UNED team at BEA 2024 Shared Task: Testing different Input Formats for predicting Item Difficulty and Response Time in Medical Exams

Alvaro Rodrigo, Sergio Moreno-Álvarez, Anselmo Peñas
NLP & IR group at UNED
Madrid, Spain
{alvarory,smoreno,anselmo}@lsi.uned.es

Abstract

This paper presents the description and primary outcomes of our team's participation in the BEA 2024 shared task. Our primary exploration involved employing transformer-based systems, particularly BERT models, due to their suitability for Natural Language Processing tasks and efficiency with computational resources. We experimented with various input formats, including concatenating all text elements and incorporating only the clinical case. Surprisingly, our results revealed different impacts on predicting difficulty versus response time, with the former favoring clinical text only and the latter benefiting from including the correct answer. Despite moderate performance in difficulty prediction, our models excelled in response time prediction, ranking highest among all participants. This study lays the groundwork for future investigations into more complex approaches and configurations, aiming to advance the automatic prediction of exam difficulty and response time.

1 Introduction

In this paper, we describe the proposals sent by our team to the BEA 2024 shared task (Yaneva et al., 2024). This task aims to predict standardized exams' difficulty (Track 1) and response time (Track 2). The data used in this task is from a high-stakes medical exam called the United States Medical Licensing Examination[1]. The exams are provided in a multiple-choice format, with answer candidates ranging from 4 to 10.

Adjusting the difficulty of exams to align with the intended level of evaluation is crucial for ensuring the validity and fairness of assessments. Educators can accurately gauge students' understanding and proficiency within the targeted subject matter by calibrating the difficulty appropriately. This practice also promotes an equitable assessment environment where students can handle their challenges, allowing for a more reliable measure of their knowledge and skills. Moreover, it encourages a more constructive learning experience, as students are motivated to engage with material that appropriately matches their abilities, fostering growth and development. Ultimately, the careful adjustment of exam difficulty supports the effectiveness and integrity of the assessment process.

Several human examiners showed us these difficulties and asked for our help, opening the possibilities for an exciting application of language technologies to this problem. This is why our group is quite interested in this problem and participated in this task. Actually, we are working on automatically predicting the difficulty of examinations for new language learners. The exams of our work are also in a multiple-choice format but, the number of options is lower (3 or 4, depending on the exam).

Our primary objective in this task was centered on the initiation of experiments utilizing transformer-based systems (Vaswani et al., 2017) to explore their applicability to the given problem domain. Instead of using the most modern generative models such as ChatGPT[2], Llama-2 (Touvron et al., 2023) or Mixtral (Jiang et al., 2024), we explored the use of several BERT-based models (Devlin et al., 2019), which require less computational resources. We experiment with different input sequences and use the same data and approaches for both tracks. While our results in Track 1 (Item Difficulty Prediction) were relatively low (13th position for our best run), we obtained good results in Track 2 (Response Time Prediction), where we ranked in 1st, 3rd, and 4th position with our proposed systems.

The paper is structured as follows: we describe the main features of our approach in Section 2, while we detail the runs submitted to the task in Section 3. Then, we analyzed our results in Section

[1] https://www.usmle.org/

[2] https://chat.openai.com/

Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications, pages 567–570
June 20, 2024 ©2024 Association for Computational Linguistics

4. Finally, we give some conclusions and future work in Section 5.

2 Systems Description

In this Section, we describe the main features of our systems. In the development period, we tested different configurations using 10% of the training collection as test data. All our experiments are based on a BERT-base model[3] fine-tuned for regression (Devlin et al., 2019). We experimented with similar models like DeBERTa (He et al., 2021) and DistilBERT (Sanh et al., 2020), obtaining the best results with BERT. We focused on the base versions of these models instead of the large ones because we wanted to study the use of simple approaches that do not require big GPU units.

We applied the same pre-processing to all our models and focused on testing the effect of using different inputs for the model. We provide more details in the next subsections.

2.1 Pre-processing

We only used text from the item and the answers as input to our systems. More in detail, we only used the following text fields provided by the organizers:

- ItemStem_Text: contains the clinical case and the question.
- Answer_N: contains the text of the n-candidate answer.
- Answer_Text: contains the text of the correct answer,

We did not apply any special pre-processing to these input texts and used the tokenizer provided by the BERT model.

We scale the target variables (Difficulty for Track 1 and Response_Time for Track 2) into the $[0, 1]$ scale using the MinMaxScaler from sklearn[4], which gave us the best results in the development period.

2.2 Input Formats

We tested different input formats in our experiments. We wanted to explore the effects of using different combinations of text and study the importance of different text elements for solving the task. We explored the following input formats:

- **All text together**: we concatenate the ItemStem_Text field with all the Answer_N fields. With this format, we wanted to study how including all the answer candidates can help predict the difficulty of the item. We tried to include the separator token before each candidate, but we had several problems. Therefore, we just concatenated the answer candidates with the text.
- **Text and correct answer**: we include the ItemStem_Text and Answer_Text fields and use the separator token to mark the separation between the two fields. With this format, we wanted to study the impact of including only the correct answer without having access to the other answer candidates.
- **Only text**: we only include the ItemStem_Text field. We wanted to study the effect of the clinical case text, without any information about the answer candidates, when predicting the difficulty and response time.

3 Submitted Runs

We submitted the same three configurations to each Track, where the only difference between tracks is the target label used for training the models. All the runs were trained using the whole training set provided by the organizers, with the hyperparameters selected in the development period. The only difference among the three runs was the input format. All the runs were trained using a V100 GPU in Google Colab, selecting the best model after ten training epochs. The details of the three runs are:

- Run 1: it uses the ”All text together” input format described in Section 2.2. Hence, this run uses the clinical text and the candidates to make predictions. We use a batch size of 8 and a learning rate of 2e-5.
- Run 2: it uses the ”Text and correct answer” input format described in Section 2.2. This run gives us information about including the correct answer without including the other candidates. We use a batch size of 8 and a learning rate of 2e-5.
- Run 3: it uses the ”Only text” input format described in Section 2.2. The objective of this run was to make the predictions without including any information about the candidates

[3] https://huggingface.co/google-bert/bert-base-uncased

[4] https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html

or the correct answer. We use a batch size of 4, a learning rate of 1e-6.

Each run's batch size and learning rate were selected based on our experiments in the development period. We use the Adam optimizer and the mean-squared error as the loss function for all our experiments, while the other hyperparameters were the default provided by the transformers[5] library.

4 Analysis of Results

The official measure for both tracks was the Root Mean Squared Error metric (RMSE), which compares the prediction with the correct value. Systems are ranked according to RMSE, with the best systems obtaining the lowest error scores. We show and discuss the results of each track in the next subsections.

4.1 Track 1: Item Difficulty Prediction

In Table 1, we show the results of our three runs, the best system, and the proposed baseline in Track 1. Our best submission in this track was Run 3, which only included the clinical text as input. Thus, it seems that, at least with our approach and this data, any answer candidate's inclusion was harmful. We think this information must be helpful and want to perform a more profound study about correctly including it. Regarding the other two runs, the best one was Run 1, which included all the answer candidates.

Concerning other participants, our Run 3 was quite close to the winner system, with several systems ranking better. Besides, only Run 3 obtained better results than the proposed baseline.

Table 1: Results in Track 1, including the best system and the proposed baseline.

System	RMSE	Rank
Best system	0.299	1
Run 3	0.308	13
Baseline	0.311	16
Run 1	0.337	35
Run 2	0.363	40

4.2 Track 2: Response Time Prediction

In Table 2, we show the results of our three runs and the proposed baseline in Track 2. Our results in this task were quite good, obtaining the best result, the third and the fourth, despite using the same approaches in Track 1.

The results of the three runs were quite similar, so we must be careful with the conclusions we draw from them. According to the scores obtained, the best submission was Run 2, which included only the clinical text and the correct answer. In contrast, the submission, including the clinical text and all the answer candidates (Run 1), ranked third. Therefore, in this track, it was pretty useful to include information from the answers (in contrast to the results obtained in Track 1).

Table 2: Results in Track 1, including the best system and the proposed baseline.

System	RMSE	Rank
Run 2	23.927	1
Run 1	24.777	3
Run 3	25.365	4
Baseline	31.68	25

5 Conclusions and Future Work

Automatic prediction of exam difficulty remains an open challenge for both humans and machines. This is why the BEA 2024 Shared Task proposed evaluating systems predicting difficulty and response time in medical exams, opening a common framework for researching this challenge.

We have tested the use of BERT-based models with different input formats. Our objective was to establish a set of first results with simple systems and continue our research with the most complex approaches in the future.

We have tested the impact of using 1) only the text containing the clinical text and the question, 2) including the correct answer, and 3) including all the candidates. Our results differ depending on the track (predicting difficulty or response time). While we obtained our best results for predicting difficulty using only the clinical text, our best results for predicting response time were obtained including the correct answer.

Comparing results with other participants, we ranked at the middle of the ranking when predicting difficulty. On the other hand, we obtained the best results among all the participants when predicting response time, with our three runs in the first four positions of the final ranking.

Future work aims to study new configurations

[5] https://huggingface.co/docs/transformers/index

for both predictions and include more systems in the study.

Acknowledgements

This work has been partially funded by the Spanish Research Agency (Agencia Estatal de Investigación), DeepInfo project PID2021-127777OB-C22 (MCIU/AEI/FEDER,UE)

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Victoria Yaneva, Kai North, Peter Baldwin, An Ha Le, Saed Rezayi, Yiyun Zhou, Sagnik Ray Choudhury, Polina Harik, and Brian Clauser. 2024. Findings from the First Shared Task on Automated Prediction of Difficulty and Response Time for Multiple Choice Questions. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, Mexico City, Mexico. Association for Computational Linguistics.