DLSU at BEA 2025 Shared Task: Towards Establishing Baseline Models for Pedagogical Response Evaluation Tasks

Mark Edward M. Gonzales*, **Lanz Kendall Lim***, **Maria Monica Manlises***
College of Computer Studies, De La Salle University
Manila, Philippines
{mark_gonzales, lanz_kendall_lim, maria_monica_manlises}@dlsu.edu.ph

Abstract

We present our submission for Tracks 3 (Providing Guidance), 4 (Actionability), and 5 (Tutor Identification) of the BEA 2025 Shared Task on Pedagogical Ability Assessment of AI-Powered Tutors. Our approach sought to investigate the performance of directly using sentence embeddings of tutor responses as input to downstream classifiers (that is, without employing any fine-tuning). To this end, we benchmarked two general-purpose sentence embedding models: gte-modernbert-base (GTE) and all-MiniLM-L12-v2, in combination with two downstream classifiers: XGBoost and multilayer perceptron. Feeding GTE embeddings to a multilayer perceptron achieved macro-F1 scores of 0.4776, 0.5294, and 0.6420 on the official test sets for Tracks 3, 4, and 5, respectively. While overall performance was modest, these results offer insights into the challenges of pedagogical response evaluation and establish a baseline for future improvements.

1 Introduction

Recent advancements in large language models (LLMs) have opened new possibilities for using AI-powered chatbots as educational tutors, providing benefits for tasks such as homework assistance, personalized learning, and skills development (Labadze et al., 2023). However, while these systems can generate human-like dialogue, assessing their pedagogical effectiveness remains a significant challenge. In the past, human evaluation has typically been used for evaluation, though reliable, this is costly and difficult to scale (Liu et al., 2023).

To address this gap, the BEA 2025 Shared Task on Pedagogical Ability Assessment of AI-powered Tutors (Kochmar et al., 2025) was organized to promote the development of automated evaluation systems for tutor responses in educational dialogues.

The shared task focused on assessing the quality of tutor responses aimed at helping students correct their mistakes in math-related dialogues. Participants were provided with dialogues that included conversation history, a student's incorrect utterance, and multiple possible tutor responses (Maurya et al., 2025). Each response was to be evaluated along four pedagogically motivated dimensions: mistake identification, mistake location, guidance provision, and actionability. These dimensions were annotated on a three-point scale: "Yes," "To some extent," or "No."

In addition to these four tracks, the shared task included a fifth track, Tutor Identification, wherein participants were asked to predict the origin of anonymous tutor responses, distinguishing between different LLMs and human tutors. This track explored whether distinct pedagogical or linguistic styles could be used to attribute responses to their source.

The organizers released a development dataset of 300 annotated dialogues and a test set of 191 dialogues. Both sets included responses from a diverse set of state-of-the-art LLMs and, in some cases, human tutors (Maurya et al., 2025).

Our contributions are as follows:

- We evaluated the performance of directly feeding sentence embeddings of tutor responses (without any fine-tuning) to downstream classifiers for Tracks 3 (Providing Guidance), 4 (Actionability), and 5 (Tutor Identification).
- We benchmarked two sentence embedding models: gte-modernbert-base (GTE) and all-MiniLM-L12-v2. Our results show that using GTE embeddings and a multilayer perceptron yielded macro-F1 scores of 0.4776, 0.5294, and 0.6420, thus providing a baseline for the performance of general-purpose sentence embeddings on multiple pedagogical response evaluation tasks.

*Contributed equally

Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications, pages 1260–1265

2 Methods

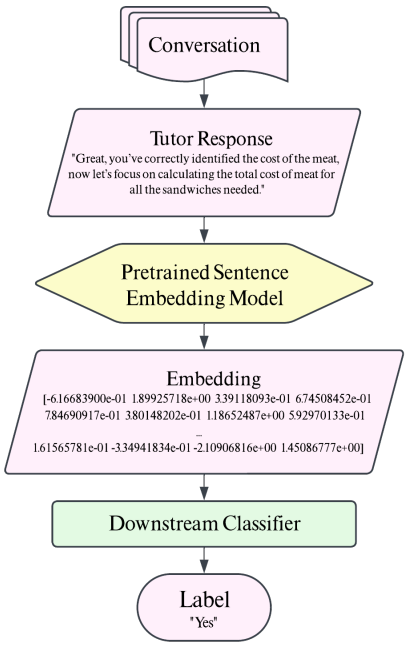

Figure 1: Methodology

Figure 1 shows our approach. First, we extracted the tutor response from each dialogue instance and fed it into a pretrained sentence embedding model to obtain a fixed-length vector representation. We used this representation as input to a classifier trained to predict the relevant task labels.

This methodology was applied across all three tracks in which we participated. We modeled Tracks 3 and 4 as multiclass classification problems where the output labels are "No," "To some extent," and "Yes." Likewise, Track 5 was also modeled as a multiclass classification problem with nine output labels: "Expert," "GPT4," "Gemini," "Llama31405B," "Llama318B," "Mistral," "Novice," "Phi3," and "Sonnet."

2.1 Sentence Embedding Model

Two embedding models were chosen from the Massive Text Embedding Benchmark (MTEB) Leaderboard[1] (Enevoldsen et al., 2025), which compares the performance of over a hundred embedding models across multiple tasks.

We first selected **gte-modernbert-base**[2] (Zhang et al., 2024) or GTE, a general-purpose embedding model built on modernBERT (Warner et al., 2024). With 149 million parameters and a context length of up to 8192 tokens, it performs strongly on the MTEB leaderboard, competitive with other models with under 1 billion parameters.

In addition, we also evaluated a more lightweight model, **all-MiniLM-L12-v2**[3], which has 33.4M parameters. Despite its compact size, it registers competitive performance on the MTEB leaderboard and on other classification tasks (Meleti et al., 2025).

2.2 Downstream Classifier

We trained two classification models: **XGBoost** and a **multilayer perceptron (MLP)** with a single hidden layer. XGBoost, a decision tree-based gradient boosting method, has been reported to achieve good performance with dense sentence embeddings as input (Muqadas et al., 2025; Chen and Guestrin, 2016). MLPs are capable of capturing nonlinear relationships and, as such, are widely used in supervised learning tasks (Goodfellow et al., 2016).

We partitioned the development set such that 80% of the data comprises the training set and the remaining 20% comprises the test set. We then performed three-fold cross-validation with grid search on the training set to tune the hyperparameters of both models; the complete hyperparameter search space is reported in Table 3. Tables 4 and 5 show the combination of hyperparameters that returned the highest macro-F1.

3 Results and Discussion

3.1 Development Set Results

Table 1 summarizes the results on the test set partition of our development set. We found that using MLP consistently outperformed using XGBoost in terms of macro-F1 across all three tasks, with the strongest gains observed in Tracks 4 (Actionability) and 5 (Tutor Identification). Pairing GTE embeddings with MLP achieved the highest macro-F1 and also the highest accuracy (except for Task 3).Confusion matrices are given in Figure 2.

3.2 Official Test Set Results

Based on the development set results, we selected the top two model combinations for final testing. For Tracks 3 and 4, we chose GTE + MLP and GTE + XGBoost. For Track 5, we selected GTE + MLP and MiniLM + MLP. The complete official test set

[1] https://huggingface.co/spaces/mteb/leaderboard

[2] https://huggingface.co/Alibaba-NLP/gte-modernbert-base

[3] https://huggingface.co/sentence-transformers/all-MiniLM-L12-v2

Task	Model	Macro-F1	Accuracy
Track 3	GTE + MLP	**0.5601**	0.6371
	GTE + XGBoost	0.5095	**0.6492**
	MiniLM + MLP	0.4675	0.6371
	MiniLM + XGBoost	0.4814	0.6310
Track 4	GTE + MLP	**0.5667**	**0.6492**
	GTE + XGBoost	0.5097	0.6552
	MiniLM + MLP	0.5504	0.6411
	MiniLM + XGBoost	0.4766	0.6431
Track 5	GTE + MLP	**0.6047**	**0.5665**
	GTE + XGBoost	0.4879	0.4476
	MiniLM + MLP	0.5333	0.4879
	MiniLM + XGBoost	0.4595	0.3992

Table 1: Macro-F1 and accuracy on the development set across Tracks 3 (Providing Guidance), 4 (Actionability), and 5 (Tutor Identification). The best performance scores are in bold.

scores for these selected model combinations are reported in Table 2.

3.3 Limitations

First, we fed the tutor responses, as is, to the sentence embedding models, that is, we did not perform any text preprocessing (such as stopword removal or punctuation stripping) prior to embedding. While this decision aligns with the intention to evaluate the raw utility of general-purpose embeddings, preprocessing might have potentially reduced noise and improved classification performance.

Second, we did not fine-tune the sentence embedding models on task-specific data. The GTE and MiniLM embeddings were used as is, without adaptation to the tutoring domain or label space. This might have limited the models' ability to capture nuanced patterns in the instructional dialogue, particularly for more subtle distinctions such as "To some extent" in Tracks 3 and 4 or between tutor personas in Track 5.

Finally, the per-class evaluation results (Figure 3) reflect the class imbalance, with the dominant class ("Yes") having noticeably higher F1 compared to "No" and "To some extent" for Tracks 3 and 4. To address this, it may be helpful to incorporate class-adjusted weights during training, perform data augmentation, or generate synthetic data.

4 Conclusion

In this paper, we investigated the performance of directly feeding sentence embeddings of tutor responses to downstream classifiers for multiple pedagogical response evaluation tasks, thus providing baseline models for future improvements in this domain.

For future work, it may be interesting to compare these baselines with domain-specific fine-tuning, as well as perform more extensive hyperparameter tuning through automated optimization techniques (such as Bayesian optimization) to further improve classification accuracy.

References

Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA. Association for Computing Machinery.

Kenneth Enevoldsen, Isaac Chung, Imene Kerboua, Márton Kardos, Ashwin Mathur, David Stap, Jay Gala, Wissam Siblini, Dominik Krzemiński, Genta Indra Winata, Saba Sturua, Saiteja Utpala, Mathieu Ciancone, Marion Schaeffer, Gabriel Sequeira, Diganta Misra, Shreeya Dhakal, Jonathan Rystrøm, Roman Solomatin, and 67 others. 2025. MMTEB: Massive multilingual text embedding benchmark. *arXiv preprint arXiv:2502.13595*.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Ekaterina Kochmar, Kaushal Kumar Maurya, Kseniia Petukhova, KV Aditya Srivatsa, Anaïs Tack, and Justin Vasselli. 2025. Findings of the BEA 2025 shared task on pedagogical ability assessment of AI-powered tutors. In *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*.

Lasha Labadze, Maya Grigolia, and Lela Machaidze. 2023. Role of AI chatbots in education: systematic literature review. *International Journal of Educational Technology in Higher Education*, 20:56.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Kaushal Kumar Maurya, Kv Aditya Srivatsa, Kseniia Petukhova, and Ekaterina Kochmar. 2025. Unifying AI tutor evaluation: An evaluation taxonomy for pedagogical ability assessment of LLM-powered AI tutors. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1234–1251, Albuquerque, New Mexico. Association for Computational Linguistics.

Task	Model	Exact F1	Exact Acc	Lenient F1	Lenient Acc
Track 3 (Providing Guidance)	GTE + MLP	**0.4776**	0.5669	0.6755	0.7382
	GTE + XGBoost	0.4545	**0.6244**	**0.6784**	**0.7712**
Track 4 (Actionability)	GTE + MLP	**0.5294**	0.6089	**0.7351**	0.7738
	GTE + XGBoost	0.4966	**0.6102**	0.7170	**0.7789**
Track 5 (Tutor Identification)	GTE + MLP	**0.6420**	**0.6231**	–	–
	MiniLM + MLP	0.5808	0.5624	–	–

Table 2: Performance on the official test sets. "F1" is shorthand for macro-F1, and "Acc" stands for accuracy. For Tracks 3 and 4, two additional metrics were additionally computed by the testing platform: lenient F1 and lenient accuracy, which consider "Yes" and "To some extent" the same class. The qualifier "exact" distinguishes the conventional metrics from their lenient variation.

Marco Meleti, Stefano Guizzardi, Elena Calciolari, and Carlo Galli. 2025. A comparative analysis of sentence transformer models for automated journal recommendation using pubmed metadata. *Big Data and Cognitive Computing*, 9(3):67.

Amara Muqadas, Hikmat Ullah Khan, Muhammad Ramzan, Anam Naz, Tariq Alsahfi, and Ali Daud. 2025. Deep learning and sentence embeddings for detection of clickbait news from online content. *Scientific Reports*, 15(1):13251.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, and 1 others. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *arXiv preprint arXiv:2412.13663*.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, and 1 others. 2024. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1393–1412.

Hyperparameter	Search Space
XGBoost	
Number of estimators	50, 100, 150
Maximum depth of a tree	3, 5, 7
Learning rate	0.01, 0.1, 0.2
Subsample ratio of the training instances	0.8, 1.0
Subsample ratio of columns when constructing each tree	0.8, 1.0
MLP	
Hidden layer size	(50,), (100,), (150,)
Activation	ReLU, tanh, logistic
Solver	Adam, SGD
L2 regularization strength	10^{-4}, 10^{-3}, 10^{-2}
Learning rate schedule	Constant, adaptive

Table 3: Hyperparameter search space

Task	Embedding	n_estimators	max_depth	learning_rate	subsample	colsample_bytree
Track 3	GTE	100	7	0.2	1.0	0.8
	MiniLM	50	5	0.2	1.0	1.0
Track 4	GTE	150	3	0.2	0.8	0.8
	MiniLM	150	7	0.1	0.8	0.8
Track 5	GTE	150	3	0.2	0.8	0.8
	MiniLM	150	5	0.1	0.8	1.0

Table 4: Optimal XGBoost hyperparameters selected via three-fold cross-validation with grid search for each task and sentence embedding model. `n_estimators` refers to the number of estimators; `max_depth`, maximum depth of a tree; `learning_rate`, learning rate; `subsample`, subsample ratio of the training instances; and `colsample_bytree`, subsample ratio of columns when constructing a tree.

Task	Embedding	Activation	L2 Reg.	Hidden Layer Size	Learning Rate Schedule	Solver
Track 3	GTE	ReLU	10^{-2}	(150,)	Constant	Adam
	MiniLM	tanh	10^{-4}	(150,)	Constant	SGD
Track 4	GTE	ReLU	10^{-4}	(50,)	Constant	Adam
	MiniLM	ReLU	10^{-4}	(150,)	Constant	Adam
Track 5	GTE	Logistic	10^{-3}	(50,)	Constant	Adam
	MiniLM	Logistic	10^{-2}	(50,)	Constant	Adam

Table 5: Optimal MLP hyperparameters selected via three-fold cross-validation with grid search for each task and sentence embedding model

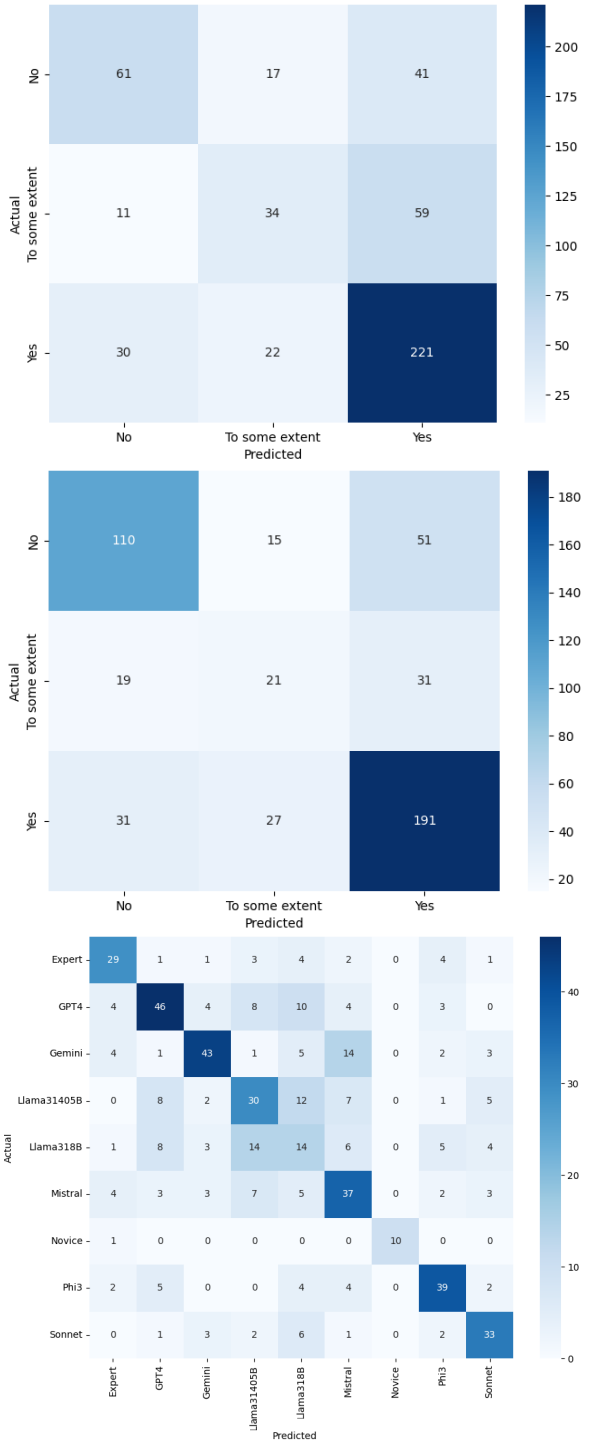

Figure 2: Confusion matrices for (a) Track 3, (b) Track 4, and (c) Track 5, obtained by pairing gte-modernbert-base and multilayer perceptron (GTE + MLP)

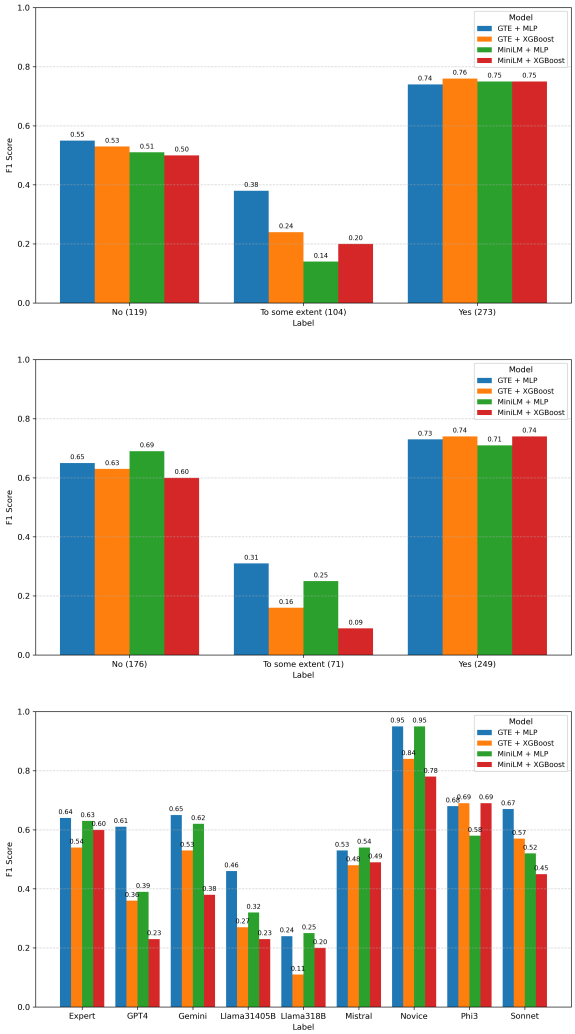

Figure 3: Per-class F1 scores for (a) Track 3, (b) Track 4, and (c) Track 5