Chinese Spelling Error Detection and Correction Based on Language Model, Pronunciation, and Shape

Junjie Yu and Zhenghua Li
Provincial Key Laboratory for Computer Information Processing Technology
Soochow University, China
20144227010@stu.suda.edu.cn; zhli13@suda.edu.cn

Abstract

Spelling check is an important preprocessing task when dealing with user generated texts such as tweets and product comments. Compared with some western languages such as English, Chinese spelling check is more complex because there is no word delimiter in Chinese written texts and misspelled characters can only be determined in word level. Our system works as follows. First, we use character-level n-gram language models to detect potential misspelled characters with low probabilities below some predefined threshold. Second, for each potential incorrect character, we generate a candidate set based on pronunciation and shape similarities. Third, we filter some candidate corrections if the candidate cannot form a legal word with its neighbors according to a word dictionary. Finally, we find the best candidate with highest language model probability. If the probability is higher than a predefined threshold, then we replace the original character; or we consider the original character as correct and take no action. Our preliminary experiments shows that our simple method can achieve relatively high precision but low recall.

1 Introduction

Spelling check is a traditional and important preprocessing task for natural language processing, since spelling errors happen in written texts, such as short messages, emails, and so on. Lots of research has been devoted to English spelling error detection and correction. In English spelling error detection and correction, the errors can be classified into "non-word" error and "real-word" error (Kukich, 1992). Unlike English, Chinese words are not separated by space and all characters in Chinese are "real-word". Therefore, automatic word segmentation need to be applied in order to produce words (Zhang et al., 2000). There are many Chinese input methods (Zhang et al., 2005). Different input methods lead to different types of spelling errors. For example, input methods based on pinyin which usually lead to spelling errors of characters sharing similar pronunciations; while input methods based on radical methods usually lead to errors related to character shapes. Huang et al. (2007) proposed a learning model based on Chinese phonemic alphabet to detect Chinese spelling errors. Yeh et al. (2013) presented a method based on N-gram ranked inverted index list to deal with this problem.

2 System Architecture

Our system includes two cascaded components: spelling error detection and spelling error correction, as shown in Figure 1.

2.1 Resources

To train our language mode, we use a portion of Chinese Gigaword version 2.0 (LDC2009T14), which contains about 12 million traditional Chinese sentences. We do not split sentence into words, but treat each character as an individual unit. In other words, our language model is based on character. In order to take advantage of the context information, we train a new language model by reversing all sentences in the corpus. So, we will calculate twice for one character based on this two language models. And the total score is the combination of both.

As misspelled characters in a sentence can only be detected in word level, we construct a word dictionary which contains about 300 thousand words collected from Internet. And the SIGHAN organizer provides a dictionary including about 5000 Chinese characters with other characters in similar pronunciation or shape which can be used in candidate generation.

2.2 Spelling Error Detection

In spelling error detection phase, we propose two methods to deal with this problem. One is to gather the characters which get a low score

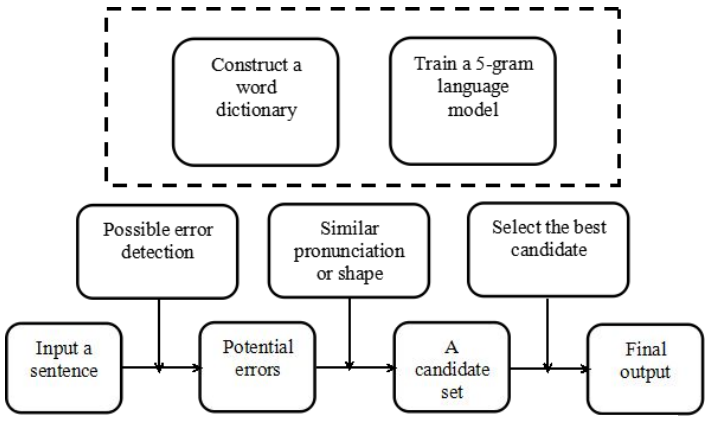

Figure 1: Framework of our proposed system

under language model. Another is to record any independent characters after automatic word segmentation. However, we find both will bring in lots of irrelevant characters though most errors have been discovered. Because Chen et al. (2011) find the average amount of errors in a learners' corpus for a student essay is only 2, we do not want to mark too many error characters to cause false-alarm problem heavily.

In order to make the best of the two methods, we prepare two steps to combine both. Step 1, we calculate the score of each character in a sentence by a forward-backward 5-gram language model. While the score is less than the threshold, the character and its location are sent to Step 2. To find as more errors as possible, we set the threshold in a quite tight value. However, this will result in more irrelevant characters which confuse the system. In Step 2, we need to filter the characters generated in Step 1. We will judge the character whether it can construct a word. Otherwise, we make the assumption that it may be a spelling error which means we are still not sure about it. Anyhow, we will send the results to next phase.

2.3 Spelling Error Correction

In spelling error correction phase, we firstly generate a candidate set for the error character. Characters of similar pronunciations are the most common source of spelling errors (Wu et al., 2013). But there still exist some errors from similar shape (Liu et al., 2011). So, the candidate generation is based on a similar pronunciation or shape dictionary. For more details about the dictionary, please refer to Yeh et al., (2013). Secondly, each character in the candidate set will be tested whether it can form a legal word with its neighbors. Here, the character which can construct a legal word with its neighbors will be left for calculating its score by the language model. After filtering, the number of candidates has been reduced which will bring two benefits: most candidates that have been cut are irrelevant characters and less candidates makes the system be more efficient. At last, the best candidate means one character gets the highest score under a forward-backward 5-gram language model and the score is higher than the threshold. If existing, the original character finally will be recognized as an error character and it will be replaced by the best candidate.

We only use the language model to choose the best candidate because we find that the language model can get a quite high accuracy if we can provide a suitable candidate set successfully.

3 Experimental Analysis

In this paper, we use 300 sentences from the final test of SIGHAN Bake-off 2013 as our training data and 1000 sentences provided by the SIGHAN organizer are our test data.

In our training data, there are 402 error characters in total. We first test the recall of the spelling error detection based on language model.

Function threshold	Language model	
	Recall(%)	#Characters
-4	26.67	2
-3	57.00	6
-2	86.67	18
-1	96.32	38

Table 1: Results on error detection

Table 1 shows that when threshold become tighter, the recall is higher. However, the average number of characters increases quickly. Average number of characters means how many

The size of window is 4, so, if the character "竟" is the target character, then it will generate such words:
逆竟 竟時 到逆竟 逆竟時 遇到逆竟 到逆竟時 遇到逆竟時
If the character in the window is a punctuation or the start or end of the sentence, the system will set the character be a new boundary.

Figure 2: Example to show how to construct a word

Run	False Positive Rate	Detection Level				Correction Level			
		Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
1	0.2524	0.4539	0.3881	**0.1601**	**0.2267**	0.4426	0.3527	**0.1375**	**0.1978**
2	**0.032**	**0.5292**	**0.7385**	0.0904	0.1611	**0.5235**	**0.7119**	0.0791	0.1424

Table 2: Results of our error detection and correction subtask

characters are marked as error characters by our system. The average length of sentences in our training data is about 70 characters. When the threshold has been set to be -1, more than half of the characters in a sentence have been marked as errors on average. Though the recall is very high in this case, too many correct characters have been recognized as errors. So we prefer to give up the high recall rather than reserve too many irrelevant characters. As we mentioned in Section 2.2, the average number of spelling errors in a sentence is quite low. Threshold = -2 only leads to a slight reduce in recall but the average number of characters have been cut down by half.

As shown in Figure 1, we firstly prepare two resources: a forward-backward 5-gram language model and a word dictionary. As described in previous sections, such two resources will be applied into both spelling check detection and correction. Then, we start to detect the error characters in a sentence. For each character in a sentence, if its score which calculated by the forward-backward 5-gram language model is less than the threshold value, it will be sent to next phase. And the threshold is set at -2 as we discussed before. Next, we will test the character for constructing a word. We set the size of the window at 4 which means the target character can be combined with its neighbors at a distance of 4 characters. For example, Figure 2 describes the details.

After the target character is combined with its neighbors, we will look up the word dictionary. While none of combinations can be found in the word dictionary, we make the assumption that the target character may be an error. In this example, none of these 7 words can be found in word dictionary. So, the character "竟" in this sentence would be marked as an error and sent to next phase.

In spelling check correction phase, we first generate candidates by similar pronunciation or shape. Then the candidates are filtered by constructing a word. This time, we reserve the candidates which can construct a word with its neighbors. At last, the rest candidates will be ranked by language model. The best candidate with its score higher than threshold will replace the original character in the sentence. Here, the threshold is the same with the value in detection level.

4 Final Results

In this bake-off, there are 1000 sentences and all sentences contain at least more than one error. Table 2 shows that the F1 score is very low because we can only find a small portion of all errors. However, the false positive rate and precision is satisfactory especially for the false positive rate. Such results are consistent with our main idea that we choose to under-correct rather than over-correct.

We can see that the performance in detection level and correction level are similar. As described in previous sections, only when the best candidate has been found, we will make the conclusion that the target character is a spelling error. The performance in correction level only has a slight decrease compared with the detection level. But the unavoidable reality is that the recall is not good.

5 Conclusions

Based on n-gram language model and judging a character whether it can form a legal word with its neighbors, a simple approach is proposed to detect and correct the spelling errors in traditional Chinese text. To find the spelling errors in sentence, the language model and a word dictionary are both used. And in order to reduce the false positive rate, the system only treats the character as a spelling error when the best candidate has been found.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Grant No. 61373095, 61333018).

Reference

Chen, Y. Z., Wu, S. H., Yang, P. C., & Ku, T. (2011). *Improve the detection of improperly used Chinese characters in students' essays with error model*. International Journal of Continuing Engineering Education and Life Long Learning, 21(1), 103-116.

Huang, C. M., Wu, M. C., & Chang, C. C. (2007). *Error detection and correction based on Chinese phonemic alphabet in Chinese text*. In Modeling Decisions for Artificial Intelligence (pp. 463-476). Springer Berlin Heidelberg.

Kukich, K. (1992). *Techniques for automatically correcting words in text*. ACM Computing Surveys (CSUR), 24(4), 377-439.

Liu, C. L., Lai, M. H., Tien, K. W., Chuang, Y. H., Wu, S. H., & Lee, C. Y. (2011). *Visually and phonologically similar characters in incorrect Chinese words: Analyses, identification, and applications.* ACM Transactions on Asian Language Information Processing (TALIP), 10(2), 10.

Wu, S. H., Liu, C. L., & Lee, L. H. *Chinese Spelling Check Evaluation at SIGHAN Bake-off 2013.* In Sixth International Joint Conference on Natural Language Processing (p. 35).

Yeh, J. F., Li, S. F., Wu, M. R., Chen, W. Y., & Su, M. C. (2013). *Chinese Word Spelling Correction Based on N-gram Ranked Inverted Index List.* In Sixth International Joint Conference on Natural Language Processing (p. 43).

Zhang, L., Huang, C., Zhou, M., & Pan, H. (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 (pp. 248-254). Association for Computational Linguistics.

ZHANG, Y. S., YU Shi-wen. (2006). *Summary of Text Automatic Proofreading Technology*. Application Research of Computers, 6.