Combining GCN and Transformer for Chinese Grammatical Error Detection

Jinhong Zhang*

School of Computer Science and Cyber Engineering, Guangzhou University, China
jhzhang_gzhu@163.com

Abstract

This paper describes our system at a task: Chinese Grammatical Error Diagnosis (CGED). The task is held by the Natural Language Processing Techniques for Educational Applications (NLP-TEA) to encourage the development of automatic grammatical error diagnosis in Chinese learning since 2014. The goal of CGED is to diagnose four types of grammatical errors: word selection (S), redundant words (R), missing words (M), and disordered words (W). The automatic CGED system contains two parts including error detection and error correction and our system is designed to solve the error detection problem. Our system is built on three models: 1) a BERT-based model leveraging syntactic information; 2) a BERT-based model leveraging contextual embeddings; 3) a lexicon-based graph neural network leveraging lexical information. We also design an ensemble mechanism to improve the single model's performance. Finally, our system achieves the highest F1 scores at detection level and identification level among all teams participating in the CGED 2020 task.

Keywords: CGED task, GCN, BERT, Ensemble mechanism

1 Introduction

The Chinese language is often recognized as unity of the utmost tough to learn. In comparison to English, Chinese does not have a singular/plural transition, nor does it have verb tense variations. Furthermore, because word boundaries are not explicitly specified in Chinese, word segmentation is frequently required prior to deeper analysis. All of these issues make learning Chinese difficult for newcomers. In recent years, an increasing number of people from various linguistic and educational backgrounds have expressed an interest in studying Chinese as a second language. To assist in identifying and correcting grammatical errors produced by these people, it is necessary to develop an automated Chinese Grammatical Error Diagnosis (CGED) tool.

Since 2014, the CGED has been chosen as one of the shared projects by Natural Language Processing Techniques for Educational Applications (NLP-TEA) to stimulate the development of automatic grammatical mistake diagnosis in Chinese learning. To tackle the CGED challenge, a variety of approaches have been offered.

In this work, we introduce our system to solve the error detection problem. In our system, we use three types of models. The first one is the BERT-GCN-LSTM-CRF, which is based on the perfect of multi-layer bidirectional transformer encoder and incorporates GCN to improve the performance. The second one is the BERT with context-LSTM-CRF, which makes use of contextualized word representations because they have the ability to efficiently capture compositional information in language. The third one is the LGN, which incorporates lexical information into Chinese NER tasks.

We also design an ensemble mechanism to progress the single model's performance. In the experiment, our system gets the highest F1 scores at detection level and identification level among all the models that participated in the NLPTEA-2020 CGED task.

2 Chinese Grammatical Error Diagnosis

Since 2014, the shared assignment for the CGED has taken place. Several sets of training data produced by CFL students have been released, many of which contain severe grammatical mistakes. The CGED provides four categories of mistakes for detection: (1) R (redundant word errors); (2) M (missing words); (3) W (word ordering errors); and (4) S (single word errors) (word selection errors). Three levels of performance are evaluated: detection, identification, and location. Missing and selection mistakes require no more than three repairs, according to the systems. In this paper, our system focuses on the error detection problem.

3 Models

Some previous works consider the error detection problem to be a sequence labeling problem. Similarly, we use BIO encoding to generate a corresponding label sequence y for a sentence x [1]. We use three models to solve the labeling problem, in which BERT and GCN information are all used.

3.1 BERT-GCN-LSTM-CRF

Previous works use LSTM-CRF model to solve the problem [2]. For better performance, we combine the BERT model and GCN model.

We use BERT [3], the multi-layer bidirectional transformer encoder [4] to encode the input sentence. As

*Corresponding Author: Jinhong Zhang; E-mail: jhzhang_gzhu@163.com
DOI: 10.53106/160792642022122307020

shown in Figure 1, given an input sequence $S = x_1, x_2, \cdots, x_N$, BERT outputs the hidden states $S' = h_1, h_2, \cdots, h_N$.

3.1.1 GCN

Previous research [2, 5] put a lot of effort into feature engineering, such as pretrained and parsing features. The most significant parsing characteristics are part-of-speech tagging (POS) and dependency information, indicating that the job is strongly related to the structure of the sentence syntactic dependence.

The Graph Convolution Network is used to better comprehend the dependence structure of an input phrase (GCN) [6-7]. Specifically, we use the graph attention networks (GAT) [8] to assign different importance to nearby nodes using masked self-attention layers. The BERT model's high-level character information and the dependency tree's adjacency matrix are accepted by the multi-layer GCN network. A GAT operation with M independent attention heads can be expressed as follows:

$$f_i' = \prod_{m=1}^{M} \sigma\left(\sum_{j \in N_i} \alpha_{ij}^m W^m f_j\right). \quad (1)$$

$$\alpha_{ij}^m = \frac{\exp\left(\text{LeakyReLU}\left(a^T [W^m f_i \| W^M f_j]\right)\right)}{\sum_{m \in N_i} \exp\left(\text{LeakyReLU}\left(a^T [W^m f_i \| W^M f_m]\right)\right)}, \quad (2)$$

where $\prod$ is the concatenation operation, σ is a nonlinear activation function, N_i is the graph's neighborhood of node i, α_{ij}^m are the attention coefficients and a is a feed-forward neural network. At the last layer, averaging will be adopted:

$$f_i^{final} = \sigma\left(\frac{1}{M} \sum_{m=1}^{M} \sum_{j \in N_i} \alpha_{ij}^m W^m f_j\right). \quad (3)$$

3.1.2 Concatenation

Following the graph convolution network, we concatenate the representation H_l for the l-th layer with the BERT hidden state as the LSTM layer's input.

3.1.3 CRF

To predict the sequence tags for each token, a CRF layer is added.

$$\text{Score}(X, Y) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} V_{i, y_i}. \quad (4)$$

$$P(Y|X) = \frac{\exp(\text{Score}(X,Y))}{\sum_{Y'} \exp(\text{Score}(X,Y'))}, \quad (5)$$

where X, Y, Y′ denotes the input system, the fact tag order, and a random label order, V signifies the discharge grooves, and A is the transition scores matrix of the CRF layer. We use Viterbi Decoding [2] to inference answers.

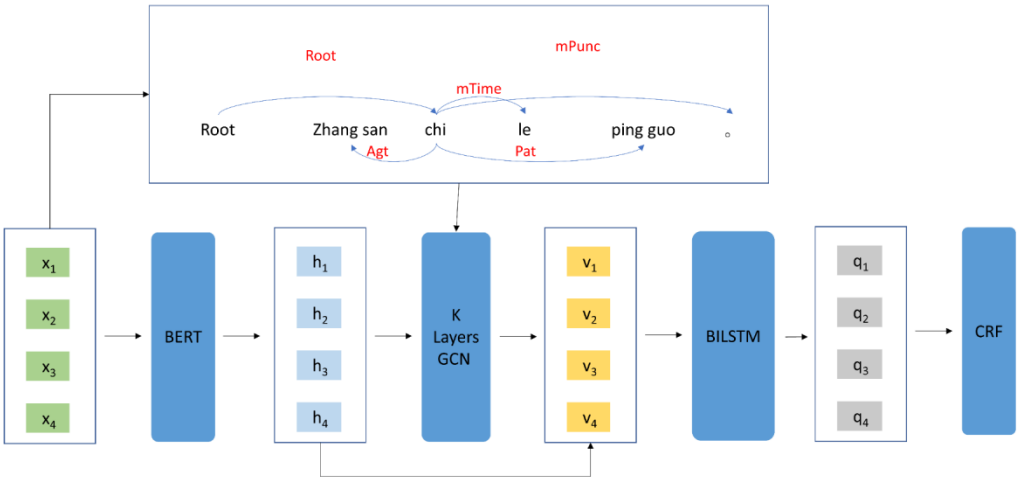

Figure 1. Architectures of BERT-GCN-LSTM-CRF for grammatical error detection

3.2 BERT with Context-LSTM-CRF

Error detection can be difficult because CGED datasets are restricted in extent and the label disseminations are extremely unbalanced. As described in [9], Contextualized word representations can detention compositional statistics in philological efficiently, and they can be augmented on bulky quantities of unproven data. Specifically, it uses ELMo, BERT and Flair embeddings as contextualized word representations. To improve the performance on this task, we similarly use the structure shown in Figure 2.

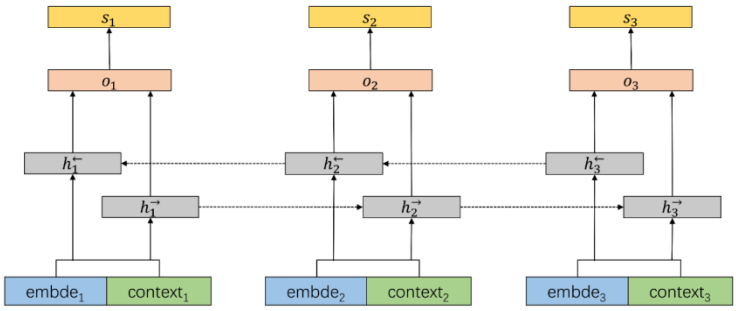

Figure 2. Architectures of BERT with context-LSTM-CRF for grammatical error detection

3.3 LGN

RNN is frequently used in Chinese named entity recognition (NER) task. However, RNN-based models are prone to word ambiguities because to their chain erection and absence of universal semantics. LGN [10] solves this problem by providing a global semantics lexicon-based graph neural network, in which dictionary acquaintance is utilised to link letters to capture the local composition, while a global relay node connects each character node and word edge to capture global sentence semantics and long-range dependency. Figure 3 shows the structure of LGN. Node c represents every character and e represents every potential word. Based on various graph-based interactions among characters, possible words, and whole-sentence semantics, the model may employ global context information to continually compare ambiguous words to handle the word ambiguities problem. LGN achieves Chinese NER as a graph node classification task. We treat error detection task as NER and use LGN to increase the diversity of prediction.

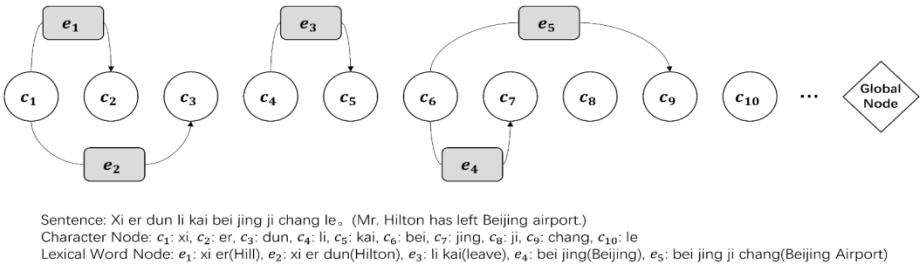

Figure 3. Architectures of LGN for grammatical error detection

4 Ensemble Mechanism

To generate better results, we train several fault recognition representations and employ a three-stage voting ensemble mechanism to get the final result by utilizing the predictions from multiple models.

Before all, we convert the BIO encoding to the format like: (error_start_position, error_end_position, error_type) and then use the ensemble mechanism:

In the first stage, we calculate the number of errors of each type. If the number of errors of a certain type is greater than θ_1 of the number of all models, we believe that there is an error of this type and the error is the largest of all predictions of this type of error. Specially, we do not include LGN when calculating the number of models for the reason that the number of predictions of LGN is small.

In the second stage, if an error appears in the predictions of more than θ_2 models, we think the error exits. Also, we do not include LGN when calculating the number of models if the error is not predicted by LGN for the same reason.

In the third stage, we calculate the number of all predicted errors. If it is great than θ_3 of the number of all models, we believe there is an error. If no errors are predicted after the first two stages, we think the error is the one which is predicted the most by the all models. Also, we do not include LGN when calculating the number of models for the same reason.

In the experiments, we select the θ_1, θ_2 and θ_3 according to the performance on the validation data.

5 Experiments

5.1 Data and Experiment Settings

We trained our sole representations using training parts that include both the incorrect and the corrected sentences from 2016 (HSK Track), 2017, 2018, 2020 training data sets, as well as 2016 (HSK Track) and 2018 testing data sets. The sentences from 2017 trying data set are used for validation and 2020 testing data set are used for test. The overall data distribution in the training data is shown in Table 1.

Table 1. Data statistics

	Error	R	M	S	W
Train	62,661	13,929	16,672	27,504	4,556
Validation	4,871	1,060	1,269	2,156	386
Test	3,660	768	862	1,701	329

For the BERT-GCN-LSTM-CRF model, As the Bert's initialization, we choose the ELECTRA discriminator [11]. We utilise the Chinese ELECTRA-Large discriminator model, which has 1024 hidden units, 16 heads, 24 hidden layers, and 324M parameters. For the GCN model, Language Technology Platform (LTP) [12] was used to generate the dependency tree, and the first layer's hidden vector size was 512 with 8 heads, while the second layer's hidden vector size was 1024 with 8 heads. The hidden size for LSTM was 2048 with one layer. For the other settings, we utilise 128-token streams, a 32-token mini-batch, a 2e-5 learning rate, and a 120-second epoch. To train 12 single models for the ensemble process, we utilise different random seeds and dropout [13] values.

For the BERT with context-LSTM-CRF model, we also select ELECTRA discriminator as the Bert's beginning, and we custom ROBERTA to get the contextual embeddings. Specially, contextual embeddings are not fine-tuned in all experiments. Other parameters are the same as above. Also, we use dissimilar arbitrary seeds and dropout standards to train 10 single models for the ensemble mechanism.

For the LGN model, the group size, learning rate, and epoch were set to 32, 2e-5, 120. Moreover, the dropout rates for embedding, attention and aggregation module were all set to 0.1. We use different random seeds to train 45 single models for the ensemble mechanism.

5.2 Metric

For the error detection task, the evaluation method includes three levels:

Detection level. Determine whether a sentence is correct or not. The sentence is incorrect if there is an error. All types of errors will be considered incorrect.

Level of identification This level may be thought of as a classification issue with several classes. For a given sort of mistake, the rectification scenario should be similar to the gold standard.

Position level. The system's outputs should be closely the equal as the gold benchmark's quadruples.

At the recognition, identification, and situation close, the three metrics precision, recall, and f1 are measured.

5.3 Validation Results

We use the three single models described above as our reference line mockups. The results of dissimilar models are listed in Table 2.

The first and second models have a good fit on the validation set. However, LGN does not perform well mainly because of the low recall, i.e., fewer errors in the sentences are identified, while the high precision ensures a better accuracy of the identified errors. Moreover, LGN also increases the distribution of results to improve the performance of the system. To reduce the effect of low recall, we do not fully include the LGN models when using the ensemble mechanism. As shown in Table 2, the ensembled model does not achieve much improved performance on the validation set, and it is mainly because the first and second models already have a good fit on the validation set and the number of training models is not large enough.

Table 2. The results of single models and ensemble model on validation dataset

Model	Detection			Identification			Position		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
BERT-GCN-LSTM-CRF	0.8695	0.8444	0.8568	0.7471	0.6529	0.6968	0.6111	0.4938	0.5462
BERT with context-LSTM-CRF	0.8848	0.8338	0.8586	0.7631	0.6333	0.6922	0.6305	0.4786	0.5441
LGN	0.8244	0.2419	0.3741	0.7313	0.1297	0.2203	0.4316	0.0631	0.1101
Ensembled model	0.8633	0.8551	**0.8592**	0.7611	0.6698	**0.7125**	0.6210	0.5054	**0.5572**

5.4 Testing Results

Table 3 illustrations the presentations on error detection. Our structure realizes the best F1 scores at the detection level and identification level by a balanced precision and recall among all teams participating in the CGED 2020 task. At the detection level, we improved the F1 value by 0.47% over the state-of-the-art [27-28], and this is because we added syntactic information of the sentences, which is much richer than the POS Score and PMI Score used by the state-of-the-art method. At the identification level, we improved the F1 value by 1.23% over the state-of-the-art [23], and we think this is because the

state-of-the-art method only adds ResNet on top of BERT, but we not only add rich information: syntactic information, contextual embeddings and lexical information, but also add CRF layer to improve the performance, so we can get better F1 value. Although we achieve the highest F1 score, there is still a significant gap in our system's ability to diagnose Chinese grammar errors.

Table 3. Error detection performances on official testing data sets

Team	Detection			Identification			Position		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
UNIPUS-Flaubert	0.8782	0.9157	0.8966	0.6507	0.6420	0.6463	0.3147	0.2739	0.2929
NJU-NLP	0.8565	0.9757	0.9122	0.5571	0.8432	0.6709	0.2097	0.4648	0.2890
OrangePlus	0.9252	0.8600	0.8914	0.7230	0.6287	0.6726	0.4428	0.3610	0.3977
Flying	0.9273	0.6213	0.6736	0.7356	0.6213	0.6736	0.4320	0.3514	0.3876
Ours	0.9037	0.9304	**0.9169**	0.6957	0.6765	**0.6859**	0.4185	0.3608	0.3875

6 Related Work

The scientists have proposed numerous dissimilar technologies to learning the detection and rectification of English grammatical errors [14-16]. However, there are few research on current Chinese grammatical mistakes. Since 2014, current Chinese grammatical mistake diagnostic tasks have been introduced to the Natural Language Processing Techniques for Educational Applications (NLPTEA). To tackle this problem, a variety of approaches have been offered [17-18]. [19] In 2016, they presented a model based on layered LSTM and CRF that enhanced automated grammatical mistake identification accuracy and recall rate. [20] used Bi-LSTM to sense the position of errors and added supplementary verbal statistics, POS, and n-gram, combining machine learning and traditional n-gram methods. [21] The work of mistake repair was viewed as a translation effort. Surface mistakes and grammatical faults are the two types of errors. Low-level mistakes are solved using a comparable phonetic table and 5-gram language model, whereas high-level errors are solved using the Transformer archetypal based on appeal roughness and term granularity [22-24]. The researchers utilised a multi-model parallel framework with three different types of models: rule-based, statistics-based, and neural network models. To solve the detection problem, [25] combined ResNet and BERT, and to increase the performance of a single model, researchers looked at stepwise ensemble selection from model libraries. [26, 29-30] leveraged syntactic information and adopted a multi-task learning framework based on BERT to progress the reference line typical to sense grammatical errors.

7 Conclusion

Our method, which associations GCN and BERT for Chinese Grammatical Error Diagnosis, is described in this article for the NLPTEA-2020 CGED problem. We also design an ensemble mechanism to maximize the model's capability. Among all teams participating in the CGED 2020 task, we achieve the highest F1 scores at detection near and identification level.

References

[1] J.-D. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, N. Collier, Introduction to the bio-entity recognition task at JNLPBA, *International Conference on Computational Linguistics*, Geneva, Switzerland, 2004, pp. 70-75.

[2] Z. Huang, W. Xu, K. Yu, Bidirectional LSTM-CRF Models for Sequence Tagging, ArXiv:1508.01991, August, 2015.

[3] J. Devlin, M. W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, *Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Minneapolis, Minnesota, 2019, pp. 4171–4186.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, *Neural Information Processing Systems*, Long Beach, CA, USA, 2017, pp. 5998–6008.

[5] R. Fu, Z. Pei, J. Gong, W. Song, D. Teng, W. Che, S. Wang, G. Hu, T. Liu, Chinese Grammatical Error Diagnosis using Statistical and Prior Knowledge driven Features with Probabilistic Ensemble Enhancement, *Natural Language Processing Techniques for Educational Applications*, Melbourne, Australia, 2018, pp. 52–59.

[6] T. N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, *International Conference on Learning Representations*, Toulon, France, 2017, pp. 1–14.

[7] D. Marcheggiani, I. Titov, Encoding sentences with graph convolutional networks for semantic role labeling, *Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, 2017, pp. 1506–1515.

[8] P. Velicković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, *International Conference on Learning Representations*, Vancouver, Canada, 2018, pp. 1–12.

[9] S. Bell, H. Yannakoudakis, M. Rei, Context is key: Grammatical error detection with contextual word representations, *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, Florence, Italy, 2019, pp. 103–115.

[10] T. Gui, Y. Zou, Q. Zhang, M. Peng, J. Fu, Z. Wei, X. Huang, A lexicon-based graph neural network for Chinese ner, *Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing*, Hong Kong, China, 2019, pp. 1040-1050.

[11] K. Clark, M. T. Luong, Q. V. Le, C. D. Manning, Electra: Pre-training text encoders as discriminators rather than

generators, *International Conference on Learning Representations*, Millennium Hall, Addis Ababa, Ethiopia, 2020, pp. 1-18.
[12] W. Che, Z. Li, T. Liu, LTP: A chinese language technology platform, *International Conference on Computational Linguistics*, Beijing, China, 2010, pp. 13-16.
[13] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, *Journal of Machine Learning Research*, Vol. 15, pp. 1929-1958, June, 2014.
[14] J. Foster, C. Vogel, Parsing ill-formed text using an error grammar, *Artificial Intelligence Review*, Vol. 21, No. 3, pp. 269-291, June, 2004.
[15] M. Gamon, Using mostly native data to correct errors in learners' writing: A meta-classifier approach, *The North American Chapter of the Association for Computational Linguistics*, Los Angeles, CA, USA, 2010, pp. 163-171.
[16] M. Rei, H. Yannakoudakis, Compositional sequence labeling models for error detection in learner writing, *The Association for Computational Linguistics*, Berlin, Germany, 2016, pp. 1181-1191.
[17] L. C. Yu, L. H. Lee, L. P. Chang, Overview of grammatical error diagnosis for learning Chinese as a foreign language, *International Conference on Computers in Education*, Nara, Japan, 2014, pp. 42-47.
[18] G. Rao, Q. Gong, B. Zhang, E. Xun, Overview of NLPTEA-2018 Share Task Chinese Grammatical Error Diagnosis, *Natural Language Processing Techniques for Educational Applications*, *Melbourne*, Australia, 2018, pp. 42-51.
[19] B. Zheng, W. Che, J. Guo, T. Liu, Chinese grammatical error diagnosis with long short-term memory networks, *Natural Language Processing Techniques for Educational Applications*, Osaka, Japan, 2016, pp. 49-56.
[20] Y. T. Shiue, H. H. Huang, H. H. Chen, Detection of Chinese word usage errors for non-Native Chinese learners with bidirectional LSTM, *Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, 2017, pp. 404-410.
[21] K. Fu, J. Huang, Y. Duan, Youdao's Winning Solution to the NLPCC-2018 Task 2 Challenge: A Neural Machine Translation Approach to Chinese Grammatical Error Correction, *Natural Language Processing and Chinese Computing*, Hohhot, China, 2018, pp. 341-350.
[22] J. Zhou, C. Li, H. Liu, Z. Bao, G. Xu, L. Li, Chinese Grammatical Error Correction Using Statistical and Neural Models, *Natural Language Processing and Chinese Computing*, Hohhot, China, 2018, pp. 117-128.
[23] M. F. Billah, N. Saoda, J. Gao, B. Campbell, BLE Can See: A Reinforcement Learning Approach for RF-based Indoor Occupancy Detection, *Proceedings of the 20th International Conference on Information Processing in Sensor Networks (co-located with CPS-IoT Week 2021)*, Nashville, TN, USA, 2021, pp. 132-147.
[24] A. Shanthini, G. Manogaran, G. Vadivu, K. Kottilingam, P. Nithyakani, C. Fancy, Threshold segmentation based multi-layer analysis for detecting diabetic retinopathy using convolution neural network, *Journal of Ambient Intelligence and Humanized Computing*, March, 2021.
[25] S. Wang, B. Wang, J. Gong, Z. Wang, X. Hu, X. Duan, Z. Shen, G. Yue, R. Fu, D. Wu, W. Che, S. Wang, G. Hu, T. Liu, Combining ResNet and Transformer for Chinese Grammatical Error Diagnosis, *Asian Chapter of the Association for Computational Linguistics*, Suzhou, China, 2020, pp. 36-43.
[26] Y. Luo, Z. Bao, C. Li, R. Wang, Chinese Grammatical Error Diagnosis with Graph Convolution Network and Multi-task Learning, *Asian Chapter of the Association for Computational Linguistics*, Suzhou, China, 2020, pp. 44-48.
[27] D. Vu, T. Nguyen, T. V. Nguyen, T. N. Nguyen, F. Massacci, P. H. Phung, A Convolutional Transformation Network for Malware Classification, *6th NAFOSTED Conference on Information and Computer Science (NICS)*, Hanoi, Vietnam, 2019, pp. 234-239.
[28] Y. Cao, L. He, R. Ridley, X. Dai, Integrating BERT and Score-based Feature Gates for Chinese Grammatical Error Diagnosis, *Asian Chapter of the Association for Computational Linguistics*, Suzhou, China, 2020, pp. 49-56.
[29] C. J. Jutinico, C. E. Montenegro-Marin, D. Burgos, R. G. Crespo, Natural language interface model for the evaluation of ergonomic routines in occupational health (ILENA), *Journal of Ambient Intelligence and Humanized Computing*, Vol. 10, No. 4, pp. 1611-1619, April, 2019.
[30] M. R. Rabbani, M. K. Hassan, S. Khan, M. A. Ali, Artificial intelligence and Natural language processing (NLP) based FinTech model of Zakat for poverty alleviation and sustainable development for Muslims in India, in: M. R. Rabbani, M. K. Hassan, S. Khan, M. A. M. Ali (Eds.), *COVID-19 and Islamic Social Finance*, Routledge, 2021, pp. 107-117.

Biography

Jinhong Zhang, a postgraduate in Guangzhou University, majoring in natural language processing, machine translation.