Question Difficulty Decision for Automated Interview in Foreign Language Test

Minghao Lin[1], Koichiro Ito[1] and Shigeki Matsubara[1,2][a]
[1] *Graduate School of Informatics, Nagoya University, Japan*
[2] *Information & Communications, Nagoya University, Japan*

Keywords: Question Generation, Computer-Assisted Language Assessment, Language Speaking Test, Oral Proficiency Interview.

Abstract: As globalization continues to progress all over the world, demand is growing for objective and rapid assessment of language proficiency in foreign language learners. While automated assessment of listening, reading, and writing skills has been proposed, little research has been done to automate assessment of speaking skills. In this paper, we propose a method of deciding the difficulty of questions generated for interview tests of skills assessment in speaking a foreign language. To address question difficulty flexibly according to the abilities of test takers, our method considers the appropriateness of responses from test takers. We implemented this method using the large-scale pre-trained language model BERT (Bidirectional Encoder Representation from Transformers). Experiments were conducted using simulated test data from the Japanese Learner's Conversation Database to confirm the effectiveness of our method in deciding difficulty.

1 INTRODUCTION

The need to evaluate proficiency in a second language is increasing, as the number of people moving overseas abroad to study or work is increasing all over the world. The four main language skills that are assessed to determine second language proficiency are reading, writing, listening, and speaking (Powers, 2010). Automated assessment methods for the first three skills have been discussed in previous studies, including question generation and machine scoring (Huang et al., 2014) (Du et al., 2017) (Yannakoudakis et al., 2011).

Assessment of second language speaking remains an under-researched area due to its complexity. The process of evaluation involves a vast amount of human labor, specialized equipment, and specialized environments. Some studies that previously investigated speaking skills assessment were conducted to alleviate this phenomenon and to accomplish the goal of evaluating the speaking skills of individuals in a second language (Litman et al., 2016).

According to (Bahari, 2021), computer-assisted language assessment studies are moving towards nonlinear dynamic assessment, which focuses on the individual learner, by introducing interactive, dynamic, and adaptive strategies. However, previous studies in this area have hitherto focused on monologue-type tests of speaking. This approach is insufficient for implementing a reliable test or for simulating a real-life test environment for human–human oral proficiency.

Oral proficiency in language assessment includes the following two parts:

- **Test delivery:** To acquire speech samples from a second language learner, the following two main types of tests have been proposed: monologue and interview. A monologue test is conducted with a static question difficulty. This means that the questions are created in advance and are not to be changed during the assessment. An interview test is a standardized, global assessment of functional speaking ability in the form of a conversation between the tester and the test taker. The questions posed in an interview test are dynamic.
- **Scoring:** The analysis of the speech sample collected in the test allows the speaking skills of the test taker to be fully assessed. Different test strategies focus on a vast variety of aspects, including pronunciation and accuracy.

[a] https://orcid.org/0000-0003-0416-3635

Lin, M., Ito, K. and Matsubara, S.
Question Difficulty Decision for Automated Interview in Foreign Language Test.
DOI: 10.5220/0011774900003393
In *Proceedings of the 15th International Conference on Agents and Artificial Intelligence (ICAART 2023) - Volume 3*, pages 760-767
ISBN: 978-989-758-623-1; ISSN: 2184-433X

According to (Bernstein et al., 2010), testers expect to see certain elements of real-life communication to be represented in the test. The interview test meets this expectation to the greatest degree among oral proficiency tests. The interview test is always conducted dynamically, in that the tester does not always pose questions with the same difficulty.

In this paper, we propose a method of deciding the proper difficulty for questions in an interview test in response to the second language speaking skill of individual test takers. This method is built upon the large-scale pre-trained language model BERT (Bidirectional Encoder Representation from Transformers) (Devlin et al., 2019). BERT was proposed to perform natural language processing and has proven effective in sentiment analysis, question answering, document summarization, and other tasks. This motivates us to imply such a model to difficulty decision tasks.

The conversational context and additional appropriateness information for responses of test takers are used in our method. We conducted our experiments using a simulated language oral speaking interview test dataset to validate our method. This method outperformed the baseline models, confirming the validity of our proposed method.

The remainder of this paper is organized as follows: Section 2 reviews recent research on the automation of the oral proficiency assessment. Section 3 describes the difficulty decision task and provides insight into the structural design of the proposed model. Section 4 presents a full account of the experimental setting. Finally, we provide a brief summary of our work and discuss potential objections to our plan for future work.

2 RELATED WORK

Many studies put effort into the automation of oral proficiency assessment with static question difficulty. The samples that were used to judge speaking skills of a test taker were collected as monologues. In (Yoon and Lee, 2019), the authors collected around one minute of spontaneous speech samples from test takers, including readings and/or answers after listening to a passage. They used a Siamese convolutional neural network (Mueller and Thyagarajan, 2016) to model the semantic relationship between the key points generated by experts and test responses. The neural network was also used to score the speaking skill of each test taker. In (Zechner et al., 2014), the authors collected restricted and semi-restricted speech from test takers. The restricted speeches involved reading and repeating a passage. In the semi-restricted speech, the test taker is required to provide sufficient remaining content to formulate a complete response, corresponding to an image or chart. The authors proposed a method that combines diverse aspects of the features of speaking proficiency using a linear regression model to predict response scores.

The studies mentioned above use a strategy that collects samples manually and then analyzes them using algorithms. Some previous studies also use machines to deliver tests and collect samples. In a Pearson Test of English (Longman, 2012), test takers are requested to repeat sentences, answer short questions, perform sentence builds, and retell passages to a machine. Their responses are analyzed by algorithms (Bernstein et al., 2010). In (de Wet et al., 2009), the authors designed a spoken dialogue system for test takers to guide them and capture their answers. The system involves reading tasks and repetition tasks. The authors used the automatic speech recognition to evaluate speaking skills of test takers, focusing on fluency, pronunciation, and repeat accuracy.

Oral proficiency tests delivered using the monologue test have been used to evaluate the speaking skill of test takers and have shown a high degree of correlation with the interview test (Bernstein et al., 2010). However, automation of the interview assessment method with dynamic question difficulty has not been developed to the extent that automation of the monologue one has. This paper seeks to fill this gap.

3 METHOD

3.1 Problem Setting

In an interview test, the tester follows the strategy below:

- First, the appropriateness of the responses of the test taker is estimated.
- Second, based on this appropriateness measure, the difficulty of the question to be given next is decided.

As noted in (Kasper, 2006), if it is difficult for the test taker to respond to the given question, the tester would change the question as the next action. The questions target a specific oral proficiency level and functioning at that level (ACTFL, 2012). This means that an automated interview test should have the ability to adjust the difficulty level of its questions during the course of an interview test. In this paper, we propose a method to decide the difficulty of the next question posed by the tester. Our method should first estimate the appropriateness of the response of the

Table 1: Examples of interview test dialogue. The appropriateness and the difficulty are shown below the responses and questions, respectively.

	utterances (*appropriateness / difficulty*)
Q_1	What is your favorite sport? (*easy)*
R_1	My favorite sport is soccer. (*appropriate)*
Q_2	Could you please explain the rule of it? (*difficult)*
R_2	Uh, it's ... a game like ... I'm sorry, it's hard to tell. (*inappropriate)*
Q_3	OK, I see. Then ... Do you find this sport funny? (*easy)*
R_3	Oh yes, I enjoy playing it. (*appropriate)*

test taker and then choose a suitable difficulty level for the next question.

We denote the entire interview test dialogue with the symbol D. D contains the following two types of utterances: questions delivered by the tester and responses performed by the test taker. The ith question in D is denoted as Q_i, while ith response in D is denoted as R_i. When there are n pairs of questions and responses in D, D is denoted as $[Q_1, R_1, Q_2, R_2, \cdots, Q_n, R_n]$. An example dialogue is shown in Table 1. The second question utterance is the question Q_2 (= "Could you please explain the rule of it?") asking the test taker to explain the rules of the sport. The test taker finds it difficult to produce a response. The test taker makes the following inappropriate response: R_2 (= "Uh, it's ... a game like ... I'm sorry, it's hard to tell."). Here, the tester should reduce the difficulty level of the questions and provide an easy question next. Conversely, if the test taker were to give the response, e.g., R'_2 (= "Basically, it is a sport in which two teams of 11 players use a single ball and kick it into each other's goal."), the test taker has successfully answered the tester's question and given an appropriate response. The tester is more likely to give a difficult question next, such as asking for the test taker's opinion of whether it is fair to the average student for athletes to be preferred for admission to the university.

3.2 Models Structure

To perform question difficulty decision in an interview test, we propose a two-step method that uses the BERT (Devlin et al., 2019), as shown in Figure 1. BERT is built upon the Transformer architecture (Vaswani et al., 2017). Some previous works have applied BERT to dialogue-related tasks (Wu et al., 2020) (Song et al., 2021) (Wang et al., 2021). Our method consists of two parts, namely, the response appropriateness estimation model and the question difficulty decision model. We define our consecutive method structure as follows: given an interview test dialogue context, the method estimates response appropriateness to identify whether the response given is appropriate as a response to the given question. Then, the method uses the appropriateness estimation result and the dialogue context mentioned above to set the difficulty of the next question.

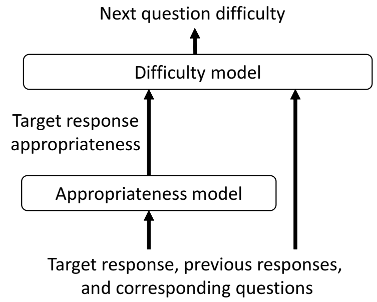
Next question difficulty
Difficulty model
Target response appropriateness
Appropriateness model
Target response, previous responses, and corresponding questions

Figure 1: Structure of our method.

3.2.1 Response Appropriateness Estimation

The main structure of our model is shown in Figure 2. To estimate the appropriateness of response R_i, two pairs of questions and responses $[Q_{i-1}, R_{i-1}, Q_i, R_i]$ are used as input for the model. Q_i is the question corresponding to response R_i, and $[Q_{i-1}, R_{i-1}]$ serve as the context. The two pairs of questions and responses are input into the tokenizer, and [SEP] tokens are inserted at the end of each tokenized utterance. The input tokens of the utterance are denoted with the corresponding lowercase letter. For example, R_i is denoted as $[r_i^1, r_i^2, \cdots, r_i^m]$, where r_i^j is the jth token in R_i.

The input token embedding is denoted as E. For example, $E_{r_i^j}$ denotes the token embedding of token r_i^j, or the jth token in R_i. The two types of segment embeddings are denoted by E_A and E_B. E_i denotes the position embedding of the ith token in the input token sequence. The last hidden states of all tokens are denoted as T. For example, $T_{r_i^j}$ denotes the last hidden states of the token r_i^j. The representation of [CLS] token $T_{[CLS]}$ was used as input to the classifier. The classifier has linear transformation layers with a softmax function, performing a response appropriateness estimation.

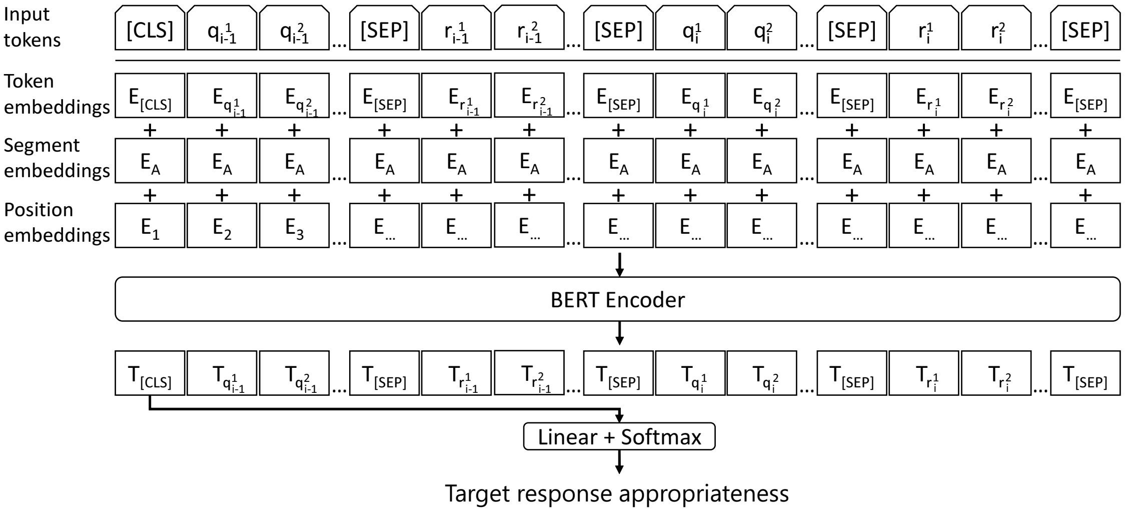

Figure 2: Structure of proposed method for response appropriateness estimation.

3.2.2 Question Difficulty Decision

Within the interview test dialogue context, our method utilizes the corresponding response appropriateness estimation result to decide on difficulty of questions. Response appropriateness estimation result is represented by a special token set $[[0],[1]]$, as it has been shown that special tokens can be useful in representing abstract concepts (Xie et al., 2021). Token [0] denotes an inappropriate response, while token [1] denotes an appropriate one. In the following, the special token is also referred to as the response appropriateness label.

To decide the difficulty of the next question Q_{i+1}, our method uses two pairs of questions and responses $[Q_{i-1}, R_{i-1}, Q_i, R_i]$ as input information. The appropriateness label of R_i was taken as additional information. With the exception of the additional appropriateness label and the input format in the sentence-pair configuration, the other parts of the model remain the same as in the appropriateness estimation model. The main structure of the difficulty decision model is shown in Figure 3.

We build our model using the label-fusing method proposed by (Xiong et al., 2021). Although their method utilizes the default sentence-pair input configuration in BERT, it uses different segment embeddings for the labels and context, without changing the original encoder structure. This means that the non-natural language labels and natural language contexts can be distinguished by segment embedding (E_A and E_B) and concatenated with a [SEP] token as input. This method, which utilizes a label embedding technique, could improve the performance of BERT in text classification while maintaining nearly the same computational cost. The appropriateness estimation information, which is represented as a special token, is added to the BERT dictionary. The appropriateness label that serves as input is denoted by L.

4 EXPERIMENTS

4.1 Dataset

To evaluate the effectiveness of our method, we performed experiments on simulation test data in the Japanese Learner's Conversation Database (JLCD)[1], published by the National Institute for Japanese Language. All of the simulated interview tests were performed according to the ACTFL Oral Proficiency Interview (OPI) standard, which assesses the ability to use language effectively and appropriately in real-life situations (ACTFL, 2012). After assessment in tests that take around twenty minutes, test takers are rated on the following four proficiency levels: *superior*, *advanced*, *intermediate*, and *novice*. All interview contents are presented as spoken transcripts created manually.

In all, there are 390 transcript data in the dataset, of which we used 59. We split these data into approximately 8:1:1 for training, development, and testing. We divided the transcript data into utterances. The transcript data were divided into 3,251 question utterances and 3,251 response utterances. We manually annotated each utterance. The response utterances spoken by the test takers were marked as appropriate or inappropriate, with reference to both the

[1] https://mmsrv.ninjal.ac.jp/kaiwa/DB-summary.html

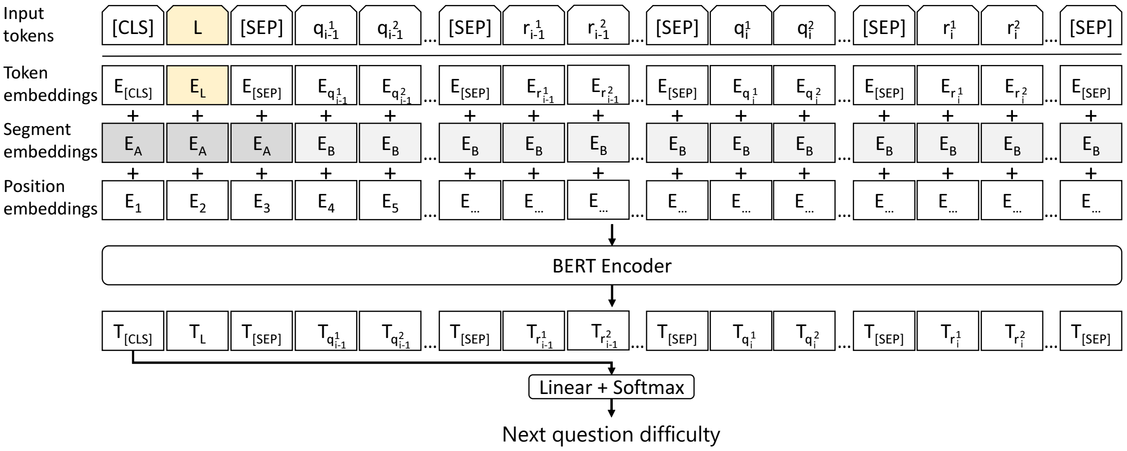
Input tokens
[CLS]
L
[SEP]
q_{i-1}^1
q_{i-1}^2
[SEP]
r_{i-1}^1
r_{i-1}^2
[SEP]
q_i^1
q_i^2
[SEP]
r_i^1
r_i^2
[SEP]
Token embeddings
$E_{[CLS]}$ E_L $E_{[SEP]}$ $E_{q_{i-1}^1}$ $E_{q_{i-1}^2}$ $E_{[SEP]}$ $E_{r_{i-1}^1}$ $E_{r_{i-1}^2}$ $E_{[SEP]}$ $E_{q_i^1}$ $E_{q_i^2}$ $E_{[SEP]}$ $E_{r_i^1}$ $E_{r_i^2}$ $E_{[SEP]}$
Segment embeddings
E_A E_A E_A E_B E_B E_B E_B E_B E_B E_B E_B E_B E_B E_B E_B
Position embeddings
E_1 E_2 E_3 E_4 E_5 $E_{...}$
BERT Encoder
$T_{[CLS]}$ T_L $T_{[SEP]}$ $T_{q_{i-1}^1}$ $T_{q_{i-1}^2}$ $T_{[SEP]}$ $T_{r_{i-1}^1}$ $T_{r_{i-1}^2}$ $T_{[SEP]}$ $T_{q_i^1}$ $T_{q_i^2}$ $T_{[SEP]}$ $T_{r_i^1}$ $T_{r_i^2}$ $T_{[SEP]}$
Linear + Softmax
Next question difficulty

Figure 3: Structure of proposed method for next question difficulty decision. Input token L represents the appropriateness label of the target response.

Table 2: Response appropriateness distribution.

label	train	dev	test
appropriate	2,069	233	292
inappropriate	491	73	93

Table 3: Question difficulty distribution.

label	train	dev	test
easy	1,829	228	287
difficult	731	78	98

response and the corresponding question. Table 2 presents the distribution of response appropriateness across datasets.

In the ACTFL-OPI standard, *novice* and *intermediate* level questions involve simple inquiries, and test takers are asked to make binary choices or provide simple factual conclusions. Then, *advanced* questions involve describing an object in detail, retelling a story, and comparing two targets. Questions on the *superior* level involve expressing personal opinions or views on social phenomena and abstract concepts. In this paper, we defined the decision for the difficulty of the next question as a binary classification task, in reference to whether the next generated question would be easy or difficult. Questions involving binary choices or providing simple factual conclusions were treated as easy questions. Questions involving describing objects in detail, presenting personal opinions, and the rest were treated as difficult questions. The question utterances spoken by testers were marked as easy or difficult. Table 3 exhibits the question difficulty distribution across datasets.

4.2 Model Settings

We implemented the response appropriateness estimation model and the question difficulty decision model by fine-tuning pre-trained BERT. We used the bert-base-japanese model[2] published by Tohoku University for the appropriateness estimation model and the difficulty decision model. Both models were set to perform binary classification tasks to complete appropriateness estimation and difficulty decision. The learning rate was 1e-5 for the appropriateness estimation model, and 2e-7 for the difficulty decision model. The batch size was set to 16. We used AdamW(Loshchilov and Hutter, 2019) as the optimizer, and we set the weight decay to 0.01. The dropout rate was kept at 0.1. The maximum input token length remained at 512. We used cross-entropy loss for the training.

The distribution of labels was imbalanced in both the response appropriateness estimation task and the question difficulty decision task. We calculated each class weight using Equation (1), and we utilized the class weight while training our model. For W_j, i.e., weight of a class j, $n_{samples}$ mark the total number of samples in the data, and $n_{classes}$ represents the total number of unique classes, while $n_{samples_j}$ gives the total number of samples in the class j.

$$W_j = \frac{n_{samples}}{n_{classes} * n_{samples_j}} \tag{1}$$

In the training of the question difficulty decision model, manually annotated appropriateness labels were used for responses. While testing, labels that were estimated using the response appropriateness estimation model were used instead of the man-

[2] https://github.com/cl-tohoku/bert-japanese

ually annotated labels. Therefore, the performance of the question difficulty decision model on the test data was affected by error in the response appropriateness estimation model. This proposed method is denoted as [ours (noisy)]. Each model was trained for ten epochs for the corresponding task. The models were evaluated in the development set at the end of each epoch, and the those with the lowest evaluation loss were saved. We report performance of those models on the test set. Training, evaluation, and testing processes of both models were done on a single NVIDIA RTX A5000 graphics card.

4.3 Baselines and Metrics

In the response appropriateness estimation task, we implemented a random prediction baseline. The rate of predicting that a target response was appropriate was set to 0.809, based on the appropriate response rate in our training data.

For the question difficulty decision task, we implemented the following four methods by comparison with the proposed method [ours (noisy)].

- [random]: This method randomly predicts the difficulty of questions. The rate of predicting whether the difficulty of the following question was easy was set to 0.714, based on the easy question rate in our training data.
- [vanilla BERT]: This method only uses dialogue context as an input in making question difficulty decisions, which means that no appropriateness label was included. This method was developed to evaluate the validity of introducing appropriateness information.
- [ours (noisy) w/o seg]: This method excludes different segment embedding in our model. The output labels for the response appropriateness model were used while testing. This method is used to verify the effectiveness of segment embeddings in distinguishing special tokens from natural language context.
- [ours (correct)]: This method uses correct appropriateness labels with manual annotations for training and testing. In this method, all labels serve as correct appropriateness information and are included in the input for the difficulty decision model. This method is intended to evaluate the performance of the difficulty decision model where the appropriateness estimation result produced by the appropriateness estimation model was absolutely correct. The performance of this method is the reference value in the ideal environment.

Table 4: Performance of appropriateness estimation method on the test set. All the results were macro-averaged.

	Precision	Recall	F1
[random]	0.473	0.480	0.472
[ours]	**0.665**	**0.653**	**0.658**

Table 5: Performance of difficulty decision method on the test set. All the results were macro-averaged.

	Precision	Recall	F1
[random]	0.500	0.500	0.499
[vanilla BERT]	0.605	0.619	0.609
[ours (noisy) w/o seg]	0.626	0.638	0.631
[ours (noisy)]	**0.629**	**0.648**	**0.635**
[ours (correct)]	0.631	0.650	0.637

We report macro-precision, macro-recall, and macro-F1 for all of our experiments, as we are not only focused on the performance of a single label, but both are important.

4.4 Experimental Results

We report the result of the evaluation of the appropriateness estimation method in Table 4. The macro-F1 results show that the appropriateness estimation for the responses were feasibly implemented by utilizing the BERT model. The accuracy of the BERT model is 0.758. Therefore, in testing the question difficulty decision method [ours (noisy)], the correct response appropriateness labels were input at a rate of 0.758.

Table 5 shows the results of the evaluation of the decision method for question difficulty. It shows that the proposed method [ours (noisy)] outperformed [random] and [vanilla BERT], in which the appropriateness information was not presented[3]. Table 6 shows an example where [ours (noisy)] was correct, but [vanilla BERT] was not correct. In this example, [ours (noisy)] correctly predicted that the question generated after R_{55} would be difficult by utilizing information that R_{55} was an appropriate response.

The F1 score of the proposed method [ours (noisy)] shows a slight decrease relative to [ours (correct)]. It is thought that this result is due to the error in the estimation model of response appropriateness. The performance of [ours (noisy) w/o seg] is degraded relative to [ours (noisy)], indicating that the different segment embeddings are effective for the importation of response appropriateness labels into the input sequences.

[3] The difference between the proposed method [ours (noisy)] and other methods were evaluated using McNemar's test with $p < 0.05$. The random prediction method showed a statistically significant difference, while the remaining methods did not.

Table 6: Examples of a sample where [vanilla BERT] provides an incorrect difficulty decision, but [ours (noisy)] is correct. The *appropriateness* of R_{55} shows the result of the response appropriateness estimation result, and this estimation result was correct. The *difficulty* columns show the true labels in the data.

	utterance	*appropriateness*	*difficulty*	[vanilla BERT]	[ours (noisy)]
Q_{54}	So you send messages with some characters that can be pronounced like alphabets?	-	*easy*	-	-
R_{54}	You mean characters that can be pronounced.	-	-	-	-
Q_{55}	Yes, do you use the Russian alphabets to write Mongolian texts, like writing in the English alphabets?	-	*easy*	-	-
R_{55}	It's similar to Russian alphabets, but not identical.	*appropriate*	-	-	-
Q_{56}	So what's the difference between Mongolian alphabets and Russian alphabets?	-	*difficult*	*easy*	*difficult*

5 CONCLUSIONS

We proposed a two-step method for question difficulty decision for automated interview test delivery using the large-scale language model BERT, including response appropriateness estimation. The experimental results show that the judgments of appropriateness estimation were useful when deciding the difficulty of the subsequent question.

This method only uses two pairs of questions and responses in the dialogue context as input. Long-term contexts containing temporal information could play an important role in dialogue-related tasks, and they should not be simply discarded. Thus, we intend to explore this further in later work. We will also examine different means of incorporating additional information that could be beneficial for interview test automation. For example, tester strategy may differ between the early and late stages of an interview. The early stage focuses on probing, and later stage focus on level checking. Therefore, the amount of time spent on the current interview will have an impact on future difficulty decisions.

REFERENCES

ACTFL (2012). Oral proficiency interview familiarization manual. https://community.actfl.org.

Bahari, A. (2021). Computer-assisted language proficiency assessment tools and strategies. *Open Learning: The Journal of Open, Distance and e-Learning*, 36(1):61–87.

Bernstein, J., Van Moere, A., and Cheng, J. (2010). Validating automated speaking tests. *Language Testing*, 27(3):355–377.

de Wet, F., Van der Walt, C., and Niesler, T. (2009). Automatic assessment of oral language proficiency and listening comprehension. *Speech Communication*, 51(10):864–874.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Du, X., Shao, J., and Cardie, C. (2017). Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352, Vancouver, Canada. Association for Computational Linguistics.

Huang, Y.-T., Tseng, Y.-M., Sun, Y. S., and Chen, M. C. (2014). Tedquiz: automatic quiz generation for ted talks video clips to assess listening comprehension. In *2014 IEEE 14th International Conference on Advanced Learning Technologies*, pages 350–354, Athens, Greece. IEEE.

Kasper, G. (2006). When once is not enough: Politeness of multiple requests in oral proficiency interviews. *Multilingua*, 25(3):323–350.

Litman, D., Young, S., Gales, M., Knill, K., Ottewell, K., Van Dalen, R., and Vandyke, D. (2016). Towards using conversations with spoken dialogue systems in the automated assessment of non-native speakers of English. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 270–275, Los Angeles.

Longman, P. (2012). *Official Guide to Pearson Test of English Academic*. Pearson Japan, London, 2nd edition.

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *Proceedings of the 7th International Conference on Learning Representations*, New Orleans, LA, USA.

Mueller, J. and Thyagarajan, A. (2016). Siamese recurrent architectures for learning sentence similarity. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, page 2786–2792. AAAI Press.

Powers, D. E. (2010). The case for a comprehensive, four-skills assessment of English-language proficiency. *R & D Connections*, 14:1–12.

Song, H., Wang, Y., Zhang, K., Zhang, W.-N., and Liu, T. (2021). BoB: BERT over BERT for training persona-based dialogue models from limited personalized data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 167–177, Online. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, P., Fang, J., and Reinspach, J. (2021). CS-BERT: a pretrained model for customer service dialogues. In *Proceedings of the 3rd Workshop on Natural Language Processing for Conversational AI*, pages 130–142, Online. Association for Computational Linguistics.

Wu, C.-S., Hoi, S. C., Socher, R., and Xiong, C. (2020). TOD-BERT: Pre-trained natural language understanding for task-oriented dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 917–929, Online. Association for Computational Linguistics.

Xie, Y., Xing, L., Peng, W., and Hu, Y. (2021). IIE-NLP-eyas at SemEval-2021 task 4: Enhancing PLM for ReCAM with special tokens, re-ranking, Siamese encoders and back translation. In *Proceedings of the 15th International Workshop on Semantic Evaluation*, pages 199–204, Online. Association for Computational Linguistics.

Xiong, Y., Feng, Y., Wu, H., Kamigaito, H., and Okumura, M. (2021). Fusing label embedding into BERT: An efficient improvement for text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1743–1750, Online. Association for Computational Linguistics.

Yannakoudakis, H., Briscoe, T., and Medlock, B. (2011). A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

Yoon, S.-Y. and Lee, C. M. (2019). Content modeling for automated oral proficiency scoring system. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 394–401, Florence, Italy. Association for Computational Linguistics.

Zechner, K., Evanini, K., Yoon, S.-Y., Davis, L., Wang, X., Chen, L., Lee, C. M., and Leong, C. W. (2014). Automated scoring of speaking items in an assessment for teachers of English as a foreign language. In *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 134–142, Baltimore, Maryland. Association for Computational Linguistics.