IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 16 December 2022, accepted 17 January 2023, date of publication 25 January 2023, date of current version 30 January 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3239693

RESEARCH ARTICLE

Neural Quality Estimation Based on Multiple Hypotheses Interaction and Self-Attention for Grammatical Error Correction

CHEN ZHANG[1], TONGJIE XU[2], AND GUANGLI WU[2]
[1]School of Foreign Studies, Gansu University of Political Science and Law, Lanzhou 730070, China
[2]School of Cyberspace Security, Gansu University of Political Science and Law, Lanzhou 730070, China

Corresponding author: Guangli Wu (272956638@qq.com)

This work was supported in part by the Natural Science Foundation of Gansu Province under Grant 21JR7RA570; in part by the Gansu University of Political Science and Law Major Scientific Research and Innovation Projects under Grant GZF2020XZDA03 and Grant 2017XQNLW12; in part by the Young Doctoral Fund Project of Higher Education Institutions in Gansu Province, in 2022, under Grant 2022QB-123; in part by the Gansu Province Innovation Fund Project under Grant 2022A-097; in part by the Science and Technology Project of Gansu Province under Grant 20CX9JA130; and in part by the Lanzhou Talents Innovation and Entrepreneurship Project under Grant 2020-RC-27.

ABSTRACT The English grammatical error correction system is suitable for the English learning environment, with the goal of accurately correcting errors in learners' writing. However, false corrections are often generated in practical applications, and many errors cannot be corrected, thus misleading learners. The quality estimation model is beneficial to ensure that learners obtain accurate grammatical error correction results and avoid misleading sentences caused by error corrections. Grammatical error correction models can generate multiple hypotheses of higher quality, but existing quality estimation models do not consider interactions between different hypotheses. Based on this, we propose a model based on multiple hypotheses interaction and self-attention, BGANet, for English grammatical error correction quality estimation. BGANet builds interactions between multiple hypotheses, extracts and aggregates grammatical error correction evidence in hypotheses through two kinds of self-attention mechanisms, and evaluates the quality of the generated hypotheses. Experiments on four grammatical error correction datasets show that BGANet has better quality estimation performance.

INDEX TERMS Attention mechanism, grammatical error correction, neural quality estimation.

I. INTRODUCTION

Grammatical Error Correction (GEC) in English is an important application in the field of natural language processing in the English environment, and its main purpose is to provide guidance for English learners. Due to the progress and popularization of machine learning and deep learning methods, the research on grammatical error correction based on deep learning methods has also made significant progress. With the progress of globalization, the demand for learning English is increasing day by day, and grammatical error correction has gradually attracted the attention of more natural language processing researchers.

By far, English is the most widely spoken language and the most spoken language as a second language in the world. As non-native English speakers learn English, they tend to make grammatical errors in their writing. Therefore, it is crucial to create a tool that can accurately and effectively correct grammatical errors of English learners, so we first focus on the task of correcting grammatical errors in English.

The best performing work in the CoNLL-2014 Shared Task [1], GECToR [2], its $F_{0.5}$, precision and recall are 65.3, 77.5, and 40.1, respectively. Considering the complexity and diversity of languages used in real-world scenarios, the challenges faced by GEC are still daunting. In the

The associate editor coordinating the review of this manuscript and approving it for publication was Yongming Li.

practical application of language learning, GEC may produce output that misleads learners. At this time, experienced language learners can manually intervene to correct errors. Estimating the quality of the GEC system's output can greatly assist learners in checking and correcting possible erroneous outputs of the system. Based on this, we propose a deep learning-based method for evaluating the quality of GEC outputs.

GEC is generally regarded as a natural language generation task [3], with Seq2Seq [4] architecture designed to correct grammatical errors and beam search to generate correction hypotheses [5]. Because of the effectiveness of transformer-based architectures in natural language generation tasks [6], they are also used to implement error correction [7]. To evaluate GEC systems, Courtney et al. [8] proposed reference-less metrics known as GBMs. Leshem et al. [9] provided another measurement for meaning preservation using a semantic annotation scheme. Large-scale pre-trained models such as BERT [10] brings opportunities to improve the performance of GEC, demonstrating its effectiveness in context learning and enabling better quality estimation [11]. In terms of natural language generation tasks, the interaction between multiple hypotheses has an important impact on quality estimation [12], but the existing quality estimation of GEC outputs do not consider the interaction among hypotheses.

To fully exploit the valuable GEC evidence from GEC hypotheses, we propose a model based on multiple hypotheses interaction and self-attention, BGANet, to enable multiple hypotheses interaction-based GEC quality estimation. Through beam search, BGANet selects K hypotheses sentences from the output of the basic GEC model, and uses the source sentence and hypothesis sentence pair as a node to construct a connectivity graph to propagate GEC evidence among multiple hypotheses, thereby establishing hypotheses interactions. Then BGANet proposes two attention mechanisms on the graph: attention based on hypothesis association and attention based on hypothesis importance, summarizing and aggregating the necessary GEC evidence from other hypotheses to estimate the quality of tokens.

The main contributions are summarized as follows:

- We introduce the method of learning GEC evidence through interactions between hypotheses for estimating the quality of the generated hypotheses.
- We propose two types of attention for interactions of different hypotheses and interactions between source sentences and hypotheses based on the self-attention.
- We conducted extensive experiments to validate the effectiveness of our method and show that it has better performance on CoNLL-2014, FCE, BEA19, and JFLEG datasets.

II. RELATED WORKS

A. GRAMMATICAL ERROR CORRECTION

Grammatical error correction (GEC) aims to enable the system to automatically correct grammatical errors in a given text, including any errors in vocabulary, syntax, and semantics that violate the standards of English usage. At first, mainstream methods used classification based methods to correct preposition or article errors. In this approach, the classifier is trained on a large number of error-free texts to predict the correct target word, taking into account language features given by the context. However, there is still some distance to correcting all the error types of goals here. With the development of deep learning, GEC systems based on neural machine translation that apply SeqSseq become mainstream, correcting all possible errors by translating sentences that do not conform to grammatical standards into correct ones. Recently, the pre-trained language model BERT has proved its effectiveness in context token representation, and some methods rely on it to achieve better performance [2].

B. GRU

Gate Recurrent Units(GRU) [13], which is adapted from the Simple Recurrent Neural Network (RNN) [14], is similar to the Long Short-term Memory (LSTM) network [15], but because the GRU has fewer parameters and has a faster convergence speed, the actual training time is much less, which can greatly speed up the iterative process of the model. Compared with RNN, the improvement of GRU lies in the addition of gating mechanism.

The reset gate and the update gate are important structures for GRU to solve the problem of long dependencies. Unlike LSTM, GRU reduces a gate. LSTM chooses to expose part of the information (the output of the hidden layer is the desired result, and the memory unit is only the carrier of information, not as the output result), while GRU chooses to display all the information, and the change of their output affects the structure of each model. The reset gate is in principle a modification of the output gate of the LSTM, since the output changes, it changes into the process of computing $\tilde{h}_t$. The reset gate will process the rules of combining the input of the current time step with the memory of the previous time step, the update gate will process the memory of the previous time step, calculates the memory that needs to be retained and save it to the current time step. The structure of the GRU unit is shown in Figure 1.

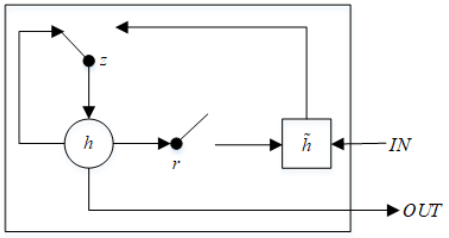


FIGURE 1. The structure of GRU unit.

At time step t, when input x_t enters a GRU unit, it first passes through the update gate.

$$z_t = \sigma\left(W^{(z)}x_t + U^{(z)}h_{t-1}\right) \tag{1}$$

In the update gate, the current input x_t and the memory of the previous moment h_{t-1} will undergo linear transformation and sum, and the result will be mapped to between 0 and 1 by Sigmoid function. If the value is close to 0, the information will be discarded, and if it is close to 1, it will be saved. The update gate determines how much information of the last time step will be retained. The reset gate determines how much information from the previous time step needs to be forgotten, for input x_t and h_{t-1} of the previous time step.

$$r_t = \sigma\left(W^{(r)}x_t + U^{(r)}h_{t-1}\right) \tag{2}$$

C. BERT

In the early days, most natural language processing systems adopted rule-based approaches [16], [17], [18], while were later replaced by machine learning models [19], [20], [21]. Machine learning models need to create features from raw data, which requires domain expertise and takes a long time. Current Pre-trained language models based on Transformer(T-PTLM) [22] have the ability to learn common language representations from large-scale unlabeled text data and transfer that knowledge to downstream tasks. When the target task is similar to the source task, transfer learning allows researchers to reuse the parameters learned from the source task to the target task.

Devlin of Google and his colleagues proposed the bidirectional encoder representation technology based on the transformer, also known as BERT, in 2018. It abandoned the decoder part of bidirectional Transformer, leaving only the remaining encoder as its model architecture. In pre-training, Masked Language Model is used to extract word-level features, and Next Sentence Prediction is used to extract sentence-level features. When pre-training model is used, large quantities of data are not needed for training, thus improving experimental efficiency.

The Multi-layer Bidirectional Transformer's encoder is the basis of BERT, and the implementation in BERT is the same as the original implementation. In the process of BERT implementation, the author expressed the number of Transformer blocks as L, the number of hidden layer neurons as H, and the number of heads in Multi-head Attention as A. In all cases, the size of the forward propagating filter is set to $4H$, and the author provides two models, simple and complex:

$$\begin{aligned} BERT_{BASE} &: L = 12, H = 768, A = 12 \\ BERT_{LARGE} &: L = 24, H = 1024, A = 16 \end{aligned} \tag{3}$$

The number of parameters of the former is 110M, and the number of parameters of the latter is 340M. BERT's input vector is the unit sum of three embedding features, which are:

- Token Embedding: Build a character vector dictionary to map each character in the text to a one-dimensional vector.
- Position Embedding: The position information of characters is also encoded as feature vectors to prevent the position information of words in sentences from being modified by self-attention.
- Segment Embedding: Used to divide sentences.

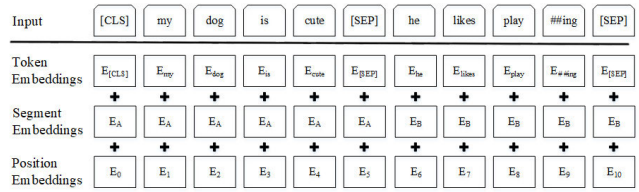


FIGURE 2. BERT's input representation.

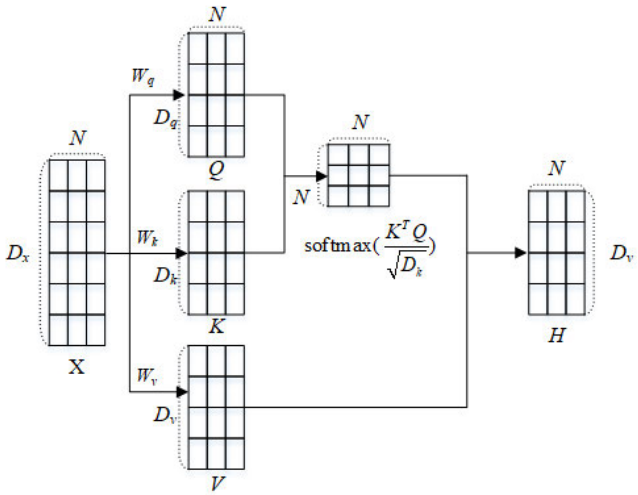


FIGURE 3. The calculation process of Self-Attention mechanism.

In the Masked Language Model, the author believes that the bidirectional depth model has stronger performance than the one-way shallow connection. In order to obtain a good bidirectional depth representation through training, the author adopts the method of randomly masking token with 15% and predicting only the masked part. The author calls this process Masked Language Model. In Next Sentence Prediction, its task is to judge whether two sentences are context or not, and generate training data by randomly extracting two consecutive sentences from parallel corpus. Half of the data is reserved for two sentences, which conform to the continuation relationship, and the other half are two sentences randomly extracted from corpus, which have no relationship with the previous sentence.

D. SELF-ATTENTION

Self-attention is evolved from Attention, it does not need much external information and is characterized by good detection of inter-data or internal correlation of features. In the field of text processing, the key role is to solve the problem of long-distance dependence in the long text by calculating the relevance between characters.

The self-attention mechanism adopts query-key-value scheme to improve the performance of the model, and the operation process is shown in the figure 3. When the input

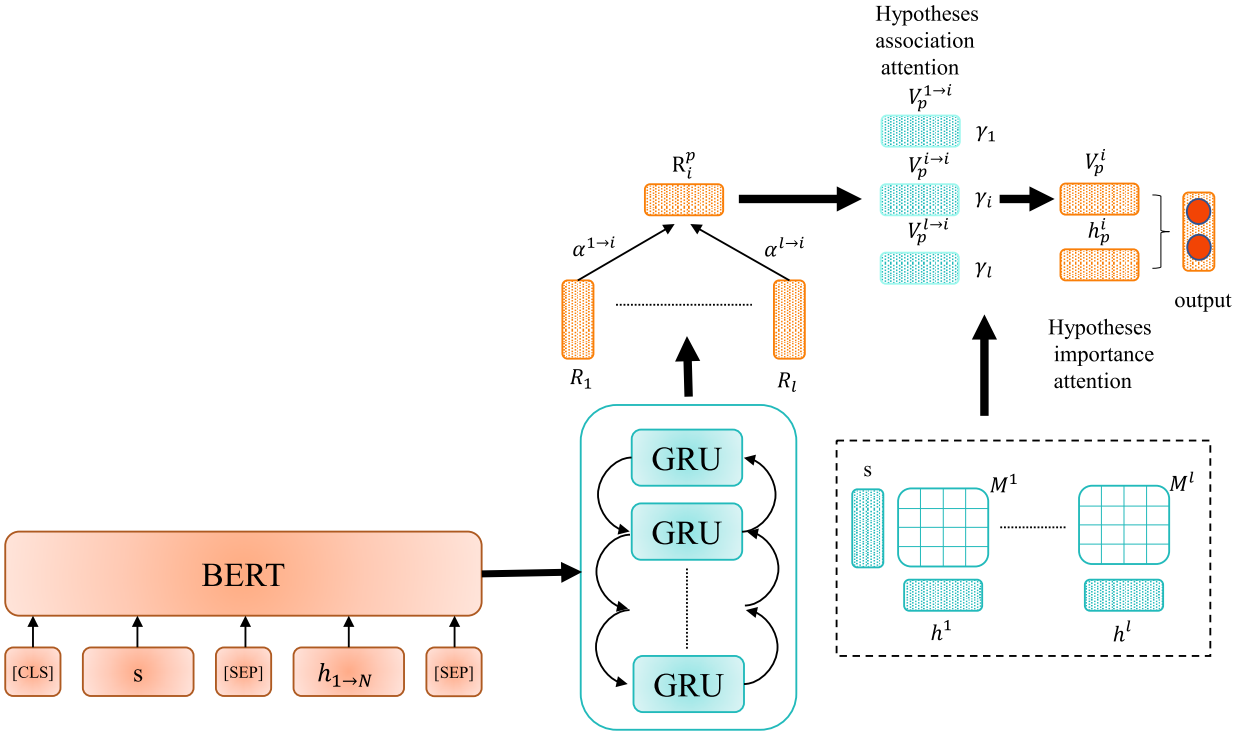


FIGURE 4. The structure of the BGANet. The source and hypothesis are spliced together and the feature representation is learned by the Seq2Seq module composed of BERT and BiGRU, then the interaction information between hypotheses is learned by hypothesis association attention to obtain GEC evidence, and the importance of nodes is discriminated by evidence aggregation attention. Finally, the model aggregates the token-level quality estimation to calculate the quality estimation results on the hypotheses.

is $X = [x_1, \cdots, x_N] \in \mathbb{R}^{D_x \times N}$, the query vector, key vector and value vector are calculated first:

$$\begin{aligned} Q &= W_q X \in \mathbb{R}^{D_q \times N} \\ K &= W_k X \in \mathbb{R}^{D_k \times N} \\ V &= W_v X \in \mathbb{R}^{D_v \times N} \end{aligned} \tag{4}$$

where W is the parameter matrix of linear mapping, $Q = [q_1, \cdots, q_N]$ is the matrix composed of query vectors, $K = [k_1, \cdots, k_N]$ is the matrix composed of key vectors, and $V = [v_1, \cdots, v_N]$ is the matrix composed of value vectors. For each query vector $q_n \in Q$, the output vector h_n can be calculated:

$$\begin{aligned} h_n &= \text{att}\left((K, V), q_n\right) \\ &= \sum_{j=1}^{N} \alpha_{n,j} v_j \\ &= \sum_{j=1}^{N} \text{softmax}(s(k_j, q_n)) v_j \end{aligned} \tag{5}$$

where $n, j \in [1, N]$ is the position of the sequence of output and input vector, and $\alpha_{n,j}$ represents the correlation degree between the j-th input and the i-th input.

III. HYPOTHESES ESTIMATION MODEL

First, we combine the source sentence s and each sentence in its corresponding hypothesis set $H = \{h_1, h_2, \ldots, h_K\}$ into a source-hypothesis pair s-h_i, where K denotes the number of hypotheses, and all s-h_i pairs are considered as nodes and connected to establish the interaction between different hypotheses. The Seq2Seq model will then obtain the feature representation of s-h_i pair, the encoder of Seq2Seq model consists of BERT and the decoder consists of Bidirectional GRU(BiGRU). Subsequently, two kinds of attention mechanisms are proposed, attention based on hypothesis association and attention based on hypothesis importance, summarizing and aggregating the necessary GEC evidence from other hypotheses to estimate token quality. Finally, the quality of the hypothesis is estimated by aggregating the token-level quality estimation scores. The model structure is shown in Figure 4.

A. INITIAL FEATURE REPRESENTATION

For the source sentence s with length L and the hypothesis h_i with length N, the s-h_i pair first passes through the encoder constituted by BERT in Seq2Seq, and then the output of the hidden layer of encoder is used as the input of decoder BiGRU to obtain the token-level feature representation $R_i = \{R_i^1, R_i^2 \ldots, R_i^{L+2}, \ldots, R_i^{L+N+3}\}$ of the i-th pair:

$$R_i = \text{BiGRU}(\text{BERT}(s\text{-}h_i)) \tag{6}$$

where R_i^1 denotes the representation of "[CLS]", R_i^{L+2} and R_i^{L+N+3} denotes the representation of "[SEP]".

B. NODE REPRESENTATION OF ATTENTION BASED ON HYPOTHESIS ASSOCIATION

Attention based on hypothesis association obtains supporting evidence of the l-th node from the i-th node, thus establishing token-level node representation $V^{l\to i}$.

For the p-th token t_i^p in the i-th node, we first calculate the association score $\alpha_q^{l\to i}$ between nodes according to the correlation between t_i^p and the q-th token t_l^q in the l-th node:

$$\alpha_q^{l\to i} = \text{softmax}((R_i^p)^T \cdot W \cdot R_l^q) \tag{7}$$

where W is a learnable parameter, R_i^p and R_i^q are feature representations of t_i^p and t_l^q respectively. Then all token representations of the l-th node are aggregated:

$$V_p^{l\to i} = \sum_{q=1}^{L+N+3} (\alpha_q^{l\to i} \cdot R_l^q) \tag{8}$$

Based on $V_p^{l\to i}$, a token-level representation of the l-th node pointing to the i-th node is further established:

$$V^{l\to i} = \{V_1^{l\to i}, \ldots, V_p^{l\to i}, \ldots, V_{L+N+3}^{l\to i}\} \tag{9}$$

C. EVIDENCE AGGREGATION ATTENTION BASED ON THE IMPORTANCE OF HYPOTHESIS

Attention based on hypothesis importance measures node importance and is used to aggregate supporting evidence from the representation $V^{l\to i}$ of the l-th node. We use attention-over-attention [23] to represent source sentence h^{ls} and hypothesis h^{lh}, which is used to calculate the attention score γ^l of the l-th node, and then obtain the verification representation V_p^i of the node according to this score.

To calculate the attention score γ^l, we establish the interaction matrix M between the source and the hypotheses of the l-th node. Calculating each element $M_{r,c}^l$ in M^l according to the correlation between the r-th token of the source sentence and the c-th token of the hypothesis sentence:

$$M_{r,c}^l = (R_l^r)^T \cdot W \cdot R_l^{L+2+c} \tag{10}$$

where W is a learnable parameter. Then, the attention score λ_r^{ls} and λ_c^{lh} are calculated along the source dimension and hypothesis dimension respectively:

$$\begin{aligned} \lambda_r^{ls} &= \frac{1}{L+2} \sum_{r=1}^{L+2} \text{softmax}\,(M_{r,c}^l) \\ \lambda_c^{lh} &= \frac{1}{N+1} \sum_{c=1}^{N+1} \text{softmax}\,(M_{r,c}^l) \end{aligned} \tag{11}$$

Then calculate the representation of the source and hypothesis:

$$\begin{aligned} h^{ls} &= \sum_{r=1}^{L+2} \lambda_r^{ls} \cdot R_l^r \\ h^{lh} &= \sum_{c=1}^{N+1} \lambda_c^{lh} \cdot R_l^{L+2+c} \end{aligned} \tag{12}$$

Finally, for evidence aggregation, the importance score γ^l of the l-th node if calculated:

$$\gamma^l = \text{softmax}(\text{Linear}((h^{ls} \circ h^{lh}); h^{ls}; h^{lh})) \tag{13}$$

where $\circ$ is the element-by-element multiplication operator, and ; is the connect operator.

The node importance attention score γ^l aggregates evidence for the verification representation V_p^i of t_i^p:

$$V_p^i = \sum_{l=1}^{K} (\gamma^l \cdot V_p^{l\to i}) \tag{14}$$

where $V^i = \{V_1^i, \ldots, V_p^i, \ldots, V_{L+N+3}^i\}$ is the verification representation of the i-th node.

D. HYPOTHETICAL QUALITY ESTIMATION

For the p-th token t_i^p in the i-th node, the probability $P(y|t_i^p)$ of quality label y is calculated using the validation representation V_p^i:

$$P(y|t_i^p) = \text{softmax}(\text{Linear}((R_i^p \circ V_p^i); R_i^p; V_p^i)) \tag{15}$$

where, $\circ$ is the element-by-element multiplication operator, and ; is the connect operator. We average all probability $P(y = 1|t_i^p)$ of token-level quality estimation as the hypothesis quality estimation score $f(s, h^i)$ for s-h_i pair:

$$f(s, h^i) = \frac{1}{N+1} \sum_{p=L+2}^{L+N+3} P(y = 1|t_i^p) \tag{16}$$

E. END-TO-END TRAINING

In this part, we use the source sentence labels and the hypothesis sentence labels to indicate the syntactic quality of the source sentence and the accuracy of GEC hypothesis.

The cross entropy loss of the p-th token t_i^p in the i-th node is calculated by using the ground truth token labels $y*$.:

$$F(t_i^p) = \text{CrossEntropy}(y*, P(y|t_i^p)) \tag{17}$$

BGANet's training loss is then calculated:

$$Loss = \frac{1}{K} \frac{1}{L+N+3} \sum_{i=1}^{K} \sum_{p=1}^{L+N+3} F(t_i^p) \tag{18}$$

IV. THE EXPERIMENT

A. DATASET

We use FCE [24], BEA19 [25], NUCLE [26], CoNLL-2014 [1] and JFLEG [27] as datasets for training and testing:

- The Cambridge Learner Corpus First Certificate in English (FCE), is a dataset composed of English short texts, which contains 77 types of errors and are manually marked with errors.
- The Building Educational Applications (BEA) 2019 Shared Task introduces a new dataset, the Write&Improve+LOCNESS corpus, which represents a broader range

of native and learner English proficiency and ability, and controls the amount of participant annotated data.
- NUCLE is a large, annotated Corpus of English learners' texts released in 2013, using about 1,400 undergraduate papers from the National University of Singapore, all of which have the errors flagged and corrected.
- Grammatical error correction is the shared task of the Eighteenth Conference on Computational Natural Language Learning in 2014 (CoNLL-2014). CoNLL-2014 allows participants to work on the same grammatical error correction task and evaluate the same blind test set using the same evaluation indicators and raters, which is a commonly used test set in the GEC field.
- JHU FLuency-Extended GUG (JFLEG) Dataset represents a broad range of language proficiency levels and uses holistic fluency edits to not only correct grammatical errors but also make the original text more native sounding.

The data volume distribution of training, validation, and testing is shown in Table 1. Under all the experiments, three of them were used for training, two for validation, and four for testing.

TABLE 1. Dataset partitioning.

Dataset	Training	Development	Test
FCE	28350	2191	2695
BEA19	34308	4384	4477
NUCLE	57151	-	-
CoNLL-2014	-	-	1312
JFLEG	-	-	747
Total	119809	6575	9231

B. EVALUATION METRICS

We introduce evaluation metrics in token-level quality estimation and sentence-level quality estimation. We use the same evaluation metrics precision, recall and $F_{0.5}$ as the previous grammatical error detection (GED) model as the evaluation metrics for token-level quality evaluation. Assuming that S is the predicted error character position, G is the real error character position, and O is the intersection of S and G, the calculation of precision and recall rate is as follows:

$$\begin{aligned} \text{pre} &= O/S \\ \text{rec} &= O/G \end{aligned} \tag{19}$$

The F measure can be calculated from precision and recall:

$$F = \frac{(1+\beta^2) \times \text{pre} \times \text{rec}}{\beta^2 \times \text{pre} + \text{rec}} \tag{20}$$

when β is equal to 0.5, F measure is $F_{0.5}$.

For sentence-level quality evaluation, we also used GLEU [8] to evaluate the performance of the model on JFLEG dataset. GLEU calculates a weighted precision of n-grams. For a hypothesis H with a corresponding source S and reference R, the modified n-gram precision for $GLEU(H, R, S)$ is shown in (21).

$$p'_n = \frac{\sum\limits_{\text{n-gram}\in C} f_{R\setminus S}(\text{n-gram}) + f_R(\text{n-gram})}{\sum\limits_{\text{n-gram}'\in C'} f_S(\text{n-gram}') + \sum\limits_{\text{n-gram}\in R\setminus S} f_{R\setminus S}(\text{n-gram})} \tag{21}$$

(22) and (23) describe how the counts are collected given a bag of n-grams B.

$$f_B(\text{n-gram}) = \sum_{\text{n-gram}'\in B} d(\text{n-gram}, \text{n-gram}') \tag{22}$$

$$d(\text{n-gram}, \text{n-gram}') = \begin{cases} 1 & if \quad \text{n-gram} = \text{n-gram}' \\ 0 & \text{otherwise} \end{cases} \tag{23}$$

The calculation of the final GLEU score is shown in (24) and (25):

$$GLEU(H, R, S) = BP \cdot exp(\sum_{n=1}^{N} w_n log p'_n) \tag{24}$$

$$BP = \begin{cases} 1 & if \quad c > r \\ e^{(}1 - c/r) & if \quad c \le r \end{cases} \tag{25}$$

where $N = 4$ and $w_n = \frac{1}{N}$, c is the length of the hypothesis and r is the effective reference corpus length.

C. EVALUATION RESULTS

We evaluate the performance of BGANet from two aspects: toke-level quality estimation and sentence-level quality estimation. Finally, an ablation study is conducted to investigate the effect of our proposed attention method on the performance of the overall model.

The experiment was carried out in the Windows 10 system, NVIDIA Tesla P100 and Pytorch 1.8.1. And we set the learning rate as 1e-5, batch size as 4. The maximum length of BERT input is set to 243 and its hidden layer size is set to 768, while the hidden layer size of GRU is also set to 768.

1) TOKEN-LEVEL EVALUATION

We first perform token-level evaluation on the model, comparing it with the previous state-of-the-art GED model MHMLA [28] and its variant MHMLA (HYP). MHMLA is a syntactic error detection model using large-scale pre-training model. A multi-head multilevel attention model is proposed to determine the appropriate layers in BERT. It integrates the information of the last layer of the model and the middle layer of the pre-trained model to detect grammatical errors, and processes the MHMLA variant MHMLA (HYP) by considering the first GEC hypothesis of beam search.

As shown in Table 2, the source and hypotheses scenarios are used to evaluate the performance of the model. As with the GED model [29], the source scenario evaluates the grammatical quality estimation ability of the model; the

TABLE 2. Evaluation on token-level. Evaluate grammatical quality estimation ability under source and hypotheses separately. Bold indicates the highest score in each column.

	Model	FCE			CoNLL-2014 ann.1			CoNLL-2014 ann.2		
		P	R	$F_{0.5}$	P	R	$F_{0.5}$	P	R	$F_{0.5}$
Source	MHMLA [28]	74.2	43.34	64.9	59.84	27.11	48.20	77.94	25.02	54.77
	BGANet	**82.64**	**45.23**	**71.07**	**62.97**	**31.23**	**52.33**	**83.67**	**29.45**	**61.15**
Hypotheses	MHMLA(HYP) [28]	80.27	40.58	67.14	74.28	34.20	60.17	66.49	27.68	51.93
	BGANet	**82.33**	**44.96**	**70.59**	**75.99**	**35.31**	**61.76**	**82.34**	**30.62**	**61.54**

TABLE 3. Evaluation on sentence-level. We use the reordered top-1 hypothesis to calculate GEC metrics. Bold indicates the highest score in each column.

Model	CoNLL-2014(M^2)			BEA19			JFLEG
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	GLEU
NQE [30]	60.62	35.77	52.23	56.83	49.47	55.19	57.22
MHMLA [28]	52.98	**52.07**	52.79	47.15	**65.09**	49.90	60.32
Multi-encoder+GED re-ranking [31]	60.4	39.0	54.4	60.8	50.8	58.5	-
BERT-fuse GED [32]	63.6	33.0	53.6	58.1	44.8	54.8	54.4
BGANet	**67.12**	42.03	**60.59**	**67.95**	59.46	**66.06**	**61.67**

TABLE 4. Results of metrics evaluation under FCE and CoNLL-2014 in ablation study.

Model	FCE			CoNLL-2014 ann.1			CoNLL-2014 ann.2		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	P	R	$F_{0.5}$
BGANet-A	77.54	**46.96**	68.61	69.34	32.89	56.76	74.72	28.44	56.38
BGANet	**82.33**	44.96	**70.59**	**75.99**	**35.31**	**61.76**	**82.34**	**30.62**	**61.54**

quality estimates ability on GEC accuracy is evaluated under hypotheses scenario.

It can be seen that in both cases BGANet shows further improvement compared to MHMLA. Under the source sentence, BGANet's $F_{0.5}$ is higher than MHMLA by more than 0.4 for all datasets. Under the hypotheses sentence, the $F_{0.5}$ of BGANet is improved by 9.61 under CoNLL-2014 ann.2. These improvements indicate the benefits of multiple hypotheses interaction for token-level quality estimation.

2) SENTENCE-LEVEL EVALUATION

In this section, we evaluate BGANet's performance in sentence-level quality estimation by reordering beam search decoding hypothesis.

Compared with MHMLA, NQE [30] and the methods of Yuan et al. [31] and Kaneko et al. [32], the NQE model is the first supervised GEC quality estimation model. Encoder-Decoder architecture is used to encode source sentence-hypothesis sentence pairs, and it is verified that the performance of GEC system can be improved by reordering n candidate hypotheses through evaluation scores. Yuan's method uses multi-class grammatical error detection system to improve grammatical error correction for English, and propose a multi-encoder GEC model and two-step training strategy, we use its added GED re-ranking mothed for comparison. Kaneko's method incorporates a pre-trained language model into an encoder-decoder model for grammatical error correction, we use its BERT-fuse GED for comparison.

As shown in Table 3, compared to quality estimation based on language models, the quality estimation model based on GEC accuracy achieved better accuracy and $F_{0.5}$, and provided more accurate feedback to users. The experiment results support our claim that beam search's multiple hypotheses provide valuable GEC evidence and contribute to a more effective quality estimation of the generated GEC hypotheses.

3) ABLATION STUDY

In order to investigate the effect of our designed attention method on the performance of GEC quality estimation, we conducted an ablation study on BGANet. BGANet: Our complete GEC quality estimation model; BGANet-A: Remove the attention based on hypothesis association and attention based on hypothesis importance, and only the Seq2Seq model composed of BERT and GRUs is used to estimate GEC quality. Table 4 shows the results of the ablation study. It can be concluded that attention based on hypothesis association and attention based on hypothesis importance can effectively aggregate GEC evidence, implement the interaction between different hypotheses to estimate the quality of tokens.

V. THREATS TO VALIDITY

A. INTERNAL VALIDITY

Internal validity concerns the strength of the experimental results, i.e., whether there are factors outside the experimental variables that influence the experimental results. An important influencing factor is the correctness of the code. In this experiment, in order to avoid the effects of subjectivity and inconsistent code implementation standards and to make it universal, we use the interface provided by the widely recognized and used Pytorch framework to implement the network model, and finally we ensure the accuracy of the code by reviewing it by multiple people.

B. EXTERNAL VALIDITY

External validity concerns the extent to which experimental results can be generalized to more general scenarios. In order to verify the degree of generality of our experimental results, we chose four publicly available datasets, CoNLL-2014, FCE, BEA19 and JFLEG, which are commonly used in the field of English grammatical error correction, and the data of these four datasets are collected from the writing contents of real writers. We have conducted sufficient experiments and achieved good performance with these four datasets, so our experimental results are to some extent general.

VI. CONCLUSION

This study presents a BGANet model for multiple hypotheses GEC quality estimation. BGANet models the interaction of multiple hypotheses, and then extracts GEC evidence with two kinds of attention: hypothesis association attention and hypothesis importance attention. They summarize and aggregate GEC evidence from multiple hypotheses to verify the quality of tokens. Experiments on four datasets show that BGANet achieves state-of-the-art quality estimation performance.

There may be some possible limitations in this study. Because the pre-trained BERT model used in this study has restrictions on the length of the input and all sentences must be of the same length, a threshold needs to be set for the length of the sentences, and sentences with lengths exceeding this threshold need to be truncated and those that are insufficient need to be filled. This preprocessing of data may affect the original semantic information of the sentences, resulting in the model not being able to learn the semantic information in the sentences completely. In addition, although the pre-trained model BERT has created good generalization ability due to the accumulation of large amount of data, it still needs to fine-tune the pre-trained model to adapt to the downstream task, and the design of the fine-tuning strategy will have an important impact on the results. Therefore, to explore how to fully learn the semantic information of the sentences and find the appropriate fine-tuning strategies to make the pre-trained model better adapted to the downstream tasks is the next focus of our research.

REFERENCES

[1] H. T. Ng, S. M. Wu, T. Briscoe, C. Hadiwinoto, R. H. Susanto, and C. Bryant, "The CoNLL-2014 shared task on grammatical error correction," in *Proc. 18th Conf. Comput. Natural Lang. Learn., Shared Task*, Baltimore, Maryland, 2014, pp. 1–14, doi: 10.3115/v1/W14-1701.

[2] K. Omelianchuk, V. Atrasevych, A. Chernodub, and O. Skurzhanskyi, "GECToR—Grammatical error correction: Tag, not rewrite," in *Proc. 15th Workshop Innov. Use NLP Building Educ. Appl.*, Seattle, WA, USA, 2020, pp. 163–170, doi: 10.18653/v1/2020.bea-1.16.

[3] K. Cho, B. Van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Doha, Qatar, 2014, pp. 1724–1734, doi: 10.3115/v1/D14-1179.

[4] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Advances Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[5] Z. Yuan and T. Briscoe, "Grammatical error correction using neural machine translation," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, San Diego, CA, USA, 2016, pp. 380–386, doi: 10.18653/v1/N16-1042.

[6] S. Chollampatt and H. T. Ng, "A multilayer convolutional encoder–decoder neural network for grammatical error correction," in *Proc. Conf. Artif. Intell. (AAAI)*, 2018, pp. 1–8.

[7] R. Grundkiewicz, M. Junczys-Dowmunt, and K. Heafield, "Neural grammatical error correction systems with unsupervised pre-training on synthetic data," in *Proc. 14th Workshop Innov. Use NLP Building Educ. Appl.*, Florence, Italy, 2019, pp. 252–263, doi: 10.18653/v1/W19-4427.

[8] C. Napoles, K. Sakaguchi, M. Post, and J. Tetreault, "Ground truth for grammaticality correction metrics," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, Beijing, China, 2015, pp. 588–593, doi: 10.3115/v1/P15-2097.

[9] L. Choshen and O. Abend, "Reference-less measure of faithfulness for grammatical error correction," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, New Orleans, Louisiana, 2018, pp. 124–129, doi: 10.18653/v1/N18-2020.

[10] J. D. M. W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, Minneapolis, Minnesota, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.

[11] M. Kaneko, K. Hotate, S. Katsumata, and M. Komachi, "TMU transformer system using BERT for re-ranking at BEA 2019 grammatical error correction on restricted track," in *Proc. 14th Workshop Innov. Use NLP Building Educ. Appl.*, Florence, Italy, 2019, pp. 207–212, doi: 10.18653/v1/W19-4422.

[12] M. Fomicheva, L. Specia, and F. Guzmán, "Multi-hypothesis machine translation evaluation," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 1218–1232, doi: 10.18653/v1/2020.acl-main.113.

[13] J. Chung, C. Gulcehre, and K. Cho, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Proc. NIPS Workshop Deep Learning*, Dec. 2014, pp. 1–9.

[14] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *Proc. 32nd Int. Conf. Int. Conf. Mach. Learn.*, 2015, pp. 2342–2350.

[15] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 802–810.

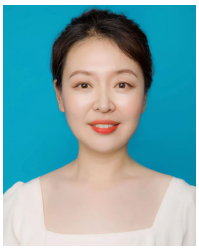
[16] G. Sidorov, "Syntactic dependency based N-grams in rule based automatic English as second language grammar correction," *Int. J. Comput. Linguistics Appl.*, vol. 4, no. 2, pp. 169–188, 2013.

[17] B. Lata, D. Gauri, and K. Manali, "Grammar checking system using rule based morphological process for an Indian language," in *Proc. Int. Conf. Comput. Commun. Syst.*, 2011, pp. 524–531.

[18] T. Vosse, "Detecting and correcting morpho-syntactic errors in real texts," in *Proc. 3rd Conf. Appl. Natural Lang. Process. Assoc. Comput. Linguistics*, 1992, pp. 111–118.

[19] G. Sidorov, F. Velasquez, E. Stamatatos, A. Gelbukh, and L. Chanona-Hernández, "Syntactic N-grams as machine learning features for natural language processing," *Exp. Syst. Appl.*, vol. 41, no. 3, pp. 853–860, Feb. 2014.

[20] R. Zbib, S. Matsoukas, R. Schwartz, and J. Makhoul, "Decision trees for lexical smoothing in statistical machine translation," in *Proc. Joint 5th Workshop Stat. Mach. Transl. Metrics (MATR)*, 2010, pp. 428–437.
[21] W. Zhang, X. Tang, and T. Yoshida, "TESC: An approach to TExt classification using semi-supervised clustering," *Knowl.-Based Syst.*, vol. 75, pp. 152–160, Feb. 2015.
[22] A. Vaswani, N. Shazeer, and N. Parmar, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[23] Y. Cui, Z. Chen, S. Wei, S. Wang, T. Liu, and G. Hu, "Attention-over-attention neural networks for reading comprehension," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, Vancouver, Canada, 2017, pp. 593–602, doi: 10.18653/v1/P17-1055.
[24] H. Yannakoudakis, T. Briscoe, and B. Medlock, "A new dataset and method for automatically grading ESOL texts," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Human Lang. Technol.*, Portland, Oregon, USA, 2011, pp. 180–189.
[25] C. Bryant, M. Felice, Ø. E. Andersen, and T. Briscoe, "The BEA-2019 shared task on grammatical error correction," in *Proc. 14th Workshop Innov. Use NLP Building Educ. Appl.*, 2019, pp. 52–75, doi: 10.18653/v1/W19-4406.
[26] D. Dahlmeier, H. T. Ng, and S. M. Wu, "Building a large annotated corpus of learner English: The NUS corpus of learner English," in *Proc. 8th Workshop Innov. Use NLP Building Educ. Appl.*, Atlanta, Georgia, 2013, pp. 22–31.
[27] C. Napoles, K. Sakaguchi, and J. Tetreault, "JFLEG: A fluency corpus and benchmark for grammatical error correction," in *Proc. 15th Conf. Eur. Chapter Assoc. Comput. Linguistics*, Valencia, Spain, 2017, pp. 229–234.
[28] M. Kaneko and M. Komachi, "Multi-head multi-layer attention to deep language representations for grammatical error detection," *Computación y Sistemas*, vol. 23, no. 3, pp. 883–891, Oct. 2019.
[29] M. Rei and A. Søgaard, "Jointly learning to label sentences and tokens," in *Proc. Conf. Artif. Intell. (AAAI)*, 2019, vol. 33, no. 1, pp. 6916–6923.
[30] S. Chollampatt and H. T. Ng, "Neural quality estimation of grammatical error correction," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 2528–2539, doi: 10.18653/v1/D18-1274.
[31] Z. Yuan, S. Taslimipoor, C. Davis, and C. Bryant, "Multi-class grammatical error detection for correction: A tale of two systems," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2021, pp. 8722–8736, doi: 10.18653/v1/2021.emnlp-main.687.
[32] M. Kaneko, M. Mita, S. Kiyono, J. Suzuki, and K. Inui, "Encoder–decoder models can benefit from pre-trained masked language models in grammatical error correction," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 4248–4254, doi: 10.18653/v1/2020.acl-main.391.



CHEN ZHANG was born in Liaocheng, Shandong, China, in 1982. She is currently pursuing the master's degree. She is also a Lecturer. Her research interests include natural language processing and corpus linguistics.

TONGJIE XU was born in Binzhou, Shandong, China, in 1999. He received the B.S. degree in computer science and technology from Qufu Normal University (QFNU), Rizhao, Shandong, in 2021. He is currently pursuing the master's degree in cybersecurity with the Gansu University of Political Science and Law. His main research interests include temporally language grounding and artificial intelligence.

GUANGLI WU was born in Weifang, Shandong, China, in 1981. He is currently pursuing the Ph.D. degree. He is also a Professor. His research interests include video content understanding and artificial intelligence. He is a member of ACM.