# Alibaba at IJCNLP-2017 Task 1: Embedding Grammatical Features into LSTMs for Chinese Grammatical Error Diagnosis Task

**Yi Yang** and **Pengjun Xie** and **Jun Tao** and **Guangwei Xu** and **Linlin Li** and **Si Luo**
{zongchen.yy, Jun.Tao, linyan.lll, luo.si}@alibaba-inc.com
chengchen.xpj, kunka.xgw@taobao.com@taobao.com

## Abstract

This paper introduces Alibaba NLP team system on IJCNLP 2017 shared task No. 1 Chinese Grammatical Error Diagnosis (CGED). The task is to diagnose four types of grammatical errors which are redundant words (R), missing words (M), bad word selection (S) and disordered words (W). We treat the task as a sequence tagging problem and design some handcraft features to solve it. Our system is mainly based on the LSTM-CRF model and 3 ensemble strategies are applied to improve the performance. At the identification level and the position level our system gets the highest F1 scores. At the position level, which is the most difficult level, we perform best on all metrics.

## 1 Introduction

Chinese is one of the old and versatile languages in the world. In its long use of history, it has accumulated a lot of difference from other languages. For example, compared to English, Chinese has neither singular/plural change, nor the tense changes of the verb. It has more flexible expression but loose structural grammar, uses more short sentences but less clauses. It also has more repetitions, while English omits a lot. e.g. “Ambition is the mother of destruction as well as of evil. 野心不仅是罪恶的根源，同时也是毁灭的根源。” In English, “the mother of” evil is completely omitted. But in Chinese, it need be expressed clearly in the sentence. All these differences bring a lot of trouble to new learners. With the surging of Chinese Foreign Language(CFL) Learners, an automatic Chinese Grammatical Error Diagnosis will be very helpful. English Grammar Check has been studied for many years, with a lot of progress being made, while Chinese Grammar Check, Error Detection and Correction study is much less until very recently. Though the two languages have a lot of difference between them, they also share similarities, such as the fixed collocation of words. Experience can be obtained from the English NLP study. The CGED Task gives Chinese NLP researchers an opportunity to build and develop the Chinese Grammatical Error Diagnosis System, compare their results and exchange their learning methods.

This paper is organized as follows: Section 2 describes the Shared Task. Section 3 introduces some related work both in English and in Chinese. Section 4 describes our methodology, including feature generation, model architecture and ensemble. Section 5 shows the data analysis and final result on the evaluation data set. Section 6 concludes the paper and shows future work.

## 2 Chinese Grammatical Error Diagnosis

The NLPTea CGED has been held since 2014. It provides several sets of training data written by Chinese Foreign Language(CFL) leaner. In these training data sets, 4 kinds of error have been labeled: R(redundant word error), S(word selection error), M(missing word error) and W(word ordering error). With a test data set provided, an automatic Chinese Grammatical Error Diagnosis System need to be developed to detect: (1) If the sentence is correct or not; (2) What kind of errors the sentence contains; (3) the exact error position. One thing need additional attention is that each sentence may contain more than one error. Some

*Proceedings of the 8th International Joint Conference on Natural Language Processing, Shared Tasks*, pages 41–46,
Taipei, Taiwan, November 27 – December 1, 2017. ©2017 AFNLP

Table 1: Typical Error Examples.

| Error Type | Original Sentence | Correct Sentence |
|---|---|---|
| M(missing word) | 我河边散步的时候。 | 我在河边散步的时候。 |
| R(redundant word) | 流行歌曲告诉我们现在的我们的心理状态。 | 流行歌曲告诉我们现在的心理状态。 |
| S(word selection) | 还有其他的人也受被害。 | 还有其他的人也受伤害。 |
| W(word order) | 听多流行歌就会对唱那首歌的歌手痴迷。 | 多听流行歌就会对唱那首歌的歌手痴迷。 |

typical examples are shown in table 1:

All these metrics are measured in the test results: False Positive Rate, Accuracy, Precision, Recall and F1.

## 3 Related Works

Grammatical Error Detection and Correction in CoNLL2013 and CoNLL2014 shared Task attracts a lot of English NLP researchers and different approaches were adopted by the participants, e.g. hand-crafted rules, statistical model, translation model and language model(Ng et al., 2014). The study on collocation also shows great improvement in Grammatical Error Detection(Chen et al., 2016; Ferraro et al., 2014). The long short term memory (LSTM) has proved its efficiency in NLP general sequence related modeling(Hochreiter and Schmidhuber, 1997) and Grammatical Error Diagnosis(Zheng et al., 2016). Chinese Grammatical Error Detection research is much less compared with English, and lack of large publicly published data also hinders its study. In 2012, a CRF based Classifier is proposed to detect the word ordering error(Yu and Chen, 2012). In 2014, Cheng etc propose a rule based Diagnosis System(Chang et al., 2014). NLPTea 2014/2015/2016 CGED shared task also provides the Chinese NLP researchers a chance to publish their progress on this topic(Yu et al., 2014; Lee et al., 2015, 2016). HIT propose a CRF+BiLSTM model based on character embedding on bigram embedding(Zheng et al., 2016). CYUT propose a CRF based model on collocation, Part-Of-Speech (POS) linguistic features(Ferraro et al., 2014).

## 4 Methodology

### 4.1 Model Description

Same with the method of most teams in 2016, we treat the CGED problem as a sequence tagging problem. Specifically, given a sentence x, we generate a corresponding label sequence y using the BIO encoding (Kim et al., 2004). The HIT team (Zheng et al., 2016) had used traditional CRF model and LSTM-based models to solve the sequence tagging problem. However, it's straightforward to combine the two models, that results to a relatively new model named LSTM-CRF (Huang et al., 2015). With the help of the CRF, the LSTM-CRF model can predict the tagging sequence better. For instance, the LSTM-CRF model can avoid predicting the 'I-X' error compared with single LSTM model. Same with the HIT team, we use the bidirectional LSTM unit as the RNN unit to model the input sequence. The model architecture is illustrated in figure 1. As you can see from the figure, the features are not specific in the architecture, we will explain them in the next section.

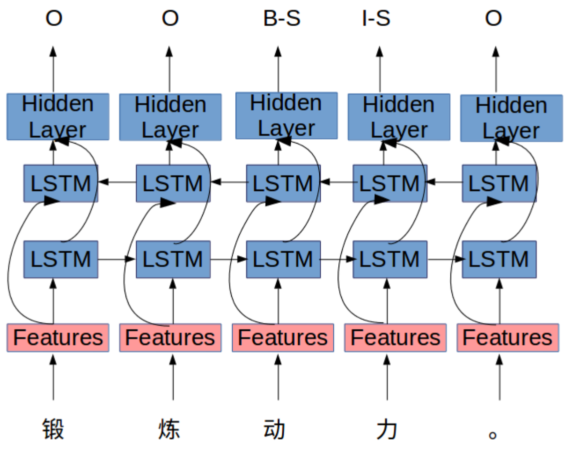

Figure 1: The LSTM-CRF model we applied. Note that we use bi-directional LSTM as the basic neural unit.

### 4.2 Feature Engineering

Since the lack of training data, the task heavily depends on the prior knowledge, such as POS feature, provided by external data or domain expert. In another word, the feature engineering is very important in this task. We designed several features. Figure 2 illustrates the features we used. Next we will introduce each feature one by one.

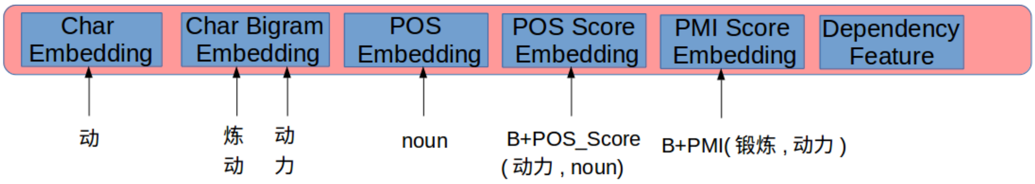

Figure 2: The features we used in this task. Use the feature at “动” in “锻炼动力。” as an example.

- Char: We solved the sequence tagging problem at the character level, it's straightforward to use the char embedding as an input feature. We randomly initialized the char embedding.

- Char Bigram: Bigram is an informative and discriminating feature in this task because it makes the model easily to learn the degree of collocation between neighbor chars. We obtained the bigram embeddings the same way as the HIT team.

- POS: Part-of-speech-tagging (POST) of words containing verb, adverb, adjective, noun. character's POS tag is decided by its word POS tag, B-pos indicating the beginning character's POS tag while I-pos indicating the middle and end characters'.

- POS Score: By analyzing the training data, we found that the POS tag of lots of error words are not the tag the word showing most of times. For example, the POS tag for the word “损伤” in the sentence “抽烟明显损伤身体健康” is VV, which is not its usual tag. This is a S error and the word should be changed to “损害”. We used the discrete probability of the word's tag as a feature. The probabilities are calculated by the Gigawords dataset (Graff and Chen, 2005). Note that we also attached position indicators to this feature in the same way as the POS feature.

- Adjacent Word Collocation: In the training data, we found that wrong collocation happens between the neighbor words. Based on this observation, we calculated a point-wise mutual information (PMI) (Church and Hanks, 1990) score using the Gigawords dataset for each adjacent words pair. The formula for calculating the PMI score is:

$$PMI(w1, w2) = log(\frac{p(w1, w2)}{p(w1) * p(w2)}) \quad (1)$$

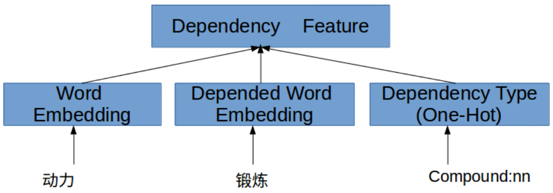

Figure 3: The dependency feature sub-network.

For example, the score of the words pair <“锻炼”, “动力”> in the sentence ”天天锻炼动力。” is very low, while the score of < “锻炼”, “能力”> is much higher. Similarly, the score of <“一部”, “电影”> is much higher than <“一台”, “电影”>. We embedded the discrete PMI score into a low dimension vector as one of input features for our LSTM-CRF model. Since we solved the task at the character level, we also added the position indicators to the discrete PMI score.

- Dependent Word Collocation: The adjacent word PMI score represents the collocation between adjacent words. However the collocation relationship is not limited to the adjacent words. For example, the word “一个” in sentence “他刚刚出版了一个新的小说.” is used to modify the word “小说”. By using dependency parser, we can get the dependent word for each word. At each position, we model the collocation relationship feature through a sub-network. The input of the sub-network is a concatenation of the word's embedding, the dependent word's embedding and the dependent type. Figure 3 illustrates the sub-network.

### 4.3 Model Ensemble

Because of random initialization, random dropout (Srivastava et al., 2014) and random training order, the model result may be different at the end of each training. After the experiments, we found that each model gave very different predicted result even they share the same set of pre-trained parameters. According to this situation, we designed 3 different ensemble methods to improve the result.

The first ensemble method was just simple merging all results. We found that the precision of a single model was much higher than the recall. It's straightforward to merge different model results to increase the recall. After applying the merging-all strategy, the recall was increased as expected, but the precision decreased greatly. So we designed a second ensemble method to balance the precision and the recall. We used the score generated by the LSTM-CRF to rank the errors generated by each model. We deleted the final ranking 20% errors for each single model and then merged the rest results. The second ensemble method can increase the precision to some extent, but it is hard to exceed the single model. In order to increase the precision further, the third ensemble method we applied was voting. Voting is really a powerful method in this task because it can greatly increase the precision while keep the recall same as long as the model quantity is big enough.

In all of our experiments, we used 4 different groups of parameters and trained 2 models for each parameter groups, so totally we used 8 models. Surprisingly, we found that the 3 different ensemble methods achieved the best F1-score respectively in the detection level, identification level and position level.

# 5 Experiments

## 5.1 Data Split and Experiment Settings

We collected data sets of year 2015, 2016, and 2017, of which 20% data of year 2017 are used for validation and the rest for training. In our experiments, we found that adding the correct sentence could improve the result, so all correct sentence were added into the training set.

We pretrained bigram-char embeddings and word embeddings using the Gigawords dateset and fixed them when training models. For other parameters, we randomly initialized them.

The performance metrics were calculated at three levels, detection-level, identification-level and position-level. For each level, false positive rate, accuracy, precision, recall and F1-score were included.

## 5.2 Experiment Results

### 5.2.1 Results on Validation Dataset

We used the validation dataset to select the best hyper-parameters for single LSTM-CRF model. Among the parameters, we chose 4 groups of parameters and trained 2 models for each parameters group to ensemble. There exists a certain degree of difference among the 4 groups, while the model performance is also good for each single parameters group. Table 2 shows the result. As we can see, the ensemble method 1 (simple merging all) has the highest recall at all 3 levels as intended. It gets the best F1-score at detection level at the same time. The ensemble method 2 (merging after removing low-score errors) has relatively good performance at all 3 levels, especially gets the highest F1-score at detection level. The ensemble method 3 (voting) gets the best precision at all 3 levels. It gets the best F1-score at position level. As intended, we can figure out that the precision is increasing from ensemble method 1 to method 3, and the recall is decreasing. Furthermore, all 3 ensemble methods are very effective compared with single model and achieve great improvement on F1-score at all 3 levels.

### 5.2.2 Results on Evaluation Dataset

When testing on the final evaluation dataset, we merged our training dataset and validation dataset, and retrained our models. Table 3 shows the results of our 3 submissions, each submission corresponds to an ensemble method.

Our system achieves the best F1 scores at identification level and position level, and achieve the best recall rates in all 3 levels. At detection level, if not taking all sentences to be error into consideration, our F1 score is also the highest. At the position level, our system perform best on all the metrics. It is a little pity that the best F1-score we have gotten at position level is just 0.2693. To some extent, it is because that the problem is very hard and the training data is not enough. However, we are optimistic about the solving the CGED problem completely.

# 6 Conclusion and Future Work

This article describes our system approach in IJCNLP TASK1, CGED. We combined some handcraft features, like the POS, dependency features and PMI Score, etc, and trained LSTM-CRF models based on these features. Later, we designed

Table 2: Results on Validation Dataset. Single Model refers to the LSTM-CRF model. Ensemble Method 1 refers to merging results of 8 models. Ensemble Method 2 refers to merging results after removing the 20% low-score errors. Ensemble Method 3 refers to voting. We keep errors occurred at least 2 times among 8 results when voting.

| Method | Detection Level | | | Identification Level | | | Position Level | | |
|---|---|---|---|---|---|---|---|---|---|
| | Precision | Recall | F1 | Precision | Recall | F1 | Precision | Recall | F1 |
| Single Model | 0.66 | 0.46 | 0.54 | 0.52 | 0.26 | 0.36 | 0.31 | 0.13 | 0.19 |
| Ensemble Method 1 | 0.55 | **0.84** | **0.6745** | 0.25 | **0.62** | 0.36 | 0.15 | **0.29** | 0.202 |
| Ensemble Method 2 | 0.56 | 0.81 | 0.66 | 0.39 | 0.58 | **0.47** | 0.17 | 0.27 | 0.21 |
| Ensemble Method 3 | **0.67** | 0.56 | 0.61 | **0.54** | 0.36 | 0.43 | **0.30** | 0.20 | **0.24** |

Table 3: Results on Evaluation Dataset

| Method | Detection Level | | | Identification Level | | | Position Level | | |
|---|---|---|---|---|---|---|---|---|---|
| | Precision | Recall | F1 | Precision | Recall | F1 | Precision | Recall | F1 |
| Ensemble Method 1 | 0.6792 | **00.8284** | **0.7464** | 0.453 | **0.6006** | 0.5164 | 0.1949 | **0.2941** | 0.2344 |
| Ensemble Method 2 | 0.686 | 0.7986 | 0.738 | 0.4791 | 0.5657 | **0.5188** | 0.2169 | 0.2752 | 0.2426 |
| Ensemble Method 3 | **0.7597** | 0.5714 | 0.6523 | **0.6007** | 0.3756 | 0.4622 | **0.3663** | 0.213 | **0.2693** |

different ensemble strategies for the 3 levels. The results show that our strategies are valid. At the identification level and the position level we get the highest F1 scores. At detection level, without taking all sentences to be error into consideration, our F1 score is also the highest. At the position level, which is the most difficult level, our accuracy, precision, recall, and F1 score are the highest.

In the future, with more training data, we hope not only to identify the error, but also directly correct the error based on models like seq2seq (Sutskever et al., 2014). Chinese Grammatic Error Correction will be more helpful for the non-native language learners to learn Chinese. Currently, We used many handcraft features in this task. In the future, we will design a more automatic neural architecture to get an end-to-end grammatical error recognition system by combining the pre-trained language model and other related multi-task models, which will help the recognition and correction of grammatical errors.

## 7 References

## References

Tao Hsing Chang, Yao Ting Sung, Jia Fei Hong, and Jen I Chang. 2014. Knged: A tool for grammatical error diagnosis of chinese sentences. In *22nd International Conference on Computers in Education, ICCE 2014*. Asia-Pacific Society for Computers in Education.

Po-Lin Chen, Shih-Hung Wu, Liang-Pu Chen, and Ping-Che Yang. 2016. Improving the selection error recognition in a chinese grammar error detection system. In *Information Reuse and Integration (IRI), 2016 IEEE 17th International Conference on*, pages 525–530. IEEE.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

Gabriela Ferraro, Rogelio Nazar, Margarita Alonso Ramos, and Leo Wanner. 2014. Towards advanced collocation error correction in spanish learner corpora. *Language resources and evaluation*, 48(1):45–64.

David Graff and Ke Chen. 2005. Chinese gigaword. *LDC Catalog No.: LDC2003T09, ISBN*, 1:58563–58230.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi, and Nigel Collier. 2004. Introduction to the bio-entity recognition task at jnlpba. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications*, pages 70–75. Association for Computational Linguistics.

Lung-Hao Lee, RAO Gaoqi, Liang-Chih Yu, XUN Endong, Baolin Zhang, and Li-Ping Chang. 2016. Overview of nlp-tea 2016 shared task for chinese grammatical error diagnosis. In *Proceedings of the 3rd Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA2016)*, pages 40–48.

Lung Hao Lee, Liang Chih Yu, and Li Ping Chang. 2015. Overview of the nlp-tea 2015 shared task for chinese grammatical error diagnosis. In *Overview of the NLP-TEA 2015 Shared Task for Chinese Grammatical Error Diagnosis*.

HT Ng, SM Wu, and Y Wu. Ch. hadiwinoto, and j. tetreault. 2013. the conll-2013 shared task on grammatical error correction. *Proceedings of CoNLL: Shared Task*.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The conll-2014 shared task on grammatical error correction. In *CoNLL Shared Task*, pages 1–14.

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

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

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

Bo Zheng, Wanxiang Che, Jiang Guo, and Ting Liu. 2016. Chinese grammatical error diagnosis with long short-term memory networks. *NLPTEA 2016*, page 49.