Chinese Grammatical Error Diagnosis with Graph Convolution Network and Multi-task Learning

Yikang Luo[†] [*], **Zuyi Bao**[‡], **Chen Li**[‡] and **Rui Wang**[‡]
[†] School of Software, Shanghai Jiao Tong University, Shanghai, China
[‡]Alibaba Group
[†]luoyikang@sjtu.edu.cn
[‡]{zuyi.bzy,puji.lc,masi.wr}@alibaba-inc.com

Abstract

This paper describes our participating system on the Chinese Grammatical Error Diagnosis (CGED) 2020 shared task. For the detection subtask, we propose two BERT-based approaches 1) with syntactic dependency trees enhancing the model performance and 2) under the multi-task learning framework to combine the sequence labeling and the sequence-to-sequence (seq2seq) models. For the correction subtask, we utilize the masked language model, the seq2seq model and the spelling check model to generate corrections based on the detection results. Finally, our system achieves the highest recall rate on the top-3 correction and the second best F1 score on identification level and position level.

1 Introduction

Chinese has become an influential language all over the world. More and more people choose Chinese as a second/foreign language (CSL/CFL). Their writings usually contain grammatical errors including spelling and collocation errors. For instance, a Japanese learner may write "我苹果喜欢" (I apple like) while its correct expression should be "我喜欢苹果" (I like the apple). The inconsistency of Chinese and Japanese grammatical structures will lead to different expression order. Grammatical structure in Chinese is different from other languages and affects expression.

The previous works used to do feature engineering including pretrained features and parsing features to improve performance. In this paper, we fertilize the representations from BERT with the syntactic dependency tree and propose a multi-task learning of error detection and correction. We employ three strategies based on BERT for correction based on detection results. Experiment shows that our system is effective on both detection and correction level. Our contributions are summarized as follows:

- We propose the graph-convolutional-network-based (GCN-based) approach to improve the baseline model's understanding of syntactic dependency and introduce the sequence-to-sequence (seq2seq) model to improve the performance of the original sequence labeling task.
- We combine three approaches including the masked language model, the seq2seq and the Chinese spelling check to correct the erroneous sentences based on the detection results.
- We get the highest recall rate of the top-3 correction and the second highest F1 score at the identification level and position level of the detection.

This paper is organized as follows. Section 2 describes the CGED task. Section 3 describes our system for grammatical error detection and correction. Section 4 reports the experimental results conducted by the proposed methods. Section 5 concludes this work.

```
<TEXT id="200205215525100007_2_2x1">
所以我认为安乐死绝对不要允许。
</TEXT>
<CORRECTION>
所以我认为安乐死绝对不能被允许。
</CORRECTION>
<ERROR start_off="11" end_off="12" type="S"></ERROR>
<ERROR start_off="13" end_off="13" type="M"></ERROR>
```

Figure 1: A sample of the training data.

2 Chinese Grammatical Error Diagnosis

The CGED shared task has been held since 2014. Several sets of training data have been released written by CFL learners which contain a lot of grammatical errors. For detection, the CGED defines four types of errors: (1) R (redundant word

[*]This work was done when Yikang Luo was an intern in Alibaba Group.

Proceedings of the 6th Workshop on Natural Language Processing Techniques for Educational Applications, pages 44–48
Suzhou, China, December 4, 2020. ©2020 Association for Computational Linguistics

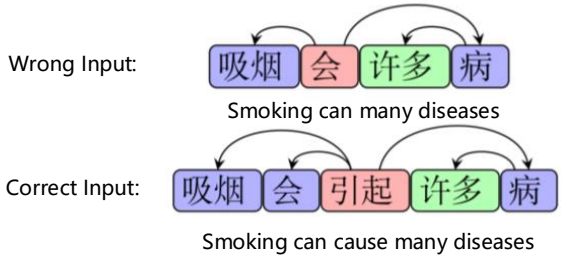

Figure 2: The different structures of syntax tree between the error sentence and right sentence.

errors);(2) M (missing words); (3) W (word ordering errors);(4) S (word selection errors) as shown in Figure 1. The performance is measured at detection level, identification level and position level. For correction, systems are required to recommend at most 3 corrections for missing and selection errors.

3 System Description

3.1 BERT-CRF

Previous works regard the detection task as the sequence labeling problem solving by the LSTM-CRF model (Huang et al., 2015). We introduce the BERT model (Devlin et al., 2018) to replace the LSTM model. For different pretrained BERT models, we choose the StructBERT (Wang et al., 2019) as our main body model. One of the reasons is that its pretraining strategy Word Structural Objective accepts sentences with wrong word order, which is similar to the word ordering errors in this task.

3.2 BERT-GCN-CRF

Previous works (Yang et al., 2017; Fu et al., 2018) spent a lot of effort in feature engineering including pretrained features and parsing features. Part-of-speech-tagging(POS), and dependency information are the most important parsing features, which indicates to us the task is closely associated with the structure of the sentence syntactic dependency. Specifically, the redundant error and the missing error sentences syntax tree are very different from the correct sentences as the Figure 2 shows.

To understand the dependency structure of an input sentence better, we introduce the Graph Convolution Network (GCN) (Kipf and Welling, 2016; Marcheggiani and Titov, 2017).

Figure 3 shows our BERT-GCN-CRF model architecture. We will explain each part in detail.

Word Dependency We split the input sentences into words and obtain the dependency relation of each word. As BERT acts on character level in Chinese, we add extra dependency edges for one word to all of characters of the word.

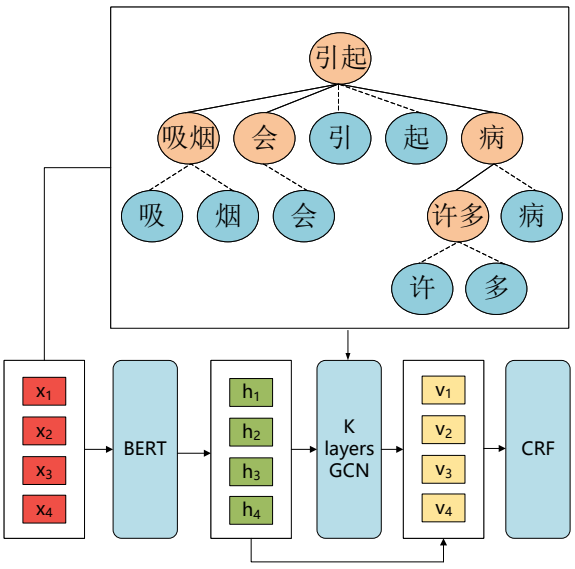

Figure 3: The structure of BERT-GCN-CRF model

Graph Convolution Network The multi-layer GCN network accepts the high-level character information obtained by the BERT model and the adjacency matrix of the dependency tree. The convolution operation is adopted for each layer.

$$f(A, H^l) = AH_l W_l^g \tag{1}$$

where $W_l^g \in R^{D\times D}$ is a trainable matrix for the l-th layer, A is the adjacency matrix of the dependency tree, $H_l = (h_1, h_2, ..., h_n)$ is the hidden state of the characters. Words use the same input representation in the network to indicate the dependency relation of the characters.

Accumulated Output After the graph convolution network, we concatenate the representation H_l for the l-th layer and the BERT hidden state passing to a linear classifier as the input of the CRF layer.

$$V = Linear(H_0 \oplus H_l) \tag{2}$$

CRF Layer A CRF layer is introduced to predict the sequence tags for each token.

$$Score(X, Y) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} V_{i,y_i} \tag{3}$$

$$P(Y|X) = \frac{\exp(Score(X, Y))}{\sum_{\hat{Y}} \exp(Score(X, \hat{Y}))} \tag{4}$$

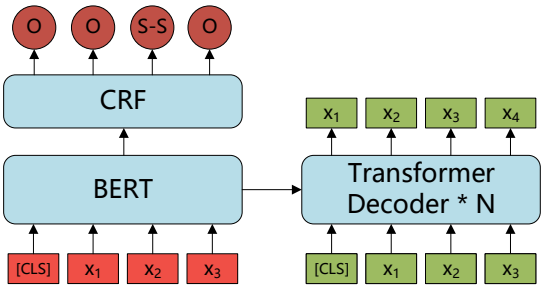

Figure 4: The structure of the multi-task learning

where X, Y, $\hat{Y}$ represents the input sequence, the truth tag sequence, and an arbitrary label sequence, V represents the emission scores, and A is the transition scores matrix of the CRF layer. The loss function is calculated as:

$$Loss_{sl} = -\log(P(Y|X)) \quad (5)$$

We use Viterbi Decoding (Huang et al., 2015) to inference answers.

3.3 Multi-task

Most previous works trained their model by the sequence tags (Yang et al., 2017; Li and Qi, 2018; Fu et al., 2018). We utilize not only tags but also correct sentences during the training process. Correct sentences are important for providing better representation in the hidden state. Moreover, with the correct sentences, the model can have a better understanding of the original meaning of the input sentence. Therefore, we introduce the seq2seq task (Sutskever et al., 2014; Vaswani et al., 2017) treating the training process as multi-task learning. As shown in Figure 4, the sequence labeling model is the encoder in our structure combined with the transformer decoders to predict the truth sentence. The sequence labeling loss and the seq2seq loss are combined by a hyper-parameter w:

$$Loss = w * Loss_{sl} + (1 - w) * Loss_{seq2seq} \quad (6)$$

During the inference phase, we use the sequence labeling module to predict answers.

3.4 Ensemble Mechanism

To take advantage of the predictions from multiple error detection models, we employ a two-stage voting ensemble mechanism.

In the first stage, predictions from multiple models are utilized to distinguish the correct sentences from the sentences with grammar errors. Specifically, we label the sentences as correct when less than θ_{det} models detect errors in the sentence.

In the second stage, an edit-level voting is applied to the predictions for the sentences with grammar errors. We only include edits that appear in the predictions of more than θ_{edit} models.

In the experiments, we use the grid search to choose the θ_{det} and θ_{edit} according to the performance on the validation data.

3.5 Correction

For the selection (S) and missing (M) errors, we introduce two methods to generate corrections.

In the first method, we insert mask tokens into the sentence and use BERT to generate correction by replacing mask tokens one by one in an autoregressive style. In the experiments, we insert 1 to 4 mask tokens to cover most of the cases and adopt the beam-search algorithm to reduce the search complexity.

In the second method, we generate the candidates by a seq2seq model trained by mapping the wrong sentences to the correct sentences. According to the detection result, we keep generating next characters until the correct character appears within the beam-search algorithm, and then replace the incorrect span.

3.6 Chinese Spelling Check

The Chinese Spelling Check (CSC) models are utilized to handle spelling errors. We combine the results from a rule-based checker and a BERT-based spelling checker learned from the CSC data (Bao et al., To appear). The rule-based checker is good at handling non-word errors. The BERT-based checker treats the CSC task as a sequence labeling problem and is good at handling real-word errors. The corrections are then segmented and aligned with the input sentences to get the edited results on the word-level. As the CSC models show a high precision on the validation data, we treat the spelling errors as word selection errors and directly merge the CSC results into the detection and correction results for our final submissions.

4 Experiments

4.1 Data and Experiment Settings

We trained our models by CGED 2015, 2016, 2017, 2018 training data and used pairs of error sentences

Method	Detection			Identification			Position		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
BERT-CRF	78.4	76.7	77.5	61.4	50.8	55.6	40.7	28.7	33.6
BERT-GCN-CRF	65.5	**91.2**	76.3	53.1	**62.7**	57.5	34.7	36.1	35.4
BERT-CRF + multi-task	65.5	90.8	75.9	52.2	60.6	55.4	36.0	36.2	36.1
StrcutBERT-CRF	72.1	89.3	79.8	60.0	60.7	60.3	42.1	36.2	38.9
StrcutBERT-GCN-CRF	77.6	84.5	80.9	64.1	58.0	60.9	45.7	35.3	39.8
StrcutBERT-CRF + multi-task	73.5	88.4	80.3	60.7	62.6	61.6	42.0	38.7	40.2
Ensembled Model	**85.5**	78.6	**81.9**	**68.1**	62.1	**65.0**	**48.0**	**41.3**	**44.4**

Table 1: The results of single models and ensemble model on validation dataset.

	Detection			Identification			Position			Correction			Top-3 Correction		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
Run#1	**92.8**	84.4	88.4	72.2	61.2	66.3	43.7	33.7	38.1	13.6	11.0	12.1	7.7	18.4	10.8
Run#2	91.6	**86.4**	89.0	71.9	54.5	62.0	42.4	27.3	33.2	**18.9**	12.5	15.0	**9.6**	17.7	12.5
Run#3	92.5	86.0	**89.1**	**72.3**	**62.9**	**67.3**	**44.3**	**36.1**	**39.8**	17.8	**15.3**	**16.5**	9.3	**22.8**	**13.3**
Top 1	85.7	97.6	91.2	73.6	62.1	67.4	47.2	35.4	40.4	28.5	14.2	18.9	32.2	13.3	18.9

Table 2: Final results on the official evaluation testing data. "Run #1" represents the ensemble model with correction. "Run #2" represents the single best model with correction. "Run #3" represents the ensemble model with correction and CSC. "Top 1" reports the highest F1 score with its precision and recall at different levels.

and correct sentences for the seq2seq training without extra data. We used the CGED-2018 testing dataset as our validation dataset. We introduced the BIOES (Ratinov and Roth, 2009) scheme for tagging.

Language Technology Plantform (LTP) (Che et al., 2010) was introduced to obtain the dependency tree. The hyper-parameters are selected according to the performance on the validation data through official metrics. For the GCN model, the hidden vector size was 256 with 2 layers. The batch size, learning rate, and GCN dropout were set to 32, 1e-5, 0.2. For the multi-task model, the batch size, learning rate and w are set to 32, 3e-5, 0.9.

Transformer decoder parameters are initialized from the BERT parameters as much as possible.

4.2 Validation Results

We use the BERT-CRF (base) and StructBERT-CRF (large) as our baseline models. The results of different methods are listed in Table 1. The StructBERT-CRF (large) overwhelms the BERT-CRF (base) model by obtaining a significantly better recall rate on all levels.

Both GCN and multi-task approaches achieve improved performance over the baseline model in identification level and position level. Thus, we select StrcutBERT-GCN-CRF and StructBERT-CRF + multi-task models for ensemble.

To obtain diverse single models for ensemble, we trained 38 StrcutBERT-GCN-CRF models and 65 StructBERT-CRF + multi-task models with different random seeds and hyper-parameters. As shown in Table 1, the proposed ensemble mechanism achieves an obvious improvement over the single models.

Model	Type	Precision	Recall	F1
BERT-CRF	R	**42.6**	28.3	34.0
BERT-GCN-CRF	R	36.2	**34.8**	**35.4**
BERT-CRF	M	**36.3**	26.6	30.7
BERT-GCN-CRF	M	32.8	**30.0**	**31.7**

Table 3: The position level performance of the BERT-CRF and BERT-GCN-CRF model on validation data. "R" denotes the redundant error and "M" denotes the missing error.

We evaluated the contribution of the GCN network of the redundant and missing error type. The experiment shows the effectiveness of the BERT-GCN-CRF model to resolve redundancy and missing errors.

4.3 Testing Results

For the final submission, we submitted three results from different strategies: (1) single best model with correction; (2) ensemble model with correction; (3) ensemble model with correction and CSC.

As shown in Table 2, our system approach achieves the second highest F1 scores at identification level and position level by a balanced precision and recall and highest recall rate at top-3 correction. One of the reasons for the detection gap is that for an error sentence there are multiple methods to modify the sentence and the modification granularity is difficult to control.

Most of the sentences in our training data contain grammar errors and the ensemble mechanism is tuned based on the F1 score on the validation data. These factors hurt the precision at detection level

as well as the False Positive Rate.

5 Conclusion

This article describes our system in the CGED shared task. We proposed two approaches including BERT-GCN-CRF model and multi-task learning to improve the baseline model to detect grammatical errors. We also designed three approaches including masked language model, seq2seq and spelling check to correct these errors. We got first place in the recall rate of the top-3 correction and got the second highest F1 scores at the identification level and position level.

References

Zuyi Bao, Chen Li, and Rui Wang. To appear. Chunk-based chinese spelling check with global optimization. In *Proceedings of the EMNLP 2020*.

Wanxiang Che, Zhenghua Li, and Ting Liu. 2010. Ltp: A chinese language technology platform. In *COLING 2010, 23rd International Conference on Computational Linguistics, Demonstrations Volume, 23-27 August 2010, Beijing, China*.

Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Ruiji Fu, Zhengqi Pei, Jiefu Gong, Wei Song, Dechuan Teng, Wanxiang Che, Shijin Wang, Guoping Hu, and Ting Liu. 2018. Chinese grammatical error diagnosis using statistical and prior knowledge driven features with probabilistic ensemble enhancement. In *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 52–59, Melbourne, Australia. Association for Computational Linguistics.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging.

Thomas N. Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks.

Changliang Li and Ji Qi. 2018. Chinese grammatical error diagnosis based on policy gradient LSTM model. In *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications*, pages 77–82, Melbourne, Australia. Association for Computational Linguistics.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 147–155, Boulder, Colorado. Association for Computational Linguistics.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Wei Wang, Bin Bi, Ming Yan, Chen Wu, Zuyi Bao, Liwei Peng, and Luo Si. 2019. Structbert: Incorporating language structures into pre-training for deep language understanding.

Yi Yang, Pengjun Xie, Jun Tao, Guangwei Xu, Linlin Li, and Luo Si. 2017. Alibaba at IJCNLP-2017 task 1: Embedding grammatical features into LSTMs for Chinese grammatical error diagnosis task. In *Proceedings of the IJCNLP 2017, Shared Tasks*, pages 41–46, Taipei, Taiwan. Asian Federation of Natural Language Processing.