Abstract

Deep learning (DL)-based approaches, such as LSTM and Transformer, have shown remarkable advancements in automated speaking assessment (ASA). Nevertheless, two challenges persist: faithful modeling of hierarchical context, such as how to portray word-to-paragraph relationships, and seamless integration of hand-crafted knowledge into DL-based model. In this work, we propose utilizing heterogeneous graph neural networks (HGNNs) as the backbone model to handle hierarchical context effectively. Furthermore, to enhance node embeddings in the HGNN, we integrate external knowledge from spoken content, such as text-based features (vocabulary profile) and speech-based features (filled pauses). Experimental results on the NICT JLE corpus validate the efficacy of our approach, achieving superior performance over the existing Transformer-based language models. Our findings also highlight the utility of our method in accurately evaluating speaking proficiency, showcasing its practical promise.

Index Terms: Automated Speaking Assessment (ASA), CEFR (Common European Framework of Reference for Languages), Graph attention network (GAT)

1. Introduction

Automated speaking assessment (ASA) is a computer-based method to evaluate, for example, the performance of English-as-a-second-language (ESL) learners on standardized tests and convert it into performance metrics. This method is valuable for assessing the speaking skills of ESL learners in a consistent, objective, and efficient manner [1]. To provide learners with interpretable and reliable feedback, widely adopted scoring references, such as the common European framework of reference (CEFR) [2], are generally given by institutes to systemically present and analyze the language learning status of students or test-takers.

The prevailing approaches to ASA of language proficiency typically employ linguistic and acoustic features derived from automatic speech recognition (ASR) systems. These features include, but are not limited to, acoustic features, recognized phones and words, and multi-granular time-aligned information, which are intricately linked to distinct proficiency domains, such as fluency, pronunciation, prosody and text complexity [3, 4, 5, 6]. In turn, the obtained linguistic and acoustic cues are channeled into the corresponding analytic graders of different aspects. However, these hand-designed features may inevitably leave out some salient information relevant to ASA. As a remedy, a prompt-aware encoder is proposed, which is based on bi-directional long short-term memory (BLSTM) and an attention mechanism to capture an response and its contextualized information, thereby conditioning on text prompts (stimulus questions) to evaluate speaking proficiency according to a predefined scoring rubric [7, 8]. Those methods excel at capturing long-range dependencies but may struggle with longer content [9]. Some other attempts used Transformers [10] with multi-head self-attention to connect words more flexibly. In [11], the Transformer-based models have been shown to outperform LSTM-based models in evaluating monologic responses. In [12], a Transformer-based language model (LM) is applied to assess the content aspect of speaking proficiency in online dialogues with non-native speakers. Despite the success of these neural methods in capturing syntactic structures, they are faced with challenges in effectively integrating hierarchical context. Heterogeneous graph neural networks (HGNNs) provide a promising solution by leveraging relay nodes to transmit information between super nodes, enabling capturing both local and global hierarchical context [13]. HGNNs also can extract fine-grained details and specific meanings from content and words. Notably, [14] is the first work that employed HGNNs to evaluate the English written ability of test-takers. To our best knowledge, research has yet to address the ASA problem by utilizing HGNNs. In view of this, we propose a novel approach: Graph-Enhanced resPonse encoder based on Transformer architecture (GEPT) for ASA, so as to improve spoken language proficiency assessment based on a graph attention network (GAT) [15] to explicitly model the dynamic interaction between responses in a meeting with conditional response information, building upon the existing Transformer-based model.

On the other hand, our proposed method performs ASA-based ASR transcripts, which meets the demand for education providers who only access text results converted from users’ audio recordings due to privacy reasons. The associated content-scoring engine typically uses sparse features, such as unigrams and bigrams, and is trained on a large set of responses. Such a regime has shown to be not inferior in performance comparable
to that based on fluency and pronunciation [16]. In [11], text transcripts generated by an off-the-shelf ASR system are used to predict overall spoken language proficiency. Similar methods in conjunction with natural language processing techniques have been applied with success in analogous fields, such as automated essay grading (AEG) [17, 18].

Nevertheless, spoken language assessments are usually conducted on spontaneous speech, in which disfluencies, such as hesitation ‘um’ shown in Figure 1, are common in spoken language [19], which does not exist in AEG. In addition, previous studies have proved that incorporating information about disfluencies can reduce learner errors’ impact on model precision [20]. Others still, language models (LMs) often prioritize lexical words over disfluencies and ignore the word difficulties tagged in Figure 1, which is a crucial proxy for evaluating spoken language proficiency and incorporating it as a supplement to assist in grading content [14]. In our proposed method, we also incorporate information on the filled pauses (FP) and vocabulary scale relationships to improve rendering the relationships between CEFR ratings, hesitations, and the content of the verbal response.

2. Proposed Method

2.1. Problem Formulation

In the corpus, a stage within an interview, which consists of distinct topics and role-play sections, holds a level of significance equivalent to a paragraph within a document. As an illustration, \((X, Z)\) is an input pair of interviewer’s and interviewee’s responses bundled to label \(Y\) predicting \(\hat{Y}\), where \(X = \{x_1, x_2, \ldots, x_{|X|}\}\), \(Z = \{z_1, z_2, \ldots, z_{|Z|}\}\). In more detail, the interviewees and interviewers responses are partitioned based on the closing delimiter marking the end of each stage; an interviewee’s response \(x = [\mu_1, \mu_2, \ldots, \mu_{|x|}]\) consists of \(|x|\) paragraphs and an interviewee’s response \(z = [\nu_1, \nu_2, \ldots, \nu_{|z|}]\) consists of \(|z|\) paragraphs. The \(\mu_i\) can be represented as a sequence of words \(\mu_i = [\mu_{i,1}, \mu_{i,2}, \ldots, \mu_{i,|\mu_i|}]\), where \(\mu_{i,j}\) denotes the \(j\)-th word of \(i\)-th interviewee’s response. The \(\nu_k\) can be represented as a sequence of words \(\nu_k = [\nu_{k,1}, \nu_{k,2}, \ldots, \nu_{k,|\nu_k|}]\), where \(\nu_{k,j}\) denotes the \(j\)-th word of \(k\)-th interviewee’s response. \(y\) and \(\hat{y}\) has the same number as \(|x|\). For the moment evaluating the performance of our proposed method, \(y'\) is calculated by an mean operation, \(y' = \frac{1}{|y|}\sum_{i=1}^{y} y[i]\), over all the paragraphs corresponding to the same speaker. Notice here that due to the restriction of no access to the interview’s audio data, we define our task to use only text-based features to predict spoken language proficiency.

2.2. Overall Architecture

Figure 2 schematically visualizes the model structure of our method, which comprises three parts: 1) a Transformer-based response encoder that exploits both the interviewer’s and interviewee’s responses to retrieve the similarity information. 2) a graph-based response encoder that extracts the contextualized embedding for the interviewee’s response. 3) a prediction head that predicts the final holistic proficiency of the interviewee’s response.

2.3. Transformer-based Response Encoder

The Transformer-based model aims to retrieve semantic-level similarity information from text content. Following [12], which concatenates interviewer and interviewee responses, passes the concatenated responses to the Transformer-based response encoder, and then evaluates the content dimension with respect to scoring rubrics [7]. Such an approach calculates the similarity between pairs of responses through a multi-head attention mechanism in a straightforward and effective manner. Following this, we adopt MpNet [21], a BERT [22] family model, as our Transformer-based response encoder.

2.4. Graph-based Response Encoder

The proposed graph-based response encoder consists of three main steps: heterogeneous graph construction, node embedding initialization, and graph attention operation. The heterogeneous graph adopted is an information network consisting of primary semantic units (word nodes) as relay nodes and other discourse units (paragraph nodes) as supernodes. The goal of the graph-based response encoder is to learn a mapping connection that projects the node embeddings initialized from \(H_{w} \in \mathbb{R}^{|V_{w}| \times d_{w}}\) of the t-time iterative updating process into a new representation \(H_{w}^{t}\) through an attention mechanism.

Heterogeneous Graph Construction. Let \(G = (V, E)\) represent an arbitrary graph, where \(V\) and \(E\) denote the node and edge sets, respectively. Our undirected heterogeneous graph can be formally defined as \(V = V_{w} \cup V_{p}\) and \(E = \{E_{w2w}, E_{w2p}, E_{p2w}\}\), where \(V_{w}\) and \(V_{p}\) denotes word and paragraph nodes from the interviewees, respectively, and \(E_{w2p}, E_{p2w}\) stand for word-to-paragraph, paragraph-to-word, and word-to-word edges, respectively.

Node Embedding Initialization. Let word node and paragraph node is denoted as \(F_{w} \in \mathbb{R}^{d_{w} \times d_{w}}\) and \(F_{p} \in \mathbb{R}^{d_{p} \times d_{p}}\) respectively. In the case of a word node, we initialize its representation by using GloVe [23]. In addition, we propose two kinds of embeddings, CEFR embedding, and FP embedding, to represent CEFR (0, A1, A2, B1, B2, C1, C2) for delivery information of each word level in the vocabulary profile and FP (0: not, 1: yes) for distinguishing filled pauses of each word. The CEFR embedding for each word is denoted as \(F_{w}^{CEFR} \in \mathbb{R}^{d_{CEFR} \times d_{w}}\) while FP embedding is denoted as \(F_{w}^{FP} \in \mathbb{R}^{d_{FP} \times d_{w}}\). Then we add them back to word embeddings in an element-wise manner. In the case of the paragraph node, the embedding of the interviewer’s response is computed as \(F_{w_{i}} = \text{CNN}(x_{i}[\mu_{i}]) \odot \text{BLSTM}(x_{i}[\mu_{i}])\). Similarly, the embedding of the interviewee’s response is \(F_{p_{i}} = \text{CNN}(z_{i}[\nu_{i}]) \odot \text{BLSTM}(z_{i}[\nu_{i}])\). That is, the concatenation operation \(\odot\) occurs at the dimension of the hidden state for two matrices, the convolutional neural network (CNN) layer to capture the local n-gram features for each paragraph, followed by the BLSTM layer capturing the paragraph-level features. Following [8], the paragraph node embedding is finally obtained.
by $H_{pe} = F_{pe} \odot F_{pre}$, where $i = k$.

**Graph Attention Operation.** Graph attention operation encapsulates the information from neighboring nodes. Given a heterogeneous graph $G$ and initialized features of each node, we refer to two samples $h_m, h_n \in \mathbb{R}^{d_v}$, $m, n \in \{1, \ldots, |V_p| + |V_r|\}$ and the neighborhood aggregation scheme can be defined with the multi-head attention head and a residual connection:

$$h'_m = \|_{k=1}^K \sigma(\sum_{j \in N_i} \alpha_{mn} W_v h_m),$$

$$h''_m = h_m + h'_m,$$

where $W_v$ and $\alpha_{mn}$ are trainable parameters, and $\alpha_{mn}$ is the non-linear attention weight transform between $h_m$ and $h_n$. The residual connection is to sidestep the gradient vanishing problem. After each GAT layer, we utilize a position-wise feed-forward layer composed of two linear convolutional transformations.

**Graph Propagation:** It is an iterative updating process that recursively aggregates and compresses node features from the local neighborhoods. After the initialization, the paragraph nodes are updated with their neighbor words nodes and paragraph nodes using GAT and the feed-forward network (FFN) layer. Then we can obtain the final paragraph node embedding $H_p$.

### 2.5. Prediction Head

The prediction head predicts a score for the interviewee’s response depending on two kinds of embedding from which the components are: the graph-based response encoder and the Transformer-based response encoder. To observe the efficacy of the embedding from the Transformer-based response encoder, we adopt a gated fusion method [24] that designed learnable gate values computed with a linear layer followed by a sigmoid function. The input features include $H_p$ and the Transformer similarity embedding $H_h$. We denote the gate vectors for similarity embedding as $g_0$:

$$g_0 = \sigma(W_0 (H_p \odot H_h) + b_0),$$

where $W_0$ and $b_0$ are trainable parameters. The fused representation $H_f$ is derived through following equations:

$$H'_{fb} = (W_{fb} h_b + b_{fb}) \odot g_b,$$

$$H_f = H_b + H'_b,$$

where $W_{fb}$ and $b_{fb}$ are trainable parameters, $\odot$ represent the element-wise multiplication and $H'_b$ represents the weight-computed embedding. Finally, the predicted score $\hat{Y}$ is derived by:

$$\hat{Y} = PH (WH_f + b),$$

where $W$ and $b$ are trainable, $PH$ is the final prediction head.

### 2.6. Training Objectives

The final layer of the prediction head is bound to the scoring scale (0-6), which is an ordinal ranking number. We regard this problem as a regression problem. Therefore, we use mean square error (MSE) loss function for the prediction task. Furthermore, to reduce the impact of imbalanced data, we adopt reweighting techniques in the loss function:

$$loss' = loss + (1 - \frac{N}{N_c})^\beta,$$

where $\beta$ is a controllable parameter, $N$ represents the total number of speakers in the training set, while $N_c$ refers to the number of speakers in each CEFR level within it.

### 3. Datasets and Experimental Setup

#### 3.1. Datasets

The NICT JLE [26] corpus comprises oral interviews conducted in English with 1,281 Japanese individuals and 20 American native individuals. It contains 2 million human-annotated words with rich annotations such as disfluencies (e.g., filled pauses), but only text-annotated transcriptions can be accessed. The interviews in the corpus involve a test-taker being interviewed by an English tutor or examiner and consist of multiple stages, including warm-up questions, single picture description, simulated conversation (role-play), story description using designated pictures, and wind-down questions. The interviews resemble conversations of specific topics, which were held with non-fixed prompt questions, and the interviewees are expected to respond promptly. Each interview has a standard speaking test score (SST), which is ranked on a scale and can be converted to corresponding CEFR levels [25]. Among them, C is a native-like fluency. For our experiments, we further assigned labels for the converted CEFR levels as shown in Table 1. We used up 1,298 interviews and split them into training, validation and test sets, as summarize in Table 2 except for three preA1 speakers. We also used the CEFR-J Wordlist [27, 14] Vocabulary Profile 1.6 to assign CEFR levels (A1 to C2)/sup to words of parts-of-speech. It is important to note that a word can have different CEFR levels depending on its phrasing. Since we will conduct evaluation separately on each topical conversation; therefore, we initialized the mean embedding value of words based on their frequency in the vocabulary profile.

### 3.2. Implementation details

The human-annotated spoken content is pre-tokenized via UD-Pipe [28] toolkit, in which the LM is trained on mass web available media text content. The filled pauses were temporarily removed before pre-tokenizing and added back after. The filled pauses and word CEFR level are then labeled to align the length

![Image of a page from a document](image_url)
of each response. In the detail of word CEFR level, the vocabulary profile is preprocessed, such as adding both United Kindom English and United States English of the same word, to avoid wrong labeling in the human-annotated spoken content.

As mentioned in our proposed HGNN grading method, we adopt [29] and modify it to fit our grading task. Moreover, the BERT model parameters were initialized with sentence-transformers/all-mpnet-base-v2 provided by HuggingFace [10], and consists of 12 multi-head attention with 12 layers of dimension 768. The training batch size is four, and the accumulation gradient is 16. For more details, our code is released here\(^4\). The model performance evaluation adopts root-mean-square error (RMSE), while further comparisons include Pearson’s correlation coefficient (PCC). Additionally, to determine the accuracy of classification, we designed margin accuracy within 0.5 (ACC05) and 1.0 (ACC10), under-estimate rate (UR), and over-estimate rate (OR). However, we also wanted to estimate the situation at individual levels. Therefore, RMSE, PCC, ACC05, ACC10, UR, and OR are individually divided into two types: micro and macro.

4. Experimental Results

The performance of our proposed method and the baseline are shown in Table 3. The performance of our proposed method and the baseline are summarized in Table 3. We make three observations from this table: the difference between the baseline and our proposed method, the effectiveness of eliminating the Transformer-based encoder, and our proposed CEFR and FP embeddings. The baseline model is based on BERT and uses both the interviewer’s and interviewee’s responses to predict overall proficiency. Our proposed model outperforms the baseline in most metrics except for OR, indicating that the graph-based response encoder helps the grading model focus on overall hierarchical context better than a Transformer-based model but may be obscure in each CEFR group, such as the difference of inclination words. Figure 3 also illustrates the same reality in another aspect. Most scores are predicted at the correct position; even a few in B2 and C-level speakers’ data can also be predicted. However, the model tends to under-score.

Removing the BERT-based encoder reduces RMSE, PCC, Macro-ACC, and UR performance, indicating that the encoder plays a role in other aspects of proficiency, such as semantic context. Additionally, we introduce CEFR and FP embeddings, added to the word vectors generated from GloVe. We evaluated their effectiveness by gradually removing them in experiments. Table 3 shows that removing FP embedding while leaving CEFR embedding alone improves RMSE, ACC05, micro-ACC10, and OR performance but slightly decreases other metrics, implying that CEFR embedding still contributes to the representation and model evaluating ability. Removing CEFR embedding while leaving FP embedding alone also reduces performance, demonstrating its significance. Finally, if both CEFR and FP embeddings are removed, the metrics show a significant drop in performance. Nonetheless, the original GEPT model maintains a balanced performance across various aspects of metrics.

5. Conclusions

In this paper, we have proposed using graph-based response encoding to learn the hierarchical context of the interviewee’s response, which incorporates CEFR and FP information to enrich the graph representation for final proficiency prediction. The experiments have revealed the efficacy of our proposed modeling mechanisms, viz. graph-based response encoder, CEFR and FP embedding, can make significant contributions to ASA performance in most cases. In the future, we will extend our method to render the dialogue behaviors existing in ASA and leverage other graph modeling techniques, meanwhile incorporating acoustic features or self-supervised learning representations whose effectiveness has been proved [30, 31], in another corpus available with audio data. In addition, we put further steps in dealing with the uncertainty distribution in CEFR levels to reduce the negative impact on ASA systems for giving more reliable feedback to users.

6. Acknowledgements

The original CEFR label in Table 1 is provided by Cambridge Assessment English. We express our gratitude to Dr. Kate Knill from the Department of Engineering, University of Cambridge, for supporting the development of ASA.

---

\(^3\)https://huggingface.co/sentence-transformers/all-mpnet-base-v2
\(^4\)https://github.com/a2d8a4v/gept

---

Table 3: The results of the grading model: Baseline, our proposed model, and ablation study. FPe and CEFRe represent FP embedding and CEFR embedding, respectively.

<table>
<thead>
<tr>
<th>Model</th>
<th>RMSE ((\downarrow))</th>
<th>Macro-RMSE ((\downarrow))</th>
<th>PCC ((\uparrow))</th>
<th>Accuracy Micro</th>
<th>Macro-Accuracy Micro</th>
<th>Over-estimate rate ((\uparrow)) Micro</th>
<th>Under-estimate rate ((\downarrow)) Micro</th>
<th>Accuracy Macro</th>
<th>Macro-Accuracy Macro</th>
<th>Over-estimate rate ((\uparrow)) Macro</th>
<th>Under-estimate rate ((\downarrow)) Macro</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0.483</td>
<td>0.820</td>
<td>0.769</td>
<td>71.034</td>
<td>97.241</td>
<td>43.539</td>
<td>75.000</td>
<td>11.724</td>
<td>8.957</td>
<td>17.241</td>
<td>47.504</td>
</tr>
<tr>
<td>GEPT</td>
<td>0.431</td>
<td>0.515</td>
<td>0.820</td>
<td>72.414</td>
<td>98.621</td>
<td>66.029</td>
<td>90.000</td>
<td>15.172</td>
<td>11.809</td>
<td>12.414</td>
<td>22.162</td>
</tr>
<tr>
<td>- BERT</td>
<td>0.460</td>
<td>0.662</td>
<td>0.809</td>
<td>72.414</td>
<td>97.241</td>
<td>42.543</td>
<td>89.009</td>
<td>6.897</td>
<td>6.106</td>
<td>20.690</td>
<td>51.350</td>
</tr>
<tr>
<td>- FPe</td>
<td>0.426</td>
<td>0.552</td>
<td>0.822</td>
<td>73.793</td>
<td>99.310</td>
<td>43.334</td>
<td>95.000</td>
<td>12.414</td>
<td>9.726</td>
<td>13.793</td>
<td>46.940</td>
</tr>
<tr>
<td>- CEFRe</td>
<td>0.457</td>
<td>0.622</td>
<td>0.824</td>
<td>72.414</td>
<td>96.552</td>
<td>41.896</td>
<td>88.239</td>
<td>4.828</td>
<td>6.087</td>
<td>22.759</td>
<td>52.017</td>
</tr>
<tr>
<td>- FPe</td>
<td>0.433</td>
<td>0.601</td>
<td>0.822</td>
<td>75.172</td>
<td>97.931</td>
<td>42.037</td>
<td>89.231</td>
<td>8.966</td>
<td>8.715</td>
<td>15.862</td>
<td>49.248</td>
</tr>
</tbody>
</table>

Figure 3: The confusion matrix of our proposed method.
7. References


