of a word "감자", "밭", "비배", "관리", *"To"* "에서", "도".

Therefore, if you wish to develop the Korean POS tagger, first of all you should divide Korean words to morphemes and then attach the POS tag to each morpheme.

On the other hand English words can be formalized as the following.

$$W = P + R + S$$

In English words prefix and suffix do not influence to the POS tagging of the word, so the issue to segment words do not raise.

As mentioned above the words in Korean sentence are segmented by space like English or Russian words, but Korean POS tagging raise the additional issue to segment words into morphemes unlike English POS tagging.

Segmentation of words are raised in some language such as Chinese and Japanese in which the sentences are not divided to words by space. Such languages do not space words, so before POS tagging it should preprocess to segment the sentence to words. But in Chinese segmentation of words is different from the one in Korean.

In Korean some characters of morphemes are changed when two or more morphemes are combined each other. For example, Korean word "헛총질해왔다" is consist of combinations of prefix "헛", the root of a word "총질하(다)", *"To"* "여", the root of a word "오(다)", *"To"* "았", "다", where the root of a word "총질하(다)" and *"To"* "여" are combined and changed into "총질해". And the root of a word "오(다)" and *"To"* "았" are combined and changed into "왔(다)". Also "아름다운" is consist of the root of a word "아름답(다)" and *"To"* "은" and the root of a word "아름답(다)" and *"To"* "은" are combined and changed into "아름다운" instead of changing into "아름답은". Therefore in Korean word segmentation the issue to find out the original morphemes of changed words, i.e. changed sound analysis is raised additionally.

The POS taggers for English or Chinese mostly employed bidirectional long short-term memory (BiLSTM) model. But it needs the additional process to use this model for Korean since in Korean the segmentation of words and the changed sound analysis are raised together.

Recently, in NLP tasks Seq2Seq model is widely used. Seq2Seq model is the DL model that inputs sequence of tokens and outputs sequence of tokens. Korean POS tagging can be carried out using this model.

As mentioned above, Korean POS tagging is to segment words into morphemes and attach the POS (include *"To"*) tag to each morpheme. In a few words, Korean POS tagging is to input words (a sequence of characters) and output the sequence of pairs of morpheme and its POS tag. To mention a word "학생들에게" as example, Korean POS tagging is to input character string "학생들에게" and output the sequence of the pairs of morphemes and its POS tags "학생<Noun>들<plural To>에게<case To>". In the rest of this paper, the sequence of pairs of morpheme and its POS tag of a word are simply named as the sequence of POS tagging.

In the sequence of POS tagging of Korean word, if POS and *"To"* tags are considered as the specific characters then Korean POS tagging is come to a Seq2Seq task that inputs the sequence of characters and outputs the sequence of characters.

In this work, we present a Korean POS tagging model using character-level transformer and compare this model with different preceding models.

First, we present the Korean POS tagging model using a character-level transformer and second, in order to improve the performance of the presented model, we combine character-level word embedding with the model.

According to experimental results, the presented model outperform the model combining conditional random fields (CRF) and BiLSTM (BiLSTM+CRF) as well as encoder-decoder model based on LSTM and character-level transformer model. The experimental results show the presented model is very efficient in Korean POS tagging.

2. Related Work

Since the POS tagging is fundamental stage of many NLP tasks, a number of approaches including rule-based and statistical approaches have been offered. Recent years many DL based POS tagging models have been offered and they outperformed the all different previous approaches and achieved the state-of-the-art accuracies.

The most works on the DL based POS tagging use BiLSTM [2-4, 9, 10, 12, 16, 17].

Applying BiLSTM tagger for English, Makarenkov, V. et al. [10] presented a novel DL based application of word choice task for writing scientific papers by non-native English speakers.

For the POS tagging of Arabic tweets, AlKhwiter, W. et al. [2] presented two supervised POS taggers that are developed based on two approaches: CRF and BiLSTM models. The BiLSTM-based POS tagger achieved the state-of-the-art results with 96.5% accuracy for the "Mixed" dataset that Modern Standard Arabic (MSA) and Arabic dialects, namely Gulf

(GLF) mixed.

In the study of Anbukkarasi et al. [3], it was addressed for the problem of POS tagging for the Tamil language, which is low resourced and agglutinative. For this work, various sequential DL models such as recurrent neural network (RNN), LSTM, Gated Recurrent Unit (GRU) and BiLSTM were evaluated. According to experiment results, among all the combinations, BiLSTM with 64 hidden states displayed the best accuracy (94%).

Pathak, D. et al. [12] presented a DL-based POS tagger for Assamese. This tagger employs the pre-trained word embeddings and BiLSTM-CRF. For presenting Assamese POS tagger, several pre-trained word embeddings are evaluated and the top-performing model FlairEmbeddings [1] is chosen. Using this embedding, they compared the performance of CRF and BiLSTM+CRF model. As a result the tagging accuracy of CRF model attained with an F1 score of 84.88%, BiLSTM+CRF model 86.52%.

In the study of Xiang, Z. et al. [16], it was proposed a Masked Part-of-Speech Model (MPoSM) using BiLSTM, inspired by the recent success of Masked Language Models (MLM). They evaluated the performance of the MPoSM on 10 diverse languages, MPoSM achieves competitive performance on most languages, but achieves the lowest performance on Indonesian and so on.

Hongwei, L. et al. [6] proposed a model based on Transformer for English POS tagging. Attending that in POS tagging, BiLSTM is commonly used and achieves good performance, but BiLSTM is not as powerful as Transformer in leveraging contextual information, since BiLSTM simply concatenates the contextual information from left-to-right and right-to-left, In this work, they obtain all possible POS tags of each token and select a single tag of the token by using rule-based methods and obtain the tag of each token in a sentence by using a masked transformer. Experimental results show that the proposed method achieves the better performance than other methods using BiLSTM.

Transformer first was proposed in the study of Vaswani, A. et al. [14], which has encoder-decoder structure. The encoder maps the input sequence to a sequence of representations, the decoder generates an output sequence from the sequence of representations obtained by encoder. At each step the model auto-regressively generates the output. Both the encoder and decoder are composed of a stack of 6 identical layers. Each layer has a multi-head self-attention mechanism and a fully connected feed-forward network (Figure 1). And the decoder also has another layer, which performs multi-head attention over the output of the encoder.

Vaswani, A. et al. [14] experimented on English-to-German and English-to-French translation tasks and achieved the new state-of-the-art performances.

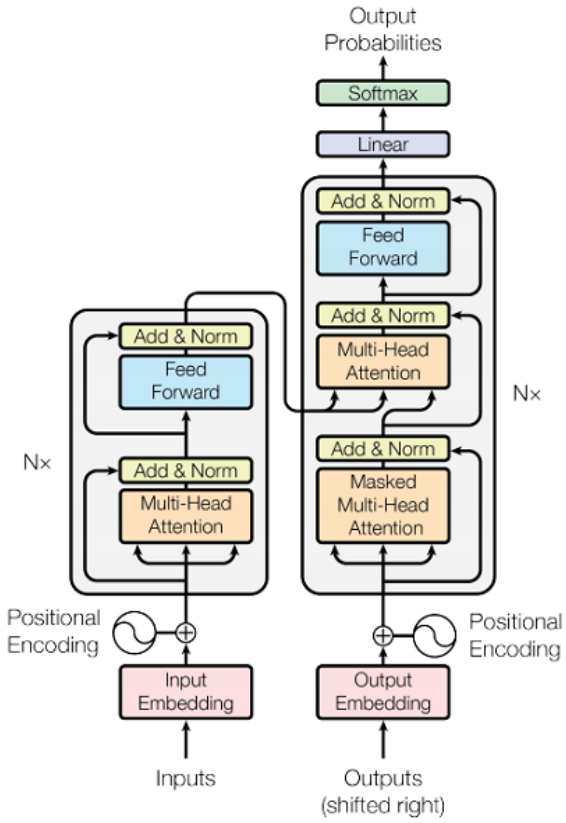

Figure 1. *The Transformer-model architecture.*

Ying, X. et al. [17] investigated Chinese word segmentation (CWS) and POS tagging for Chinese clinical text. They compared two state-of-the-art methods, i.e. CRF and BiLSTM with a CRF layer. As a result, if only CWS was considered, CRF achieved higher performance than BiLSTM+CRF. When both CWS and POS tagging were considered, CRF also gained an advantage over BiLSTM. CRF outperformed BiLSTM+CRF by 0.14% in F-measure on CWS and by 0.34% in F-measure on POS tagging. The CWS information brought a greatest improvement of 0.34% in F-measure, while the CWS&POS information brought a greatest improvement of 0.74% in F-measure.

Furthermore, Dangguo, S. et al. [5] proposed a domain-specific CWS based on BiLSTM model. First, the word segmentation model is obtained by using the BiLSTM model to train on metallurgical training set and the open data set (MSRA corpus). Then the metal-BiLSTM model and msr-BiLSTM model are obtained respectively. Second, the label probability of the two models are combined to obtain the probability of each Chinese character label. Then, the probability of each label of the combined Chinese characters is calculated by Viterbe algorithm. The final probability of each label is obtained. Then, the probability values of each Chinese character under each label are compared. Finally, the maximum probability label is used as the one of each Chinese character. Thus the Chinese word segmentation is completed. Experimental results showed that the accuracy of the word segmentation method proposed in this paper is 97.6% and this method can achieve a better word segmentation effect.

In the study of Dangguo, S. et al. [5], the general structure of CWS based on neural network was showed (Figure 2).

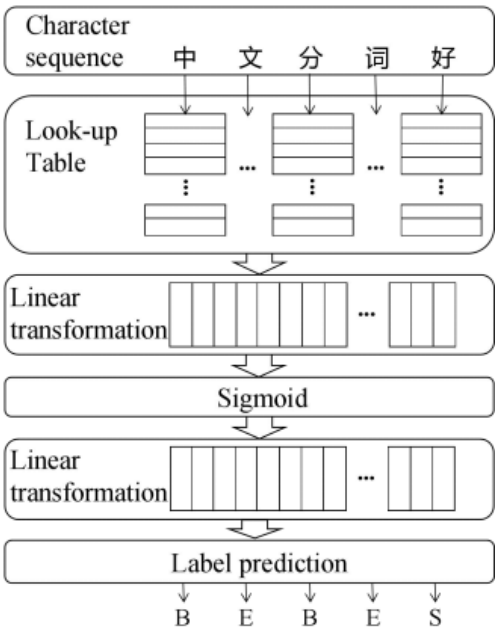

Figure 2. *The general structure of CWS based on neural network (labels "B", "E", "S" represent the start character, the end character of word and the single character, respectively).*

In the study of Dangguo, S. et al. [5], a BiLSTM was used instead of two Linear transformation in the general neural network structure. As figure show, each Chinese character is attached only a label. But in Korean there are cases that more than two characters combine into one character due to sound-changed morphemes. In these cases it is not able to attach only a label to each character.

In DL based methods, how to obtain the vector representation of word affects the performance of the method. The same is true of the case with POS tagging. From importance of word representation, Priyadarshi, A. et al. [13] proposed POS tagging using pre-trained word embeddings. To evaluate the importance of the vector representation of words in POS tagging, first POS tagger using CRF classifier was trained on given dataset and the accuracy of 82.67% was achieved. Secondly word embeddings were trained using the word2vec Continuous Bag Of Words (CBOW) model on the corpora created using Wikipedia Maithili dumps and other web resources. Incorporating generated word vectors into the system, the accuracy was improved to 85.88%. This paper shows intuitively the importance of word embeddings in POS tagging.

Yoon, K. et al. [18] proposed character-level language model, a notable approach is to use character-level word embeddings. In this work, the vector representation of input word is obtained by employing a convolutional neural network (CNN) and a highway network, whose output is given to a LSTM (Figure 3). The proposed model outperforms word-level/morpheme-level LSTM baselines, again with fewer parameters.

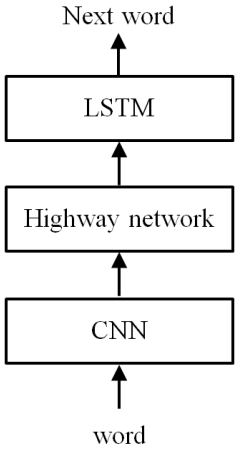

Figure 3. *A architecture of character-level language model.*

3. Proposed Approach

3.1. Architecture of Korean POS Tagger Based on Neural Network

In NLP model based on neural network which is choosed as token is the important issue.

In Korean, token is able to be one of character and subword, morpheme, word and according to which of them is used as token, the models can be classified to character-level, subword-level, morpheme-level and word-level model.

In word-level or morpheme-level models Out of Vocabulary (OOV) issue is raised and segmentation issue is raised in subword-level models. So in this paper we use character-level model.

LSTM is a well-known effective structure to address long-term dependency issue in sequence models. Although LSTM can capture long-term dependencies, Transformer is much better at capturing long-term dependencies because its attention mechanism sees all context directly. Transformer was shown to outperform LSTM and is now replacing LSTM in all sequence models [8].

In this work we employ a character-level transformer for Korean POS tagging. The character-level transformer follows the structure proposed in the study of Vaswani, A. et al. [14] as it was. It is different from the study of Vaswani, A. et al. [14] only in that both of input and output are Korean characters. Figure 4 shows the structure of the character-level transformer for POS tagging (POS tagging Transformer; PTT).

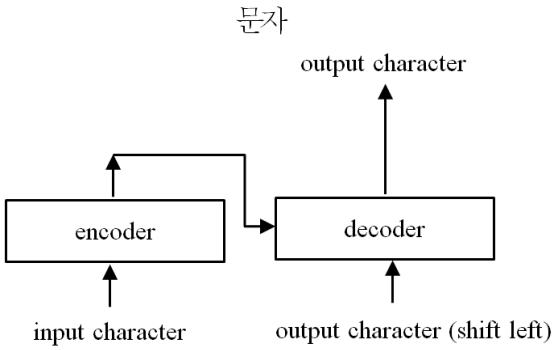

Figure 4. *The architecture of PTT.*

3.2. Character-level Model to Generate Word Embeddings

PTT proposed above only depends on characters of a word and does not use a bit the context information of the word in a sentence. In a sentence words are not independent of other words and have the relationships with other words in context of the one. Only if one consider the context information of words, the sense of words can be identified correctly and POS tagging can be carried out more correctly. Therefore, in this work we obtain the word embeddings impacted from context information of the one and improve the accuracy of Korean POS tagging adding the word embeddings to PTT.

In order to obtain the word embeddings impacted from context information, we employ the structure proposed in the study of Yoon, K. et al. [18].

Figure 5 shows the structure of Character-level model to generate word embeddings (Word Embedding Network; WEN). The aim of the study of Yoon, K. et al. [18] is to propose character-level language model, but our work aims at obtaining character-level word embeddings. That is, in this work output of Highway network is used for word embeddings of input words instead of using the output of character-level model to generate word embeddings. Therefore we replace LSTM by a Feed-Forward Network (FFN) so that the probability distribution of context words are simply obtained.

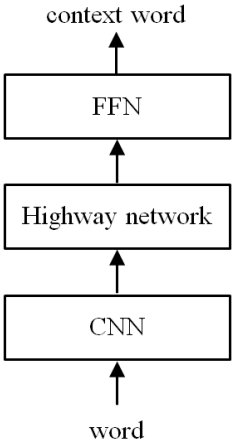

Figure 5. *The structure of Character-level model to generate word embeddings.*

3.3. The Combination of WEN and PTT

Given a word $w = (c_1, \ldots c_n)$, we assume that the word embedding of w from WEN is $x \in R^m$ and the output vector of c_i from encoder in PTT is $y_i \in R^l$. Then y_i is modified to z_i as following.

$$z_i = x \oplus y_i \tag{1}$$

The $z_i \in R^{m+l}$ obtained in this way is become to the final representation of c_i from encoder to be inputted to decoder (Figure 6).

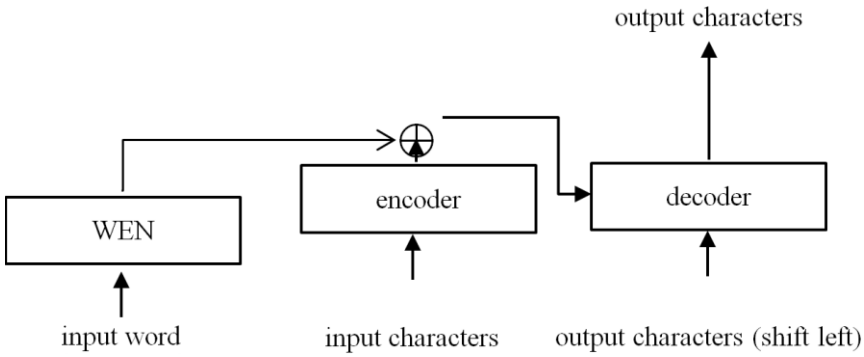

Figure 6. The architecture of the improved PTT.

4. Experiments

4.1. Methods

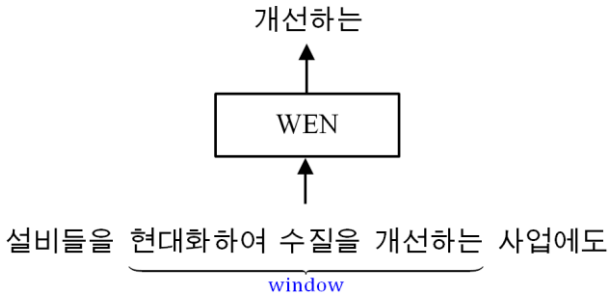

Figure 7. The training of WEN using skip gram (size of window = 3).

Before training the PTT, WEN are trained to generate word embeddings by using skip gram. The size of window of skip gram was setted to 3 and extracting the pairs of words and their context words from Korean text corpus, the train dataset is prepared. The WEN is inputted the words and outputs the context words (Figure 7).

After training the WEN, the PTT is trained using word embeddings generated from WEN. The train data have the type of (word embedding, character string of word, sequence of POS tagging), where word embedding and character string of word are input data, and sequence of POS tagging is output data.

4.2. Data

4.2.1. Tag Set

D. P. R. K. Standard tag set has been declared the national standard for annotating Korean language data. It contains 60 tags with 25 top-level categories. The description of all tag sets is presented in Table 1.

Table 1. D. P. R. K. Standard tag set.

No	POS, To	tag	English
1	명사	Nn	Noun
1-1	고유명사	nnp	proper noun
1-2	불완전명사	nnb	bound noun
2	수사	nm	numeral
3	대명사	pr	pronoun
4	동사	vb	verb
4-1	보조동사	vbx	auxiliary verb
5	형용사	aj	adjective
5-1	보조형용사	ajx	auxiliary adjective

No	POS, To	tag	English
6	부사	ad	adverb
6-1	접속부사	adc	conjunctive adverb
7	관형사	pn	pre-noun
8	감동사	ij	interjection
9	접두사	pf	prefix
9-1	명사접두사	pfn	noun prefix
9-2	수사접두사	pfm	numeral prefix
9-3	동사접두사	pfv	verbal prefix
9-4	형용사접두사	pfj	adjective prefix
9-5	부사접두사	pfd	adverbial prefix
10	접미사	sf	suffix
10-1	명사접미사	sfn	noun suffix
10-2	수사접미사	sfm	numeral suffix
10-3	동사접미사	sfv	verb suffix
10-4	형용사접미사	sfj	adjective suffix
10-5	부사접미사	sfd	adverb suffix
11	격토	tc	case particle
11-1	주격토	tcs	subjective particle
11-2	대격토	tco	objective particle
11-3	속격토	tcp	possessive particle
11-4	여격토	tcd	dative particle
11-5	위격토	tcl	locative particle
11-6	조격토	tci	instrumental particle
11-7	구격토	tcc	company particle
11-8	호격토	tcv	vocative particle
12	도움토	tx	auxiliary particle
13	복수토	tu	plural particle
14	바꿈토	th	change particle
15	상토	tv	voice particle

No	POS, To	tag	English
16	존경토	tr	honorific particle, respected
17	시간토	tt	time particle
18	규정토	td	determinative particle
18-1	현재규정토	tds	determinative particle-present
18-2	과거규정토	tdp	determinative particle-past
18-3	미래규정토	tdf	determinative particle-future
19	접속토	tj	conjunctive particle
19-1	병렬접속토	tjp	parallel conjunctive particle
19-2	대립접속토	tjo	opposite conjunctive particle
19-3	선택접속토	tjn	optional conjunctive particle
19-4	원인접속토	tjc	causative conjunctive particle
19-5	조건접속토	tjd	conditional conjunctive particle
20	상황토	tm	modifiable particle
21	종결토	tp	predicate particle
21-1	알림종결토	tpd	declarative predicate particle
21-2	물음종결토	tpq	interrogative predicate particle
21-3	시킴종결토	tpi	imperative predicate particle
21-4	추김종결토	tpa	advisory predicate particle
22	기호	sb	symbol
23	외국말표기	fc	foreign character
24	전문용어	ht	technical terms
25	성구	im	idiom

We except "성구(idiom)" category from 25 top-level categories showed above and we merge "외국말표기(foreign character)" category and "전문용어(technical terms)" category into "명사" category. As a result, we have 22 top-level categories.

4.2.2. Vocabulary of Characters

Both of WEN and PTT use the same vocabulary of character.

The vocabulary of character is consisted of 2,978 characters and symbols in all including 2,605 Korean characters in common use and number, English characters, Russian characters, Greek characters, unit symbols, declarative symbols, scientific symbols, punctuation marks and POS tags such as "nn", "pr", "tx", "tc".

4.2.3. Dataset

WEN: The WEN was trained on 415MB text data, which includes 5,360k sentences, 61,240k words, 5,620k single

words. The vocabulary of words is consisted of 80k words which are the most frequent words among 5,620k single words. The frequency of a word which has the lowest frequency in vocabulary is 61(Figure 8).

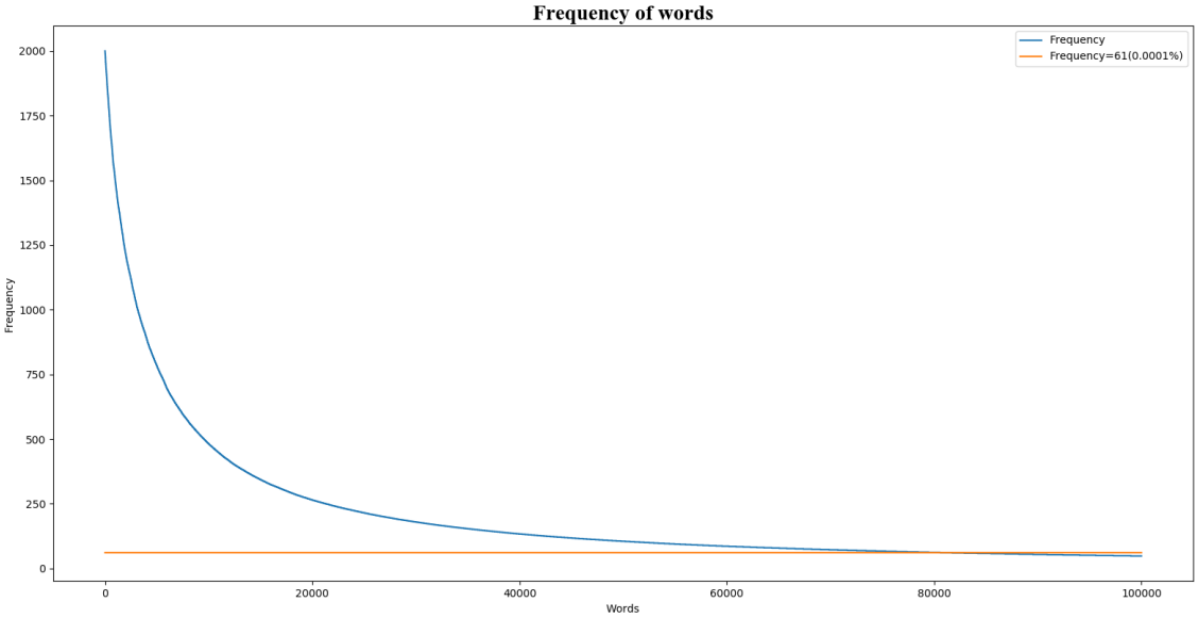

Figure 8. *The frequency of words in train dataset.*

PTT: In order to make dataset for training PTT, we extracted 3,200K single words and their sequences of POS taggings from annotated text corpus (includes 3,800k sentences, 47,000k words).

The maximum of length of 3,200K single words and their sequences of POS taggings is 38. So we set 40(be added start and end mark of words) as the maximum of length of input and output sequences in PTT. Figure 9 shows the distribute of length of words and sequences of POS tagging included in input data, output data and both of them, respectively.

Dataset is divided into train dataset (80%: 2,560k), validation dataset (10%: 320k) and test dataset (10%: 320k).

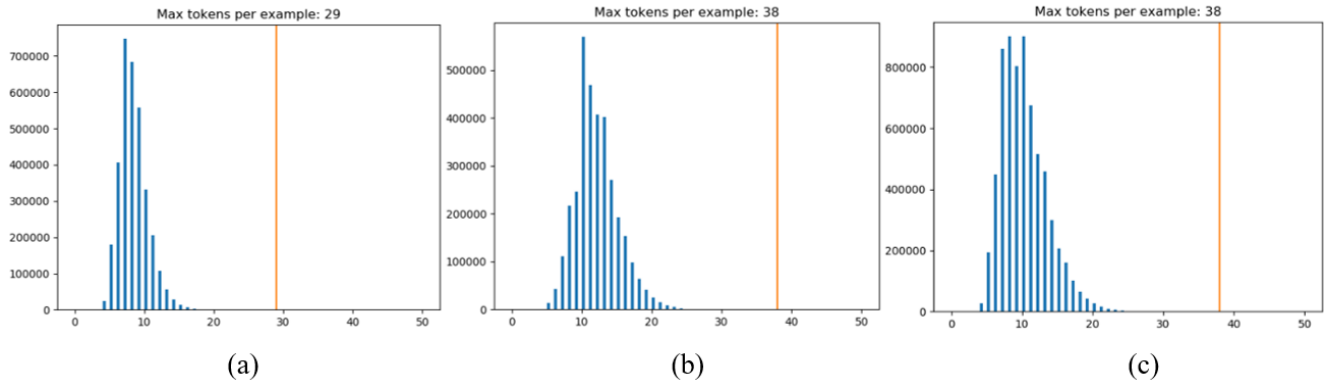

Figure 9. *(a) The distribute of length of Korean words in dataset; (b) The distribute of length of sequences of POS tagging in dataset; (c) The distribute of length of input and output sequences in dataset.*

4.3. Model

WEN

WEN was designed by combining default parameters in [18] and customized parameter. Since the dimensionality of word embeddings generated from WEN equals the sum of hs, the counts of filters, in experiments we set 3*w as the counts of filters with width w= [1-7] and 20 as the count of filters with width w = 9, so that the sum of counts of filters become 128. In other words, we designed the dimensionality of word embeddings generated from WEN to be 128.

The details of parameters of CNN, Highway network and FFN in WEN are showed in Table 2.

Training was performed using NCE (Noise contrastive estimation) loss function [11] and the Adam optimizer and a batch size of 64 was used.

Table 2. *The hyperparameters of WEN.*

Layer	Hyperparameter	Value
NN	size of vocabulary of characters	2,978
	dimensionality of character embeddings	128
	width of filters	[1-9]
	count of filters	[3, 6, 9, 12, 15, 18, 20, 21, 24]
	nonlinearity function	tanh
Highway	number of layers	1
	nonlinearity function	ReLU
FFN	number of layers	1
	size of vocabulary of words	80,000

PTT

Dimensionality of input and output of encoder, decoder in PTT were set to 128, 256, respectively. That is, dimensionality of input and output of encoder was set to 128, so that the dimensionality of final output of encoder which the output vector of encoder was concatenated with word embeddings (dimensionality of 128) from WEN become 256, which is the same dimensionality as input and output of decoder.

Dimensionality of inner layer of FFN was set to 512, and maximum length of input and outpour of PTT was set to 40.

In Table 3, we show the hyperparameters of PTT.

Training was performed using Adam optimizer and categorical cross-entropy as the loss function.

Table 3. *Hyperparameters of PTT.*

Hyperparameter	Value
Size of vocabulary of characters	2,978
Maximum length of input and output tokens	40
Dimensionality of input and output	encoder: 128, decoder: 256
Number of layers	encoder: 4, decoder: 4
Dimensionality of inner layer of FFN	512
Number of heads	8
Dropout rate	0.1
Batch size	64

5. Results

WEN

In training WEN, we obtained minimum average loss of 7.6 at 597,000 step and loss was not decreased no longer. In this time, words which have the closest representation to one of a given word are showed in Table 4.

Table 4. *Words close to the given word.*

	given word	close words
In Vocabulary	위한	위하여, 위하는, 위하시는, 위해주는, 위협하며, 위해, 지휘한, 결사옹위하는
	있는	있기는, 있지는, 있든, 있던, 있긴, 있으리라는, 지식있는, 있은,
	그러나	그러다간, 그런데다가, 그러는, 그러다가, 이러나, 이러나저러나, 이러다간, 그러던,
	높이	높이높이, 긍지높이, 높이는, 드높이, 기치높이, 불길높이, 소리높이, 높아지면서,
	나라의	나라와의, 나리의, 나라사이의, 나라들의, 마누라의, 나라에서의, 나라들사이의, 제자의,
	젊은이들에게	군사들에게, 자기들에게, 녀성들에게, 시민들에게, 군인들에게, 독자들에게, 마을사람들에게, 처녀들에게,
	그러구보니	그러거나말거나, 그랬더니, 어머니, 그러고는, 그러더니, 아바이, 이러니, 오니,
Out-Of-Vocabulary	성진제강련합기업소에서	흥남비료련합기업소에서, 락원기계련합기업소, 평양국제비행장, 상원세멘트련합기업소, 경제에서, 흥남비료련합기업소의, 형제산구역, 경제적, 순천세멘트련합기업소, 황해제철련합기업소, 련합기업소에서는,
	태천군	배천군, 강동군, 운전군, 온천군, 강남군, 온성군, 성천군, 출판보도부문,
	천명하시였다	표명하시였다, 치하하시였다, 바치시였다, 지시하시였다, 밝혀주시였다, 밝히시였다, 제시하시였다, 가르치시였다,

PTT

Training PTT was performed for 20 epochs. After 10 epochs, the accuracy on validation dataset reached a maximum 98.86% on the validation dataset and loss was 0.0543.

We compared the performance of PTT combined WEN to LSTM based encoder-decoder model and BiLSTM+CRF [5], only PTT. LSTM based encoder-decoder model has two LSTM layers in encoder and decoder respectively and attention to output of encoder in decoder.

Table 5. *Test dataset accuracy of different methods.*

Method	Accuracy (%)
BiLSTM+CRF	94.4
Encoder-Decoder	93.1
PTT	94.7
PTT+WEN	96.1

Table 5 shows the accuracies of compared methods on test dataset. BiLSTM+CRF model cannot deal with sound-changed words. Therefore, in the table the accuracy of BiLSTM+CRF is to obtain by evaluating only on sound-unchanged words. Nevertheless, our PTT+WEN outperformed BiLSTM+CRF in POS tagging accuracy.

The third column in the Table 5 shows that performance of proposed method is the highest among all compared methods. Also it shows that performance of PTT is higher 1.4% when it is combined with WEN than when it is not.

6. Analysis

In order to analysis the performance of our PTT in more detail, we experimented on another dataset, which consists of 20,000 words not included in the train dataset, validation dataset and test dataset. In this dataset there are proper nouns of 6,761, compound of nouns and pronouns, numerals (compound *Cheon* in Korean) of 7,686, compound of verbs and adjectives (compound *Yongon* in Korean) of 3,328, foreign words of 1,784, spacing errors of 326 and so on. The experiment results on each word class are same as Table 6.

Table 6. *Accuracy of POS tagging on different word classes.*

Word class	Accuracy (%)
proper noun	99.3
compound *Cheon*	98.2
compound *Yongon*	99.7
foreign word	87.3
spacing error	68.4

Table 7. *Examples of sequence of POS tagging obtained by PTT.*

Word	Sequence of POS tagging
왕명귀가족일행 (proper noun)	왕명귀<nn>가족<nn>일행<nn>
왕창욱대표의 (proper noun)	왕창욱<nn>대표<nn>의<tc>
조선장애어린이회복중심은 (compound *Cheon*)	조선<nn>장애<nn>어린이<nn>회복<nn>중심<nn>은<tx>
어린아이취급하고있다고 (compound *Yongon*)	어린아이<nn>취급<nn>하<vb>고<tj>있<vb>다고<tj>
겹쳐놓은것을 (compound *Yongon*)	겹치<vb>여<tj>놓<vb>은<td>것<nn>을<tc>
선생님모습그대로 (spacing error)	선생님<nn>모습<nn>그대로<ad>
전략이새로운 (spacing error)	전략<nn>이새롭<aj>은<td>
전쟁이지상에서 (spacing error)	전쟁<nn>이<th>지상<tj>에서<tc>

As shown in Table 6, the accuracy of PTT on proper noun, compound *Cheon* and compound *Yongon* is very high as 98%. In specialty, as shown in Table 7, our PTT performs correctly the tagging of relatively long length of compound *Cheons* and compound *Yongons*. This indicates that our PTT address long-term dependency issue very well.

Furthermore, our PTT address well segmentation and tagging for the sound-changed words.

And since the input unit of our PPT is character, PPT does not suffer from OOV issue and deals well with unknown words unlike word-level or morpheme-level models and previous rule-based or statistical models do.

While PPT has many benefits, PPT does not work well with spacing errors. As shown in Table 7, PTT tagged incorrectly "전략이새로운" with "전략<nn>이새롭<aj>은<td>"(correct tag is "전략<nn>이<th>새롭<aj>은<td>") and "전쟁이지상에서" with "전쟁<nn>이<th>지상<tj>에서<tc>"(correct tag is "전쟁<nn>이<th>지상<nn>에서<tc>"). But our PTT correctly tagged 68.4% of total spacing errors like tagging "선생님모습그대로" with "선생님<nn>모습<nn>그대로<ad>". And since spacing errors appear very infrequently in a text, the low accuracy of POS tagging on spacing errors little effect the accuracy of tagging on to overall text.

7. Conclusion

In this work, we trained and evaluated the Korean POS tagging model using character-level word embeddings and Transformer. First, we trained character-level word embedding network using the Convolution network and Highway network. Secondly, we trained the POS tagging model using character-level Transformer and the word embedding from the word embedding network. Finally, we evaluated experimentally the proposed model on two dataset: test dataset and analysis dataset. Our model has 96.1% of accuracy of POS tagging on test dataset and outperforms the other compared models.

Most of Korean words have only one POS tagging sequence corresponding to each word. In other words, POS tagging sequence of a Korean word has no change in any Korean sentence. From this specificity, it is able to design and train the model obtaining directly the sequences of POS tagging from words in this work.

However, there are some Korean words have more than two POS tagging sequences. For example, "가지는" has two POS tagging sequences: "가지<nn>는<tx>" and "가지(다)<vb>는<td>".

In order to perform POS tagging correctly for all Korean words including words like above, we should use additional information of other words in a sentence. In the future, we will study the Korean POS tagger using information of sentence.

Abbreviations

BiLSTM	Bidirectional Long Short-term Memory
CBOW	Continuous Bag of Words
CNN	Convolutional Neural Network
CRF	Conditional Random Fields
CWS	Chinese Word Segmentation
DL	Deep-Learning
FFN	Feed-Forward Network
LSTM	Long Short-term Memory
NCE	Noise Contrastive Estimation
NLP	Natural Language Processing
OOV	Out of Vocabulary
POS	Part-of-Speech
PTT	POS Tagging Transformer
Seq2Seq	Sequence-to-Sequence
WEN	Word Embedding Network

Author Contributions

Pong-Gol You: Conceptualization, Methodology, Writing – original draft

Chun-Sik So: Investigation, Validation

Song-Min Choe: Investigation, Validation

Yong-Hak Lee: Writing – original draft

Chol-Jun O: Writing – review & editing

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Conflicts of Interest

The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

[1] Akbik, A., Blythe, D., & Vollgraf, R. (2018). Contextual string embeddings for sequence labeling. in Proceedings of the 27th international conference on computational linguistics. Association for Computational Linguistics. (pp. 1638–1649).

[2] AlKhwiter, W., & Al-Twairesh, N. (2020). Part-of-speech Tagging for Arabic Tweets using CRF and BiLSTM. In Computer Speech & Language (pp. 1-21), https://doi.org/10.1016/j.csl.2020.101138

[3] Anbukkarasi, S., & Varadhaganapathy, S. (2021). Deep Learning based Tamil Parts of Speech (POS) Tagger. In Bulletin Of The Polish Academy Of Sciences Technical Sciences. Volume. 69(6). (pp. 1-6), https://doi.org/10.24425/bpasts.2021.138820

[4] Chotirat, S., & Meesad, P., (2021). Part-of-Speech tagging enhancement to natural language processing for Thai wh-question classification with deep learning. In Heliyon 7. (pp. 1-13), https://doi.org/10.1016/j.heliyon.2021.e08216

[5] Dangguo, S., Na Z., Zhaoqiang Y., Zhenhua C., Yan X., Yantuan X., & Zhengtao Y. (2019). Domain-Specific Chinese Word Segmentation Based on Bi-directional Long-Short Term Memory Model. In IEEE Access. (pp. 1-10), https://doi.org/10.1109/ACCESS.2019.2892836

[6] Hongwei, L., Hongyan, M., & Jingzi W. (2022). Part-of-Speech Tagging with Rule-Based Data Preprocessing and Transformer. In Electronics 2022, 11, 56. (pp. 1-14), https://doi.org/10.3390/electronics11010056

[7] Jason, L., Kyunghyun C., & Hofmann, T. (2017). Fully Character-Level Neural Machine Translation without Explicit Segmentation. In Transactions of the Association for Computational Linguistics. volume. 5. (pp. 365-378).

[8] Jinyu, L. (2022). Recent Advances in End-to-End Automatic Speech Recognition, arXiv preprint arXiv: 2111.01690. (pp. 1-27).

[9] Khan, W., Daud, A., Khan, K., Nasir, J. A., Basheri, M., Aljohani, N., & Alotaibi, F. S. (2019). Part of Speech Tagging in Urdu: Comparison of Machine and Deep Learning Approaches. In IEEE Access volume 7. (pp. 38918-38936), https://doi.org/10.1109/ACCESS.2019.2897327

[10] Makarenkov, V., Rokach, L., & Shapira, B. (2019). Choosing the right word: Using bidirectional LSTM tagger for writing support systems. In Engineering Applications of Artificial Intelligence 84. (pp. 1–10), https://doi.org/10.1016/j.engappai.2019.05.003

[11] Andres-Ferrer, J., Bodenstab, N., & Vozila, P. (2018). Efficient language model adaptation with Noise Contrastive Estimation and Kullback-Leibler regularization. In Interspeech 2018. (pp. 3368-3372).

[12] Pathak, D., Nandi, S. & Sarmah, P. (2022). AsPOS: Assamese Part of Speech Tagger using Deep Learning Approach. (pp. 1-9).

[13] Priyadarshi, A., & Saha, S. K. (2020). Towards the first Maithili part of speech tagger: Resource creation and system development. In Computer Speech & Language 62. (pp. 1-9), https://doi.org/10.1016/j.csl.2019.101054

[14] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. Attention is all you need. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.

[15] Wentao M. et al. (2020). CharBERT: Character-aware Pre-trained Language Mode. In Proceedings of the 28th International Conference on Computational Linguistics. (pp. 39-50).

[16] Xiang, Z., Shiyue, Z., & Bansal, M. (2022). Masked Part-Of-Speech Model: Does Modeling Long Context Help Unsupervised POS-tagging? arXiv preprint arXiv: 2206.14959.

[17] Ying, X., Zhongmin, W., Dehuan, J., Xiaolong, W., Qingcai, C., Hua X., Jun, Y. & Buzhou, T. (2019). A fine-grained Chinese word segmentation and part-of-speech tagging corpus for clinical text. In BMC Medical Informatics and Decision Making 2019, 19(Suppl 2): 66. (pp. 179-184), https://doi.org/10.1186/s12911-019-0770-7

[18] Yoon, K., Jernite, Y., Sontag, D., & Alexander M. R. (2015). Character-aware neural language models. In Proceedings of the 30th AAAI Conference on Artificial Intelligence, pages 2741–2749.