Detecting Word Ordering Errors in Chinese Sentences for Learning Chinese as a Foreign Language

Chi-Hsin Yu, Hsin-Hsi Chen
Department of Computer Science and Information Engineering, National Taiwan University
#1, Sec.4, Roosevelt Road, Taipei, 10617 Taiwan
jsyu@nlg.csie.ntu.edu.tw, hhchen@ntu.edu.tw

ABSTRACT

Automatic detection of sentence errors is an important NLP task and is valuable to assist foreign language learners. In this paper, we investigate the problem of word ordering errors in Chinese sentences and propose classifiers to detect this type of errors. Word n-gram features in Google Chinese Web 5-gram corpus and ClueWeb09 corpus, and POS features in the Chinese POS-tagged ClueWeb09 corpus are adopted in the classifiers. The experimental results show that integrating syntactic features, web corpus features and perturbation features are useful for word ordering error detection, and the proposed classifier achieves 71.64% accuracy in the experimental datasets.

協助非中文母語學習者偵測中文句子語序錯誤

自動偵測句子錯誤是自然語言處理研究一項重要議題，對於協助外語學習者很有價值。在這篇論文中，我們研究中文句子語序錯誤的問題，並提出分類器來偵測這種類型的錯誤。在分類器中我們使用的特徵包括：Google 中文網路 5-gram 語料庫、與 ClueWeb09 語料庫的中文詞彙 n-grams及中文詞性標注特徵。實驗結果顯示，整合語法特徵、網路語料庫特徵、及擾動特徵對偵測中文語序錯誤有幫助。在實驗所用的資料集中，合併使用這些特徵所得的分類器效能可達 71.64%。

KEYWORDS : ClueWeb09, computer-aided language learning, HSK corpus, word ordering error detection
KEYWORDS IN L_2 : ClueWeb09, 電腦輔助語言學習, HSK 語料庫, 語序錯誤偵測

Proceedings of COLING 2012: Technical Papers, pages 3003–3018,
COLING 2012, Mumbai, December 2012.

1 Introduction

Non-native language learners usually encounter problems in learning a new foreign language and are prone to generate ungrammatical sentences. NLP systems that detect and correct grammatical errors are important and invaluable to language learners. Error detection, which tells if there exists a special type of errors in a given sentence, is the first step for this kind of applications (Leacock, Chodorow, Gamon, & Tetreault, 2010).

Sentences with various types of errors are written by language learners of different backgrounds. The distribution of errors varies across different learner groups. For example, the most frequent error for native English learners is missing comma after introductory element, while the most frequent error for ESL (English as a Second Language) students is comma splice, which joins two sentences by using comma instead of a conjunction (Leacock et al., 2010). In addition, learners may generate sentences with multiple types of errors. This complicates the error detection. To simplify the problem and evaluate the performance of the proposed methods, researchers usually detect one type of error at a time, such as common grammatical errors (Islam & Inkpen, 2011; Wagner, Foster & Genabith, 2007) and prenominal adjective ordering errors (Lin et al., 2009; Malouf, 2000).

A training dataset is indispensable for learning an error detection system. In English, there are many well-established corpora for this purpose such as Cambridge Learner Corpus (CLS). In Chinese, Beijing Language and Culture University built an HSK dynamic composition corpus (动态/dynamic 作文/composition 语料库/corpus) and announced a publicly accessible online system to query this corpus[1]. The HSK corpus contains Chinese compositions written by Chinese language learners in the university. The students' compositions were collected and annotated with different error types. In this paper, we will deal with the word ordering errors in the HSK corpus.

This paper is organized as follows. In Section 2, we survey the related work. In Section 3, we introduce the HSK corpus and specify the type of word ordering errors in Chinese. In Section 4, we describe a web-scale Chinese POS-tagged dataset developed from the ClueWeb09 dataset[2] for this study. In Section 5, we present our feature extraction approaches in detail. Section 6 specifies the experimental setups, and Section 7 presents and analyzes the results. The uses of POS bigrams in the detection of different error types are discussed. Finally, we conclude our study.

2 Related Work

Word ordering errors (WOEs) are defined to be the cases where words are placed in the wrong places in sentences. Word ordering error detection is not only useful for language learning, but also beneficial for many applications such as machine translation. For example, we can use a word ordering error detection module to assess the quality of the machine generated sentences. Lee et al. (2007) employ machine translated sentences as training data to rank the fluency of sentences written by non-native learners in English. Their approaches result in accuracy 76.2%.

[1] http://202.112.195.192:8060/hsk/login.asp ; accessd 2012/08/14.
[2] http://lemurproject.org/clueweb09/ ; accessd 2012/08/14.

Leacock et al. (2010) give thorough surveys in automated grammatical error detection for language learners (native and second language learners). Error types, available corpora, evaluation methods, and approaches for different types of errors are specified in the book.

Several approaches have been proposed to detect English grammatical errors (Chodorow & Leacock, 2000) and Japanese Learners' English Spoken Data (Izumi, Uchimoto, Saiga, Supnithi, & Isahara, 2003). Chodorow and Leacock's approach is tested only on 20 specific English words. 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. Their classification accuracy with decision tree is 66.0%. 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. Word information such as heads of noun phrases and verb phrases, and POS n-grams in a given context is adopted for error detection. A POS context is defined to be the left and the right of a given preposition. The detection performance varies across different error types. A language model is trained on English Gigaword corpus (Linguistic Data Consortium [LDC], 2003) for error correction. The corrected version whose score is higher than a threshold is proposed as the result.

In addition to newswire text data, a large scale web corpus is also explored. Bergsma, Lin, and Goebel, (2009) adopt Google English Web 1T 5-gram corpus (Brants & Franz, 2006) to compute the frequencies of word n-grams and achieve 95.7% in spelling correction and 75.4% accuracy in preposition selection. Islam and Inkpen (2010) adopt the same corpus for unsupervised preposition error correction. An improved version of Google Web 1T 5-gram corpus called Google N-gram V2 (Lin et al., 2009; Lin et al., 2010) is constructed by adding POS information. Google N-gram V2 contains POS sequence information for each word N-gram pattern. Consider the following unigram example. A word "files" occurs as NNS 611,646 times among all its occurrences:

files 1643568 NNS|611646 VBZ|1031992

Lin et al. (2009) present tools for processing Google N-gram V2 and report tasks that can be benefited by using this corpus, such as base noun phrase parsing and prenominal adjective ordering.

The above research shows that features extracted from English POS-tagged Web corpus are useful. In this paper, we will explore the usefulness of a Chinese POS-tagged web corpus in the application of word ordering error detection for learning Chinese as a foreign language. In addition to the traditional POS and parsing features extracted from this corpus, we will study the effects of different Chinese segmentation methods and perturbation of terms in training the error detector.

The major contributions of this paper cover the following four aspects: (1) application aspect: detecting a common type of Chinese written errors of foreign learners with HSK corpus; (2) language aspect: considering the effects of character-based (without segmentation) and word-based (with segmentation) features in Chinese sentences; (3) resource aspect: exploring the feasibility of using a web-scale word and POS-tagged corpus; and (4) technology aspect: exploring the syntactic features and the web corpus features, and comparing the effects of training corpora by foreign learners and native writers.

3 Word Ordering Errors in Chinese

This section introduces the HSK corpus in our study, and specifies the type of word ordering errors in Chinese.

3.1 HSK Corpus

HSK corpus was built in a project led by Cui Xiliang from 1992 to 2005. This corpus contains 4.24 million characters and 2.83 million words in 11,569 students' compositions. Those students come from different countries to study Chinese in Beijing Language and Culture University. Their compositions are scanned, translated to texts, and annotated with different error types. This corpus also contains students' metadata such as age, country, and language skill level. Since its announcement in 2006, this corpus inspires many research efforts in different fields such as Chinese language teaching and learning.

In the HSK corpus, total 46 error types are labeled. The errors range from character level, word level, sentence level, to discourse level. For some error types, such as missing word error, correct answers are also annotated, so that they can be employed to investigate error detection and correction problems.

3.2 Word Ordering Errors in Chinese

The types of word ordering errors (WOEs) in Chinese are different from that in English. In English, characters are meaningless, while each Chinese character has its own meaning in context. 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. In the HSK corpus, there are 35,884 errors at sentence level, and WOEs are the most frequent type of errors at this level. Table 1 lists the top 10 error types on sentence level in HSK. They belong to different error categories defined in HSK.

#	Error Category	Error Type	Count
1	Other Error Types	Word ordering error	8,515
2	Missing Component	Adverb	3,244
3	Missing Component	Predicate	3,018
4	Grammatical Error	"Is (是)… DE(的)" sentence	2,629
5	Missing Component	Subject	2,405
6	Missing Component	Head Noun	2,364
7	Grammatical Error	"Is (是)" sentence	1,427
8	Redundant Component	Predicate	1,130
9	Other Error Types	Uncompleted sentence	1,052
10	Redundant Component	Adverb	1,051

TABLE 1 – Top 10 types of sentence level errors in HSK

In the HSK website, 8,515 sentences contain WOEs. We collect these sentences and sample a subset of 200 sentences for deep analysis. In this subset, we further classify these WOEs into five categories, including adverb ordering error, subject/verb/object ordering error, prepositional phrase (PP) position error, prenominal adjective ordering error, and others. C*overt error*

sentences (Carl, 1998), which can be interpreted correctly in some specific way, belong to the "others" category. Table 2 lists an example for each category along with its distribution.

Adverb ordering error (35.0%)	也 她 很 关心 also she (is) very concerned (她也很关心)
Subject/verb/object ordering error (32.0%)	就这样 我 休学 大学 来 中国 了 therefore I drop out university (and) come to China (就这样我大学 休学 来中国了)
PP position error (20.5%)	我 留学 在 贵国 I (am) studying in your country (我 在贵国 留学)
Prenominal adjective ordering error (6.0%)	我 遇到了 才貌双全的 一位 女人 I meet beautiful and wise a woman (我遇到了一位 才貌双全的女人)

TABLE 2 – Categories of word ordering errors

The word(s) in an erroneous position are underlined. English words are added to make the examples easy to read. In addition, the correct Chinese sentence is parenthesized below the English one. We can see that adverb ordering and subject/verb/object ordering errors are the two most frequent error categories. Because grammatical error is one category of word ordering errors, parsing may be helpful in this task.

The second and third examples are two interesting examples. In the second example, learner translate "drop out" to 休学 and "university" to 大学 in the same order, but, in Chinese, the reversed order 大学 休学 is more fluent. In the third example, learner translate "am studying" to 留学 and "in your country" to 在贵国, but, in Chinese, changing the position of the PP 在贵国 results in a more fluent sentence.

4 A Web-Scale Chinese POS-Tagged Corpus

We use a publicly available Chinese Web POS 5-gram corpus (CP5g) (Yu, Tang & Chen, 2012) for our experiments. The Chinese POS-tagged corpus has been developed based on the ClueWeb09 dataset, which is a huge web document collection crawled by LTI at CMU. The ClueWeb09 corpus contains documents in ten languages including English, Chinese, and so on. There are 177,489,357 Chinese pages in the ClueWeb09. Although most of the Chinese records are crawled from mainland China, there are still some pages in other languages, such as Japanese. In addition, a Chinese web page may be in different encodings, e.g., Big5, CNS 11643, GBK, GB2312, and Unicode. Thus, the encoding detection and language identification problems have to be dealt with in the development of the CP5g dataset.

After identifying the encoding scheme and the written language, they use the Stanford Chinese word segmenter (Tseng, Chang, Andrew, Jurafsky, & Manning, 2005) to segment the text, and adopt the Stanford POS tagger (Toutanova, Klein, Manning, & Singer, 2003) to tag Chinese sentences. After all n-gram (n=1, 2, 3, 4, 5) patterns are extracted, those patterns that occur less than 40 times are filtered out. The output format is similar to Google N-gram V2.

We also use the Google Chinese Web 5-grams (GC5g) (Liu et al., 2010) corpus for comparison. The size of GC5g is larger than that of CP5g, but GC5g does not contain linguistic information. In addition, GC5g and CP5g are developed by different segmentation systems, i.e., a maximum probability segmenter and a CRF-based segmenter, respectively. That may result in different word patterns and introduce uncertainty into the systems. We will consider these issues when discussing feature extraction of our approaches.

5 Detection of Word Ordering Errors

The detection of word ordering errors (WOEs) is formulated as a binary classification problem. We employ LIBSVM (Chang & Lee, 2011) to build a classifier which detects whether there is a word ordering error in a given Chinese sentence. Different types of features are extracted from various sources in the following sections.

5.1 Syntactic Features

Syntactic features include tagging and parsing results of a sentence. The motivation behind this approach is that the sentence with word ordering errors may result in abnormal POS and parsing patterns, and these linguistic clues are useful for WOE detection.

For the application of tagging features, a sentence is segmented and tagged by using the Stanford segmenter and tagger (Toutanova et al., 2003; Tseng et al., 2005). The performance of POS tagging is 94.13%. The POS *n*-grams (n = 2, 3, 4) in the sentence are extracted, and used as in the bag-of-words approach. In other words, these POS n-grams are treated like words and considered as the features for classification. We use B_2, B_3 and B_4 to refer to the approaches of POS bi-gram, tri-gram, and 4-gram, respectively.

For the applications of parsing features, the Stanford parser is used to analyze the dependency relations in a sentence. Consider the following example.

绿色/JJ 食品/NN 当然/AD 是/VC …
green food definitely is

The Stanford parser generates a set of relations as follows for this example, where the number after each word denotes the word position in the sentence.

amod(*食品-2, 绿色-1*) {*amod*(*food-2,green-1*)}
advmod(*是-4,当然-3*) {*advmod*(*is-4,definitely-3*)}
…

The word in a relation is replaced by its POS tag. For example, *amod*(*食品-2*, *绿色-1*) {*amod*(*food-2*,*green-1*)} is converted into *amod*(*NN*, *JJ*). These kinds of relations are collected, and used as features similar to the bag-of-words approach, in which the relation between two tags is regarded as a word. We use B_p to denote this approach.

Parsing features can be used individually, or integrated with all tagging features to form a larger feature vector B = (B_2, B_3, B_4, B_p) for a sentence. We will explore the effectiveness of all the alternatives in the later experiments.

5.2 Web Corpus Features

Web corpus features are extracted from two reference corpora, i.e., GC5g and CP5g corpora. Intuitively, the sentences with word ordering errors are less likely to occur in the language usages of real world. To measure the probability of a sentence with respect to a reference corpus, we use point-wise mutual information (PMI) (Church & Hanks, 1990). The PMI of a word pair (w_1, w_2) and the score c_g for a sentence that uses GC5g corpus are defined as follows.

$$PMI\ (w_1, w_2) = \log \frac{p(w_1, w_2)}{p(w_1)\,p(w_2)} \quad (1)$$

$$c_g = \frac{1}{L-1} \sum_{l=1}^{L-1} \log \frac{p(w_l, w_{l+1})}{p(w_l)\,p(w_{l+1})} \quad (2)$$

In (2), L is the sentence length in terms of the number of words. We divide the sum of PMIs by sentence length minus 1 for normalization.

To avoid the zero count problems in the above computation, we need some smoothing method. Section 4 specifies that those word pairs occurring less than 40 have been removed from the reference corpora. Thus, there are two cases for an unknown pair: a pair does not appear before, or it occurs, but its total occurrences are less than 40. However, we cannot know which case an unknown pair belongs to. Here, we adopt the same smoothing approach for these two cases. We introduce a $k \in \{1, 10, 20, 30, 39\}$ to replace the zero count, where $k = 1$ means we ignore this unknown word pair, and $k = 39$ is the largest value because the minimum occurrence in the reference corpora is 40.

This approach results in a 5-tuple feature vector $C_g = (c_{g,k})$, $k \in \{1, 10, 20, 30, 39\}$ for a sentence, when the GC5g corpus is regarded as a reference corpus. Similarly, the feature vector $C_b = (c_{b,k})$ is used if the CP5g corpus is adopted to compute the occurrence frequency.

The two feature vectors, C_g and C_b, can be directly input to an SVM classifier. In this way, there is no need to set a fixed threshold. As a usual preprocessing step of SVM training, we also scale each dimension to range [-1, 1].

In addition to word level features, we also consider POS level features. We utilize the tag information in the CP5g corpus to derive a probability of a sentence. The probability is defined as follows.

$$c_{p1} = \frac{1}{L} \sum_{l=1}^{L} \log\ p(t_l \mid w_l) = \frac{1}{L} \sum_{l=1}^{L} \frac{p(t_l, w_l)}{p(w_l)} \quad (3)$$

$$c_{p2} = \frac{1}{L-1} \sum_{l=1}^{L-1} \log \frac{p(t_l, t_{l+1}, w_l, w_{l+1})}{p(w_l, w_{l+1})} \quad (4)$$

where t_l is the POS tag of word w_l, c_{p1} is a normalized probability of a sentence to be tagged as $t_1 t_2 \ldots t_L$ by POS unigrams, and c_{p2} is a normalized probability of a sentence by POS bigrams. Similar to C_g, we have to face the unknown pair problem. Here, we adopt the similar smoothing approach, i.e., let $k \in \{1, 10, 20, 30, 39\}$ be the occurrence count of an unknown pair. We represent a sentence by three alternatives: a 5-tuple vector $C_{p1} = (c_{p1,k})$, a 5-tuple vector $C_{p2} = (c_{p2,k})$, or a 10-tuple vector $C_p = (C_{p1}, C_{p2})$.

5.3 Perturbation of Terms in a Sentence

Motivated by Gamon et al.'s work (2009) that calculates scores derived from a language model to filter out improper correction, we perturb a sentence to see how the features respond to the perturbation. We randomly choose two words and switch their positions to simulate the word ordering errors generated by language learners. After that, we extract different types of features from the perturbed sentences as the procedures described in Sections 5.1 and 5.2.

Let pB_n be a perturbed version of B_n for the perturbed sentence. Similarly, we have all perturbed versions of features. As an alternative to pB_n, we consider the difference to the original sentence and use $\triangle pB_n = pB_n - B_n$ as the features.

5.4 Combining Features

We combine features from different sources to get better models. At first, we cosine-normalize a vector to a unit vector. For example, C_b is normalized by the following formula.

$$\overline{C}_b = \frac{C_b}{\| C_b \|} \tag{5}$$

We then combine normalized vectors for experiments. Here, we equally emphasize different types of features. For example, when combining vectors B_2 and C_b, we normalize B_2 and C_b into $\overline{B}_2$ and $\overline{C}_b$, and combine them to a new vector $(\overline{B}_2, \overline{C}_b)$.

6 Experimental Setups

We collect all 8,515 sentences containing WOEs from the HSK web site. We restrict the sentence length to 6~60 Chinese characters, filter out sentences with multiple error types, and remove duplicate sentences. There are 1,100 error sentences with WOEs for this study.

Considering the ratio of correct sentences, we use balanced datasets in the experiments. The most important reason is that the percentage of WOEs in the real world is relatively small. There are 2.83 million words in HSK. Assume a sentence contains 10 words on average. There are 0.283 million sentences in HSK. If we build an experimental dataset in this ratio, i.e., 8,525 vs. 283,000, the number of correct sentences will dominate the system's performance. Therefore, we build a balanced dataset to prevent this bias and to have a fair examination.

In the study of automatic error detection, collecting linguistic errors in the real world is indispensable, but time-consuming. It needs a lot of efforts to tag the error types and the possible corrections. Therefore, we plan to best utilize the error sentences collected in the HSK corpus to get more results for discussion. From the HSK corpus, we collect 4,400 correct sentences to generate four balanced datasets, each containing 1,100 error sentences and 1,100 correct sentences. These four datasets are called the HSK-HSK datasets.

In addition to using HSK correct sentences as positive sentences, we consider sentences by native Chinese writers as positive sentences to analyze which training set is useful to detect error sentences written by non-native language learners. We collect another 4,400 sentences from Chinese articles written by native Chinese writers, which talk about the same themes as those in HSK corpus, i.e., health, life, job application, classmate and friendship. The sentences from native learners are assumed to be correct. Those 4,400 web sentences are used to generate

another four balanced datasets. The resulting four datasets are named as the NAT-HSK datasets. In total, we have eight datasets in two groups for our experiments.

For each dataset, the corresponding 2,200 sentences are further randomly split into five folds. The experimental results are reported by averaging the 20 folds (4 datasets by 5 folds). In this way, we decrease the variations from the dataset generation, and make more reliable analyses about our approaches. We use RBF kernel and adopt grid search to determine the best SVM parameters in training set.

7 Experimental Results

In this section, we report the average accuracy of HSK-HSK datasets and NAT-HSK datasets. Paired Student's *t*-test is used for significant test. The null hypothesis is reject if $|t| > t_{19,\ 0.975} = 2.093$, where there are 19 degrees of freedom and the probability that *t* is less than 2.093 is 0.975. Because the WOE problem is formulated as a binary classification problem and we use balanced datasets, we adopt accuracy as our evaluation metric, which is more adequate than precision and recall. In addition, we ignore the analysis of confusion matrix in the results because we find that there are no big differences between the results of positive and negative samples.

7.1 Basic Features

Firstly, we want to know the performance of the syntactic features. Table 3 shows the experimental results. B_2, B_3, and B_4 denote feature vectors of POS *n*-grams (*n*=2, 3, 4) and B_p denotes feature vector of dependency relations. (B_2,B_3,B_4,B_p) denotes the combined syntactic feature vectors of POS n-grams (n=2, 3, 4) and parsing feature B_p. The resulting model is named model B.

Features	HSK-HSK		NAT-HSK	
	accuracy (%)	stdev.	accuracy (%)	stdev.
B_2	62.63	2.12	67.61	2.17
B_3	60.81	2.14	65.47	2.24
B_4	57.83	1.98	61.28	2.18
B_p	61.86	2.21	63.17	2.57
B=(B_2,B_3,B_4,B_p)	**63.89**	2.17	**69.61**	2.04

TABLE 3 –Performance of basic features

The accuracy of using POS 2-gram features (i.e., B_2) is better than that of using other individual features significantly except the parsing feature (i.e., B_p) in HSK-HSK datasets. It shows that POS 2-grams are very useful in detecting Chinese word ordering errors.

Another interesting finding is the usefulness of parser. The accuracy of using POS 2-gram features (i.e., B_2) is better than that of using parsing features (i.e., B_p) in NAT-HSK datasets significantly. Its accuracy reaches 67.61%. That reflects the effect of training with native Chinese writers' sentences. Intuitively, the sentences of native Chinese learners are more fluent than those of foreign learners, so that it is easier to detect wrong sentences with POS 2-grams learned from the correct sentences.

Because we use balanced datasets, the trivial baseline is 50% accuracy. Using all the above features outperform the baseline significantly. The combined model $B=(B_2,B_3,B_4,B_p)$ using all syntactic features outperforms the features using only one syntactic feature in both HSK-HSK and NAT-HSK datasets significantly.

7.2 Web Corpus Features

In Chinese, characters are segmented into a sequence of words and then the word frequencies are counted. In the next set of experiments, we aim to know how word segmentation influences the results of error detection. Two issues, including the performance difference without/with segmentation, and the effects of different segmentation systems, are considered.

When calculating $p(w_i,w_{i+1})$ in Equation (2), we use different approaches to get the frequencies of word pairs (w_i,w_{i+1}). The first approach ignores the word boundary between w_i and w_{i+1}, regards them as a single string, and uses string matching to get its count. This is a without-segmentation approach $C_{\bullet W}$, where the subscript character $\bullet$ denotes a reference corpus, i.e., GC5g or CP5g in our experiments. The second approach regards w_i and w_{i+1} as two separate words, and uses exact word matching to count their co-occurrences. This is a with-segmentation approach $C_{\bullet S}$. As a result, when we compute frequencies of word pairs, there are six combinations C_{gS}, C_{gW}, C_{bS}, C_{bW}, C_{p2S}, and C_{p2W}, where the subscript g means the Google Chinese Web 5-gram corpus (GC5g) is adopted, the subscript b means the Chinese Web POS 5-gram corpus (CP5g) is adopted, and the subscript p2 means Equation (4) and CP5g are adopted. We compare the effects of segmentation and the use of different kinds of corpora to see if they are complementary. Table 4 shows the results.

Features	HSK-HSK		NAT-HSK	
	accuracy (%)	stdev.	accuracy (%)	stdev.
C_{gW}	50.84	2.26	63.17	1.54
C_{gS}	52.64	2.01	65.01	2.23
$\overline{\overline{C}}_g=(C_{gW}, C_{gS})$	52.59	2.21	64.77	2.29
C_{bW}	50.90	2.94	63.90	1.81
C_{bS}	51.95	2.67	65.09	1.69
$\overline{\overline{C}}_b=(C_{bW}, C_{bS})$	56.99	1.98	65.93	2.01
C_{p1}	49.33	2.59	53.33	2.16
C_{p2W}	53.19	2.06	60.32	2.19
C_{p2S}	52.10	2.17	59.68	1.53
$\overline{\overline{C}}_p=(C_{p1},C_{p2W},C_{p2S})$	57.01	1.35	61.27	1.68
$(\overline{\overline{C}}_g, \overline{\overline{C}}_b, \overline{\overline{C}}_p)$	59.35	2.48	66.69	1.92
$B=(B_2,B_3,B_4,B_p)$	63.89	2.17	69.61	2.04
$(B, \overline{\overline{C}}_g)$	63.70	1.84	69.82	1.66
$(B, \overline{\overline{C}}_b)$	64.11	2.13	69.85	1.99
$(B, \overline{\overline{C}}_p)$	63.80	2.25	70.23	2.15
$(B, \overline{\overline{C}}_g, \overline{\overline{C}}_b, \overline{\overline{C}}_p)$	**64.34**	2.35	**71.18**	2.29

TABLE 4 –Performance of web corpus features

Comparing the uses of the individual features, i.e., C_{gS}, C_{gW}, C_{bS}, C_{bW}, C_{p1}, C_{p2S}, and C_{p2W}, on the HSK-HSK and NAT-HSK datasets, we find the performance of using NAT-HSK datasets is better than that of using HSK-HSK datasets. That meets our expectation because the correct sentences in NAT-HSK datasets are selected from native Chinese writers' web pages. Thus, we fix the datasets to NAT-HSK in the next discussion.

Because two different segmentation systems are applied in the development of the two reference corpora, we compare C_{gS} and C_{bS} features to see if the segmentation results have different effects on the performance. Their performance, 65.01% vs. 65.09%, does not have a significant difference in NAT-HSK datasets. Thus, we conclude that the two different segmentation systems do not have different effects.

In addition, we compare the performance of C_{gW} (63.17%) vs. C_{gS} (65.01%), and C_{bW} (63.90%) vs. C_{bS} (65.09%) to see if segmentation is necessary. The accuracies using word-based matching (i.e., with segmentation $C_{\cdot S}$) are better than those using string-based matching (i.e., without segmentation $C_{\cdot W}$) significantly. This phenomenon also holds in HSK-HSK datasets. Thus, we conclude that using segmentation systems result in better performance.

Next we analyze the usefulness of individual POS features. Table 4 shows that the performance of using C_{p1} features is 49.33% and 53.33% in HSK-HSK and NAT-HSK datasets, respectively. It seems that POS unigram information does not help. Comparatively, the performance of C_{p2W} shows that POS bigrams are more useful than POS unigrams, but POS-based features (C_{p2W}) cannot compete with word-based features (C_{gW} and C_{bW}). We analyze the eight datasets and find that only 7.9% of the tuple $(t_l,t_{l+1},w_l,w_{l+1})$ in Equation (4) can be found in the CP5g corpus, while 77.9% and 78.6% of the word pair (w_l,w_{l+1}) can be found in the GC5g corpus and the CP5g corpus, respectively. Therefore, we conjecture that with a larger dataset, POS N-grams will make a significant difference in this task.

Finally, we examine the complementary effects of the individual features by different integration. Integrating the individual web corpus features $\overline{\overline{C}}_g$, $\overline{\overline{C}}_b$, and $\overline{\overline{C}}_p$ with all syntactic features B outperforms using only the syntactic features B. The accuracy (71.18%) of the approach integrating all web corpus features ($\overline{\overline{C}}_g$, $\overline{\overline{C}}_b$, $\overline{\overline{C}}_p$) with all syntactic features B is significantly better than the accuracy (69.61%) of the approach using B. This confirms that the web corpus features are useful in WOEs detection.

7.3 Perturbation

Table 5 shows parts of perturbation experiments. The system $B=(B_2,B_3,B_4,B_p)$ significantly outperforms all the perturbation features. It seems that the perturbation does not help in all variations. One possible explanation is that we perturb words, but the word ordering errors usually involve many words such as the example "我 留学 在 贵国" (I am studying in your country) in Table 1. The word-based swapping "在 留学" (in studying) cannot capture the PP "在 贵国" (in your country). A phrase-based swapping should be adopted.

Features	HSK-HSK		NAT-HSK	
	accuracy (%)	stdev.	accuracy (%)	stdev.
B_2	62.63	2.12	67.61	2.17
pB_2	60.83	3.03	64.49	1.81

TABLE 5 –Performance of perturbations

ΔpB_2	51.14	3.30	53.28	2.08
C_b	56.99	1.98	65.93	2.01
pC_b	57.25	2.29	65.32	1.54
ΔpC_b	57.23	2.14	65.57	1.67
$B=(B_2,B_3,B_4,B_p)$	**63.89**	2.17	**69.61**	2.04
$(pB_2, \Delta pB_2, pC_b, \Delta pC_b)$	62.34	2.91	66.33	1.99

TABLE 5 –Performance of perturbations (*continued*)

7.4 Combined Features

We combine all features, including syntactic features (B), web corpus features (C_g, C_b, and C_p), and perturbation features, and report their performance in Table 6. We can see that using all features results in the highest accuracy (71.64%) in NAT-HSK datasets. It is better than the accuracy (69.61%) of the system B significantly. Also, using all features also results in the highest accuracy (64.81%) in HSK-HSK datasets.

Features	HSK-HSK		NAT-HSK	
	accuracy (%)	stdev.	accuracy (%)	stdev.
All features	**64.81**	3.45	**71.64**	1.85
$(B, \overline{C}_g, \overline{C}_b, \overline{C}_p)$	64.34	2.35	71.18	2.29
$B=(B_2,B_3,B_4,B_p)$	63.89	2.17	69.61	2.04

TABLE 6 –Performance of combining features

7.5 Analyses

At first, we want to know the relationship between sentence length and accuracy. We apply the models trained by using all features and one of HSK-HSK datasets, as well as one of NAT-HSK datasets to gather the statistics. These two datasets are denoted as HSK-HSK-1 and NAT-HSK-1, respectively. Figure 1 shows the results. We can see that the average accuracies in both two sets are high when the sentence length is in range [10, 19] characters. Moreover, the performance of using NAT-HSK-1 dataset is better than that of using HSK-HSK-1 dataset in all various lengths. That confirms our conclusion in Section 7.1.

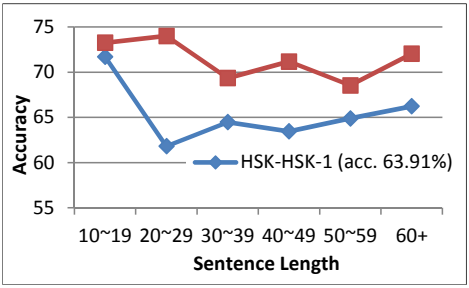

FIGURE 1 – Accuracy analysis by sentence length

Next, we want to know which category of errors is more difficult to detect. Among 200 sentences in Table 1, 197 WOE sentences are used in our experiments. We use HSK-HSK-1 and NAT-HSK-1 to gather the performance in different categories. The overall accuracy of these 197 sentences in HSK-HSK-1 dataset is 74.11% and is 70.56% in NAT-HSK-1 dataset. The detail result is listed in Table 7. The overall average accuracy of HSK-HSK-1 is 63.91% (in Figure 1 and Table 7), which is smaller than the average accuracy 74.11% of 197 sentences. Therefore, the 197 sentences may not be a good representation of HSK-HSK-1 dataset. Comparatively, the overall accuracy of 197 sentences in NAT-HSK-1 dataset is 70.56%, which is near 71.22%. Thus, these 197 sentences may be representative in NAT-HSK-1 dataset. We use the last column to analyze the difficulties of error detection in different categories. Of the former 3 major categories of errors, the accuracy of detecting PP position errors is higher than the overall accuracy. This indicates that the fluent sentences written by native Chinese learners are especially useful for detecting PP position errors in the sentences by non-native learners.

Category	#sentences	HSK-HSK-1	NAT-HSK-1
		accuracy (%)	accuracy (%)
Adverb ordering error	70	77.14	67.14
S/V/O ordering error	65	70.77	67.69
PP position error	41	73.17	**80.49**
Prenominal adjective error	10	**80.00**	80.00
Others	11	72.73	63.64
accuracy (total 197 sentences)		74.11	70.56
accuracy (total 2,200 sentences)		63.91	71.22

TABLE 7 –Performance analysis by category

Finally, we want to know the error detection performance with respect to the students' nationality. We collect the nationality information of the 197 sentences from the HSK website. The results are shown in Table 8. We can find that Korea and Japanese students are the major Chinese learners in the 197 WOE sentences. In HSK-HSK-1 dataset, 77.78% of WOEs from Japanese students can be detected. But this trend is not the same in NAT-HSK-1 dataset. In NAT-HSK-1 dataset, nationality does not make large difference on the error detection performance.

Nationality	#sentences	HSK-HSK-1	NAT-HSK-1
		accuracy (%)	accuracy (%)
Korea	73	69.86	**72.60**
Japan	72	**77.78**	68.06
Other countries	52	75.00	71.15
accuracy (total 197 sentences)		74.11	70.56
accuracy (total 2,200 sentences)		63.91	71.22

TABLE 8 –Performance analysis by nationality

Conclusion

In this paper, we deal with the detection of Chinese word ordering errors for learning Chinese as a foreign language from the application, language, resource and technology aspects. Different categories of word ordering errors in the sentences written by non-native language learners are

analyzed. The experiments show that syntactical features, web corpus features and perturbation features are all useful in detecting word ordering errors. The system using all the features from different sources has the best accuracy, 71.64%, when the native writers' sentences are selected as positive sentences. The proposed system is significantly better than the baseline systems.

In Chinese, word segmentation is inherent in many applications. Even though word ordering errors in Chinese sentences may result in incorrect segmentations, the experiments show that the word-based approach (i.e., with segmentation) is better than the character-based approach (i.e., without segmentation) irrespective of the use of different segmentation systems and reference corpora. On the other hand, although the Chinese Web POS 5-gram corpus is smaller than the Google Chinese Web 5-gram corpus, the experiments show that they have similar performances on the WOE detection. With this important finding, we plan to use linguistic information in the CP5g corpus for other Chinese NLP tasks such as sentiment analysis.

The use of web scale linguistic information provides a solid basis of further error detection and correction. In the future, extending this task to detect other types of errors, using web-scale linguistic corpus to identify the potential error positions and proposing the correct sentences are the works to be investigated.

Acknowledgments

This research was partially supported by Excellent Research Projects of National Taiwan University under contract 101R890858.

References

Bergsma, S., Lin, D. and Goebel, R. (2009). Web-Scale N-gram Models for Lexical Disambiguation. In *the 21st International Joint Conference on Artificial Intelligence*, pages 1507–1512, Pasadena, California, USA.

Brants, T. and Franz, A. (2006). Web 1T 5-gram Version 1. Linguistic Data Consortium, Philadelphia.

Carl, J. (1998). Errors in Language Learning and Use: Exploring Error Analysis. Addison Wesley Longman.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3), 27:1–27:27.

Chodorow, M. and Leacock, C. (2000). An Unsupervised Method for Detecting Grammatical Errors. In *the 1st North American chapter of the Association for Computational Linguistics conference*, pages 140–147, Seattle, Washington, USA.

Church, K. W. and Hanks, P. (1990). Word Association Norms, Mutual Information, and Lexicography. *Computational Linguistics*, 16(1), 22–29.

Gamon, M., Leacock, C., Brockett, C., Dolan, W. B., Gao, J., Belenko, D. and Klementiev, A. (2009). Using Statistical Techniques and Web Search to Correct ESL Errors. *CALICO Journal*, 26(3), 491–511.

Islam, A. and Inkpen, D. (2010). An Unsupervised Approach to Preposition Error Correction. In *the 6th IEEE International Conference on Natural Language Processing and Knowledge Engineering*, pages 1–4, Beijing, China.

Islam, A. and Inkpen, D. (2011). Correcting Different Types of Errors in Texts. In *the 24th Canadian Conference on Advances in Artificial Intelligence*, pages 192–203, St. John's, Canada.

Izumi, E., Uchimoto, K., Saiga, T., Supnithi, T. and Isahara, H. (2003). Automatic Error Detection in the Japanese Learners' English Spoken Data. In *the 41st Annual Meeting on Association for Computational Linguistics*, pages 145–148, Sapporo, Japan.

Leacock, C., Chodorow, M., Gamon, M. and Tetreault, J. (2010). Automated Grammatical Error Detection for Language Learners. Morgan and Claypool Publishers.

Lee, J., Zhou, M. and Liu, X. (2007). Detection of Non-Native Sentences Using Machine-Translated Training Data. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, pages 93–96, Rochester, NY.

Lin, D., Church, K., Ji, H., Sekine, S., Yarowsky, D., Bergsma, S., Patil, K., Pitler, E., Lathbury, R., Rao, V., Dalwani, K. and Narsale, S. (2009). Unsupervised Acquisition of Lexical Knowledge from N-grams: Final Report of the 2009 JHU CLSP Workshop.

Lin, D., Church, K., Ji, H., Sekine, S., Yarowsky, D., Bergsma, S., Patil, K., Pitler, E., Lathbury, R., Rao, V., Dalwani, K. and Narsale, S. (2010). New Tools for Web-Scale N-grams. In *the Seventh Conference on International Language Resources and Evaluation*, pages 2221–2227, Malta.

Linguistic Data Consortium (LDC). (2003). English Gigaword.

Liu, F., Yang, M. and Lin, D. (2010). Chinese Web 5-gram Version 1. Linguistic Data Consortium, Philadelphia.

Malouf, R. (2000). The Order of Prenominal Adjectives in Natural Language Generation. In *the 38th Annual Meeting on Association for Computational Linguistics*, pages 85–92, Hong Kong.

Toutanova, K., Klein, D., Manning, C. D. and Singer, Y. (2003). Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 173–180, Edmonton, Canada.

Tseng, H., Chang, P., Andrew, G., Jurafsky, D. and Manning, C. (2005). A Conditional Random Field Word Segmenter. In *Fourth SIGHAN Workshop on Chinese Language Processing pages*, pages 168–171, Jeju, Korea.

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

Yu, C.-H., Tang, Y. and Chen, H.-H. (2012). Development of a Web-Scale Chinese Word N-gram Corpus with Parts of Speech Information, In *the Eighth International Conference on Language Resources and Evaluation*, pages 320–324, Istanbul, Turkey.