# Correcting Chinese Word Usage Errors for Learning Chinese as a Second Language

Yow-Ting Shiue[1], Hen-Hsen Huang[1], and Hsin-Hsi Chen[1,2]

[1]Department of Computer Science and Information Engineering,
National Taiwan University, Taipei, Taiwan
[2]MOST Joint Research Center for AI Technology and All Vista Healthcare, Taipei, Taiwan
orina1123@gmail.com, hhhuang@nlg.csie.ntu.edu.tw,
hhchen@ntu.edu.tw

## Abstract

With more and more people around the world learning Chinese as a second language, the need of Chinese error correction tools is increasing. In the HSK dynamic composition corpus, word usage error (WUE) is the most common error type. In this paper, we build a neural network model that considers both target erroneous token and context to generate a correction vector and compare it against a candidate vocabulary to propose suitable corrections. To deal with potential alternative corrections, the top five proposed candidates are judged by native Chinese speakers. For more than 91% of the cases, our system can propose at least one acceptable correction within a list of five candidates. To the best of our knowledge, this is the first research addressing general-type Chinese WUE correction. Our system can help non-native Chinese learners revise their sentences by themselves.

## Title and Abstract in Chinese

非中文母語學習者中文寫作用詞錯誤之更正

以中文爲第二語言的學習者與日俱增，遍及全球，對於中文更正工具的需求也相應而生。在HSK動態作文語料庫中，用詞錯誤（WUE）是中文學習者最常犯的錯誤類型。在這篇論文中，針對中文用詞的更正，我們建立了專屬的神經網路模型，同時參酌目標錯誤詞彙及其前後文資訊，以產生推薦的更正方式。經過中文母語人士的評估，在91%的測試案例中，系統所推薦的前五名候選詞彙，至少有一個是適當的更正方式。據我們所知，這是目前第一個全面性的中文用詞錯誤更正之研究。我們的系統可以幫助非中文母語人士自主修正其所寫的中文句子，促進華語文教學之成效。

## 1 Introduction

Grammatical error correction (GEC) tools can help language learners revise their writing. Chinese GEC tools are in high demand since Chinese has become an increasingly popular second language worldwide. Despite the increasing need, most of the existing studies on GEC are based on English learner data. The method of correcting sentences in Chinese, a language which differs substantially from English in major aspects such as the morphological structure and the distribution of learner errors, has not yet been fully developed.

This paper focuses on the correction of Chinese word usage errors (WUEs). According to the definition of Shiue and Chen (2016), a WUE refers to an incorrect token that involves morphological, syntactical, or semantical problems. The token is either an incorrect word form, or a correct existent word that is improper for its context.

Given a token in a sentence segment that is known to be erroneous, we aim to generate a suitable correction for it. The criteria for a suitable correction are:

- **Correctness**: After substituting the erroneous token with the correction token, the result is a syntactically and semantically correct Chinese sentence segment.





*Proceedings of the 27th International Conference on Computational Linguistics*, pages 2410–2422
Santa Fe, New Mexico, USA, August 20-26, 2018.

• **Similarity**: The meaning of the correction is close to the writer's intended meaning.

We discuss the criteria with the following example sentence segments.

(E1-1) *生活方式已經**猛烈**地改變了 ( The way of living has been **fiercely** changed. )

(E1-2) *生活方式已經**暴烈**地改變了 ( ... has been **overpoweringly** changed. )

(E1-3) 生活方式已經**緩慢**地改變了 ( ... has been **slowly** changed. )

(E1-4) 生活方式已經**劇烈**地改變了 ( ... has been **dramatically** changed. )

For wrong segment (E1-1), (E1-2) is not a correction since it is incorrect itself. The adverb "暴烈地" (overpoweringly) does not collocate with the verb "改變" (change). (E1-3) is grammatical, but its meaning differs from the original meaning of (E1-1), so neither is it suitable. (E1-4) is a good correction that meets both criteria.

Nevertheless, there are some cases in which the similarity criterion is hard to meet. For example, the intended meaning of (E2-1) is very difficult to recognize. (E2-2) is the ground-truth correction, but the association between the original erroneous token "情緒" (emotion) and the correction "因素" (factor) is unclear.

(E2-1) *發生這種情況的**情緒**很多 ( Many **emotions** happen this situation. )

(E2-2) 發生這種情況的**因素**很多 ( Many **factors** can lead to this situation. )

When two criteria cannot be met at the same time, the correctness criterion should have higher priority, since an incorrect sentence can confuse the language learner.

In this paper, *target* refers to the original erroneous token written by the language learner, and *context* refers to other words in the sentence segment. A pair consisting of the target and its corresponding correction is called a *correction pair*. In example (E1) and (E2), "猛烈" and "情緒" are targets, and (猛烈, 劇烈) and (情緒, 因素) are correction pairs. According to the previous discussions, the major challenge of the WUE correction task lies in the derivation of the intended semantics and the generation of valid sentence segments.

We treat WUE correction as a candidate selection problem and propose a neural network-based model considering both target and context to rank correction candidates. Our main contributions are: (1) Though the detection of Chinese WUE and correction of certain types have been studied, this is the first research dealing with the correction of all types of WUEs. (2) To consider alternative corrections, we perform human evaluation and show that our system can propose at least one acceptable correction within the top five candidates for more than 91% of the cases. (3) We release the HSK WUE dataset with additional human annotations.

## 2 Related Work

English GEC is a rather mature field of study in NLP. Several shared tasks have been conducted for English GEC (Dale and Kilgarriff, 2011; Dale et al., 2012; Ng et al., 2013; Ng et al., 2014). Language models, machine learning classifiers, rule-based classifiers, and machine translation models are used. The machine translation approach has the advantage that there is no need to explicitly formulate the types of the errors. A series of English GEC studies are based on the phrase-based statistical machine translation (SMT) framework (Dahlmeier and Ng, 2011; Chollampatt et al., 2016b; Chollampatt et al., 2016a; Chollampatt and Ng, 2017).

Nevertheless, the satisfactory performance of the SMT approach cannot be reached without sufficient training data. In fact, Chollampatt et al. (2016a) have shown that the model trained with smaller training data from writers with the same first language (L1) as writers of the test data performs even worse than the model trained with larger training data from writers whose L1 differs from that of the test data. The amount of available Chinese learner data are even less sufficient than that of English ones. Therefore, as a preliminary research in Chinese WUE correction, we impose restrictions on our setting that there is exactly one error in a sentence segment, the error position is known, and the error can be corrected by replacing the erroneous token with an appropriate word.

The distribution of errors of non-native Chinese differ a lot from that of non-native English. In the CoNLL 2013 shared task (Ng et al., 2013), the most frequent error types are article, preposition, noun number, verb form, and subject-verb agreement. These error types are mostly in violation of English

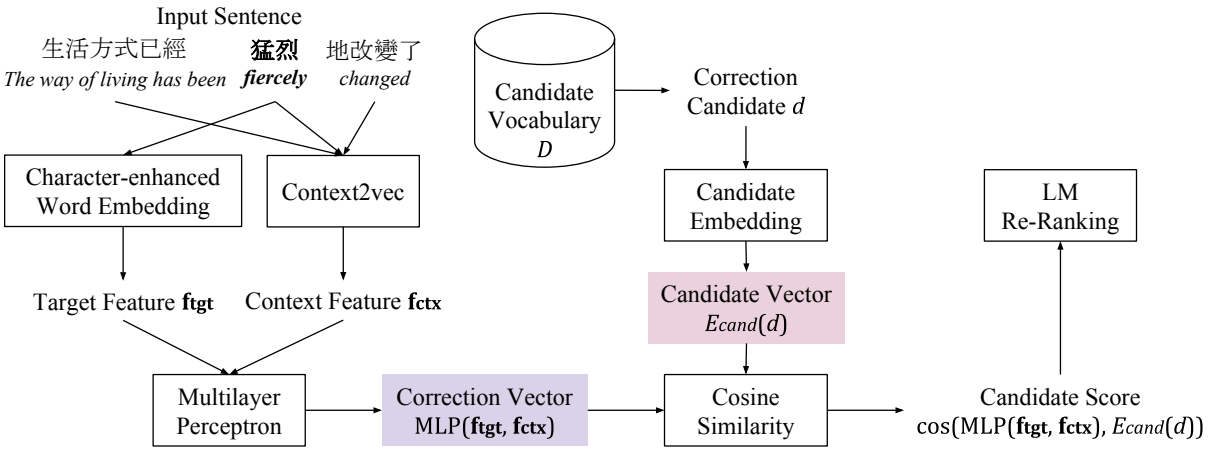


Figure 1: Overview of our correction generation model. The score of a correction candidate $d$ is predicted for replacing the target erroneous word "猛烈" (fiercely) with the target and the context information.

grammar rules, so it is relatively easy to grasp some patterns of correction. In contrast, in the HSK dynamic composition corpus built by Beijing Language and Culture University, which is the largest available Chinese learner corpus at the time of this study, WUE is the most frequent lexical-level error. In most cases, a wrong usage of a word does not lead to violation of syntactic rules. Instead, its incorrectness cannot be determined without understanding the meaning of the whole sentence. As a result, besides directly adopting the techniques used for English GEC, many aspects of Chinese GEC are worth studying.

The Chinese spelling check task (Wu et al., 2013; Yu et al., 2014b; Tseng et al., 2015; Fung et al., 2017) evaluates the detection and correction of character errors. The Shared Task for Chinese Grammatical Error Diagnosis (Yu et al., 2014a; Lee et al., 2015; Lee et al., 2016; Rao et al., 2017) extends the above task to word errors, including redundant word, missing word, word disorder and word selection. Nevertheless, these tasks only deal with detection but not correction.

Some researchers focus on certain types of Chinese writing errors. For example, Yu and Chen (2012) identify word ordering errors (WOEs), and Cheng et al. (2014) further recommend word ordering correction candidates with the use of ranking support vector machine (RankSVM).

Previous researches on Chinese WUE include segment-level (Shiue and Chen, 2016) and token-level detection (Shiue et al., 2017). Huang et al. (2016) study the Chinese preposition selection problem, which is subsumed by WUE correction. Gated recurrent unit (GRU)-based models are trained to select the most suitable one from a closed set of 43 Chinese prepositions in a context.

Nevertheless, it is still worth investigating how to treat WUEs involving other types of words such as verbs and nouns. Correcting errors of such open-set types could be much more difficult since the set of candidates can be huge. To the best of our knowledge, this is the first research dealing with general-type Chinese WUE correction.

## 3 Neural Network-based Correction Generation Model

As shown in Figure 1, our correction generation model is a multilayer perceptron (MLP) neural network that takes target feature vector $\mathbf{f_{tgt}}$ and context feature vector $\mathbf{f_{ctx}}$ as input and outputs a *correction vector* MLP($\mathbf{f_{tgt}}, \mathbf{f_{ctx}}$). $\mathbf{f_{tgt}}$ and $\mathbf{f_{ctx}}$ are derived either from the segment itself or with the help of some external resources. Section 5 will give the details for several kinds of features we used.

Every candidate word $d$ in a set $D$ of all possible corrections is mapped to an embedding $E_{cand}(d)$. The details of candidate embedding model $E_{cand}$ will be elaborated in Section 4. The training objective of our model is to make MLP($\mathbf{f_{tgt}}, \mathbf{f_{ctx}}$) as close to $E_{cand}(\hat{d})$ as possible, where $\hat{d}$ is the ground-truth correction. While MLP contains a bunch of trainable parameters, $E_{cand}$ is fixed to preserve the ability to generalize the corrections that are unseen in the training data.

When testing, given the original erroneous sentence segment, our model selects the most probable correction $d*$:

$$d* = \underset{d \in D}{\operatorname{argmax}} \cos(\text{MLP}(\mathbf{f_{tgt}}, \mathbf{f_{ctx}}), E_{cand}(d)) \quad (1)$$

In fact, this model can propose several candidates, ranked by their cosine similarities to the correction vector. The goal is to rank the ground-truth correction toward the top of the list.

## 4 Candidate Embedding

For the candidate embedding model $E_{cand}$, we use the character-enhanced word embedding model (CWE), in which word vectors (hereafter WE) and character vectors (hereafter CE) are learned jointly (Chen et al., 2015). This model is developed based on one idiosyncratic characteristic of the Chinese writing system, that meanings of individual characters in a word usually contribute to the meaning of the word. Therefore, character-level information can help capture the meaning of Chinese words much better.

On the other hand, one common case of WUE is that the characters within a word is wrongly chosen or permuted. For instance, "決解" (jué jiě) is a misused form of "解決" (solve). Though the misused form is a non-existent word, its character components serve as an important clue for discovering what the writer originally means and help provide more suitable correction.

We adopt the position-based CWE (CWE+P), which keeps three embeddings for each character according to the character's position. This variant is designed to capture different morphological functions of a Chinese character when it is at different positions in a word. A character vector is denoted as $\text{CE}(c, p)$, where $c$ is the character and $p = s, m, e$ depending on the position (start, middle or end, respectively) of $c$ within a certain word. The word representation is the word vector plus the average of the component character vectors. For example, the representation of "農產品" (agricultural product) is $\text{CWE}_w(\text{農產品}) = \text{WE}(\text{農產品}) + \frac{1}{3}[\text{CE}(\text{農}, s) + \text{CE}(\text{產}, m) + \text{CE}(\text{品}, e)]$. If we ignore the internal structure of the word, a misused form "農作品" ('nóng zuò pǐn') is simply an out-of-vocabulary (OOV) word and has no association with the correction "農產品". With the use of CWE vectors, it is more possible for the model to learn a transformation from the incorrect token to its correction.

## 5 Input Features

To meet the two criteria for suitable WUE correction, we adopt several target and context features as input to the correction generation model. The division of $\mathbf{f_{tgt}}$ and $\mathbf{f_{ctx}}$ is just for indicating the source of information. These two feature vectors are concatenated and fed to the MLP model.

### 5.1 Target CWE+P Word Embedding

This set of features is derived from the same embedding model as we used for the candidate embedding. The way of composing a representation of an existent word is also the same. For OOV word, the representation is the average of the vectors of all characters. By doing so, among the three character vector terms in the representation of "農作品", $\text{CE}(\text{農}, s)$ and $\text{CE}(\text{品}, e)$ are shared with the correction vector $\text{CWE}_w(\text{農產品})$. The word and character vectors used to calculate this set of features are in the same space with the candidate embedding, enabling the model to directly learn a transformation between a correction pair. We use $\text{CWE}_w$ to denote this set of features.

### 5.2 Target CWE Position-Insensitive Character Embedding

Although a character's position in a word could reflect its morphological function, non-native Chinese learners might not be so familiar with Chinese morphology. One common type of morphological WUE is incorrect ordering of characters within a word. For example, (*決解, 解決) is the tenth frequent correction pair in our dataset. With the use of CWE+P embeddings, the similarity between these two tokens might be underestimated since all the character vector terms are different.

To cope with this problem, we experimented with the CWE variant that only keeps one vector for each Chinese character regardless of its position, but the performance is not as good as the model with

CWE+P features. Alternatively, we design a separate set of character embedding features $\text{CWE}_c$ which is the sum of the character embedding of all positions divided by the number of characters in the word. For instance, $\text{CWE}_c(\text{決解}) = \frac{1}{2}\sum_{p=s,m,e}[\text{CE}(\text{決}, p) + \text{CE}(\text{解}, p)]$. As can be seen, $\text{CWE}_c(\text{決解})$ will contain $\text{CE}(\text{解}, s)$ and $\text{CE}(\text{決}, e)$, which are the terms of $\text{CWE}_w(\text{解決})$.

### 5.3 Context2vec Features

Context2vec (Melamud et al., 2016) is a bidirectional LSTM-based model that can encode a "context" into a real-valued vector. A context is a sequence of words with a certain position blanked out. For instance, (E3-1) is a context:

(E3-1) 可是 每 個 人 的 [ ] 都 千差萬別 ( but everyone's [ ] is different )

The representation of a context is a combination of the sequence of words before and after the blank:

$$\text{C2V}_{ctx}(w_1...w_{p-1}[\ ]w_{p+1}...w_L) = \text{LSTM}(w_1...w_{p-1}) \oplus \text{LSTM}(w_{p+1}...w_L) \quad (2)$$

where each $w_i$ is a token, $L$ is the number of tokens, $p$ is the index of the blank, and $\oplus$ is the vector concatenation operation.

Context2vec also keeps the embeddings of individual words, which are called target embeddings[1] by Melamud et al. (2016). We use $\text{C2V}_{tgt}$ to denote the vector of target word. Both target embeddings and the parameters in the LSTM layers are updated during training. The objective of the model is to predict the target word that actually occurs in the training sentence, given the encoded context vector.

The formulation of context makes Context2vec suitable for the sentence completion task. A candidate to fill the blank can be selected according to how similar its vector is to the context vector. That is, given a context where the $p$-th position is the blank, the best candidate $d*$ would be:

$$d* = \underset{d \in D}{\text{argmax}} \cos(\text{C2V}_{tgt}(d), \text{MLP}(\text{C2V}_{ctx}(w_1...w_{p-1}[\ ]w_{p+1}...w_L))) \quad (3)$$

where MLP is a non-linear projection that maps the context representation to a vector with dimensionality same as that of the target embeddings. Melamud et al. (2016) have shown the promising results of Context2vec in several sentence completion benchmarks. For the example context (E3-1), the best candidate selected in this way using our trained model is "境況", which can be put into the blank and the result is a correct sentence segment.

(E3-2) 可是 每 個 人 的 [境況] 都 千差萬別 ( but everyone's [situation] is different )

In fact, (E3-1) is extracted from a wrong segment in our dataset. The original erroneous segment and the corresponding correction are shown in (E3-3).

(E3-3) 可是每個人的(*對應, 反應)都千差萬別

( but everyone's (correspondence, reaction) is different )

Given the original segment, one can conclude that the candidate "境況" (situation) selected by Context2vec is less suitable compared to the ground-truth "反應" (reaction), according to the similarity criterion. Therefore, WUE correction is different from sentence completion in that if the model ignores the word originally written by the language learner, it is likely to generate a correction that changes the meaning of the original sentence segment.

To take both context and target information into account, we include two feature vectors $\text{C2V}_{tgt}$ and $\text{C2V}_{ctx}$ , which belong to target and context features, respectively. $\text{C2V}_{tgt}$ refers to the embedding of the original erroneous token, which can reveal important information about the writer's intended meaning as we discussed above.

### 5.4 Target POS Features

We analyze the part-of-speech (POS) tags before and after correction, and show the most frequent POS changes in the validation set in Table 1. The POS tagging is performed by using Stanford CoreNLP

[1]Note that the definition of "target" in the Context2vec paper is slightly different from ours. In our definition, target only refers to the original erroneous token, while for Context2vec, target can refer to any word to be put into the blank, regardless of whether the result is a correct sentence.

| Original POS | Correction POS | # instances | Frequency |
|---|---|---|---|
| Unchanged | | 722 | 68.70% |
| VV | NN | 27 | 2.57% |
| NN | VV | 21 | 2.00% |
| P | VV | 17 | 1.62% |
| DEC | DEV | 15 | 1.43% |
| VV | P | 13 | 1.24% |
| AD | VV | 10 | 0.95% |
| VV | VA | 10 | 0.95% |

Table 1: Part-of-speech changes occurring at least 10 times.

(Manning et al., 2014). As can be seen, the POS tag does not change after the correction in nearly 70% of the cases. Besides, there are some systematic changes. For example, non-native Chinese learners seem to confuse some nouns with some verbs, so we can observe the interchanging phenomenon of the VV and NN tags. The case of VV and P is similar. These systematic changes indicate that it is possible to reduce the candidate vocabulary. We tried to limit the candidates to the POS transitions observed in the training and validation set. However, this results in slightly lower accuracy and MRR. Therefore, instead of modifying the candidate set directly, we encode the POS of the erroneous token in a one-hot vector and feed this feature to the correction generation model. This allows the model to learn different transformation function for different source POS, that is, the POS of the erroneous token.

## 6 Language Model Re-ranking

One drawback of our MLP correction generation model is that the correctness criterion does not explicitly take priority over the similarity criterion. In our experiments, we found that our model sometimes generates segments that seriously violate the correctness criterion, since it can bias toward the similarity criterion. (E4) is an example.

*Wrong segment:*

(E4-1) *到山頂**之間**路走得不容易 ( The road **between** the hilltop was not easy to walk. )

*Model prediction:*

(E4-2) *到山頂**期間**路走得不容易 ( The road **period** the hilltop ... )

*Ground-truth correction:*

(E4-3) 到山頂**的**路走得不容易 ( The road **to** the hilltop ... )

The candidate "期間" (period) is selected since it is similar to the target "之間" (between), but the result sentence segment is incorrect. It is necessary to deal with this problem since the correctness criterion is more important, as we previously discussed in the introduction.

It is expected that this kind of unsuitable candidates can be eliminated by a language model (LM), if we assume that LM probability reflects the level of correctness of a sentence segment. Therefore, we emphasize the correctness criterion by incorporating LM scores of traditional n-gram LM or Recurrent Neural Network Language Model (RNNLM) (Mikolov et al., 2011) into the candidate selection process. One possible approach is to apply a probability cut-off and discard candidates that result in segments with low probability. Nevertheless, the "acceptable" LM probability varies from sentence to sentence since it can be affected by, for instance, length and lexical complexity of the sentence. Though we can let the cutoff be a function of various factors, it is difficult to design such a function explicitly.

Therefore, instead of performing combination of scores, we combine the rank proposed by the LM with the rank based on our correction generation model. One advantage of this approach is that the range of ranks is the same for all instances given fixed candidate vocabulary size, so the ranks can be evaluated with the same standard across different instances. For a candidate correction, let $r_{LM}$ be its rank based on the LM probability, and $r_{MLP}$ be the rank based on its cosine similarity to the correction vector generated by our MLP model. We adopt the following weighted harmonic mean to obtain a new "rank" for the candidate.

$$r_{com} = \frac{1}{\frac{\alpha}{r_{LM}} + \frac{1-\alpha}{r_{MLP}}} \tag{4}$$

where $\alpha$ is a parameter that can be tuned with the validation set (actual values will be given in Section 8.2). Preliminary experiments show that harmonic mean performs better than arithmetic and geometric mean. Though $r_{com}$ may not be an integer, it can be interpreted as a rank. The correction with smaller $r_{com}$ is considered better.

## 7 Experimental Settings

We adopt the HSK WUE dataset released by Shiue et al. (2017)[2] and follow their train/validation/test split. We use Stanford CoreNLP (Manning et al., 2014) for Chinese word segmentation and POS tagging. For each split, we filter the instances where the correction is not within the top 50,000 frequent words in the Chinese part of the ClueWeb corpus[3] (Yu et al., 2012). This decision is made based on the fact that the vocabulary used by non-native language learners is limited. After filtering, the number of instances in the train, validation, and test sets are 8,205, 1,026, and 1,025, respectively. With punctuation marks and English words eliminated, the candidate vocabulary size $|D|$ is 48,394.

Our MLP model has two hidden layers of size 1,024. The activation function is Rectified Linear Unit (ReLU) and the dropout rate is set to 0.2. The parameters are optimized with Adagrad (Duchi et al., 2011) under a cosine proximity objective function. CWE (embedding size 400), Context2vec (300 units), 5-gram LM and RNNLM (size-128 GRU) are all trained on Chinese ClueWeb. Please refer to Appendix A for detailed parameter settings.

## 8 Automatic Evaluation

### 8.1 Result before LM Re-ranking

We first evaluate our correction generation model with the single ground-truth correction per segment. The results are shown in Table 2. The baselines include the two LMs and the Context2vec sentence completion method, which selects a candidate most similar to the context but ignores the original erroneous token. The 5-gram LM is the strongest baseline for the WUE correction task.

The second part of Table 2 shows the result of a set of experiments with Context2vec features. The MLP model with only C2V$_{ctx}$ features differs from the Context2vec baseline in that it is trained with the WUE dataset. Learning a transformation from the erroneous token to the correction seems to be easier than guessing a correction only from context, probably because some common correction pairs can be learned. The model using only C2V$_{tgt}$ achieves performance substantially better than that using only C2V$_{ctx}$. Note that the Context2vec vectors does not lay in the same space with the CWE+P vectors we used for $E_{cand}$. This indicates that our model is indeed capable of learning a transformation, not just copying the vector terms from the input features. Combining C2V$_{tgt}$ and C2V$_{ctx}$ can further enhance the performance.

The third part of Table 2 shows another set of experiments, in which different kinds of features are included incrementally starting from the CWE+P target features. The last row indicates the performance when all features are used. The model with CWE$_w$ features performs better than that with C2V$_{tgt}$, since CWE+P composes a vector representation for OOV targets such as "農作品", giving the model more clues for generating the correction. The position-insensitive character feature CWE$_c$ slightly improves the performance over CWE$_w$. After including Context2vec context and target features, the model can consider the context and reaches accuracy 0.3512. Finally, incorporating POS information further enhances the accuracy to 0.3717 ($p < 0.05$ significance) and MRR to 0.4378.

### 8.2 Effect of LM Re-ranking

We apply LM re-ranking to the candidate ranks generated by the best correction generation model. The results of two alternative LMs are shown in Table 3. Although there is only slight improvement on the accuracy, the MRR and hit rates increase substantially after LM re-ranking is applied. The 5-gram LM

[2]http://anthology.aclweb.org/attachments/P/P17/P17-2064.Datasets.zip ; corresponding ground-truth corrections retrieved from: http://202.112.195.192:8060/hsk/login.asp

[3]http://www.lemurproject.org/clueweb09/

| | Target | Context | Accuracy | MRR | Hit@5 | Hit@10 | Hit@50 |
|---|---|---|---|---|---|---|---|
| Baselines (No training on the WUE dataset) | - | 5-gram LM | 0.1659 | 0.2438 | 0.3268 | 0.4029 | 0.5951 |
| | - | RNNLM | 0.1468 | 0.2208 | 0.2847 | 0.3611 | 0.5793 |
| | - | $C2V_{ctx}$ | 0.0714 | 0.1170 | 0.1575 | 0.2114 | 0.3611 |
| Correction Generation Model with Context2vec Features | - | $C2V_{ctx}$ | 0.1249 | 0.1746 | 0.2273 | 0.2741 | 0.4010 |
| | $C2V_{tgt}$ | - | 0.2507 | 0.3030 | 0.3561 | 0.3932 | 0.5024 |
| | $C2V_{tgt}$ | $C2V_{ctx}$ | 0.3249 | 0.3891 | 0.4566 | 0.4976 | 0.6185 |
| Correction Generation Model with CWE + Other Features | $CWE_w$ | | 0.2898 | 0.3545 | 0.4195 | 0.4693 | 0.5971 |
| | + $CWE_c$ | | 0.2946 | 0.3570 | 0.4234 | 0.4722 | 0.6078 |
| | + $C2V_{tgt}$ | + $C2V_{ctx}$ | 0.3512 | 0.4250 | 0.5024 | 0.5571 | 0.6800 |
| | + POS | | 0.3717 | 0.4378 | 0.5063 | 0.5688 | 0.6956 |

Table 2: Performance of the correction generation model with various target and context features.

| Model | Accuracy | MRR | Hit@5 | Hit@10 | Hit@50 | Hit@100 |
|---|---|---|---|---|---|---|
| Best MLP | 0.3717 | 0.4378 | 0.5063 | 0.5688 | 0.6956 | 0.7415 |
| + 5-gram LM | **0.3727** | **0.4605** | **0.5561** | **0.6439** | **0.8039** | 0.8488 |
| + RNNLM | **0.3727** | 0.4527 | 0.5278 | 0.6205 | 0.7808 | 0.8302 |

Table 3: Performance with LM re-ranking.

gives slightly better MRR than RNNLM. The optimal $\alpha$ for 5-gram LM and RNNLM are 0.355 and 0.255, respectively.

(E5) is an example in which LM re-ranking helps promote the rank of the answer. Though sharing a common Chinese character, the meaning of "一起" (together) and "一直" (always, all the time) are not quite similar. Thus, the MLP rank is very low. In contrast, the LM rank is high, since "就...都不..." (have always not...) is a suitable context for "一直". The higher combined rank leads to enhanced MRR.

(E5) 我從上小學起成績就(*一起, 一直)都不理想

( Since I began elementary school, my grade has (together, always) been unsatisfactory. )

$r_{LM} = 7, r_{MLP} = 1284 \rightarrow$ Combined rank: 19

## 9 Human Evaluation

In the automatic evaluation, there is only one answer for each test instance. However, correction can be highly subjective and alternatives may exist. Moreover, since the HSK WUE dataset is composed of sentence segments, the model has no access to the context outside of the segment. This results in difficulties in making the choice among several candidate corrections that are different in meaning but all seem to be acceptable. Table 4 shows an example. The erroneous token "定心" ('dìng xīn') is not a valid noun in Chinese. The meaning of character "定" is related to "stable, fixed", and the meaning of "心" is related to "mind". The top five candidates proposed by our system are all differ from the ground-truth correction. However, except for the rank 3 candidate, all other four candidates are acceptable corrections. This example shows that the single-answer automatic evaluation can underestimate the performance of our system. Therefore, we perform human evaluation, in which the top candidates proposed by our model are judged by annotators.

| Wrong segment | 不過我們要以堅定的**定心**與病對抗 (but we should fight against the disease with strong **'dìng xīn'**.) |
|---|---|
| System 1st | 不過我們要以堅定的**自信**與病對抗 (... with strong **self-confidence**.) |
| System 2nd | 不過我們要以堅定的**信念**與病對抗 (... with strong **faith**.) |
| System 3rd | 不過我們要以堅定的**理智**與病對抗 (... with strong **rationality**.) |
| System 4th | 不過我們要以堅定的**自信心**與病對抗 (... with strong **sense of self-confidence**.) |
| System 5th | 不過我們要以堅定的**毅力**與病對抗 (... with strong **persistence**.) |
| Ground-truth | 不過我們要以堅定的**決心**與病對抗 (... with strong **determination**.) |

Table 4: An example of alternative corrections.

### 9.1 Annotation Guideline

Each instance of annotation consists of two sentence segments:

(S0): the original wrong segment

(S1): a correction segment, which is either the ground-truth or one of the top $k$ proposed candidates

We set $k = 5$ ; however, only the candidates ranked before the ground-truth need annotation. For example, if our system gives rank 3 to the ground-truth correction, only the ground-truth, the first candidate and the second candidate need to be judged by human. For the 1,025 test segments, a total of 3,692 annotation instances are generated.

An annotator is asked to answer (at most) two questions for each annotation instance. The questions and the instructions given to the annotators are shown in Table 5. The answers to both questions are either Yes (1) or No (0). If the answer to Q1 is No, the answer to Q2 must be No, since (S1) violates the correctness criterion; we will skip Q2 in such cases. As long as the meaning of (S1) is similar to that of (S0), the annotator should answer Yes, regardless of whether (S0) is incorrect. In case the meaning of (S0) is not understandable, as illustrated by (E2-1), the annotator should answer Yes. This corresponds to our previous claim that the correctness criterion is more important.

| Q1 (*is_g*): Is (S1) a correct sentence segment? |
|---|
| 1. If (S1) is ungrammatical, please answer No.<br>2. If (S1) is grammatical but its semantics is not logical, or violates some common-sense knowledge, please also answer No.<br>3. Since we split sentences into segments by punctuation marks, if you encounter an "incomplete" sentence, please answer Yes to it, if it is itself correct and can be completed in some reasonable way. |
| Q2 (*is_c*): Is (S1) a correction of (S0)? |
| 1. This question will be presented only if you answered Yes to Q1, which means the grammaticality of candidate correction (S1) has been confirmed.<br>2. If the meaning of (S1) is the intended meaning of (S0), please answer Yes; otherwise please answer No. |

Table 5: WUE correction annotation instructions.

There are 10 annotators who are native speakers of Chinese. They are properly trained by giving examples selected from the validation set for every case in the instructions. Each annotation instance is assigned to two annotators randomly, so we can assume that the difficulty of the data assigned to each annotator is the same. Therefore, we calculate the average inter-annotator agreement of all pairs of annotators. The average Cohen's Kappa for Q1 and Q2 are 0.4205 and 0.4070 respectively, showing moderate level of agreement. If the two annotators disagree on either question, a third annotator is introduced to break the tie. We use majority voting to determine the final answer.

### 9.2 Evaluation with Human Annotation

Total 95.60% of the ground-truth corrections and 82.73% of the system top candidates are judged as correct by the annotators. This indicates that the grammaticality of the system output is acceptable. We use the updated rank of the test instances, which is the rank of the highest ranked candidate that meets both correctness and similarity criteria according to annotation results, to re-evaluate our best model. The performance before and after applying annotation results are shown in Table 6.

As can be seen, there is large performance increase in all metrics, verifying the existence of alternative corrections. Both accuracy and MRR increase by more than 30%. Moreover, the hit@5 rate is above 91%, which means that for most of the test data, at least one of the top five candidates is an acceptable correction. A language learner can choose the one that is close to his or her intended meaning from the list of candidates. In fact, only the writer knows the exact "intended meaning", so it is nearly impossible for a system to guess the right meaning all the time. We argue that a fairly short list of candidates can be helpful for learning to write in foreign languages.

| Evaluation | Accuracy | MRR | Hit@5 | Hit@10 | Hit@50 | Hit@100 |
|---|---|---|---|---|---|---|
| Ground-truth | 0.3727 | 0.4605 | 0.5561 | 0.6439 | 0.8039 | 0.8488 |
| + Annotation | 0.6829 | 0.7784 | 0.9122 | 0.9171 | 0.9502 | 0.9600 |

Table 6: Human evaluation results.

## 10 Error Analysis

We analyze the human evaluation result according to different POS tags of the erroneous token. The results of POS tags that occur more than 20 times are shown in Table 7. The most frequent POS tags, VV, NN and AD, which are open-set word types, contribute the most difficult cases. The accuracy is less than 70% and MRR is less than 80%. On the other hand, for closed-set word types such as prepositions (P), our system performs very well, reaching accuracy 0.81 and MRR 0.88. DEV is the POS tag of Chinese adverb marker "地". The marker inherits very regular usage, and regular pattern of learner errors, so the system can achieve perfect performance. This also indicates that our system is capable of handling various kind of errors, and there is no need to include a separate rule-based module for a specific error type.

| POS | # Tests | Accuracy | MRR | Hit@5 | Hit@10 | Mean rank | Std. |
|---|---|---|---|---|---|---|---|
| VV | 316 | 0.67 | 0.77 | 0.91 | 0.92 | 26.12 | 6.75 |
| NN | 277 | 0.64 | 0.73 | 0.88 | 0.88 | 73.97 | 11.50 |
| AD | 130 | 0.65 | 0.75 | 0.88 | 0.89 | 96.16 | 13.41 |
| P | 62 | 0.81 | 0.88 | 0.95 | 0.95 | 3.10 | 1.92 |
| VA | 45 | 0.60 | 0.76 | 0.98 | 0.98 | 1.98 | 1.08 |
| DEV | 23 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.00 |
| PN | 21 | 0.71 | 0.80 | 0.95 | 0.95 | 2.33 | 1.40 |

Table 7: Performance on most frequent POS tags.

## 11 Conclusion

In this paper, given a Chinese sentence segment with a known error position, we aim to generate correction candidates that not only result in a correct segment, but also preserve the original meaning of the writer. Our MLP correction generation model takes target and context features as input and outputs a correction vector, which can be compared against the vectors of the candidate vocabulary. We apply LM re-ranking to put emphasis on correctness, avoiding misleading corrections.

In the single-ground-truth automatic evaluation, we achieve accuracy 0.3727 and MRR 0.4605. With human evaluation, in which the top five proposed candidates are judged by native Chinese speakers to include some alternative acceptable corrections, the accuracy increases to 0.6829 and MRR to 0.7784. Moreover, the hit@5 rate reaches 0.9122. Since a list of five candidates is rather short, a language learner can choose a suitable one from the candidate list and revise his or her sentences even without the help of a language teacher.

To enable future investigations, the dataset we used is attached. We have dealt with semantic similarity and similarity in overlapping Chinese characters in this paper. Nevertheless, we have not handled phonetical similarity. For example, in the correction pair (影響, 印象), the source of confusion is that the pronunciation of "影響" (influence 'yǐng xiǎng') and "印象" (impression 'yìn xiàng') are very similar. Pronunciation information can serve as clues for selecting the suitable correction.

## Acknowledgements

This research was partially supported by Ministry of Science and Technology, Taiwan, under grants MOST-105-2221- E-002-154-MY3, MOST-106-2923-E-002-012-MY3 and MOST-107-2634-F-002-011-.

## A Detailed Model Settings

### A.1 MLP Model Parameters

We implement our MLP correction generation model with Keras (Chollet, 2015). Table 8 shows the parameters. Models with this setting generally perform the best on the validation set across different combinations of features. The activation function is not applied at the output layer, so that the model output can fit better to the candidate embedding of the ground-truth correction.

When the validation accuracy does not increase for two consecutive epochs, the training process is terminated. In most cases, our model converges in 5 to 9 epochs. We choose the model with the highest validation accuracy for each feature combination to evaluate on the test set.

| Parameter | Value |
|---|---|
| Hidden layer size | 4096 |
| Number of hidden layers | 2 |
| Activation function | ReLU |
| Dropout rate | 0.2 |
| Cost function | cosine proximity |
| Optimizer | Adagrad |
| Initial learning rate | 0.01 |
| Batch size | 32 |

Table 8: Parameter settings of our MLP model.

### A.2 CWE Parameters

We use the publicly released implementation of CWE[4] to train the CWE+P model on the Chinese ClueWeb corpus. We set the embedding size to 400 and train for 20 iterations. All other hyperparameters are left default.

### A.3 Context2vec Parameters

We use the publicly available toolkit[5] to obtain the Context2vec context and target representation model. The training is also performed on the Chinese ClueWeb corpus. We set the number of units to 300 and train for 5 epochs.

### A.4 N-gram LM Settings

We fit a 5-gram language model with KenLM[6] on Chinese ClueWeb. The modified Kneser-Ney smoothing (Heafield et al., 2013) is applied to handle unseen n-grams.

### A.5 RNNLM Parameters

We use the Faster RNNLM toolkit[7], which speeds up the training process of the original RNNLM by using the Hierarchical Softmax (HS) or Noise Contrastive Estimation (NCE). We choose NCE because it gives better performance on our WUE validation set.

We process the ClueWeb corpus before training. The words whose frequency is less than 10 are replaced with a "<unk>" token. When testing, an OOV word is treated as "<unk>". We split 10% of the corpus for validation. The toolkit automatically adjusts the learning rate and early-stops the training process based on validation entropy. The hyperparameter settings are shown in Table 9.

| Hyperparameter | Value |
|---|---|
| Layer type | GRU |
| Layer size | 128 |
| Number of negative samples | 20 |

Table 9: Hyperparameter settings of our RNNLM.

## References

Xinxiong Chen, Lei Xu, Zhiyuan Liu, Maosong Sun, and Huan-Bo Luan. 2015. Joint learning of character and word embeddings. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 1236–1242.

[4]https://github.com/Leonard-Xu/CWE
[5]https://github.com/orenmel/context2vec
[6]https://github.com/kpu/kenlm
[7]https://github.com/yandex/faster-rnnlm

Shuk-Man Cheng, Chi-Hsin Yu, and Hsin-Hsi Chen. 2014. Chinese word ordering errors detection and correction for non-native chinese language learners. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 279–289. Dublin City University and Association for Computational Linguistics.

Shamil Chollampatt and Hwee Tou Ng. 2017. Connecting the dots: Towards human-level grammatical error correction. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 327–333.

Shamil Chollampatt, Duc Tam Hoang, and Hwee Tou Ng. 2016a. Adapting grammatical error correction based on the native language of writers with neural network joint models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1901–1911. Association for Computational Linguistics.

Shamil Chollampatt, Kaveh Taghipour, and Hwee Tou Ng. 2016b. Neural network translation models for grammatical error correction. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2768–2774.

François Chollet. 2015. Keras. `https://github.com/fchollet/keras`.

Daniel Dahlmeier and Hwee Tou Ng. 2011. Correcting semantic collocation errors with l1-induced paraphrases. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 107–117. Association for Computational Linguistics.

Robert Dale and Adam Kilgarriff. 2011. Helping our own: The hoo 2011 pilot shared task. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 242–249. Association for Computational Linguistics.

Robert Dale, Ilya Anisimoff, and George Narroway. 2012. Hoo 2012: A report on the preposition and determiner error correction shared task. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 54–62. Association for Computational Linguistics.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159.

Gabriel Fung, Maxime Debosschere, Dingmin Wang, Bo Li, Jia Zhu, and Kam-Fai Wong. 2017. Nlptea 2017 shared task – chinese spelling check. In *Proceedings of the 4th Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA 2017)*, pages 29–34, Taipei, Taiwan, December. Asian Federation of Natural Language Processing.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified kneser-ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 690–696. Association for Computational Linguistics.

Hen-Hsen Huang, Yen-Chi Shao, and Hsin-Hsi Chen. 2016. Chinese preposition selection for grammatical error diagnosis. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 888–899. The COLING 2016 Organizing Committee.

Lung-Hao Lee, Liang-Chih Yu, and Li-Ping Chang. 2015. Overview of the nlp-tea 2015 shared task for chinese grammatical error diagnosis. In *Proceedings of the 2nd Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA 2015)*, pages 1–6. Association for Computational Linguistics.

Lung-Hao Lee, Gaoqi Rao, Liang-Chih Yu, Endong Xun, Baolin Zhang, and Li-Ping Chang. 2016. Overview of nlp-tea 2016 shared task for chinese grammatical error diagnosis. In *Proceedings of the 3rd Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA 2016)*, pages 40–48. The COLING 2016 Organizing Committee, December.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning generic context embedding with bidirectional lstm. In *CoNLL*, pages 51–61.

Tomas Mikolov, Stefan Kombrink, Anoop Deoras, Lukar Burget, and Jan Cernocky. 2011. Rnnlm-recurrent neural network language modeling toolkit. In *Proceedings of the 2011 ASRU Workshop*, pages 196–201.

Tou Hwee Ng, Mei Siew Wu, Yuanbin Wu, Christian Hadiwinoto, and Joel Tetreault. 2013. The conll-2013 shared task on grammatical error correction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–12. Association for Computational Linguistics.

Tou Hwee Ng, Mei Siew Wu, Ted Briscoe, Christian Hadiwinoto, Hendy Raymond Susanto, and Christopher Bryant. 2014. The conll-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14. Association for Computational Linguistics.

Gaoqi Rao, Baolin Zhang, Endong Xun, and Lung-Hao Lee. 2017. Ijcnlp-2017 task 1: Chinese grammatical error diagnosis. In *Proceedings of the IJCNLP 2017, Shared Tasks*, pages 1–8, Taipei, Taiwan, December. Asian Federation of Natural Language Processing.

Yow-Ting Shiue and Hsin-Hsi Chen. 2016. Detecting word usage errors in chinese sentences for learning chinese as a foreign language. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 220–224. European Language Resources Association (ELRA), may.

Yow-Ting Shiue, Hen-Hsen Huang, and Hsin-Hsi Chen. 2017. Detection of chinese word usage errors for non-native chinese learners with bidirectional lstm. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 404–410.

Yuen-Hsien Tseng, Lung-Hao Lee, Li-Ping Chang, and Hsin-Hsi Chen. 2015. Introduction to sighan 2015 bake-off for chinese spelling check. *ACL-IJCNLP 2015*, page 32.

Shih-Hung Wu, Chao-Lin Liu, and Lung-Hao Lee. 2013. Chinese spelling check evaluation at sighan bake-off 2013. In *Proceedings of the 7th SIGHAN Workshop on Chinese Language Processing*, pages 35–42.

Chi-Hsin Yu and Hsin-Hsi Chen. 2012. Detecting word ordering errors in chinese sentences for learning chinese as a foreign language. In *Proceedings of COLING 2012*, pages 3003–3018. The COLING 2012 Organizing Committee.

Chi-Hsin Yu, Yi jie Tang, and Hsin-Hsi Chen. 2012. Development of a web-scale chinese word n-gram corpus with parts of speech information. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, pages 320–324. European Language Resources Association (ELRA), may.

Liang-Chih Yu, Lung-Hao Lee, and Li-Ping Chang. 2014a. Overview of grammatical error diagnosis for learning chinese as a foreign language. In *Proceedings of the 1st Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA 2014)*, pages 42–47.

Liang-Chih Yu, Lung-Hao Lee, Yuen-Hsien Tseng, Hsin-Hsi Chen, et al. 2014b. Overview of sighan 2014 bake-off for chinese spelling check. In *Proceedings of the 3rd CIPSSIGHAN Joint Conference on Chinese Language Processing (CLP'14)*, pages 126–132.