Chinese Word Ordering Errors Detection and Correction for Non-Native Chinese Language Learners

Shuk-Man Cheng, Chi-Hsin Yu, Hsin-Hsi Chen
Department of Computer Science and Information Engineering
National Taiwan University, Taipei, Taiwan
{smcheng,jsyu}@nlg.csie.ntu.edu.tw; hhchen@ntu.edu.tw

Abstract

Word Ordering Errors (WOEs) are the most frequent type of grammatical errors at sentence level for non-native Chinese language learners. Learners taking Chinese as a foreign language often place character(s) in the wrong places in sentences, and that results in wrong word(s) or ungrammatical sentences. Besides, there are no clear word boundaries in Chinese sentences. That makes WOEs detection and correction more challenging. In this paper, we propose methods to detect and correct WOEs in Chinese sentences. Conditional random fields (CRFs) based WOEs detection models identify the sentence segments containing WOEs. Segment point-wise mutual information (PMI), inter-segment PMI difference, language model, tag of the previous segment, and CRF bigram template are explored. Words in the segments containing WOEs are reordered to generate candidates that may have correct word orderings. Ranking SVM based models rank the candidates and suggests the most proper corrections. Training and testing sets are selected from HSK dynamic composition corpus created by Beijing Language and Culture University. Besides the HSK WOE dataset, Google Chinese Web 5-gram corpus is used to learn features for WOEs detection and correction. The best model achieves an accuracy of 0.834 for detecting WOEs in sentence segments. On the average, the correct word orderings are ranked 4.8 among 184.48 candidates.

1 Introduction

Detection and correction of grammatical errors are practical for many applications such as document editing and language learning. Non-native language learners usually encounter problems in learning a new foreign language and are prone to generate ungrammatical sentences. Sentences with various types of errors are written by language learners of different backgrounds. In the HSK corpus, which contains compositions of students from different countries who study Chinese in Beijing Language and Culture University (http://nlp.blcu.edu.cn/online-systems/hsk-language-lib-indexing-system.html), there are 35,884 errors at sentence level. The top 10 error types and their occurrences are listed below: Word Ordering Errors (WOE) (8,515), Missing Component (Adverb) (3,244), Missing Component (Predicate) (3,018), Grammatical Error ("Is … DE") (2,629), Missing Component (Subject) (2,405), Missing Component (Head Noun) (2364), Grammatical Error ("Is" sentence) (1,427), Redundant Component (Predicate) (1,130), Uncompleted Sentence (1,052), and Redundant Component (Adverb) (1,051). WOEs are the most frequent type of errors (Yu and Chen, 2012).

The types of WOEs in Chinese are different from those in English. A Chinese character has its own meaning in text, while individual characters are meaningless in English. Learners taking Chinese as a foreign language often place character(s) in the wrong places in sentences, and that results in wrong word(s) or ungrammatical sentences. Besides, there are no clear word boundaries in Chinese sentences.

Word segmentation is fundamental in Chinese language processing (Huang and Zhao, 2007). WOEs may result in wrong segmentation. That may make WOEs detection and correction more challenging.

This paper aims at identifying the positions of WOEs in the text written by non-native Chinese language learners, and proposes candidates to correct the errors. It is organized as follows. Section 2 surveys the related work. Section 3 gives an overview of the study. Section 4 introduces the dataset used for training and testing. Sections 5 and 6 propose models to detect and correct Chinese WOEs, respectively. Section 7 concludes this study and propose some future work.

2 Related Work

There are only a few researches on the topic of detection and correction of WOEs in Chinese language until now. We survey the related work from the four aspects: (1) grammatical errors made by non-native Chinese learners, (2) word ordering errors in Chinese language, (3) computer processing of grammatical errors in Chinese language, and (4) grammatical error correction in other languages.

Leacock et al. (2014) give thorough surveys in automated grammatical error detection for language learners. Error types, available corpora, evaluation methods, and approaches for different types of errors are specified. Several shared tasks on grammatical error correction in English have been organized in recent years, including HOO 2011 (Dale and Kilgarriff, 2011), HOO 2012 (Dale et al., 2012) and CoNLL 2013 (Ng et al., 2013). Different types of grammatical errors are focused: (1) HOO 2011: article and preposition errors, (2) HOO 2012: determiner and preposition errors, and (3) CoNLL 2013: article or determiner errors, preposition errors, noun number errors, verb form errors, and subject-verb agreement errors. In Chinese, spelling check evaluation was held at SIGHAN Bake-off 2013 (Wu et al., 2013). However, none of the above evaluations deals with word ordering errors.

Wang (2011) focuses on the Chinese teaching for native English-speaking students. He shows the most frequent grammatical errors made by foreigners are missing components, word orderings and sentence structures. One major learning problem of foreign learners is the influence of negative transfer of mother tongue. Lin (2011) studies the biased errors of word order in Chinese written by foreign students in the HSK corpus. Sun (2011) compares the word orderings between English and Chinese to figure out the differences in sentence structures. Yu and Chen (2012) propose classifiers to detect sentences containing WOEs, but they do not deal with where WOEs are and how to correct them.

Wagner et al. (2007) deal with common grammatical errors in English. They consider frequencies of POS n-grams and the outputs of parsers as features. Gamon et al. (2009) identify and correct errors made by non-native English writers. They first detect article and preposition errors, and then apply different techniques to correct each type of errors. Huang et al. (2010) propose a correction rule extraction model trained from 310,956 sets of erroneous and corrected pairwise sentences. Some studies related to word orderings are specific to the topic of pre-processing or post-processing of statistical machine translation, such as Galley and Manning (2008), Setiawan et al. (2009), and DeNero and Uszkoreit (2011).

The major contributions of this paper cover the following aspects: (1) application aspect: detecting and correcting a common type of Chinese written errors of foreign learners with HSK corpus; (2) language aspect: considering the effects of words and segments in Chinese sentences; and (3) resource aspect: exploring the feasibility of using a Chinese web n-gram corpus in WOE detection/correction.

3 Overview of a Chinese Word Ordering Detection and Correction System

Figure 1 sketches an overview of our Chinese WOE detection and correction system. It is composed of three major parts, including dataset preparation, WOE detection, and WOE correction. At first, a corpus is prepared. Sentences containing WOEs are selected from the corpus and corrected by two Chinese native speakers. This corpus will be used for training and testing. Then, a sentence is segmented into a sequence of words, and chunked into several segments based on punctuation marks. Regarding words and segments as fundamental units reduce the number of reordering and limit the reordering scope. The segments containing WOEs are identified by using CRF-based models. Finally, the candidates are generated by reordering and ranked by Ranking SVM-based models. To examine the performance of WOE correction, two datasets, C_{ans} and C_{sys}, consisting of error segments labelled by human and detected by our system, respectively, are employed.

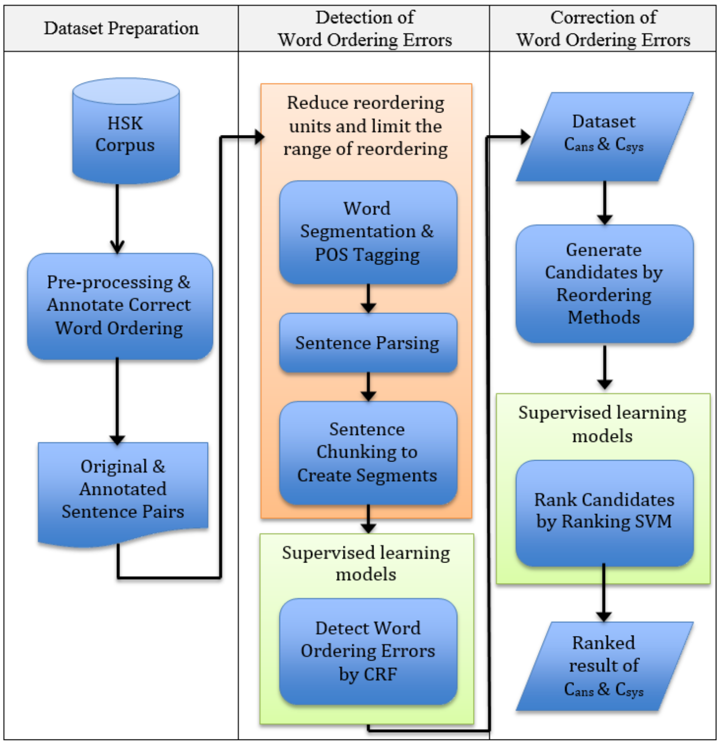
Dataset Preparation
Detection of Word Ordering Errors
Correction of Word Ordering Errors
HSK Corpus
Pre-processing & Annotate Correct Word Ordering
Original & Annotated Sentence Pairs
Reduce reordering units and limit the range of reordering
Word Segmentation & POS Tagging
Sentence Parsing
Sentence Chunking to Create Segments
Supervised learning models
Detect Word Ordering Errors by CRF
Dataset C_ans & C_sys
Generate Candidates by Reordering Methods
Supervised learning models
Rank Candidates by Ranking SVM
Ranked result of C_ans & C_sys

Figure 1: Overview of word ordering error detection and correction.

The example shown below demonstrates the major steps. This sentence is composed of three segments. The second segment contains a WOE, i.e., 今年夏天毕业了大学 (Graduated college this summer). The correct sentence should be 今年夏天大学毕业了 (Graduated from college this summer).

(1) Reduce the number of reordering units in a sentence by using word segmentation.

我	叫	王大安	，	今年	夏天	毕业	了	大学	，	现在	找	工作	。

(I / am /Wang Daan/ , /this /summer /graduated/le /college /, /now /look for/job /.)

(2) Chunk a sentence into segments by punctuation marks.

我	叫	王大安	，	今年	夏天	毕业	了	大学	，	现在	找	工作	。

(3) Detect the possible segments containing WOEs in a sentence by CRF-based methods.

我	叫	王大安	，	今年	夏天	毕业	了	大学	，	现在	找	工作	。

(4) Reorder words in an erroneous segment and generate candidates.

我	叫	王大安	，	今年	夏天	毕业	大学	了	，	现在	找	工作	。

…

我	叫	王大安	，	今年	夏天	大学	毕业	了	，	现在	找	工作	。

(5) Rank candidates and suggest correct word ordering by Ranking SVM-based methods.

我	叫	王大安	，	今年	夏天	大学	毕业	了	，	现在	找	工作	。

…

4 A Word Ordering Errors (WOEs) Corpus

HSK dynamic composition corpus created by Beijing Language and Culture University is adopted. It contains the Chinese composition articles written by non-native Chinese learners. There are 11,569 articles and 4.24 million characters in 29 composition topics. Composition articles are scanned into text and annotated with tags of error types ranging from character level, word level, sentence level, to discourse level. There are 35,884 errors at sentence level, and WOEs are the most frequent type at this level. Total 8,515 sentences are annotated with WOEs. We filter out sentences with multiple error types and remove duplicate sentences. Total 1,150 error sentences with WOEs remain for this study.

Two Chinese native speakers are asked to correct the 1,150 sentences. Only reordering operation is allowed during correction. A dataset composed of 1,150 sets of original sentence S and its two corrections A1 and A2 is formed for training and testing in the experiments. A1 may be different from A2. The following shows an example. Without context, either A1 or A2 is acceptable.

S: 她我们兄弟姊妹鼓励学音乐和外语。
(She we encouraged to study music and foreign languages.)
A1: 我们兄弟姊妹鼓励她学音乐和外语。
(We encouraged her to study music and foreign languages.)
A2: 她鼓励我们兄弟姊妹学音乐和外语。
(She encouraged us to study music and foreign languages.)

In some cases, A1 and/or A2 may be equal to S. That is, the annotators may think S is correct. That may happen when context is not available. Finally, 327 of 1,150 sets contain different corrections. Both A1 and A2 are equal to S in 27 sets. Total 47 sentences corrected by one annotator are the same as the original sentences, and total 65 sentences corrected by another annotator are the same as the original sentences. This corpus is available at http://nlg.csie.ntu.edu.tw/nlpresource/woe_corpus/.

Figure 2 shows the Damerau Levenshtein distance between the original sentences S and the corrections A1 and A2. It counts the minimum number of operations needed to transform a source string into a target one. Here the operation is the transposition of two adjacent characters. Total 823 sets of A1 and A2 have a distance of 0. It means 71.5% of sentences have the same corrections by the two Chinese native speakers. The distances between S and A1 are similar to those between S and A2. Total 850 sets of original sentences and the corrections have a distance below 10 characters and 1,014 sets of sentences have a distance below 20. We can also observe that the number of sentences with even distances is larger than that of sentences with odd distances because most of the Chinese words are composed of two characters.

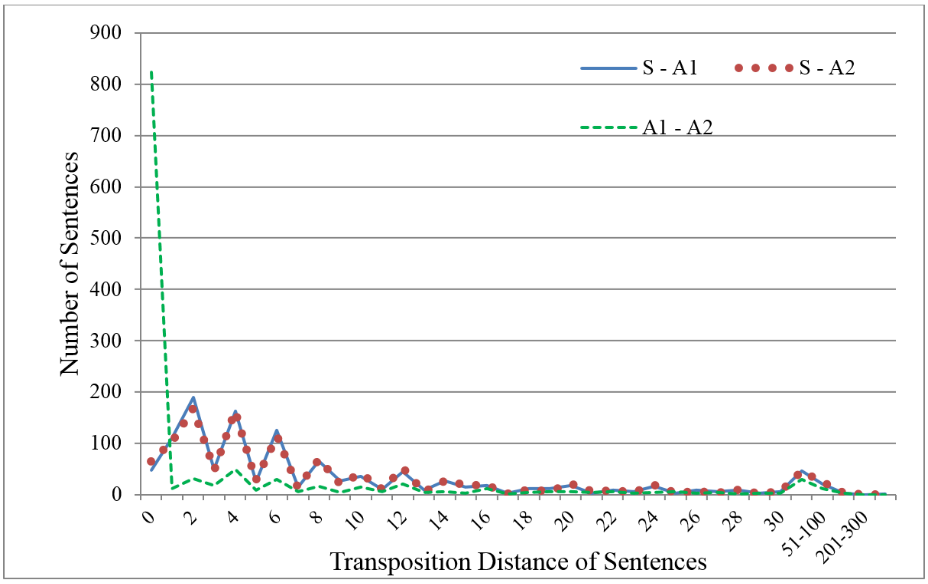

Figure 2: Transposition distance among the original sentences and two corrections.

5 Detection of Word Ordering Errors

This section first defines the fundamental units for error detection, then introduces the error detection models along with their features, and finally presents and discusses the experimental results.

5.1 Fundamental Units for Reordering

Permutation is an intuitive way to find out the correct orderings, but its cost is very high. Unrestrictive permutation will generate too many candidates to be acceptable in computation time. What units to be reordered in what range under what condition has be considered. Chinese is different from English in that characters are the smallest meaningful units, and there are no clear word boundaries. Computation cost and segmentation performance is a trade-off to select character or word as a reordering unit. On the one hand, using words as the reordering units will reduce the number of candidates generated. On the other hand, word segmentation results will affect the performance of WOE detection and correction. The following two examples show that reordering the words cannot generate the correct answers. In these two examples, a word in the original sentence (S) is segmented into two words in the correct sentence (A). These words are underlined. Because a word is regarded as a unit for reordering, the correct sentence cannot be generated by word reordering only in these two cases.

S: 他 / 教给 / 学生们 / 英语 /。
(He / teach to / students / English / .)
A: 他 / 给 / 学生们 / 教 / 英语 / 。
(He / for / students / teach / English / .)
S: 最近 / 我 / 开始 / 学 / 中国 / 的 / 做菜。
(Recently / I / start to / learn / China / 's / cooking cuisine.)
A: 最近 / 我 / 开始 / 学 / 做 / 中国 / 的 / 菜。
(Recently / I / start to / learn / cooking / China / 's /cuisine.)

Total 76 sets of sentences belong to such cases. They occupy 6% of the experimental dataset. Considering the benefits of words, we still adopt words as reordering units in the following experiments.

To prevent reordering all the words in the original sentences, we further divide a sentence into segments based on comma, caesura mark, semi-colon, colon, exclamation mark, question mark, and full stop. Sentence segments containing WOEs will be detected and words will be reordered within the segments to generate the candidates for correction. In our dataset, there are only 31 sets of sentences (i.e., 2.7%) with WOEs across segments. The following shows two examples. The underlined words are moved to other segments.

S: 其实，我还是做事情的时候，不怎么老实。
(In fact, when I am still working, I am not honest.)
A: 其实，我做事情的时候，还是不怎么老实。
(In fact, when I am working, I am still not honest.)
S: 所以有绝对的导游工作经验，无须再培训。
(Therefore we have absolute guide work experience, we do not need retraining.)
A: 有绝对的导游工作经验，所以无须再培训。
(We have absolute guide work experience, therefore we do not need retraining.)

In summary, the upper bound of the correction performance would be 91.3%. That is, 6%+2.7% of sentences cannot be resolved.

5.2 Word Ordering Errors Detection Models

Conditional random fields (CRFs) (Lafferty, 2001) are used to implement the WOE detection in sentence segments. Segments with WOEs are labelled with answer tags before training. The original sentence S written by non-native Chinese learner is compared with the annotated correct sentence A. Characters are compared from the start and the end of sentences, respectively. The positions are marked ERR_{start} and ERR_{end} once the characters are different. All words within ERR_{start} and ERR_{end} are marked ERR_{range}. The longest common subsequence (LCS) within ERR_{range} of S and ERR_{range} of A are excluded from ERR_{range} and the remaining words are marked ERR_{words}. Figure 3 shows an example. We use BIO encoding (Ramshaw and Marcus, 1995) to label segments with WOEs. Segments containing words in *ERR_{words}* are defined to be segments with WOEs. The leftmost segment with WOEs is tagged B, and the following segment with WOEs are tagged I. Those segments without WOEs are tagged O.

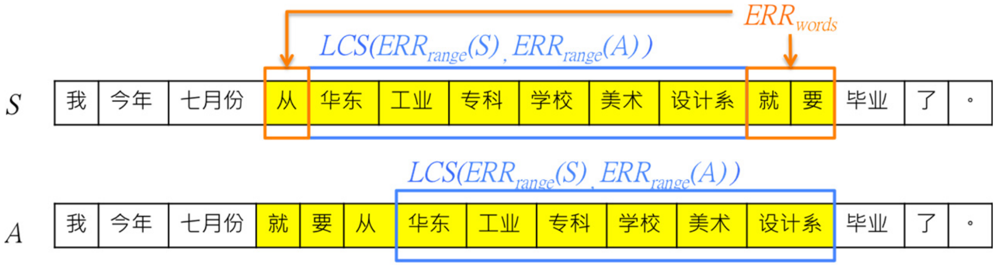
ERR_words
LCS(ERR_range(S) , ERR_range(A))
S
我 今年 七月份 从 华东 工业 专科 学校 美术 设计系 就 要 毕业 了 。
LCS(ERR_range(S) , ERR_range(A))
A
我 今年 七月份 就 要 从 华东 工业 专科 学校 美术 设计系 毕业 了 。

Figure 3: An example for ERR_{range} and ERR_{words}.

Table 1 lists the distribution of B, I and O segments. Recall that two Chinese native speakers are asked to correct the 1,150 sentences, thus we have two sets of B-I-O tagging.

Tagging→	B Tag		I Tag		O Tag		Total
Statistics→	#Segments	Percentage	#Segments	Percentage	#Segments	Percentage	Segments
Annotator 1	1111	40.6%	53	1.9%	1572	57.5%	2736
Annotator 2	1097	40.1%	59	2.2%	1580	57.7%	2736

Table 1: Distribution of B, I, and O segments.

Five features are proposed as follows for CRF training. Google Chinese Web 5-gram corpus (Liu, Yang and Lin, 2010) is adopted to get the frequencies of Chinese words for f_{PMI}, f_{Diff} and f_{LM}.

(1) Segment Pointwise Mutual Information (f_{PMI})

$PMI(Seg_i)$ defined below measures the coherence of a segment Seg_i by calculating PMI of all word bigrams in Seg_i. To avoid the bias from different lengths, the sum of PMI of all word bigrams is divided by n-1 for normalization, where n denotes the segment length. The segment PMI values are partitioned into intervals by equal frequency discretization. Feature f_{PMI} of the segment Seg_i reflects the label of the interval to which $PMI(Seg_i)$ belongs.

$$PMI(Seg_i) = \frac{1}{n-1} \sum_{k=1}^{n-1} log \frac{P(w_k, w_{k+1})}{P(w_k)P(w_{k+1})}$$

(2) Inter-segment PMI Difference (f_{Diff})

Feature f_{Diff} captures the PMI difference between two segments $Seg_{j\text{-}1}$ and Seg_j. It aims to measure the coherence between segments. The feature setting is also based on equal frequency discretization.

(3) Language Model (f_{LM})

Feature f_{LM} uses bigram language model to measure the log probability of the words in a segment defined below. Labels of interval are also determined by equal frequency discretization.

$$LM(Seg_i) = log\ \left[P(w_1) \prod_{k=1}^{n-1} \frac{P(w_k, w_{k+1})}{P(w_k)}\right] = log\,P(w_1) + \sum_{k=1}^{n-1} log \frac{P(w_k, w_{k+1})}{P(w_k)}$$

(4) Tag of the previous segment (f_{Tag})

Feature f_{Tag} reflects the tag B, I or O of the previous segment.

(5) CRF bigram template (f_B)

Feature f_B is a bigram template given by SGD-CRF tool[1]. Bigram template combines the tags of the previous segment and current segment, and generates T*T*N feature functions, where T is number of tags and N is number of strings expanded with a macro.

[1] http://leon.bottou.org/projects/sgd

5.3 Results and Discussion

WOE detection models will annotate the segments of a sentence with labels B, I or O. These labels will determine which segments may contain WOEs. In the experiments, we use 5-fold cross-validation to evaluate the proposed models. Performance for detecting WOEs is measured at the segment and the sentence levels, respectively. The metrics at the segment level are defined as follows. Here set notation is adopted. The symbol $|S|$ denotes the number of elements in the set S which is derived by the logical formula after vertical bar. $TAG_{pred}(SEG)$ and $TAG_{ans}(SEG)$ mean the labels of segment SEG tagged by WOE detection model and human, respectively. The symbol m denotes total number of segments in the test set.

$$Accuracy = \frac{|\{SEG \mid TAG_{pred}(SEG) = TAG_{ans}(SEG)\}|}{m}$$

$$Recall = \frac{|\{SEG \mid TAG_{pred}(SEG) \in (\mathrm{B,I}) \cap TAG_{ans}(SEG) \in (\mathrm{B,I})\}|}{|\{SEG \mid TAG_{ans}(SEG) \in (\mathrm{B,I})\}|}$$

$$Precision = \frac{|\{SEG \mid TAG_{pred}(SEG) \in (\mathrm{B,I}) \cap TAG_{ans}(SEG) \in (\mathrm{B,I})\}|}{|\{SEG \mid TAG_{pred}(SEG) \in (\mathrm{B,I})\}|}$$

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

The metrics at the sentence level are defined as follows:

$$Accuracy = \frac{|\{SENT \mid \forall\, SEG \in SENT,\ TAG_{pred}(SEG) = TAG_{ans}(SEG)\}|}{1150}$$

$$Correctable\ Rate$$
$$= \frac{|\{SENT \mid\ \nexists\, SEG \in SENT,\ TAG_{pred}(SEG) = \mathrm{O}\ AND\ TAG_{ans}(SEG) \in (\mathrm{B,I})\}|}{1150}$$

Accuracy and F1-score measure whether the models can find out segments with WOEs. *Correctable Rate* of sentences measures whether it is possible that the candidates of the correct word order can be generated by the WOE correction models. If a segment without WOEs is misjudged to be erroneous, the word order still has a chance to be kept by the WOE correction models. However, if a segment with WOEs is misjudged to be correct, words in the misjudged segment will not be reordered in the correction part because the error correction module is not triggered. A sentence is said to be "correctable" if no segments in it are misjudged as "correct". The ratio of the "correctable" sentences is considered as a metric at the sentence level.

Table 2 shows the performance of WOE detection. Five models are compared. We regard tagging all the segments with the labels B and O respectively as two baselines. Clearly, the recall at the segment level and the correctable rate at the sentence level are 1 by the all-tag-B baseline. However, its accuracy at the segment and the sentence levels are low. The all-tag-O baseline has better accuracy at the segment level than the all-tag-B baseline, but has very bad F1-score, i.e., 0. The proposed models are much better than the two baselines. Among the feature combinations, $f_{PMI} f_{Diff} f_{Tag} f_B$ show the best performance. The accuracy at the segment level is 0.834, and the correctable rate is 0.883. The best detection result will be sent for further correction.

Model	Segment				Sentence	
	Accuracy	Recall	Precision	F1-Score	Accuracy	Correctable Rate
Baseline (*all tag B*)	0.404	1.000	0.424	0.595	0.271	1.000
Baseline (*all tag O*)	0.576	0.000	0.000	0.000	0.074	0.074
$f_{PMI} f_{LM} f_{Tag} f_B$	0.830	0.781	0.802	0.791	0.787	0.862
$f_{PMI} f_{Diff} f_{Tag} f_B$	**0.834**	**0.795**	0.805	**0.800**	**0.788**	**0.883**
$f_{PMI} f_{Diff} f_{LM} f_{Tag} f_B$	0.831	0.769	**0.823**	0.795	0.777	0.850

Table 2: Performance of word ordering error detection

6 Correction of Word Ordering Errors

This section deals with generating and ranking candidates to correct WOEs. Two datasets, C_{ans} and C_{sys}, are explored in the experiments. We evaluate the optimal performance of the WOE correction models with the C_{ans} dataset, and evaluate WOE detection and correction together with the C_{sys} dataset.

6.1 Candidate Generation

Instead of direct permutation, we consider three strategies shown as follows to correct the error sentences. The complexity of generating candidates by permutation is O(n!). The complexity of using these three strategies decreases to O(n^2).

(1) Reorder single unit (R_{single})

R_{single} strategy reorders only one reordering unit (i.e., a word) to n-1 positions within a segment containing n words. Total $(n\text{-}1)^2$ candidates can be generated by this strategy. The following shows an example.

S: 今天 / 学校 / 去
(Today / school / go to)
A: 今天 / 去 / 学校
(Today / go to / school)

(2) Reorder bi-word ($R_{bi\text{-}word}$)

$R_{bi\text{-}word}$ is similar to R_{single}, but two reordering units are merged into a new word before reordering. Because n-1 bi-words can be generated in a segment and n-2 positions are available for each merged bi-word, (n-1)(n-2) candidates are generated by $R_{bi\text{-}word}$. The following shows an example.

S: 早/就/一家/公司/找/我/工作
(before / already / one / company / employ / me / work)
A: 一家/公司/早/就/找/我/工作
(one / company / before / already / employ / me / work)

(3) Reorder tri-word ($R_{tri\text{-}word}$)

$R_{tri\text{-}word}$ works similarly to $R_{bi\text{-}word,}$ but three reordering units are merged before reordering. Total (n-2)(n-3) candidates are generated by $R_{tri\text{-}word}$. The following shows an example.

S: 我/需要/工作/的/经验/在/您/的/公司。
(I / need / working / (de) / experience / in / your / (de) / company.)
A: 我/需要/在/您/的/公司/工作/的/经验。
(I / need / in / your / (de) / company / working / (de) / experience.)

Table 3 shows the recall rate of each candidate generation strategy. With the C_{ans} dataset, correct word ordering can be generated for 85.8% of the original sentences by fusing R_{single}, $R_{bi\text{-}word}$ and $R_{tri\text{-}word}$. The candidates generated by using the C_{sys} dataset cover 69.7% of the correct word orderings. The difference would probably be due to the error propagation of word ordering error detection specified in Section 5.3. Furthermore, 6% of correct word orderings are unable to be generated by using the reordering units due to the word segmentation issue as mentioned in Section 5.1. We can also find that 72.3% of sentences with WOEs can be corrected by the R_{single} strategy using the C_{ans} dataset. It means most of the WOEs made by non-native Chinese learners can be corrected by moving only one word.

Strategy\Dataset	C_{ans}	C_{sys}
R_{single}	0.723	0.577
$R_{bi\text{-}word}$	0.365	0.308
$R_{tri\text{-}word}$	0.239	0.217
$R_{single} \cup R_{bi\text{-}word} \cup R_{tri\text{-}word}$	**0.858**	**0.697**

Table 3: Recall of candidate generation strategies

6.2 Candidate Ranking

We use Ranking SVM (Joachims, 2002) for candidates ranking. Because WOEs may produce abnormal POS sequence, POS bigrams and POS trigrams are considered as features for Ranking SVM. We use a k-tuple feature vector for each candidate sentence, where k is the number of features. In each dimension, binary weight is assigned: 1 if the feature exists in a candidate, and 0 otherwise. Score for each candidate is assigned by a binary classifier: 1 if the candidate is the same as either of the annotated corrections, and 0 otherwise.

6.3 Results and Discussion

Mean Reciprocal Rank (MRR) defined below is used for performance evaluation. The reciprocal rank is the multiplicative inverse of the rank of the first correct answer. MRR is the mean of reciprocal rank for all sentences *S*, value from 0 to 1. The larger MRR means the correct answer more closes to the top ranking.

$$MRR = \frac{1}{|S|}\sum_{i=1}^{|S|}\frac{1}{rank_1}$$

Percentage of answers having rank 1 is another metric. Five-fold cross-validation is used for training and testing. In the C_{ans} and C_{sys} datasets, 182.03 and 184.48 candidates are proposed by the approach of fusing the results of R_{single}, $R_{bi\text{-}word}$, and $R_{tri\text{-}word}$ on the average. Experimental results are listed in Table 4. The proposed candidate ranking method achieves an MRR of 0.270 in the C_{ans} dataset. It means the correct candidates are ranked 3.7 on the average. In contrast, the MRR by using the C_{sys} dataset is 0.208. It means the correct candidates are ranked 4.8 on the average when error detection and correction are performed in pipelining.

Metric\Dataset	C_{ans}	C_{sys}
MRR	0.270	**0.208**
% of rank 1	0.195	**0.144**

Table 4: Performance of candidate ranking

There are some major types of errors shown as follows in WOE correction.

(1) Word ordering errors across segments
Section 5.1 mentions there are 31 sets of sentences (i.e., 2.7%) with WOEs across segments. Our algorithm cannot capture such kinds of sentences.

(2) Propagation errors from candidate generation
Table 3 shows the recall of word ordering error detection using the C_{ans} dataset is 0.858. Besides, 6% of sentences mentioned in Section 5.1 cannot be reordered to correct word ordering due to word segmentation issue.

(3) Limitation of our models
In the fused n-gram models, only one n-gram can be moved. It reduces the number of candidates to be generated, but some types of reorderings are missed. An example is shown as follows. The 2-gram 出生 / 于 (was born in) and the unigram 于 (on) have to be exchanged.

S：我 / 出生 / 于 / 1968 年 10 月 25 日 / 在 / 维也纳。
(I / was born / in / 25 October 1968 / on / Vienna.)
A：我 / 在 / 1968 年 10 月 25 日 / 出生 / 于 / 维也纳。
(I / on / 25 October 1968 / was born / in / Vienna.)

7 Conclusion

In this paper, we consider words as the reordering units in WOE detection and correction. Sentences are chunked into segments based on punctuation marks and the CRF technique is used to detect segments that possibly contain WOEs. The best error detection model achieves an accuracy of 0.834. Three reordering strategies are further proposed to generate candidates with correct word ordering and reduce the numerous number of candidates generated by permutation. If the segments containing WOEs are known, 85.8% of correct sentences can be generated by our approach. Finally, Ranking SVM orders the generated candidates based on POS bigrams and POS trigrams features, and achieves an MRR of 0.270 when all erroneous segments are given and an MRR of 0.208 when both detection and correction modules are considered.

Using words as the reordering unit reduces the cost to generate numerous candidates, but 6% of sentences are unable to reorder due to the word segmentation issue. How to balance the trade-off has to be investigated further. In the candidate ranking, selection of proper weights for POS bigram and trigram features may improve the ranking performance. Since the corpus of WOEs in Chinese is still in a limited size, expanding the related corpus for further research is also indispensable.

Acknowledgements

This research was partially supported by National Taiwan University and Ministry of Science and Technology, Taiwan under grants 103R890858, 101-2221-E-002-195-MY3 and 102-2221-E-002-103-MY3. We are also very thankful to the anonymous reviewers for their helpful comments to revise this paper.

References

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 7th Workshop on the Innovative Use of NLP for Building Educational Applications, pages 54–62, Montre´al, Canada.

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* (*ENLG*), pages 242–249, Nancy, France.

John DeNero and Jakob Uszkoreit. 2011. Inducing Sentence Structure from Parallel Corpora for Reordering. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, Edinburgh, Scotland, UK.

Michel Galley and Christopher D. Manning. 2008. A Simple and Effective Hierarchical Phrase Reordering Model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 848–856, Honolulu.

Michael Gamon, Claudia Leacock, Chris Brockett, William B. Dolan, Jianfeng Gao, Dmitriy Belenko, and Alexandre Klementiev. 2009. Using Statistical Techniques and Web Search to Correct ESL Errors. *CALICO Journal*, 26(3):491–511.

An-Ta Huang, Tsung-Ting Kuo, Ying-Chun Lai, and Shou-De Lin. 2010. Discovering Correction Rules for Auto Editing. *Computational Linguistics and Chinese Language Processing*, 15(3-4):219-236.

Chang-ning Huang and Hai Zhao. 2007. Chinese Word Segmentation: A Decade Review. *Journal of Chinese Information Processing*, 21(3):8-19.

Thorsten Joachims. 2002. Optimizing Search Engines using Clickthrough Data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133-142, Edmonton, Alberta, Canada.

John Lafferty, Andrew McCallum, and Fernando C.N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labelling Sequence Data. In *Proceedings of the 18th International Conference on Machine Learning* (*ICML 2001*), pages 282-289, San Francisco, CA, USA.

Claudia Leacock, Martin Chodorow, Michael Gamon, and Joel Tetreault. 2014. *Automated Grammatical Error Detection for Language Learners*. 2nd Edition. Morgan and Claypool Publishers.

Jia-Na Lin. 2011. *Analysis on the Biased Errors of Word Order in Written Expression of Foreign Students*. Master Thesis. Soochow University.

Fang Liu, Meng Yang, Dekang Lin. 2010. *Chinese Web 5-gram Version 1*. Linguistic Data Consortium, Philadelphia. http://catalog.ldc.upenn.edu/LDC2010T06.

Hwee Tou Ng, Siew Mei 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, Sofia, Bulgaria.

Lance A. Ramshaw and Mitchell P. Marcus. 1995. Text Chunking Using Transformation-based Learning. In *Proceedings of Third Workshop on Very Large Corpora*. Pages 82-94.

Hendra Setiawan, Min-Yen Kan, Haizhou Li, and Philip Resnik. 2009. Topological Ordering of Function Words in Hierarchical Phrase-based Translation. In *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pages 324–332, Suntec, Singapore.

Li-Li Sun. 2011. *Comparison of Chinese and English Word Ordering and Suggestion of Chinese Teaching for Foreign Learners*. Master Thesis. Heilongjiang University.

Joachim Wagner, Jennifer Foster, and Josef van Genabith. 2007. A Comparative Evaluation of Deep and Shallow Approaches to the Automatic Detection of Common Grammatical Errors. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 112–121, Prague, Czech Republic.

Zhuo Wang. 2011. *A Study on the Teaching of Unique Syntactic Pattern in Modern Chinese for Native English-Speaking Students*. Master Thesis. Northeast Normal University.

Shih-Hung Wu, Chao-Lin Liu, and Lung-Hao Lee. 2013. Chinese Spelling Check Evaluation at SIGHAN Bake-off 2013. In *Proceedings of the Seventh SIGHAN Workshop on Chinese Language Processing* (SIGHAN-7), pages 35–42, Nagoya, Japan.

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 the 24th International Conference on Computational Linguistics*, pages 3003-3018, Mumbai, India.