Natural Language Engineering (2023), **29**, pp. 1341–1363
doi:10.1017/S1351324922000298

CAMBRIDGE UNIVERSITY PRESS

ARTICLE

Neural automated writing evaluation for Korean L2 writing

KyungTae Lim[1,†], Jayoung Song[2†] and Jungyeul Park[3,4,*]

[1]Hanbat National University, Daejeon 34158, South Korea, [2]Pennsylvania State University, State College, PA 16801, USA, [3]The University of British Columbia, Vancouver, BC V6T 1Z4, Canada, and [4]University of Washington, Seattle, WA 98195, USA
*Corresponding author. E-mail: jungyeul@mail.ubc.ca

(Received 30 January 2021; revised 3 June 2022; accepted 6 June 2022; first published online 7 July 2022)

Abstract

Although Korean language education is experiencing rapid growth in recent years and several studies have investigated automated writing evaluation (AWE) systems, AWE for Korean L2 writing still remains unexplored. Therefore, this study aims to develop and validate a state-of-the-art neural model AWE system which can be widely used for Korean language teaching and learning. Based on a Korean learner corpus, the proposed AWE is developed using natural language processing techniques such as part-of-speech tagging, syntactic parsing, and statistical language modeling to engineer linguistic features and a pre-trained neural language model. This study attempted to determine how neural network models use different linguistic features to improve AWE performance. Experimental results of the proposed AWE tool showed that the neural AWE system achieves high reliability for unseen test data from the corpus, which implies metrics used in the AWE system can help differentiate different proficiency levels and predict holistic scores. Furthermore, the results confirmed that the proposed linguistic features–syntactic complexity, quantitative complexity, and fluency–offer benefits that complement neural automated writing evaluation.

Keywords: Automated writing evaluation; L2 writing; Korean; Learner corpus

1. Introduction

Technology is applied to all aspects of foreign language learning and teaching including assessments. Among these technologies, there has been an increase in the use of automated writing evaluation (AWE) for writing assessment. Natural language processing (NLP) and machine learning are employed in AWE systems to provide language learners with automated corrective feedback (Li, Dursun, and Hegelheimer 2017) and more accurate and objective scoring, which can otherwise be biased when performed by test raters. Because automated scoring is faster and more cost-effective compared to human scoring, it is used to help language teachers easily assess endless essays. Owing to these benefits, many scholars developed and implemented AWE systems for various languages including English (Shermis and Burstein 2003), Japanese,[a] Bahasa Malay, Chinese, Hebrew, Spanish, and Turkish.[b]

[†]KyungTae Lim and Jayoung Song contributed equally.
[a]Presented at the *Tokyo Chapter of the Japan Association for Language Teaching* in 2003, http://www.eltcalendar.com/events/details/1179.
[b]Bahasa Malay, Chinese, Hebrew, Spanish, and Turkish are available by IntelliMetric®, http://www.intellimetric.com/.

Despite the large number of pre-existing AWE systems, AWE for Korean L2 writing remains unexplored. Based on the Modern Language Association (MLA) report, Korean is the only language that demonstrated a sharp increase in enrollment over the past few years compared to other foreign languages. Furthermore, Korean has been consistently ranked as the 15th most commonly taught foreign languages in US colleges and universities between 2013 and 2016. Therefore, it is necessary to develop AWE for Korean to provide innovative resources in the growing field of Korean language education.

In the most basic terms, AWE is defined as "the process of evaluating and scoring written prose via computer programs" (Shermis and Burstein 2003). With the advent of automatic scoring in the 1960s (Page 1966), advanced language processing technologies and statistical methods led to the development of various AWE systems (Li *et al*. 2017). The first computerized scoring system called *project essay grader*™ (PEG™) could detect syntactic errors and predict scores that were comparable to those of human raters (Page and Petersen 1995).

More advanced AWE systems were developed in the 1990s; the *intelligent essay assessor*™ (IEA) utilized latent semantic analysis to move beyond the capability of scoring and include feedback on semantics (Foltz, Laham, and Landauer 1999). Recently, several scoring engines with more sophisticated language processing techniques and statistical methods have been developed (Li *et al*. 2014). *E-rater*®, *Knowledge analysis technologies*™, and *IntelliMetric*™ analyze a wide range of text features at lexical, semantic, syntactic, and discourse levels.

E-rater (developed by ETS) is an early AWE scoring engine designed to evaluate essays written by nonnative English learners; it is still widely used for TOEFL and GMAT, which are high-stakes tests for undergraduate admission or graduate business admission in the United States (Burstein, Tetreault, and Madnani 2013). E-rater identifies and extracts several feature classes for model building and scoring using statistical and rule-based NLP (Attali and Burstein 2006). Some of the feature classes include (1) grammatical errors (e.g., subject–verb agreement errors); (2) word usage errors (e.g., *here* versus *hear*); (3) errors in mechanics (e.g., spelling and punctuation); (4) presence of discourse elements (e.g., thesis statement, supporting details, and concluding paragraphs); (5) development of discourse elements; (6) style (e.g., repeated use of the same word); (7) content-vector analysis (CVA)-based features to evaluate topical word usage; (8) features associated with the correct usage of prepositions and collocations (e.g., *powerful* versus *strong*); and (9) a variety of sentence structure formation (Burstein *et al*. 2013). After measuring these features, the e-rater provides a holistic score that corresponds with human-rated scores. A randomly selected sample of human-scored essays is run through the e-rater, after which a variety of linguistic features are extracted and converted to numerical values. Using a regression modeling approach, the values obtained from this sample are used to determine the weight for each feature. To score a new essay, the e-rater extracts the set of features and converts the features to a vector value, and then, these values are multiplied by the weights relevant to each feature. Finally, the sum of the weighted feature is computed to predict the final score, which represents the overall quality of an essay (Attali, Bridgeman, and Trapani 2010).

Another important scoring engine is IntelliMetric, which uses the same holistic scoring approach employed by human raters (Schultz 2013). Similar to the training requirements for human raters to score a specific prompt, the IntelliMetric system needs to be trained with a set of previously scored responses from human raters. The system then internalizes the features of the responses linked to each score point and applies it to score essays with unknown scores. The IntelliMetric system uses a multistage process to score essays. First, the essays need to be provided in an electronic form. After the information is received and prepared for analysis, the text is then parsed to understand the grammatical and syntactic structure of the language. Each sentence is identified in terms of parts of speech, vocabulary, sentence structure, and expression. After all the information is collected from the text, statistical techniques are employed to translate the text into a numerical form. Then, IntelliMetric uses virtual raters (mathematical models) to assign scores. Each virtual rater attempts to link the features extracted from the text to the scores assigned in

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

the training set to ensure accurate scoring for essays with unknown scores. IntelliMetric finally integrates the information received from the virtual rates to present a single and reliable score.

Powered by these above-mentioned scoring engines, AWE tools such as Criterion and MYAccess! have been developed. These AWE tools can provide writing scores and feedback instantly, and students can benefit from these tools by practicing writing and receiving immediate feedback from the tools. In the context of writing instructions, AWE tools can assist instructors by providing immediate scoring and feedback, especially in large classroom scenarios.

In general, AWE studies have focused on the validity and reliability of AWE tools (Dikli and Bleyle 2014). Previous validation studies reported high agreement rates between the AWE tools and human raters (Burstein *et al*. 1998; Landauer, Laham, and Foltz 2003; Chodorow, Gamon, and Tetreault 2010). For example, Shermis *et al*. (2002) showed that PEG^{TM} achieved scores that were highly correlated with human scores ($r = 0.82$) compared with human inter-rater reliability ($r = 0.71$). Furthermore, Enright and Quinlan (2010) found high agreement indices between ratings provided by two human raters and those provided by e-rater and one human in TOEFL iBT. E-rater proved to be a reliable complement to human ratings under specific testing contexts (Burstein *et al*. 1998; Powers *et al*. 2000; Burstein 2003; Chodorow and Burstein 2004; Attali 2007; Lee, Gentile, and Kantor 2008).

Neural models have dominated current AWE systems. Ke and Ng (2019), Ramesh and Sanampudi (2021), and Uto (2021) have summarized recent neural models well. For automatic essay scoring, there are two main model types. Firstly, in RNN-based models, the RNN output is sent to mean-over-time to aggregate the input to the fixed length vector and a linear layer for the scalar value (Taghipour and Ng 2016) or a simple BiLSTM to the linear layer is used for predicting essay scores (Alikaniotis, Yannakoudakis, and Rei 2016). Secondly, transformer-based models, for example, BERT with BiLSTM with attention (Nadeem *et al*. 2019) or BERT concatenated with handcrafted features (Uto, Xie, and Ueno 2020), can be used to predict the score. Fine-tuning BERT using multiple losses including regression loss and reranking loss for constraining automated essay scores has been shown to produce state-of-the-art results (Yang *et al*. 2020).

Although there are many studies that explore AWE tools and their validation, a majority of the studies focus on AWE systems developed for native English-speaking writers (Powers *et al*. 2001; Rudner, Garcia, and Welch 2006; Wang and Brown 2007) or English as a second language (ESL) writers (Chen and Cheng 2008; Choi and Lee 2010). Only a few studies investigate the use of the AWE system for less commonly taught languages, and to the best of our knowledge, there are no studies that investigate AWE for Korean as a foreign language (KFL) because of the lack of available AWE tools. This study aims to extend the scope of research in this area by introducing a state-of-the-art AWE system that is developed based on the Korean learner corpus for Koreans.

The goal of this study is to develop a neural Korean AWE engine and validate it in terms of its capacity to distinguish the developmental level of second language learners. In this paper, we address the question of how recent advancements in neural network models can help improve automatic writing evaluation, and how neural network models can use different linguistic features to improve AWE performance using linguistic features for AWE in a complementary manner. This paper includes a description of the automated essay scoring system, its natural language processing-centered approach within the neural system, and details on the validation of the AWE system in terms of predicting the proficiency level and holistic score simultaneously of the learners.

The rest of this paper is organized as follows. First, the paper presents the Korean learner corpus used to develop the Korean AWE program and discusses how we define features in the learner corpus (Section 2). Next, the basic AWE model is presented (Section 3), followed by a proposed neural AWE model that was designed to compensate for the limitations of the basic model (Section 4). Finally, the results from an experiment are reported with detailed discussions (Section 5) and future perspectives for the AWE model in the conclusion (Section 6).

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

```
A100007_v02.xml
<level>초급1</level>
<nationality>중국</nationality>
<gender>여자</gender>
<term>기말고사</term>
<date>2013년 가을</date>
<topic>주말 이야기</topic>
<score>70</score>
<p><s>저는 일요일에 도서관에서 갔습니다.</s>
<s>저는 친구하고 도서관에서 갔습니다.</s>
<s>도서관에 책을 있었습니다.</s>
<s>도서관에서 공부했습니다.</s>
<s>저는 책을 읽었습니다.</s>
<s>그래서 일요일에 재미있습니다.</s></p>
```

Figure 1. Example of the Korean learner corpus: `<level>` = Level 1, `<nationality>` = Chinese, `<gender>` = female, `<term>` = final examination, `<date>` = Fall 2013, `<topic>` = my weekend, and `<score>` = 70. The present example of Korean writing can roughly be translated into *I went to the library on Sunday. I went to the library with a friend. There was a book in the library. I studied in the library. I read a book. So it's fun on Sunday.*

2. Korean learner corpus

2.1 Learner corpus dataset

We use the dataset from the Korean learner corpus (Park and Lee 2016); this database contains proficiency levels (from Level 1 to Level 6) (`<level>`), native language by nationality (`<nationality>`), gender (`<gender>`), teacher-attributed score (`<score>`), and text. Figure 1 shows an example of the Korean learner corpus dataset which indicates the learner's proficiency level = Level 1 (A1), L1 = Chinese, gender = F, and score = 70. Furthermore, it shows the title of the text (`<topic>`) and the entire text where the sentence is delimited using `s` (the beginning of a sentence) and `/s` (the end of a sentence), and the paragraph using `p` (the beginning of a paragraph) and `/p` (the end of a paragraph).

The Common European Framework of Reference for Languages (CEFR) suggest common reference levels divided into three level groups: A1 and A2 (basic), B1 and B2 (independent), and C1 and C2 (proficient) users. The Korean proficiency test divides students into beginner, intermediate, and advanced groups, which are further divided into levels based on each student's ability. These groups are subdivided into Levels 1 (A1) and 2 (A2) for the beginner levels (초급 *chogeub*, literally "beginner"), Levels 3 (B1) and 4 (B2) for the intermediate levels (중급 *junggeub*, "intermediate"), and Levels 5 (C1) and 6 (C2) for the advanced levels (고급 *gogeub*, "advanced"). The minimum requirement in universities for foreign students whose first language is not Korean should be at least Levels 3 and 4 respectively admission and completing their university degree regardless of their major. For students in Korean studies, Levels 5 and 6 are required for admission and degree completion, respectively.

Although they hailed from over 80 different countries, the majority of the learners were from Asian countries where Chinese and Japanese are the first and second most spoken languages. Writing examples for L1 Mandarin Chinese and Japanese in the corpus represent 38.27% and 21.09%, respectively. If we place students from China, Hong Kong, and Taiwan together, the percentage of learners who speak Chinese as L1 increases to 49.72%, and thus, half of the writing tests can be said to be produced by Chinese L1 learners.

A total of 2523 learners participated in a writing examination to produce 4094 writing examples. All examinees provided their native language (L1) and gender; there were 700 men, 1822 women, and a participant who did not specify their gender. The corpus also specified that all students were high school graduates, and over 60% were university graduates. In the learner corpus, the beginner levels (Levels 1 and 2) represent almost 50% of the corpus. Writing examples represent about 75% of the corpus if Level 3 (intermediate level) is also considered. Table 1

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

Table 1. Examples of most frequent prompts and their number of instances in the learner corpus

Prompts	Level 1	Level 2	Level 3	Level 4	Level 5	Level 6	(total)
My weekend	261	–	–	–	–	–	261
Seasons and weather in my country	–	171	–	–	–	–	171
The day that I remember the most	–	–	129	–	94	–	223
My future plans	24	96	–	–	–	–	120
My hobby	–	50	144	–	–	–	194
Other prompts	396	828	963	260	389	289	3125
(total)	681	1145	1236	260	483	289	4094

There are over 100 prompts which are used by only a small number of writing examples ("Other prompts" row in Table 1). Most prompts are only for specific proficiency levels, such as *My weekend*, *Seasons and weather in my country*.

presents the most frequently used prompts in the learner corpus. While some writing prompts are given only to learners at a specific proficiency level (e.g., *My weekend* requested only for Level 1), other topics can be used for different proficiency levels (e.g., *The day that I remember the most* for Level 3 and Level 5).

There are over 100 writing prompts. Twenty-one writing prompts are given to multiple proficiency levels, and these prompts represent 42.96% of the dataset. For the proposed AWE system, we use `<level>` and `<score>` as target classes, and extract various linguistic features only from sentences. Although other annotations in the learner corpus would be target classes for other learner corpus-related applications, such as `<nationality>` for native language identification, we do not use them in this study.

2.2 Features in the learner corpus

We explore various automatic metrics that aim to describe the characteristics of the learner corpus, and we find relevant features for the classification tasks. Such characteristics are represented in terms of complexity, fluency, and accuracy features. These features can be used for learner corpus-related applications such as automated assessment and language proficiency classification. All metrics described here should be measured and extracted automatically from the corpus. Therefore, they are evaluated without any human intervention to assess writing quality and classify language proficiency automatically.

2.2.1 Complexity features

Complexity features use quantitative measures such as the number of words and sentences in the text with their numbers and mean lengths. The length of the written text is considered as an important feature in the learner corpus. Most previous work on proficiency classification focused on the number of words (Ortega 2003; Vajjala and Loo 2013; Alfter *et al*. 2016). Since many official writing tests for proficiency levels define the number of words for each level, the quantitative measures of text in the learner corpus become the most obvious feature for learner corpus applications.

We use a part-of-speech (POS) tagging system for Korean morphological analysis to count the number of morphemes instead of eojeols (a blank-separated word unit in Korean). The POS tagger can attribute POS tag information while performing the segmentation task for the word in Korean. For example, the following sentence in (1b) is morphologically analyzed and segmented in (1c). Although the number of tokens differs based on basic units such as eojeols and morpheme, we can deal with compound words in which these units may appear with or without

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

hajiman	('however')	*hajiman*/MAJ
bili	('Billy')	*bili*/NNP
ssi-hago	('Mr.-CONJ')	*ssi*/NNB+*hago*/JKB
naoko	('Naoko')	*naoko*/NNP
ssi-neun	('Ms.-TOP')	*ssi*/NNB+*neun*/JX
modu	('all')	*modu*/MAG
sajingi-ga	('camera-NOM')	*sajingi*/NNG+*ga*/JKS
eobs-eoss-eoyo.	('do_not_have-PAST-DECL')	*eobs*/VA+*eoss*/EP+*eoyo*/EF+./SF

Figure 2. Example of Sejong corpus-style POS tagging analysis: MA{J|G} are for adverbs, NN{P|B|G} for nouns, J{KB|X|KS} for postpositions, E{P|F} for verbal endings, VA for adjectives, and SF for punctuations.

a blank space, in which case we can tokenize Korean words into morphemes to obtain a consistent number of tokens for compound words regardless of the blanks. For example, for two identical but differently segmented compound nouns *hakseubja kopeoseu* and *hakseubjakopeoseu* ("a learner corpus")—both of which are correct and grammatical—the number of morphemes can be homogeneously counted as two using the proposed counting scheme. This scheme performs counting based on what the compound word or phrase semantically represents instead of its surface segmentation, which can be different. Therefore, this scheme counts both as two tokens (as for *hakseubja kopeoseu*) instead of one token (as for *hakseubjakopeoseu*).

(1) a. 하지만 빌리 씨하고 나오코 씨는 모두 사진기가 없었어요.
hajiman bili ssi-hago naoko ssi-neun modu sajingi-ga eobs-eoss-eoyo.
However, Billy Mr.-CONJ Naoko Ms.-TOP all camera-NOM do_not_have-PAST-DECL.
"However, Mr. Billy and Ms. Naoko, both of them do not have a camera."

b. *hajiman bili ssi-hago naoko ssi-neun modu sajingi-ga eobs-eoss-eoyo.* (# of tokens by word = 8)

c. *hajiman bili ssi -hago naoko ssi -neun modu sajingi -ga eobs -eoss -eoyo.* (# of tokens by a morpheme = 13, punctuations excluded)

A type/token ratio is calculated using $\frac{\#\text{ of types}}{\#\text{ of tokens}}$, where the number of types represents the unique number of tokens, and the number of tokens represents the number of morphemes. This ratio can help measure the vocabulary richness of a corpus between 0 and 1. Within this range, 0 and 1 indicate low and high lexical variation, respectively. We use the morphological analysis and POS tagging model described in Park and Tyers (2019), which can generate POS tagging results, as shown in Figure 2.

Complexity features can also measure syntactic complexity in L2 writing (Polio 1997; Ortega 2003; Lu 2010), whereas first language syntactic complexity measures include Yngve's depth algorithm (Yngve 1960), Frazier's local non-terminal numbers (Frazier 1985), and the D-level scale (Rosenberg and Abbeduto 1987; Covington *et al.* 2006), we do not consider them in this manuscript for second language learning. A tree structure obtained by constituent parsing can show linguistic discrepancy. For example, if the subject is omitted in the sentence, a tree structure of the parsing result has a VP node as a root. A standard tree has an S node as a root as shown in Figure 3. If the root node is a VP, we may consider it as a syntactic complexity feature. We note that a VP root sentence also may be a grammatically relevant sentence in Korean. We use the phrase-structure models described in Kim and Park (2022), which trained the Sejong treebank for Korean using the Berkeley neural parser (Kitaev, Cao, and Klein 2019) with the pre-training of deep bidirectional transformers (Devlin *et al.* 2019). For syntactic complexity features, we add the distribution of grammatical morphemes such as the number of verbal endings and prepositions.

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

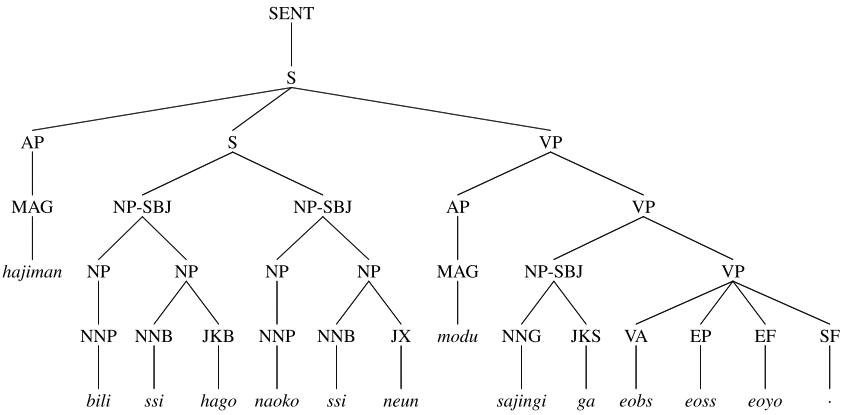

Figure 3. Example of phrase-structure analysis.

2.2.2 Fluency features

We define fluency as the capability of producing language effortlessly. Fluency is the potential of a language learner to apply their knowledge of grammar to produce intelligible speech and writing. This plays an important role in language production. We differentiate between the language fluency of a learner by observing their level of comfort when using that language and identifying if they can efficiently express themselves verbally and in text. Pauses in production and the length of written text are good indicators of fluency (Towell, Hawkins, and Bazergui 1996; Ge, Wei, and Zhou 2018; Martindale and Carpuat 2018; Qiu and Park 2019). Previous work defined various metrics for fluency. Two metrics defined in previous work and an additional fluency metric by the unigram language model are given below.

1. Fluency by Asano, Mizumoto, and Inui (2017): $f(h) = \dfrac{\log P_m(h) - \log P_u(h)}{|h|}$
2. Fluency by Ge *et al.* (2018): $f(h) = \dfrac{1}{1 + H(x)}$ where $H(x) = -\dfrac{\log P_m(h)}{|h|}$
3. Fluency by the unigram language model: $f(h) = -\dfrac{\log P_u(h)}{|h|}$

here P_m represents the probability of the sentences given by the language model, and P_u denotes the unigram probability of the sentences.

We collect a very large monolingual dataset for Korean, which contains over 9.6 M sentences and 130.6 M eojeols, to create a language model: Korean Wikipedia[c] (5.3 M sentences and 71.8 M eojeols), the Sejong morphologically analyzed corpus (3.0 M and 40.0 M), and articles from *The Hankyoreh* daily newspaper during 2016 (1.2 M and 18.6 M, previously presented in Park 2017). After preprocessing the raw text into morpheme-segmented text using the POS tagging system (Park and Tyers 2019), we create a linearly interpolated trigram model and implement the fluency metrics described in Asano *et al.* (2017) and Ge *et al.* (2018), and the fluency feature counted by the unigram language model. As indicated in (2), we attach the POS label to the morpheme-segmented lexicon and explicitly include a + symbol for consecutive morphemes. A raw text collection for creating a language model is available at http://doi.org/10.5281/zenodo.4317288 by authors of the manuscript.

[c]https://dumps.wikimedia.org/kowiki. We used a version of 20201101.

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

```
S gohyang yeseo jumal e chingu wa manna seo eseo eoyo.
A 1 2|||R:ADP|||eseo|||REQUIRED|||-NONE-|||0
A 7 8|||R:ADP|||si|||REQUIRED|||-NONE-|||0
```

Figure 4. Example of an M2 file for the Korean learner corpus.

(2) <bos> 하지만/MAJ 빌리/NNP 씨/NNB +하고/JKB 나오코/NNP 씨/NNB +는/JX
hajiman *bili* *ssi* *+hago* *naoko* *ssi* *+neun*
However Billy Mr. +CONJ Naoko Ms. +TOP
모두/MAG 사진기/NNG +가/JKS 없/VA +었/EP +어요/EF +./SF <eos>
modu *sajingi* *+ga* *eobs* *+eoss* *+eoyo*
1. all cameras +NOM do_not_have +PAST +DECL

2.2.3 Accuracy features

Thus far, we discussed features that can be extracted automatically from the learner corpus. Now, we define accuracy as a feature in the learner corpus. This feature represents the ability to produce correct sentences using correct grammar and vocabulary. However, such a learner corpus requires linguistic information such as grammatical error categories and error correction (e.g., the NUS learners corpus Dahlmeier, Ng, and Wu 2013 or the treebank of learner English Berzak *et al*. 2016). These errors are annotated based on target expressions that a native speaker would produce given the identical context, and they are used to distinguish non-standardized linguistic expressions in the learner corpus. Figure 4 shows a conceptual example of the annotated sentence described in (3) from the Korean learner. `S` represents the learner's sentence, and `A` represents the error correction annotation. `1 2` indicates the path of the tokens where the correction needs to be introduced. The value `R:ADP` indicates the type of error. For example, `yeseo`, a functional morpheme (`ADP`) at `1 2`, should be replaced by `eseo` according to the annotation.

(3) a. *고향에서 주말에 친구와 만나섰어요.
*gohyang **yeseo** jumal e chingu wa manna **seo** eseo eoyo.*
"ø (met) a friend (in the hometown) on weekend."
b. *gohyang **eseo** jumal e chingu wa manna **si** eoss eoyo.*
hometown LOC weekend AJT friend CJT meet HON PAST IND.
"ø met a friend in the hometown on weekend."

The correct sentence is presented in (3b). This example illustrates functional morpheme errors, which are among the most common errors: specifically, these errors involve postposition and honorific morphemes, which we denote as adpositions (ADP) for functional morphemes using a universal part-of-speech tagset (Petrov, Das, and McDonald 2012). Using the error-annotated learner corpus, it is possible to perform a grammatical error correction (GEC) process by automatically detecting and correcting grammatical errors in the text. In recent years, the consistent increase in the number of foreign language learners, especially learners of Korean, and the demand to facilitate their learning with timely feedback have resulted in GEC becoming increasingly popular and attracting considerable attention in both academia and industry. However, because the learner corpus needs to be in another form, that is, an error-annotated corpus instead of the current version of the corpus because of the lack of the error correction dataset in the learner corpus for Korean L2 writing, a task such as GEC including accuracy features is beyond the scope of this study, and we leave it as future work.

2.2.4 Summary

We summarize the list of features, including the bag of morphemes, in Table 2, which also shows examples of feature values for the learner corpus presented in Figure 1, which contains six sentences. We present several quantitative complexity features, such as the mean length

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

Table 2. Example of features and teir values for te learner's writing in Figure 1

	Features	Values
	bag of morp	저/PRON +는/JX 일요일/NNG +에/JKB 도서관/NNG +에서/JKB 가/VV +았/EP +습니다/EF 저/PRON +는/JX 친구/NNG +하고/JC 도서관/NNG +에서/JKB 가/VV +았/EP +습니다/EF 도서관/NNG +에/JKB 책/NNG +을/JKO 있/VV +었/EP +습니다/EF 도서관/NNG +에서/JKB 공부/NNG +하/XSV +았/EP +습니다/EF 저/PRON +는/JX 책/NNG +을/JKO 읽/VV +었/EP +습니다/EF 그래서/MAJ 일요일/NNG +에/JKB 재미있/VA +습니다/EF
complexity	# of sent	6
	# of para	1
	# of tok	43
	sent by morp	7.166666666666667
	wd by morp	2.263157894736842
	type/token ratio	0.46511627906976744
	bag of funct	+는/JX +에/JKB +에서/JKB +는/JX +하고/JC +에서/JKB +에/JKB +을/JKO +에서/JKB +는/JX +을/JKO +에/JKB
	# of vp eads	0
fluency	Asano *et al*. (2017)	0.16437636932893357
	Ge *et al*. (2018)	0.1491520664089971
	unigram LM	5.868943218961787
accuracy		not available
target	score	70
	proficiency level	Level 1

Bag of morph = bag of morphemes; # of sent = number of sentences; # of para = number of paragraphs; # of tok = number of tokens; sent by morph = mean number of morphemes per sentence; wd by morph = mean number of morphemes per word; type/token ratio = ratio of morpheme types to tokens; bag of funct = bag of functional morphemes; # of vp heads = number of vp heads. The fluency assessment by Asano *et al*. (2017) uses $f(h) = \frac{\log P_m(h) - \log P_u(h)}{|h|}$, the fluency assessment by Ge *et al*. (2018) uses $f(h) = \frac{1}{1+H(x)}$ where $H(x) = -\frac{\log P_m(h)}{|h|}$, and the fluency by the unigram language model uses $f(h) = -\frac{\log P_u(h)}{|h|}$.

of sentence by morpheme, mean length of word by morpheme, and morpheme type versus token ratio. In addition, the table shows statistical complexity features such as the number of sentences, number of paragraphs, and number of tokens using morphemes. We consider the bag of functional morphemes as a morpho-syntactic complexity feature and the number of vp heads as a syntactic complexity feature. We denote both the morpho-syntactic and syntactic complexity features as syntactic complexity features for convenience, so that they are differentiated from quantitative complexity features.

3. Baseline statistical automated writing evaluation models

First, we propose the use of a statistical automated writing evaluation system as a baseline system. Statistical automated writing evaluation systems use linear and logistic regression models.

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

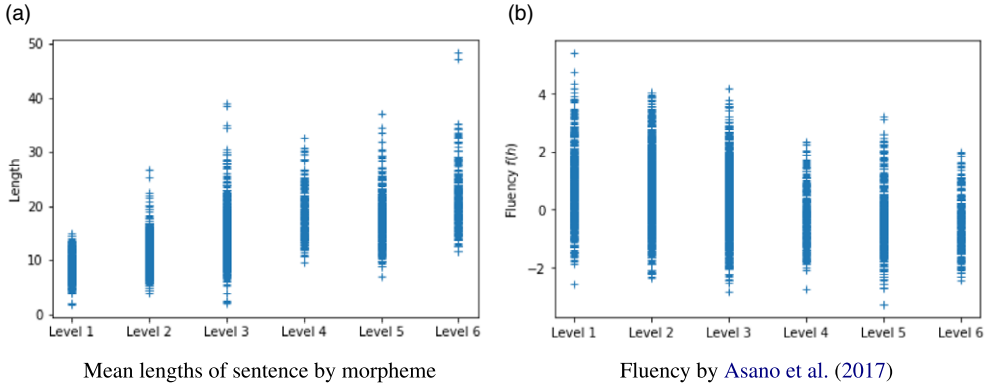

Figure 5. Distribution of sample features per level.

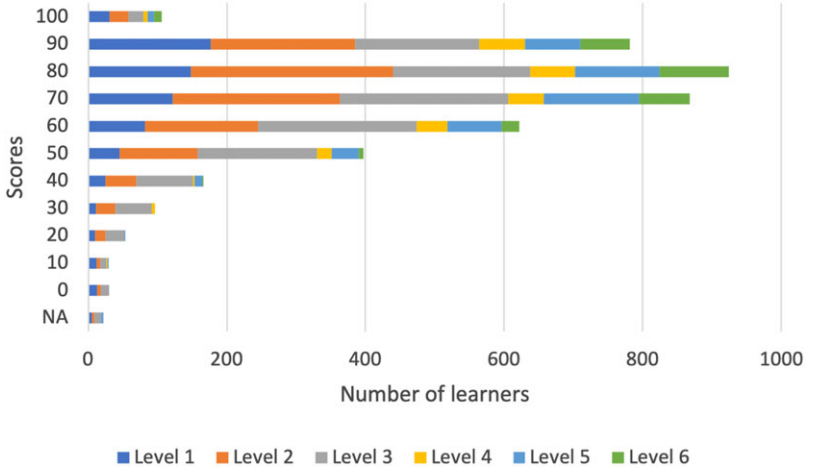

Figure 6. Distribution of learner's scores between levels.

Thus, we separately implemented two independent systems to predict proficiency levels and scores instead of using a single integrated system. Figure 5 shows a distribution of criterial features for each level, including quantitative measures (mean lengths of sentence by morpheme and fluency by Asano *et al.* 2017). This figure corresponds to the logistic regression model for classifying proficiency levels. Figure 6 presents a distribution of learners' scores between levels. Note that there are 22 writing examples for which scores are either not provided or annotated as not specified (`NA`).

We evaluated scores using 5-fold cross-validation with accuracy for proficiency classification as in (1) and mean squared error regression loss to assess writing quality as in (2).

$$\text{ACC} = \frac{\text{correctly classified number of examples}}{\text{total number of examples}} \tag{1}$$

$$\text{MSE}(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2}$$

where n denotes the total number of examples. Table 3 shows the results of the baseline statistical models. Our experiments using the baseline statistical models followed the experimental settings suggested in previous work by either predicting a score or classifying the proficiency of a learner, independently. We obtained rudimentary initial results using the basic statistical models, which

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

Table 3. Statistical AWE system results

Task	Result
proficiency classification (ACC)	53.64 (± 1.41)
assessing writing quality (MSE)	15.30 (± 4.17)

Average results of 5-fold cross-validation with the standard deviation.

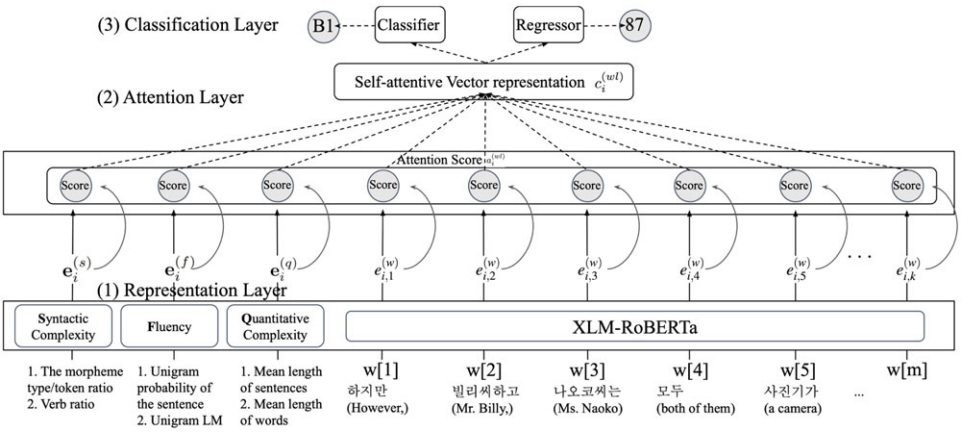

Figure 7. System structure of the proposed deep learning model. Three linguistic features are applied: syntactic complexity, fluency, and quantitative complexity, in addition to the sequence of token representations. Each token is transformed into a vector representation based on XLM-RoBERTa.

have the following limitations. First, it is difficult to determine the effect of each feature. Second, it has low performance compared to current deep learning-based systems. Third, it requires two separate systems that predict the score and classify the proficiency level; this makes it difficult to use these models for general purposes as a complete AWE system. Therefore, we propose a state-of-the-art neural model for automated writing evaluation which is able to assess proficiency levels and scores simultaneously, and we aim to introduce a system that can be widely used throughout Korean language teaching classrooms.

4. Neural automated writing evaluation models

We propose a state-of-the-art neural Korean AWE model and provide a deeper investigation into each feature proposed in Section 2.2. Our system applies XLM-Roberta to represent word forms as word representations along with the multitask learning (MTL) approach that trains several tasks simultaneously (Hashimoto *et al.* 2017; Lim *et al.* 2020). The details of the XLM-Roberta feature representation method and our MTL approach is depicted in Figure 7.

4.1 Representation of words

Machine learning (ML)-based grammar checking (Soni and Thakur 2018) and AWE (Persing, Davis, and Ng 2010; Taghipour and Ng 2016; Yang, Xia, and Zhao 2019) have been proposed and widely used in recent years because of their outstanding performance. The main idea behind ML-based AWE is applying deep learning techniques for automated essay scoring. To compute the score of writing in terms of machine learning, the system has to learn from a training dataset

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

T that comprises a pair of essays x_i and scores y_i, where $(x_i, y_i) \in T$. In the deep learning-based AWE such as in Yang *et al.* (2019), the sequence of words from the essay x_i is represented as a sequence of vector representations (i.e., word embeddings). Therefore, the essay x_i is composed of m words such that $x = (w_{i,1}, \cdots, w_{i,m})$, and the system creates a set of sequences of word embeddings $e^w_{i,1}, \cdots, e^w_{i,m}$. This vector representation of a word $e^w_{i,j}$ is trained to capture syntactic and semantic meanings of a word in a sentence (Pennington, Socher, and Manning 2014). We apply a bidirectional encoder representations from transformers (BERT)-like word representation method that is trained using a masked language model (MLM). Many MLM pre-learning methods such as BERT (Devlin *et al.* 2019) perform training by replacing certain input words with `[MASK]` and restoring them to the original token by training a deep neural network. For example, let the input text be *I have no clue*; then, the system selects tokens randomly and replaces them as *I have [MASK] clue*. This process makes the system predict the masked word based on its surrounding words. During training, the system may struggle to learn the best parameters by comparing its prediction and the masked word.

BERT is a pre-trained word representation model that is trained with large quantities of Wikipedia text as input and over 110 million parameters. RoBERTa (Liu *et al.* 2019) is an extended version of BERT, which consumes 270 million parameters and a bigger input dataset, and XLM-RoBERTa (Conneau *et al.* 2020) is a multilingual model of Roberta trained in 100 different languages. These pre-trained models are effective when transferred to a downstream NLP task because they capture a deep contextual representation of words. In this study, we apply the multilingual XLM-Roberta model to transform the Korean text into a sequence of word representations as

$$E_i^{(w)} = XLMRoberta(w_{i,1}, .., w_{i,m}) \tag{3}$$

where $E_i^{(w)}$ is a matrix that denotes a set of vector representation of words, and it comprises k subwords as $E_i^{(w)} = (e_{i,1}^{(w)}, \cdots, e_{i,k}^{(w)})$. This is because XML-Roberta tokenizes a word into several subwords to handle character-level subword information.

For example, the word *joyful* turns into two subwords, *joy* and *ful* using XLM-Roberta; therefore, the number of words m in an essay is always equal to or smaller than the number of XLM-Roberta representations k. We implement our word representation model using the pre-trained XLM-Roberta provided by Huggingface.[d]

4.2 Representation of linguistic features

Quantitative complexity, syntactic complexity, and fluency of the learner's writing are features that are traditionally important to predict essay scores to assess writing, and they can be transformed into vector representations in ML applications using a simple linear transformation method. First, we concatenate the features in Section 2.2 for each **q**uantitative complexity $t_i^{(q)}$, **s**yntactic complexity $t_i^{(s)}$, and **f**luency $t_i^{(f)}$. Then, we transform each concatenated output based on a linear model with an activation function *Relu* as

$$e_i^{(q)} = G^{(q)} Relu(U^{(q)} t_i^{(q)}) + b^{(q)} \tag{4}$$

$$e_i^{(s)} = G^{(s)} Relu(U^{(s)} t_i^{(s)}) + b^{(s)} \tag{5}$$

$$e_i^{(f)} = G^{(f)} Relu(U^{(f)} t_i^{(f)}) + b^{(f)} \tag{6}$$

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

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

where $e_i^{(q)}$, $e_i^{(s)}$, and $e_i^{(f)}$ denote a vector representation of each feature, G and U represent learnable parameters, and b indicates a bias. We concatenate the representation of features with the vector representation of words. Finally, we unify the representation between the word and the linguistic features as

$$E_i^{(wl)} = (e_i^{(q)}, e_i^{(s)}, e_i^{(f)}, e_{i,1}^{(w)}, \cdots, e_{i,k}^{(w)}) \tag{7}$$

where k denotes the number of words in the learner's writing. The proposed unified representation is commonly used with BERT-like models. For example, Prakash and Madabushi (2020) designed an enhanced version of contextual representation based on count-based features (BERT with a term frequency), and Xue *et al.* (2019) investigated the effect of relational features with BERT for the Chinese NER task. To combine pairs of features, a simple concatenation method was applied. However, the concatenation method may not be the best method in our case because our model uses diverse features simultaneously. To investigate this issue, we applied an attention-based method to form unified representations.

4.3 Self-attentive representations

The vector representations for each word (Section 4.1) and linguistic features (Section 4.2) are independent representations although they are concatenated. Thus, no co-relational information can be represented between two representations. The self-attention mechanism is adequate to address this issue. Self-attention involves applying a linear transformation (Cao and Rei 2016) over the matrix of the unified representation $E_i^{(wl)}$ for which the attention weights $a_i^{(wl)}$ are computed as

$$a_i^{(wl)} = Softmax(R^{(wl)} E_i^{(wl)}) \tag{8}$$

$$c_i^{(wl)} = a_i^{(wl)} \cdot E_i^{(wl)} \tag{9}$$

where $R^{(wl)}$ denotes a learnable parameter. The attention weight $a_i^{(wl)}$ corresponds to the most informative word w_j $(1 \leq j \leq k)$ in the learner's writing and linguistic features. The system obtains a self-attentive vector representation $c_i^{(wl)}$ through the dot-product between the attention weight and unified representation. Intuitively, the attention weight denotes a probability score that represents "how much our system focuses on a specific word or linguistic feature that we propose." Given an input, when a specific word or expression is important, the system provides more weight to build a self-attentive representation. The attention weight is discussed in Section 5.3.

4.4 Prediction of a proficiency level and a score

Our final goal is to build a system that can automatically measure the proficiency level and the score of a learner's writing. We use a linear classifier to measure the proficiency level of the essay and use another linear regressor for scoring.

$$e_i^{(c)} = P^{(c)} Relu(D^{(c)} c_i^{(wl)}) + b^{(c)} \tag{10}$$

$$\hat{y}_i^{(level)} = \underset{z}{\operatorname{argmax}}\, e_{i,z}^{(c)} \tag{11}$$

$$\hat{y}_i^{(score)} = P^{(r)} Relu(D^{(r)} c_i^{(wl)}) + b^{(r)} \tag{12}$$

z denotes an index number of levels where `level` $=$ {Level 1, . . ., Level 6}, and $P^{(c)}$ and $P^{(r)}$ are learnable parameters. The classification result $\hat{y}_i^{(c)}$ is computed by the selection of the maximum

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

Table 4. Hyperparameters

Component	Value
Train/Test split ratio	8:2
$e_{i,j}^{(w)}$ (XLM-RoBERTa) dim.	768
G, U (parameters) dim.	768
R, P, D (parameters) dim.	400
Dropout	0.3
Learning rate	0.00002
β_1, β_2	0.9, 0.99
Epoch	100
Batch size	6
Gradient clipping	5.0
Optimizer	AdamW

value of $e_{i,z}^{(c)}$. During the training phase, our system learns by backpropagation of the prediction errors over the entire training dataset T. Because we train two different classification and regression tasks, we use the individual *CrossEntropy* objective function for predicting the proficiency level and the MSE function for assigning the score of the learner's writing.

$$loss = \sum_{(x_i,y_i)\in T} CrossEntropy(\hat{y}_i^{(level)}, y_i^{(level)}) + MSE(\hat{y}_i^{(score)}, y_i^{(score)}) \tag{13}$$

where $(x_i, y_i) \in T$ denotes an element from the training set T, y_i denotes a set of gold labels (y_i^{level}, y_i^{score}), and $\hat{y}_i$ represents a set of predicted results.

5. Results of neural AWE models and discussion

5.1 Experiment setup

As presented in Section 4, we evaluate the scores using 5-fold cross-validation with the proposed regression loss to assess writing quality and the prediction accuracy for its proficiency level. Table 4 lists our hyperparameter settings. We apply 768 dimensions for parameters U and Q in (4) and set 400 dimensions for P and D in (10). We run through 80% of the training dataset during the learning phase using an epoch with a batch size of 6 randomly selected sentences. The remaining 20% is used as the test dataset. We report the best performance on the test dataset within 100 epochs over five times for the 5-fold cross-validation.

5.2 Experiment results

Table 5 summarizes our results on how we use different linguistic information to improve AWE results using XLM-RoBERTa. The linguistic features are syntactic complexity features (S), fluency features (F), quantitative features (Q), and self-attention mechanism (A). To investigate the effect of LMs on AWE performance, we compare results between multilingual BERT (M) and XLM-RoBERTa (X). Besides word representation methods, we also evaluate performance that is solely based on linguistic features without the pre-trained language model. For the models without self-attention, we applied a weighted average of the BERT word representations and linguistic features

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

Table 5. Experiment results

Model	ACC	MSE
(M)	95.83 (± 0.66)	12.11 (± 0.52)
(X)	96.16 (± 0.48)	12.46 (± 0.46)
(X) + (A)	96.14 (± 0.51)	11.98 (± 0.70)
(X) + (S)	**96.85** (± 0.91)	**11.4** (± 0.62)
(X) + (F)	96.06 (± 0.46)	12.01 (± 0.78)
(X) + (Q)	96.40 (± 0.17)	12.43 (± 0.73)
(X) + (A) + (S) + (F) + (Q)	96.71 (± 0.30)	11.96 (± 0.46)
(A) + (S) + (F) + (Q)	50.98 (± 1.27)	13.02 (± 1.02)

Accuracy for predicting a proficiency level and MSE for assigning a score for the learner's writing: (M) multilingual BERT only, (X) XLM-RoBERTa only, (X) + (A) XLM-RoBERTa and attention, (X) + (S) XLM-RoBERTa and syntactic complexity features, (X) + (F) XLM-RoBERTa and fluency features, (X) + (Q) XLM-RoBERTa and quantitative complexity features, (X) + (A) + (S) + (F) + (Q) XLM-RoBERTa and all features, and (A) + (S) + (F) + (Q) w/o pre-trained LMs.

as $c_i^{(wl)} = \frac{1}{(k+3)} (e_i^{(q)} + e_i^{(s)} + e_i^{(f)} + \sum_{j=1}^{k} e_j^{(w)})$. Note that the dimension of linguistic features is identical to that of the BERT embedding.

Overall observations. XLM-RoBERTa and syntactic complexity features outperform other experimental settings for in terms of predicting both the proficiency level and the score. The features described in Section 2.2 only narrowly impact the overall results, and linguistic features without the pre-trained language model result in a severely limited performance.

Effect of different BERT-like pre-trained language models. The model based on XLM-Roberta naturally outperforms the multilingual BERT system, wherein the former was empirically evaluated for result gains including the trade-offs between positive transfer and capacity dilution (Conneau *et al.* 2020).

Effect of linguistic features for AWE We observed a meaningful improvement in the results when using linguistic features compared to that between `only XLM-RoBERTa` and `XLM-RoBERTA` and all other features, as listed in Table 5. Among the three different linguistic features, syntactic complexity is found to be the most impactful factor in both assessing the proficiency level and the score. Furthermore, we found that quantitative complexity features have a positive effect on our empirical experiment; however, fluency features lead to performance degradation of about -0.1 points.

Effect of self-attention. In practice, there are no result gains from using self-attention: A -0.02 accuracy for predicting a proficiency level (a negative result) and -0.48 MSE for assigning a score (a positive result) were observed. This may be attributed to the multi-head self-attention, which computes several attentions simultaneously (Vaswani *et al.* 2017), being already applied in the XLM-RoBERTa model; therefore, our attention representation is relatively less effective than expected.

5.3 Analysis

In the previous section, we showed the performance of our model using different feature selection scenarios. Among the proposed features, syntactic complexity features are relatively more important than other features. However, these observations are based on empirical experiments, and thus, one cannot explain why the neural model makes such a decision. To gain a better understanding of the decision making process of the system, we conduct additional experiments to

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

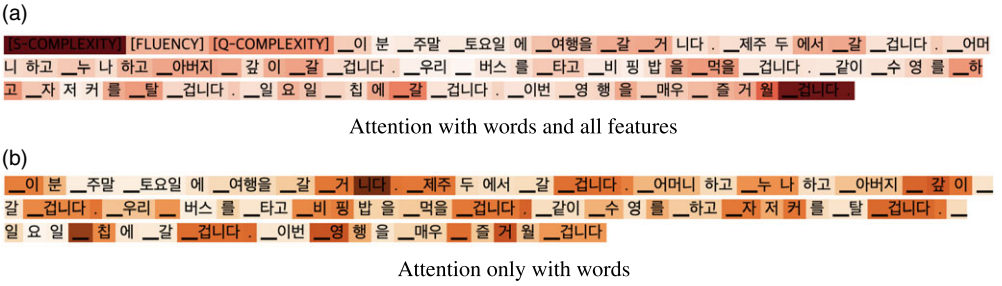

Figure 8. Visualization of the attention score proposed in Eq (8).

visualize the attention score added on the top of feature representations. The visualization of the attention score is the most powerful explainable AI (XAI) method where the results of the solution are understood by humans (Park *et al*. 2016).

During the training, the attention score of the i-th learner's writing—$a_i^{(wl)}$ in (8)—is computed as a probability distribution where $\sum_{j=1}^{k+3} a_{i,j}^{(wl)} = 1$, k denotes the number of subtokens, and three different types of linguistic features—syntactic complexity, quantitative complexity, and fluency features—are proposed. Intuitively, the attention score, therefore, represents the importance assigned by our system to each linguistic feature and the word to yield results for predicting a proficiency level and for assigning a score in a learner's writing.

Attention results on all features. Visualization results in Figure 8 show the attention score on the learner's writing number 1. In the figure, the darker the color, the more attention points of the element are assigned. Figure 8a shows the result of applying three different linguistic features as well as words. We find two interesting observations in Figure 8a compared to those in Figure 8b, where there is attention only with words. First, we observe that the system focuses on `[S-COMPLEXITY]` (syntactic complexity features). This result is in line with the result reported in Table 5, where the accuracy of our system was improved by 0.69 points when syntactic complexity features were introduced. Second, the system lacks interest in focusing on misspelled words. In this figure, there are several misspelled words such as 이분 (*ibun*) instead of 이**번** (***ibeon***, 'this time'), 제주두 (*jejudu*) instead of 제주**도** (*jeju**do***, 'Jeju Island'); 갚이(*gapi*) instead of **같**이 (***gat**i*, 'together'); 비핑밥 (*bipingbab*) instead of 비**빔**밥 (*bi**bim**bab*); and 자청거 (*jacheonggeo*) instead of 자**전**거 (*ja**jeon**geo*, 'bicycle'). Since we do not use accuracy features provided by human annotation, our system can be considered to be sound for the following reasons: (1) The attention mechanism focuses on the proposed linguistic features based on automatic metrics, and (2) a pretrained large language model can be associated with more proper words instead of spelling errors to yield classification and predicting results.

Attention results on only words. As reported, our system tends to focus on syntactic complexity features when all linguistic features are available. Then, what happens if the system can only see words? Figure 8b presents the results when we apply only words as the (X) + (A) model in Table 5. We found that the higher attention score is assigned to verbs such as 갑니다 (*gabnida*, "be going"). However, the distribution of attention scores on words varies based on the input dataset. Therefore, it is difficult to find a specific word or an expression that can directly affect the score of the learner's writing.

Attention results on only linguistic features. Table 5 shows that our system predicts a score and a proficiency level of the learner's writing only with the proposed linguistic features. We are interested in linguistic features that are the most important. Figure 9 presents attention scores of the (A) + (S) + (F) + (Q) model in Table 5 for three sample instances in the dataset. This model does not have any word information, that is, it is without the pre-trained language model. By observing the graph on `Essay Number 1` and `Essay Number 2`, the syntactic complexity is found to be the

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

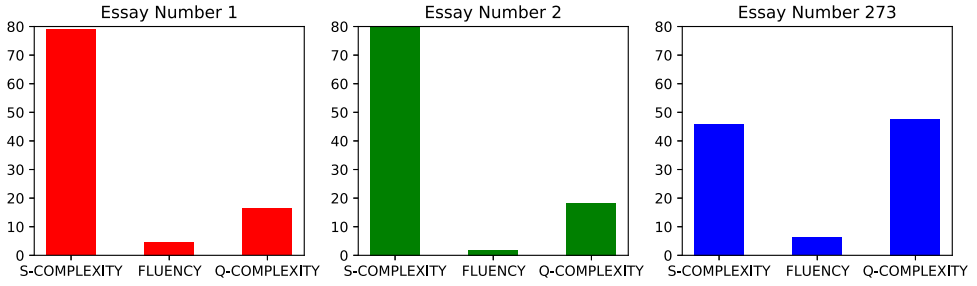

Figure 9. Visualization of the attention scores proposed in (8): `[S-COMPLEXITY]`, `[FLUENCY]`, and `[Q-COMPLEXITY]` for syntactic complexity, fluency, and qualitative complexity features, respectively.

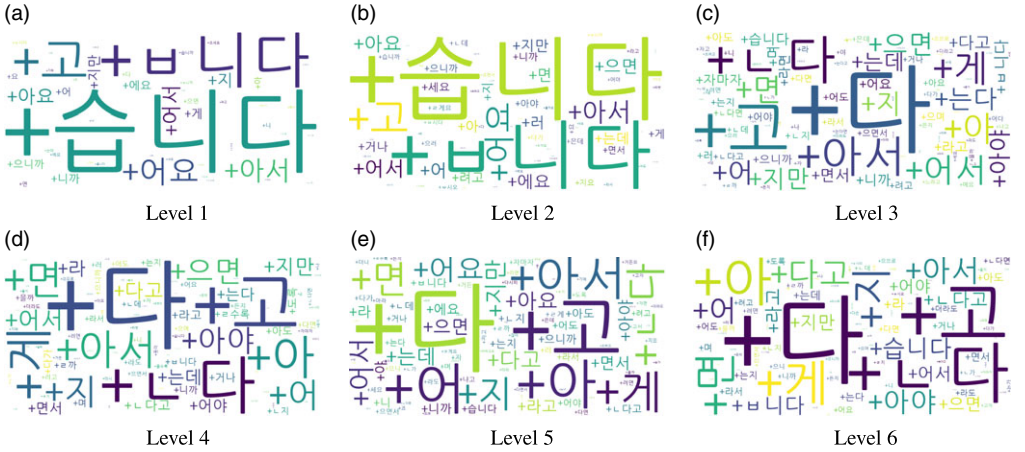

Figure 10. The usage of verbal endings based on its proficiency levels.

most significant feature. For 82.7% of essays in the test dataset, the mean attention score of syntactic complexity features is more than 0.8 out of 1. However, we also observe that the quantitative complexity is a more crucial feature for decision-making for some essays such as `Essay Number` 273. We assume that the attention mechanism attempts to capture quantitative complexity features if it fails to utilize syntactic complexity features. However, in any case, the fluency weight does not exceed 6.7% or more of its attention score. Thus, we can assume that fluency is relatively the least important property for AWE when the system have other complexity information.

The most frequent and important words based on proficiency level. In most Korean textbooks, polite verbal ending 요 (*yo*) is introduced first because it is the most commonly used ending in everyday context. Then, deferential ending 습니다 (*seubnida*) is introduced in the upper beginner level, followed by plain ending 다 (*da*) in the intermediate level. Accordingly, Figure 10 shows the distributions of verbal endings based on learners' proficiency levels.

Discussion of the usage of Korean monolingual BERT. Table 5 shows that XLM-RoBERTa outperforms the multilingual BERT. However, the proposed multilingual BERT and the XLM-RoBERTa models are designed for multilingual purposes. There are several publicly available Korean monolingual BERT models, such as KLUE-RoBERTa,[e] KoBERT,[f] DistilBERT,[g] and KoELECTRA.[h] Because these models have been trained with different amounts of training data, their parameters

[e] https://github.com/KLUE-benchmark/KLUE.
[f] https://github.com/SKTBrain/KoBERT.
[g] https://github.com/monologg/DistilKoBERT.
[h] https://github.com/monologg/KoELECTRA.

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

Table 6. Result comparison using different Korean monolingual BERTs

Model	ACC	Model size	Training data
Multilingual BERT (base)	95.83	641MB	–
XLM-RoBERTa (base)	96.16	1.2GB	–
KLUE-RoBERTa (base)	96.06	110MB	63GB
KoBERT (base)	95.94	351MB	10GB
DistilKoBERT	95.72	108MB	10GB
KoELECTRA (small-V2)	96.06	255MB	14GB
KoELECTRA (small-V3)	96.02	255MB	34GB

We evaluated the model using only the BERT model (i.e., we did not apply the proposed linguistic features). `Training Data` denotes the size of the Korean corpus used for training BERT.

also vary. We additionally investigate the performance of Korean AWE using these monolingual BERT models for following reasons. First, we are interested in whether monolingual Korean BERT models perform better than multilingual BERTs. Second, we must determine the importance of the different hyperparameters in the monolingual BERT models, as well as the optimally cost-effective BERT model size. Table 6 provides data from the ablation study on multilingual and Korean monolingual BERT models. Overall, we did not observe performance improvement by using the monolingual BERTs. Instead, we observed that the model size is more important for monolingual BERT models when comparing KoBERT and DistilKoBERT. One interesting result of the experiment is that comparing KoELECTRA small-V2 and small-V3 shows almost identical results, even with different sizes of training data. Among the monolingual models, KLUE-RoBERTa (Park *et al.* 2021) showed the best performance regardless of their model sizes.

Feature comparison with previous work. We compare our linguistic features with others previously proposed and utilized. Most previous work focused on complexity features by our criteria such as statistical features (e.g., length and n-gram)) or style-based features (e.g., part-of-speech labels, sentence structure, and other lexical patterns) (Ramesh and Sanampudi 2021). There are also content-based features (e.g., similarities between sentences and prompt overlapping), in which the similarity metric is introduced: for example, Sakaguchi, Heilman, and Madnani (2015) used BLEU, Word2vec similarity and WordNet similarity for their reference-based approach, and Dong and Zhang (2016) counted the number of words and their synonyms in the essay appearing in the prompt. Due to the availability of spell checker for English, spelling, punctuation, and capitalization errors could also be utilized as accuracy features (Persing and Ng 2013; Sakaguchi *et al.* 2015; Dong and Zhang 2016; Cummins, Zhang, and Briscoe 2016; Dong, Zhang, and Yang 2017). Table 7 shows a summary of handcrafted features in previous work. We used more detailed quantitative measures (token ratio; length of morphemes, words, and sentences for lexical diversity) and linguistic features by POS tagging and syntactic parsing. We also introduced fluency measures, which no previous work has considered. As we mentioned, in future work we are planning to include a grammar error correction system where we can obtain accuracy features beyond simple spelling errors.

6. Conclusion

In this paper, we explored several types of linguistic features in the learner corpus: quantitative complexity, syntactic complexity, and fluency. These features can be used for learner corpus-related applications that make use of machine learning techniques in addition to pre-trained language models for the neural system.

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

Table 7. Features utilized in previous work

Reference	Features	
	complexity	accuracy
PN13	keywords, POS, n-gram, FrameNet roles	spelling errors
SH15	length, n-gram, dependency relation, PropBank roles	character-based spelling error
DZ16	length, POS	spell corrected bag of words
CZ16	length, parse tree, cohesion between sentences	error rate by tri-grams
DE18	POS, dependency relation, ratio, cohesion between sentences, psychological categories	–
UE20	length, POS, n-gram, readability	spelling errors
RE21	POS	–

PN13 (Persing and Ng 2013), SH15 (Sakaguchi *et al*. 2015), DZ16 (Dong and Zhang 2016), CZ16 (Cummins *et al*. 2016), DE18 (Dasgupta *et al*. 2018), UE20 (Uto *et al*. 2020), RE21 (Ridley *et al*. 2021).

We used various metrics that were automatically measured for these features. Therefore, these metrics could be evaluated without any human intervention to assess the proficiency and holistic score of writing automatically. The proposed neural-based state-of-the-art system applied the transformer-based multilingual masked language model and XLM-RoBERTa. In addition, based on the proposed attention mechanism score, we observed how the proposed linguistic features benefit AWE in a complementary manner for neural systems, and we analyzed which sequence of words and expression can be focused on in the neural system.

Because our AWE system could provide a reliable holistic score while simultaneously detecting students' proficiency levels, it could offer potential solutions for Korean language instructors who might be struggling with the workload. Furthermore, it can be used as a resource for grading student essays in large classes or placement tests that need to be graded accurately and promptly. Furthermore, the AWE system can benefit Korean language learners in their writing practice. Learners can use the AWE system to self-grade their essays before submission and learn how their scores change as they change vocabulary, syntactic structure, etc. in their writing.

Although the proposed neural AWE engine can judge the grammaticality of the learner's writing using linguistic features and a pre-trained neural language model, the current AWE tool has several limitations. One is that it does not "read" students' essays. That is, the program can detect syntactic complexity and fluency, but does not make judgment on its content whether it is written according to the given writing topic. Similarity between the content and the topic can be estimated by defining the distance between words in the content and the concept of the topic. While previous work has proposed content-based features to calculate similarities with the prompt or reference text (Sakaguchi *et al*. 2015; Dong and Zhang 2016), we have left this for future work. Another limitation is that our approach can possibly show biased performance on limited topics that are included in the training data set. However, we observed that this issue can be mitigated by utilizing the pre-trained neural language model. Lastly, the current model does not provide specific error feedback to students. Although learners could check their scores and proficiency level with the AWE tool, they cannot check their errors, thus making it hard for them to learn from their errors.

Given that adding error types to the learner corpus has been presented for multiple grammatical (either morphological or syntactic) levels and for several languages (Ramos *et al*. 2010; Boyd 2010; Han *et al*. 2010; Seo *et al*. 2012; Dickinson and Ledbetter 2012), our next goal is to add error

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

annotations in the Korean learner corpus to broaden the usage of our AWE system. As the current NLP systems used for feature extraction are developed for the standard Korean language, it is expected that the automatic processing system may produce errors. This error-annotated learner corpus can lead to grammatical error correction (GEC) as a preprocessing step for learner corpus applications. We hope that the additional GEC task will improve learner corpus applications. It is important that the writing be relevant to the given subject, which is an aspect we cannot deal with using the proposed system. To the best of the authors' knowledge, this has not been presented in previous literature on leaner corpus applications, and we will consider this problem for future work.

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

References

Alfter D., Bizzoni Y., Agebjörn A., Volodina E. and Pilán I. (2016). From distributions to labels: A lexical proficiency analysis using learner corpora. In *Proceedings of the Joint Workshop on NLP for Computer Assisted Language Learning and NLP for Language Acquisition*, Umeå, Sweden. LiU Electronic Press, pp. 1–7.

Alikaniotis D., Yannakoudakis H. and Rei M. (2016). Automatic text scoring using neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany. Association for Computational Linguistics, pp. 715–725.

Asano H., Mizumoto T. and Inui K. (2017). Reference-based metrics can be replaced with reference-less metrics in evaluating grammatical error correction systems. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Taipei, Taiwan. Asian Federation of Natural Language Processing, pp. 343–348.

Attali Y. (2007). Construct Validity of e-rater in Scoring TOEFL Essays. Technical report, ETS, Princeton, NJ.

Attali Y., Bridgeman B. and Trapani C. (2010). Performance of a generic approach in automated essay scoring. *The Journal of Technology, Learning and Assessment* **10**(3), 1–17.

Attali Y. and Burstein J. (2006). Automated essay scoring with e-rater V.2. *The Journal of Technology, Learning and Assessment* 4(3), 1–31.

Berzak Y., Kenney J., Spadine C., Wang J.X., Lam L., Mori K.S., Garza S. and Katz B. (2016). Universal dependencies for learner English. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany. Association for Computational Linguistics, pp. 737–746.

Boyd A. (2010). EAGLE: an error-annotated corpus of beginning learner German. In *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA), pp. 19–21.

Burstein J., Tetreault J. and Madnani N. (2013). The E-rater automated essay scoring system. In Shermis M.D. and Burstein J. (eds), *Handbook of Automated Essay Evaluation : Current Applications and New Directions*, chapter 4. London, UK: Taylor & Francis Group, pp. 55–67.

Burstein J.C. (2003). The E-rater scoring engine: Automated essay scoring with natural language processing. In Shermis M.D. and Burstein J.C. (eds), *Automated Essay Scoring*, chapter 7. New York: Routledge, pp. 113–121.

Burstein J.C., Braden-Harder L., Chodorow M., Hua S., Kaplan B., Kukich K., Lu C., Nolan J., Rock D. and Wolff S. (1998). Computer Analysis of Essay Content for Automated Score Prediction: A Prototype Automated Scoring System for GMAT Analytical Writing Assessment Essays. Technical Report 1, ETS, Princeton, NJ.

Cao K. and Rei M. (2016). A joint model for word embedding and word morphology. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, Berlin, Germany. Association for Computational Linguistics, pp. 18–26.

Chen C.-F.E. and Cheng W.-Y.E. (2008). Beyond the design of automated writing evaluation: Pedagogical practices and perceived learning effectiveness in EFL writing classes. *Language Learning & Technology* **12**(2), 94–112.

Chodorow M. and Burstein J. (2004). Beyond Essay Length: Evaluating e-rater's Performance on TOEFL Essays. Technical report, ETS, Princeton, NJ.

Chodorow M., Gamon M. and Tetreault J. (2010). The utility of article and preposition error correction systems for English language learners: Feedback and assessment. *Language Testing* **27**(3), 419–436.

Choi J. and Lee Y. (2010). The use of feedback in the ESL writing class integrating Automated Essay Scoring (AES). In Gibson D. and Dodge B. (eds), *Proceedings of Society for Information Technology & Teacher Education International Conference 2010*, San Diego, CA, USA. Association for the Advancement of Computing in Education (AACE), pp. 3008–3012.

Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek, G., Guzmán, F., Grave E., Ott M., Zettlemoyer L. and Stoyanov V. (2020). Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics, pp. 8440–8451.

Covington M.A., **He C.**, **Brown C.**, **Naci L. and Brown J.** (2006). How complex is that sentence? A proposed revision of the Rosenberg and Abbeduto D-Level Scale. Technical report, University of Georgia, Athens, Georgia.

Cummins R., **Zhang M. and Briscoe T.** (2016). Constrained multi-task learning for automated essay scoring. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany. Association for Computational Linguistics, pp. 789–799.

Dahlmeier D., **Ng H.T. and Wu S.M.** (2013). Building a large annotated corpus of learner English: The NUS corpus of learner English. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, Atlanta, Georgia. Association for Computational Linguistics, pp. 22–31.

Dasgupta T., **Naskar A.**, **Dey L. and Saha R.** (2018). Augmenting textual qualitative features in deep convolution recurrent neural network for automatic essay scoring. In *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications*, Melbourne, Australia. Association for Computational Linguistics, pp. 93–102.

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, Minnesota. Association for Computational Linguistics, pp. 4171–4186.

Dickinson M. and Ledbetter S. (2012). Annotating errors in a Hungarian learner corpus. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, Istanbul, Turkey. European Language Resources Association (ELRA), pp. 1659–1664.

Dikli S. and Bleyle S. (2014). Automated essay scoring feedback for second language writers: How does it compare to instructor feedback? *Assessing Writing* **22**, 1–17.

Dong F. and Zhang Y. (2016). Automatic features for essay scoring – An empirical study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas. Association for Computational Linguistics, pp. 1072–1077.

Dong F., **Zhang Y. and Yang J.** (2017). Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, Vancouver, Canada. Association for Computational Linguistics, pp. 153–162.

Enright M.K. and Quinlan T. (2010). Complementing human judgment of essays written by English language learners with e-rater scoring. *Language Testing* **27**(3), 317–334.

Foltz P.W., **Laham D. and Landauer T.K.** (1999). The intelligent essay assessor: Applications to educational technology. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning (IMEJ)* **1**(2), 939–944.

Frazier L. (1985). Syntactic complexity. In Dowty D.R., Karttunen L. and Zwicky A.M. (eds), *Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives*. Cambridge: Cambridge University Press, pp. 129–189.

Ge T., **Wei F. and Zhou M.** (2018). Fluency boost learning and inference for neural grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia. Association for Computational Linguistics, pp. 1055–1065.

Han N.-R., **Tetreault J.**, **Lee S.-H. and Ha J.-Y.** (2010). Using an error-annotated learner corpus to develop an ESL/EFL error correction system. In *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA), pp. 763–770.

Hashimoto K., **Xiong C.**, **Tsuruoka Y. and Socher R.** (2017). A joint many-task model: Growing a neural network for multiple NLP tasks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark. Association for Computational Linguistics, pp. 1923–1933.

Ke Z. and Ng V. (2019). Automated essay scoring: A survey of the state of the art. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, Macao. International Joint Conferences on Artificial Intelligence Organization, pp. 6300–6308.

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

Kitaev N., **Cao S. and Klein D.** (2019). Multilingual constituency parsing with self-attention and pre-training. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy. Association for Computational Linguistics, pp. 3499–3505.

Landauer T.K., **Laham D. and Foltz P.W.** (2003). Automated scoring and annotation of essays with the intelligent essay assessor. In Shermis M.D. and Burstein J.C. (eds), *Automated Essay Scoring*, chapter 6. New York: Routledge, pp. 87–112.

Lee Y.-W., **Gentile C.A. and Kantor R.** (2008). Analytic Scoring of TOEFL CBT Essays: Scores From Humans and E-rater. Technical report, ETS, Princeton, NJ.

Li Z., **Dursun A. and Hegelheimer V.** (2017). Technology and L2 writing. In Chapelle C.A. and Sauro S. (eds), *The Handbook of Technology and Second Language Teaching and Learning*, chapter 6. Hoboken, New Jersey, USA: John Wiley & Sons, Inc., pp. 77–92.

Li Z., **Link S.**, **Ma H.**, **Yang H. and Hegelheimer V.** (2014). The role of automated writing evaluation holistic scores in the ESL classroom. *System* **44**, 66–78.

Lim K., **Lee J.Y.**, **Carbonell J. and Poibeau T.** (2020). Semi-supervised learning on meta structure: Multi-task tagging and parsing in low-resource scenarios. In *Proceedings of The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20)*, New York, USA. Palo Alto, California, USA: AAAI Press, pp. 8344–8351.

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L. and Stoyanov V. (2019). Roberta: A robustly optimized BERT pretraining approach. CoRR, abs/1907.11692.

Lu X. (2010). Automatic analysis of syntactic complexity in second language writing. *International Journal of Corpus Linguistics* **15**(4), 474–496.

Martindale M.J. and Carpuat M. (2018). Fluency over adequacy: A pilot study in measuring user trust in imperfect MT. In *Proceedings of the 13th Biennial Conference Organized by the Association for Machine Translation in the Americas (AMTA 2018)*, Boston, Massachusetts, USA. Association for Machine Translation in the Americas, pp. 13–25.

Nadeem F., Nguyen H., Liu Y. and Ostendorf M. (2019). Automated essay scoring with discourse-aware neural models. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, Florence, Italy. Association for Computational Linguistics, pp. 484–493.

Ortega L. (2003). Syntactic complexity measures and their relationship to L2 proficiency: A research synthesis of college-level L2 writing. *Applied Linguistics* **24**(4), 492–518.

Page E. and Petersen N.S. (1995). The computer moves into essay grading: Updating the ancient test. *Phi Delta Kappan* **76**(7), 561–565.

Page E.B. (1966). The imminence of. . . grading essays by computer. *The Phi Delta Kappan* **47**(5), 238–243.

Park D.H., Hendricks L.A., Akata Z., Schiele B., Darrell T. and Rohrbach M. (2016). Attentive Explanations: Justifying Decisions and Pointing to the Evidence. arXiv:1612.04757.

Park J. (2017). Segmentation granularity in dependency representations for Korean. In *Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017)*, Pisa, Italy. Association for Computational Linguistics, pp. 187–196.

Park J. and Lee J. (2016). A Korean learner corpus and its features. *Journal of the Linguistic Society of Korea* **75**, 69–85.

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, Italy. 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, Qatar. Association for Computational Linguistics, pp. 1532–1543.

Persing I., Davis A. and Ng V. (2010). Modeling organization in student essays. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, Cambridge, MA. Association for Computational Linguistics, pp. 229–239.

Persing I. and Ng V. (2013). Modeling thesis clarity in student essays. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Sofia, Bulgaria. Association for Computational Linguistics, pp. 260–269.

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, Turkey. European Language Resources Association (ELRA), pp. 2089–2096.

Polio C.G. (1997). Measures of linguistic accuracy in second language writing research. *Language Learning* **47**(1), 101–143.

Powers D.E., Burstein J., Chodorow M., Fowles M.E. and Kukich K. (2000). Comparing the Validity of Automated and Human Essay Scoring. Technical report, ETS, Princeton, NJ.

Powers D.E., Burstein, J., Chodorow M., Fowles M.E. and Kukich K. (2001). Stumping E-rater: Challenging the validity of automated essay scoring. Technical report, ETS, Princeton, NJ.

Prakash A. and Madabushi H.T. (2020). Incorporating count-based features into pre-trained models for improved stance detection. arXiv preprint arXiv:2010.09078.

Qiu M. and Park J. (2019). Artificial error generation with fluency filtering. In *Proceedings of the 14th Workshop on Innovative Use of NLP for Building Educational Applications (BEA2019)*, Florence, Italy. Association for Computational Linguistics, pp. 87–91.

Ramesh D. and Sanampudi S.K. (2021). An automated essay scoring systems: A systematic literature review. *Artificial Intelligence Review*, 1–33.

Ramos M.A., Wanner L., Vincze O., del Bosque G.C., Veiga N.V., Suárez E.M. and González S.P. (2010). Towards a motivated annotation schema of collocation errors in learner corpora. In *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA), pp. 3209–3214.

Ridley R., He L., Dai X.-y., Huang S. and Chen J. (2021). Automated cross-prompt scoring of essay traits. *Proceedings of the AAAI Conference on Artificial Intelligence* 35(15), 13745–13753.

Rosenberg S. and Abbeduto L. (1987). Indicators of linguistic competence in the peer group conversational behavior of mildly retarded adults. *Applied Psycholinguistics* **8**(1), 19–32.

Rudner L.M., Garcia V. and Welch C. (2006). An evaluation of IntelliMetric essay scoring system. *The Journal of Technology, Learning and Assessment* **4**(4), 1–22.

https://doi.org/10.1017/S1351324922000298 Published online by Cambridge University Press

Sakaguchi K., **Heilman M. and Madnani N.** (2015). Effective feature integration for automated short answer scoring. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Denver, Colorado. Association for Computational Linguistics, pp. 1049–1054.

Schultz M.T. (2013). The IntelliMetric automated essay scoring engine – A review and an application to Chinese essay scoring. In Shermis M.D. and Burstein J. (eds), *Handbook of Automated Essay Evaluation: Current Applications and New Directions*, chapter 6. London, UK: Taylor & Francis Group, pp. 89–98.

Seo H., **Lee K.**, **Lee G.G.**, **Kweon S.-O. and Kim H.-R.** (2012). Grammatical error annotation for Korean learners of spoken English. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, Istanbul, Turkey. European Language Resources Association (ELRA), pp. 1628–1631.

Shermis M.D. and Burstein J.C. (2003). Introduction. In Shermis M.D. and Burstein J.C. (eds), *Automated Essay Scoring: A Cross-disciplinary Perspective*. New York: Routledge, pp. xiii–xvii.

Shermis M.D., **Koch C.M.**, **Page E.B.**, **Keith T.Z. and Harrington S.** (2002). Trait ratings for automated essay grading. *Educational and Psychological Measurement* **62**(1), 5–18.

Soni M. and Thakur J.S. (2018). A systematic review of automated grammar checking in English language. CoRR, abs/1804.00540.

Taghipour K. and Ng H.T. (2016). A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas. Association for Computational Linguistics, pp. 1882–1891.

Towell R., **Hawkins R. and Bazergui N.** (1996). The development of fluency in advanced learners of French. *Applied Linguistics* **17**(1), 84–119.

Uto M. (2021). A review of deep-neural automated essay scoring models. *Behaviormetrika* **48**(2), 459–484.

Uto M., **Xie Y. and Ueno M.** (2020). Neural automated essay scoring incorporating handcrafted features. In *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona, Spain (Online). International Committee on Computational Linguistics, pp. 6077–6088.

Vajjala S. and Loo K. (2013). Role of Morpho-Syntactic features in estonian proficiency classification. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, Atlanta, Georgia. Association for Computational Linguistics, pp. 63–72.

Vaswani A., **Shazeer N.**, **Parmar N.**, **Uszkoreit J.**, **Jones L.**, **Gomez A.N.**, **Kaiser L. and Polosukhin I.** (2017). Attention is all you need. In Guyon I., Luxburg U.V., Bengio, S., Wallach H., Fergus R., Vishwanathan S. and Garnett R. (eds), *Advances in Neural Information Processing Systems 30*. Curran Associates, Inc., pp. 6000–6010

Wang J. and Brown M.S. (2007). Automated essay scoring versus human scoring: A comparative study. *The Journal of Technology, Learning and Assessment* **6**(2), 1–29.

Xue K., **Zhou Y.**, **Ma Z.**, **Ruan T.**, **Zhang H. and He P.** (2019). Fine-tuning bert for joint entity and relation extraction in chinese medical text. In 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 892–897.

Yang R., **Cao J.**, **Wen Z.**, **Wu Y. and He X.** (2020). Enhancing automated essay scoring performance via fine-tuning pre-trained language models with combination of regression and ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, Online. Association for Computational Linguistics, pp. 1560–1569.

Yang Y., **Xia L. and Zhao Q.** (2019). An automated grader for Chinese essay combining shallow and deep semantic attributes. *IEEE Access* **7**, 176306–176316.

Yngve V.H. (1960). A model and an hypothesis for language structure. *Proceedings of the American Philosophical Society* 104(5), 444–466.

Cite this article: Lim KT, Song J and Park J (2023). Neural automated writing evaluation for Korean L2 writing. *Natural Language Engineering* **29**, 1341–1363. https://doi.org/10.1017/S1351324922000298