Introduction to BIT Chinese Spelling Correction System at CLP 2014 Bake-off

Min Liu
School of Computer Science and Technology, Beijing Institute of Technology
luis328@foxmail.com

Ping Jian
School of Computer Science and Technology, Beijing Institute of Technology
pjian@bit.edu.cn

Heyan Huang
School of Computer Science and Technology, Beijing Institute of Technology
hhy63@bit.edu.cn

Abstract

This paper describes the Chinese spelling correction system submitted by BIT at CLP Bake-off 2014 task 2. The system mainly includes two parts: 1) N-gram model is adopted to retrieve the non-words which are wrongly separated by word segmentation. The non-words are then corrected in terms of word frequency, pronunciation similarity, shape similarity and POS (part of speech) tag. 2) For wrong words, abnormal POS tag is used to indicate their location and dependency relation matching is employed to correct them. Experiment results demonstrate the effectiveness of our system.

1. Introduction

Spelling check, which is an automatic mechanism to detect and correct human spelling errors, is a common task in every written language. The number of people learning Chinese as a Foreign Language (CFL) is booming in recent decades and this number is expected to become even larger for the years to come. However, unlike English learning environment where many learning techniques have been developed, tools to support CFL learners are relatively rare, especially those that could automatically detect and correct Chinese spelling and grammatical errors. For example, Microsoft Word® has not yet supported these functions for Chinese, although it supports English for years. In CLP Bake-off 2014, essays written by CFL learners were collected for developing automatic spelling checkers. The aims are that through such evaluation campaigns, more innovative computer assisted techniques will be developed, more effective Chinese learning resources will be built, and the state-of-art NLP techniques will be advanced for the educational applications.

By analyzing the training data released by the CLP 2014 Bake-off task2[1] and the test data used in SIGHAN Bake-off 2013[2], we find that the main errors focus on two types: One is wrong characters which result in "**non-words**" that are similar to OOV (out-of-vocabulary). For example, the writer may misspell "身邊" as "生邊", and "根據" as "根處" (The former appears because of the words' similar pronunciation and the latter comes up due to their similar shape). These are even not words and of course do not exist in the vocabulary. The other type is words which are correct in the dictionary but incorrect in the sentence. Some of them may be misspelled, like "情愛" in phrase "情愛的王宜家", which is a misspelling of word "親愛". But we can find "情愛" in the dictionary and it is not a non-word. Others are words which are not used correctly. This usually happens when the writer does not understand their meaning clearly. For example, writers often confuse "在" and "再", such as "高雄是再台灣南部一個現代化城市". Here, it is "在" but not "再" the right one. Different from non-words, we call these words "**wrong words**". According to the statistics obtained from the training data of CLP 2014 Back-off, there are nearly 3,400 wrong words which are about twice more than non-words, 1,800 ones.

Spelling check and correction is a traditional task in natural language processing. Pollock and Zamora (1984) built a misspelling dictionary for spelling check. Chang (1995) adopted a bi-gram language model to substitute the confusing character. Zhang et al. (2000) proposed an approximate word matching method to detect and correct spelling errors. Liu et al. (2011)

[1] http://www.cipsc.org.cn/clp2014/webpage/cn/four_bakeoffs/Bakeoff2014cfp_ChtSpellingCheck_cn.htm

[2] http://tm.itc.ntnu.edu.tw/CNLP/?q=node/27

Proceedings of the Third CIPS-SIGHAN Joint Conference on Chinese Language Processing, pages 179–185,
Wuhan, China, 20-21 October 2014

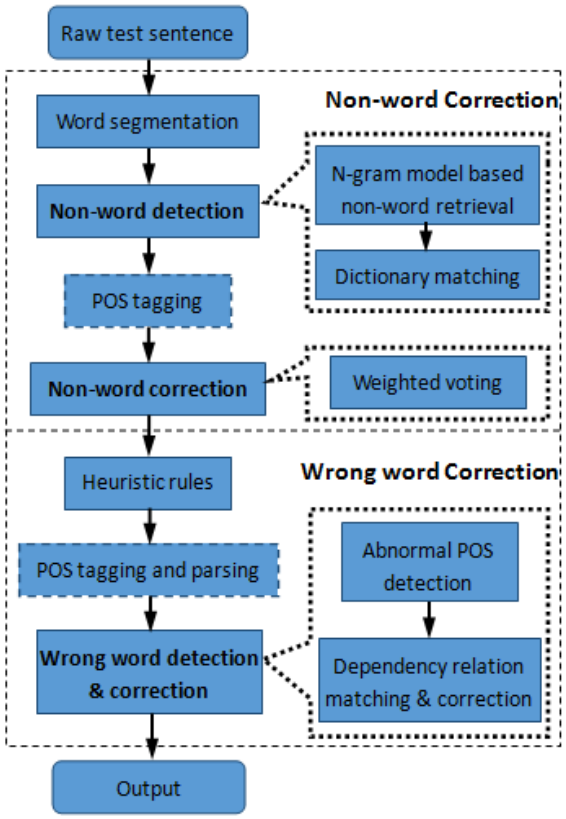


Figure 1: System architecture

extended the principles of decomposing Chinese characters with the Cangjie codes to judge the visual similarity between Chinese characters. SIGHAN Bake-off 2013 for Chinese spelling check inspired a variety of spelling check and correction techniques (Wu et al., 2013). Typical statistical approaches such as maximum entropy model and machine translation model performed well assisted by rule based model and other language analysis techniques.

Compared with the test data in SIGHAN Bake-off 2013, there are more wrong words and the text is more colloquial in the current Bake-off, which make the correction task more challenging.

2. System Architecture

In terms of the error types of the task, our system is mainly composed by two stages: non-word correction and wrong word correction. In detail, stage one consists of several parts: word segmentation, non-word detection, POS (part of speech) tagging and non-word correction. The second stage is conducted by heuristic rules correction, POS tagging & parsing, and wrong word detection & correction. The figure 1 shows the architecture of our system.

2.1 Preparations

To cater to the need of error correction system for linguistic resources, three dictionaries/bases are constructed: a dictionary, a word-POS base and a dependency relation base.

We use Tsai's list of Chinese words[3] collected by Chih-Hao Tsai as a basic dictionary and make use of Sinica Corpus[4] to add frequency for each word in it. Considering that Pinyin[5] can be useful in pronunciation similarity spelling error detection and correction, we add it to each word in the dictionary with the help of TagPinyin[6] developed by International R&D Center for Chinese Education. Since this tool can only tag Pinyin for simplified Chinese, we use OpenCC[7] to make the conversion between traditional Chinese and simplified Chinese. By this way, we obtain the dictionary like the example below:

```
胛 胛 jia 1    胛骨 jia gu 1
慚 慚 can 3    慚色 can se 1 慚愧 can kui  58
```

There are more than 239,000 words totally in the dictionary. The words have the same first character are put in one line and they are indexed by their first character to boost the efficiency of searching. Each item consists of three parts: the word (“慚愧”), the Pinyin (“can kui”), and the frequency (“58”).

Penn Chinese Treebank7.0 (CTB7.0) (Xue et al., 2005) is employed to build the word-POS base and the dependency relation base. In this way, the word category information and candidates for correct words are provided. Taking domain and area stuff into consideration, we extract the mz (news magazine from Sinorama), bc (broadcast conversation from New Tang Dynasty TV etc.) and wb (weblogs) parts of CTB7.0, which form a dependency corpus including 30,861 sentences. The simplified characters in the corpus are also converted by OpenCC. We get about 42,000 items in the word-POS base and the format is as following:

```
揭露    JJ  1   NN  1   VV  16
揮      VV  4
揮之不去 VV  2
```

[3] http://technology.chtsai.org/wordlist/
[4] http://app.sinica.edu.tw/kiwi/mkiwi/
[5] Pinyin is the standard system of romanized spelling for transliterating Chinese.
[6] http://nlp.blcu.edu.cn/downloads/download-tools/
[7] http://code.google.com/p/opencc/

In the example above, the first column is the word and the following are all the POSes and their frequencies by counting the corpus.

The dependency relation base is made up of dependency relations extracted from the CTB corpus. It includes one word with all its head words and the corresponding frequencies in each line. The following is an example:

抗議 ROOT 4 事件 3 以示 1 …

Here, "抗議" is headed by "ROOT" which means that it is the root word in the sentence. By this way, more than 300,000 dependency relations were extracted from the corpus.

Originally, we considered Sinica Treebank[8], which is a traditional Chinese corpus in nature, as the more proper one to generate the POS and dependency base. However, the POS category and the dependency relation type of the bank are too trivial. In addition, the parsing unit in Sinica Treebank is not a natural sentence but segments divided by punctuations, which results in lack of dependency types. Many relations between segments degenerate to "ROOT" in the Treebank.

2.2 Non-word Correction

This stage mainly includes non-word detection & correction stage and it starts from the segmentation of raw error sentences. When segmentation is done for the input file, we find that the words involving misspelled character might be separated into serial characters. For example, sentence "這個學期已經過了兩個裡拜了。" will be segmented into "這 個 學期 已經 過了 兩個 裡 拜 了 。" and potential non-word "裡拜" is impossible to be found as a word. Dictionary based non-word detection would not work in this case. We utilize a simple n-gram model here to retrieve the missing words. The method in detail is described as following: The uncommon co-occurrence of adjacent characters after segmentation can be found by pre-trained character n-gram model. The retrieving begins at the first single-character word with low probability, and combines it with one single-character word before or after it. To further confirm whether the combination is reasonable or not, we traverse the dictionary to find if there is a "dependable" candidate word which can make sure that the retrieved non-word can be substituted by a real word in the dictionary. For simplicity, we only consider words who have the same Pinyin form with the retrieved word as the "dependable" words.

After the non-word retrieval, a dictionary matching is competent to detect the non-words in the sentences. In the step of non-word correction, the word which cannot be matched from the dictionary completely will be substituted by a word in the dictionary. A weighted voting approach is employed here to select the most possible candidate word.

$$\hat{w}(w_{\text{non}}) = \arg\max_{w_i \in \text{Dic}} Score(w_i, w_{\text{non}}), \quad (1)$$

$$Score(w, w_{\text{non}}) = logFr_w + Sim(w, w_{\text{non}}), \quad (2)$$

$$Sim(w, w_{\text{non}}) = \alpha_1 Sim_{\text{pro}} + \alpha_2 Sim_{\text{shap}} + \alpha_3 Sim_{\text{POS}}, \quad (3)$$

$$Sim_{\text{pro}} = \begin{cases} 1 & \text{same pronunciation} \\ 0 & \text{otherwise} \end{cases}, \quad (4)$$

$$Sim_{\text{shap}} = \begin{cases} 1 & \text{same shape} \\ 0 & \text{otherwise} \end{cases}, \quad (5)$$

$$Sim_{\text{POS}} = \begin{cases} 1 & \text{same category} \\ 0 & \text{otherwise} \end{cases}, \quad (6)$$

where, w_{non} represents the non-word to be substituted while w is the candidate word. Fr in the formulation indicates the frequency of the word in the dictionary. Besides the frequency, three types of similarity measures are considered in our system: the pronunciation similarity, the shape similarity and the lexical category similarity. If the candidate word in the dictionary has the same or similar pronunciation with the target word, Sim_{pro} is set 1, else it is set 0. The setting of Sim_{shap} is the same. Because characters of similar pronunciations are the most common source of errors in the training set, the weight coefficient α_1 is set 2 and α_2 and α_3 are both set 1 in our system. The similar pronunciation and similar shape character set offered by SIGHAN Bake-off 2013 are employed to scope the candidates.

As for the category similarity, it is known that there is no lexical category for an out-of-dictionary word. To predict the probable class of the target non-word (more precisely, it's the class of the location where the non-word locates), a sequential labeling POS tagger is applied. We believe that the tagger will label a known word depending more on the word itself but label an unknown word relying more on its context. Experiments and analyses on the training data show that about 80% non-words are

[8] http://rocling.iis.sinica.edu.tw/CKIP/engversion/treebank.htm

specified with the category which is valid for the corresponding correct words. For instance, sentence "我 已經 *其待 了 很久" is tagged as following:

我_PN 已經_AD *其待_VV 了_AS 很久_NN

For the non-word "*其待", tag "VV" is marked, which indicates that it needs a verb there in accordance with the context. "VV" is also one of the possible categories of the word "期待", which is the word that there was supposed to be. In the weighted voting module, candidates who own the same POS tag with the target word are preferred to be selected.

In consideration of all the measures, candidate word with the highest score will be chosen as the correction result.

2.3 Wrong Word Correction

After all the non-words are substituted by in dictionary words, several heuristic rules are utilized to deal with some phenomena with strong regularity. These rules include:

- Replace "門" by "們": if there is any word in a predefined set or its first-class similar words in HIT-CIR TongyiciCilin (Extended)[9] (Che et al., 2010) appearing before "門", it should be "們". The set used in the task is:
  {我, 你, 妳, 他, 她, 人, 同學, 兄弟, 親人, 客人, 對手, 成員, 公司, 工廠, 企業}.
- Correction of interjections: if "阿", "把", "巴", "拉" and "麻" etc. locate before a dot mark (。？！ ，、 ； ：) and segmented as a single character word, it should be "啊", "吧", "啦" or "嘛".
- The gender related correction: correct "他", "她", "你", "妳" into the one appears more frequently in the context (within the sentences owning the same Pid). Here is an example: "妳" will be corrected by "你" in sentence "我希望，你會妳自己發現怎麼做。可是我覺得你得問朋友怎麼辦。所以我覺得你上課的時候不應該喝酒。而且喝酒對你的身體不好，害你很容易感冒。"
- Correction of "De" ("De" refers to one of the word "的", "地" and "得"): which "De" will be used depends on the category of its head word located after it in the sentence.
  If the category of the head word is adjective or adverb, it should be "得".
  If the one is noun or punctuation, it should be "的".
  If the one is verb, it should be "地".
  To make use of the dependency structure this rule should be carried out after the POS tagging and parsing step.

For common errors, a novel method comprising abnormal POS detection and dependency relation matching is designed.

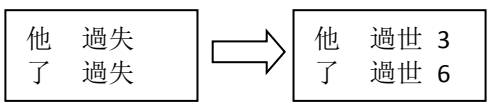
It is found that the POS tag of some words in a sentence may look strange when there is a wrong word in the sentence. Two examples are as following:

他_PN 過失_VV 已經_AD 三_CD 年_M 多_AD 了_SP 再_AD 台灣_VV 生活_NN 怎麼樣_VA

The existence of wrong word "過失" and "再" confuses the sequential POS tagger and abnormal labeling comes up. Sometimes it happens on the wrong words themselves, such as "過失" being labeled with an impossible class "VV" (verb); sometimes other words around are affected by the wrong word, such as "台灣" being tagged as a verb due to the wrongly used word "再" before it.

To locate and correct these wrong words, a dependency parsing is carried out following the POS retagging and all the dependency pairs involving the abnormal word are extracted to be examined. The left side in Figure 2 shows the dependency pairs related with "過失". Distinct with the first one, POS tagging at this stage is conducted on the sentence where the non-words has been replaced by in-dictionary ones. This is hoped to achieve a higher tagging precision.

By traversing the dependency base, if there is no exact matching of these dependencies but similar ones (by pronunciation or by shape) in the base, we have reason to believe that the matched similar pairs imply the answer we expect. The right side of Figure 2 exhibits the matched pairs in the dependency base. In the example, the wrong word "過失" is to be changed with "過世". In the same way, "再 台灣" will be corrected by "在 台灣" since the latter is frequent in the base.



Dependency relations in the error sentence

Dependency relations in the dependency base

Figure 2: Wrong word correction via dependency relation matching

[9] http://www.ltp-cloud.com/

3. Experiments

In this section, several experiments are conducted to verify the proposed methods described in Section 2. The final official provided test dataset consists of 1,062 sentences with or without spelling errors in traditional Chinese. Since the released training data are hardly employed to train models in our system, we regard it as a development set where some parameters are settled.

3.1 Training N-gram, Word Segmentation, POS Tagging and Parsing Models

Sinica Corpus was used to train the CRF based word segmentation model implemented by CRF++[10], while the final test sets released by SIGHAN Bake-off 2013 were used to train the single character word n-gram model. The POS tagger and parser were trained at the extracted part of CTB7.0 (the same part where the dependency base is built). The texts were converted into tradition Chinese by OpenCC. Like word segmentation, CRF based sequential labeling model is utilized for the POS tagging. It can achieve an accuracy more than 93% when trained and tested at CTB. Dependency trees of the test sentences were obtained by a fast parser, the Layer-based dependency parser[11], which considers hierarchical parsing as sequence labeling (Jian and Zong, 2009).

3.2 Metrics

The criteria for judging correctness are:

(1) **Detection level**: all **locations** of incorrect characters in a given passage should be completely identical with the gold standard.
(2) **Correction level:** all **locations** and corresponding **corrections** of incorrect characters should be completely identical with the gold standard. The following metrics are measured in both levels with the help of the confusion matrix.

- False Positive Rate (FPR) = FP / (FP+TN)
- Accuracy=(TP+TN)/ (TP+TN+FP+FN)
- Precision= TP/(TP+FP)
- Recall=TP/(TP+FN)
- F1-Score=2*Precision*Recall/(Precision+Recall)

Confusion Matrix		System Result	
		Positive (With Errors)	Negative (Without Errors)
Gold Standard	Positive	**TP** (True Positive)	**FN** (False Negative)
	Negative	**FP** (False Positive)	**TN** (True Negative)

Table 1: Confusion Matrix

3.3 Experiment Design

There are some different settings in our previous experiments on the development set (the released training data) and we apply three of them to the final test file.

BIT Run1: All modules are employed except the abnormal POS detection and dependency relation matching. The threshold of the n-gram transfer probability at non-word retrieval step is set as 0.008. The frequency threshold of the "dependable" word is set as 80. That is to say the quasi non-word will not be retrieved if its "dependable" word appears less than 80 times in the dictionary.

BIT Run2: Abnormal POS detection and dependency relation matching are included.

BIT Run3: "De" is a frequently used word in Chinese texts. Due to the low parsing accuracy, plenty of "De" were wrongly replaced in our experiments. To avoid this type of noise, the heuristic rules about the correction of "De" are removed in Run3. Moreover, the transfer probability and the frequency threshold is changed to 0.001 and 100 respectively to tighten the retrieval.

3.4 Final Results

We get three evaluation results (shown in Table 2 and Table 3) from the organizer. Run1 and Run2 are the ones submitted to the Bake-off.

Considering that nearly two-thirds of the errors are wrong word errors, Run1 which doesn't employ any wrong word detection strategies performs poorly on recall. Another reason of the low recall is that the non-word detection module in our system lies on the assumption that there is no more than one wrong character in a non-word. In this way, words such as "勞刀" (嘮叨) and "花鑽品" ("化妝品") are missed.

[10] http://crfpp.googlecode.com/svn/trunk/doc/index.html?source=navbar

[11] http://www.openpr.org.cn/index.php/NLP-Toolkit-for-Natural-Language-Processing/30-Layer-Based-Dependency-Parser/View-details.html

	Approaches	Resources and knowledge	Toolkits
Word segmentation	CRFs based sequential labeling	Sinica corpus	TagPinyin OpenCC CRF++ LDPar
POS tagging	CRFs based sequential labeling	Part of Penn CTB7.0	
Parsing	Layer-based dependency parsing	Part of Penn CTB7.0	
Non-word detection	Word segmentation n-gram based non-word retrieval	SIGHAN Bake-off 2013 test set Word base (Sinica corpus and Tsai's list of Chinese words) Training data released	
Non-word correction	Weighted votes	Word base Pinyin similar pronunciation character set similar shape character set POS tag	
Heuristic rules	Rule-based correction	Training set HIT-CIR Tongyici Cilin (Extended)	
Wrong word detection & correction	POS tagging Dependency parsing Abnormal POS detection Dependency relation matching	Word-POS base Dependency relation base	

Table 4: A summary of approaches and resources employed in our correction system

BIT Run1				
FPR	Accuracy	Precision	Recall	F1-Score
0.3352	Detection Level			
	0.4313	0.3710	0.1977	0.2580
	Correction Level			
	0.4115	0.3206	0.1582	0.2119
BIT Run2				
FPR	Accuracy	Precision	Recall	F1-Score
0.3277	Detection Level			
	0.4482	0.4061	**0.2241**	0.2888
	Correction Level			
	0.4303	0.3650	**0.1883**	0.2484

Table 2: The results of the submitted two runs

BIT Run3				
FPR	Accuracy	Precision	Recall	F1-Score
0.1582	Detection Level			
	0.5245	**0.5670**	0.2072	**0.3034**
	Correction Level			
	0.5122	**0.5359**	0.1827	**0.2725**

Table 3: The results of Run3

According to the results of Run2, wrong word correction based on the knowledge of POS tag and dependency relation shows positive effects both on precision and recall. Since only the POS tag is adopted to detect possible wrong words in the current strategy, the misusage of words which are in the same category will escape. “哪裡” and “那裡” is an typical example. Both of them act as pronouns at most of the time. A broader context and more complex semantic knowledge are required to distinguish them.

Management of the auxiliary word “De” is not given enough attention in our system. Although the corresponding rules designed are delicate and clear, many unexpected cases and poor performance of Chinese language analysis techniques make it not work well in practice. Results of Run3 reveal that the accuracy and precision are improved a lot when heuristic rules for correction of “De” are removed, although the recall decreases to some extent.

Results of Run3 also illustrate that the stricter thresholds for retrieval in non-word detection are helpful to improve the performance. This implies that the perplexity of non-words in this task is not very high and it is not a big problem to differentiate them from correct ones.

4. Conclusion

In this paper we propose a hybrid system for Chinese spelling mistake correction. The n-gram based non-word retrieval, abnormal POS tag based wrong word detection and dependency relation matching based wrong word correction are the key techniques of our system. All the approaches, linguistic resources and toolkits involved are gathered in Table 4.

To further improve the performance of our system, we will try to extend our work in the following aspects: 1) Make full use of the

training data, such as modeling the correct and the incorrect syntactic structures of the data; 2) Apply semantic collocations to elevate the wrong word detection and correction precision.

Acknowledgments

Heyan Huang was supported by the National Program on Key Basic Research Project (973 Program) (No.2013CB329303) and the National Natural Science Foundation of China (No. 61132009). Ping Jian was supported by the National Natural Science Foundation of China (No. 61202244).

References

Chao-Huang Chang. 1995. A new approach for automatic Chinese spelling correction. In *Proceedings of Natural Language Processing Pacific Rim Symposium*, pages 278-283, Seoul, Korea.

Wanxiang Che, Zhenghua Li, and Ting Liu. 2010. LTP: A Chinese Language Technology Platform. In *Proceedings of the Coling 2010: Demonstration Volume*, pages 13-16, Beijing, China.

Ping Jian and Chengqing Zong. 2009. Layer-based Dependency Parsing. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC 23)*, pages 230-239, Hong Kong.

Chao-Lin Liu, Min-Hua Lai, Kan-Wen Tien, Yi-Hsuan Chuang, Shih-Hung Wu, and Chia-Ying Lee. 2011. Visually and phonologically similar characters in incorrect Chinese words: Analyses, identification, and applications. In *ACM Transactions on Asian Language Information Processing (TALIP)*, 10(2): 1-39.

Joseph J. Pollock and Antonio Zamora. 1984. Automatic spelling correction in scientific and scholarly text. In *Communications of the ACM*, 27(4): 358-368.

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.

Nianwen Xue, Fei Xia, Fu-Dong Chiou, and Martha Palmer. 2005. The Penn Chinese TreeBank: Phrase Structure Annotation of a Large Corpus. *Natural Language Engineering*, 11(2): 207-238.

Lei Zhang, Changning Huang, Ming Zhou, and Haihua Pan. 2000. Automatic detecting/correcting errors in Chinese text by an approximate word-matching algorithm. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 248-254.