Generative-AI Powered TA BOT for Enhanced Personalized Support in Software Engineering Education

Sudam Rohanadeera and K Priyantha Hewagamage

November 25, 2024

Generative-AI-powered TA BOT for Enhanced Personalized Support in Software Engineering Education

Sudam Lankesha Rohanadeera, K P Hewagamage

University of Colombo School of Computing,
35, Reid Avenue, Colombo, Sri Lanka
slr@ucsc.cmb.ac.lk, kph@ucsc.cmb.ac.lk

Abstract. The rising demand for higher education has strained resources, particularly in providing instructional support through Teaching Assistants (TAs). Traditional TAs are vital for personalized learning, but their scarcity in high-enrollment courses hampers the 'one-to-one' interaction essential for student success. This research develops and evaluates a Generative AI-based Teaching Assistant BOT (TA BOT) tailored to specific course content, leveraging large language models (LLMs) to enhance personalized interactions within Learning Management Systems (LMSs). By fine-tuning OpenAI's GPT-3.5-Turbo model using domain-specific data through a simplified Retrieval-Augmented Generation approach, we created a TA BOT for a software engineering course. The TA BOT was assessed by comparing it with generic ChatGPT and through student feedback. Results show that the TA BOT delivers more focused and contextually relevant responses, aligning closely with course objectives and enhancing students' understanding of core concepts. User feedback revealed high satisfaction, with 93.1% recognizing the TA BOT as a valuable educational tool. This study demonstrates the effectiveness of fine-tuning LLMs to create domain-specific educational chatbots that improve personalized learning experiences.

Keywords: TA BOT, Software Engineering, Generative AI, Learning Management Systems, Personalized Learning

1 Introduction

The rising demand for higher education has significantly challenged institutions to maintain quality, especially in large-scale programs [10]. As university enrollments increase globally, institutions are under pressure to accommodate more students without compromising educational standards [1]. This has particularly strained resources in providing instructional support, where Teaching Assistants (TAs) traditionally play a key role.

TAs are vital for enhancing learning experiences through personalized support, feedback, and bridging communication between students and faculty [5]. However, the scarcity of TAs in high-enrollment courses limits their ability to provide individualized assistance, negatively affecting student engagement and learning outcomes [10].

Learning Management Systems (LMS) have tried to address this issue by offering platforms that facilitate interactions through various modes, including one-to-one support, which is essential for personalized learning [10].

To mitigate the shortage of TAs and enhance personalized interactions, there is increasing interest in leveraging Generative Artificial Intelligence (AI). Advances in Generative-AI, such as Large Language Models (LLM) like OpenAI's GPT-3, offer opportunities to provide consistent, tailored support at scale [3]. However, existing Generative-AI lacks the course-specific context necessary for effective guidance [11, 14]. This research develops and evaluates a Generative AI-based Teaching Assistant BOT tailored to specific course content to enhance student support, using a software engineering course as a case study. By fine-tuning AI models to align with specific curricula, this study aims to bridge the gap between scalability and the need for individualized support.

2 Related Work and Literature Review

The evolution of digital teaching assistants (TA bots) has progressed significantly, from early conversational AI systems like ELIZA to sophisticated generative AI models capable of providing personalized educational support. Early agents like ELIZA, developed by Weizenbaum, used pattern matching to simulate dialogue, demonstrating the initial potential of computers in education [12]. These systems, however, lacked true comprehension and adaptability, limiting their usefulness for complex educational interactions.

Intelligent Tutoring Systems (ITS) represented a major advancement by providing personalized feedback and adapting to individual learner needs [6]. Despite this progress, ITS struggled with flexibility and handling unexpected queries, necessitating more dynamic solutions for delivering course-specific content. A significant milestone was the development of "Jill Watson," a virtual teaching assistant designed to reduce the workload on human TAs at Georgia Tech [5]. While effective, Jill Watson's reliance on predefined datasets restricted its ability to address nuanced or novel questions. Similarly, Edubot provided support for Ordinary Level Chemistry in Sri Lanka, but its limitations in accuracy and adaptability hindered its overall impact compared to modern generative models [9]

The advent of large-scale generative AI models, such as GPT-3 and GPT-4, has transformed TA bots by enabling them to generate coherent, context-aware responses [3]. These models leverage deep learning and extensive training data, allowing for educational chatbots that align closely with the curriculum. Studies like those by Essel et al. (2022) [4] demonstrated the positive impact of virtual teaching assistants on student engagement and learning outcomes. Integrating generative AI-powered bots within LMSs further enhances the 'one-to-one' interaction model, providing personalized support to students, instant feedback, and explanations tailored to individual understanding

[8]. However, challenges such as ethical considerations, ensuring critical thinking, and addressing academic integrity issues remain key concerns [11, 12].

Despite significant advancements, notable gaps remain in the comprehensive evaluation of TA bots, particularly regarding user feedback. Edubot, for example, lacked user-centered evaluation, which hindered a complete understanding of its effectiveness [9]. Addressing these gaps is crucial for refining chatbot functionalities to ensure they meet educational needs effectively. Future research should focus on integrating user evaluations and fine-tuning AI models to specific course content, thereby fully realizing the potential of generative AI-powered TAs in enhancing personalized learning within LMSs.

3 Methodology

3.1 Fine-Tuning Large Language Models (LLMs)

Large Language Models (LLMs) like OpenAI's GPT-3.5 and GPT-4 are deep neural networks capable of generating human-like text by learning complex language patterns from large datasets [2, 3]. Fine-tuning LLMs for educational applications involves adapting pre-trained models to specific domains by retraining them on domain-specific datasets, enhancing their ability to provide specialized support [4, 7]. This involves optimization processes such as AdamW and gradient-based adjustments to refine model weights through backpropagation [2]. Regularization techniques, like weight decay and dropout, are used to prevent overfitting, ensuring the model's ability to generalize effectively [7].

Different fine-tuning strategies are applied based on resource availability and specificity requirements. Full model fine-tuning updates all parameters, which is resource-intensive, while layer-wise fine-tuning adjusts only selected layers to save computational resources [7]. Parameter-efficient techniques, such as Low-Rank Adaptation (LoRA), selectively introduce trainable parameters, balancing efficiency and performance [2]. In this research, OpenAI's models were chosen due to their high accuracy, ease of deployment via cloud infrastructure, and continuous updates, making them ideal for educational adaptation [3].

However, fine-tuning OpenAI models presents challenges due to proprietary constraints. The optimization process is conducted through a "black box" API, which affects transparency and reproducibility [14].

3.2 Data Pre-processing

Data pre-processing was a crucial step that transformed raw content into a clean dataset ready for model training [2]. The core educational material for this project comprised the first nine chapters of "Software Engineering: Seventh Edition" by Ian Sommerville

[13]. Irrelevant content such as "Further Reading" sections and exercises were manually removed using an online PDF splitting tool, ensuring the final dataset was concise and aligned with the course syllabus.

After pre-processing, the dataset for fine-tuning was generated using a simplified Retrieval-Augmented Generation (RAG) approach [8]. This involved dividing the content into chunks and generating simulated student queries to create prompt-completion pairs. LlamaIndex, an open-source tool [15], was used to manage data ingestion and indexing, which improved the model's contextual understanding and its ability to provide educational responses without embedding-based retrieval [14]. Details on the modified RAG approach can be found *https://tinyurl.com/mten8xkv*.

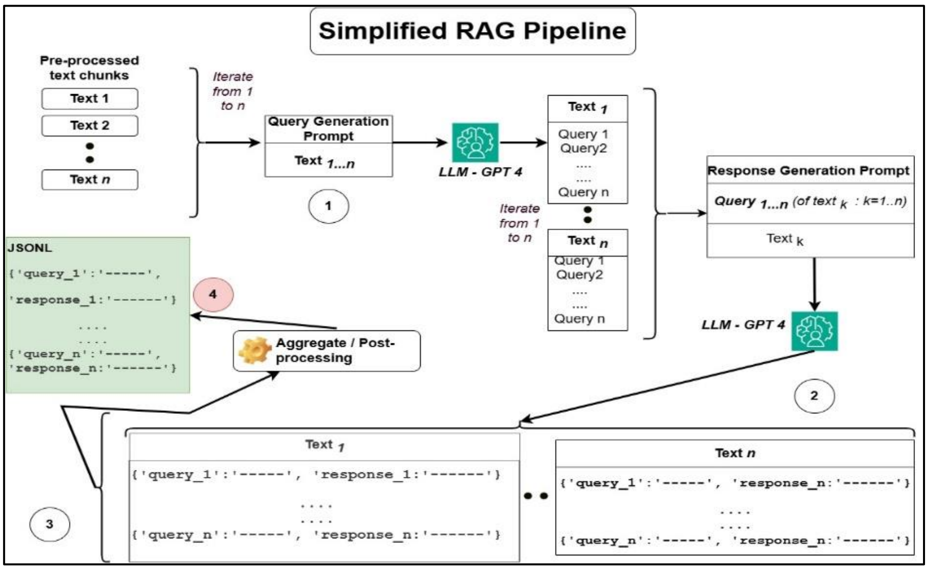

Figure 1: Simplified RAG Pipeline

3.3 Fine-Tuning the GPT-3.5-Turbo LLM

The GPT-3.5-Turbo LLM was fine-tuned using the curated dataset of query-response pairs, allowing the general-purpose model to be adapted specifically for software engineering education. Fine-tuning helps the model internalize particular content patterns, thereby making it more effective at handling tasks such as understanding course-specific terminology and responding to student inquiries with context-specific answers [7]. Ultimately, this ensures the TA BOT can deliver high-quality, context-aware responses that mimic the capabilities of a human teaching assistant, enhancing personalized learning within LMS environments.

3.4 Experimental System Architecture and Design

The TA BOT architecture comprises three layers: frontend, backend API server, and data storage. The frontend, developed with React, manages user interactions, while the backend uses Python Flask for integration with Redis Cache and MongoDB. Redis manages real-time conversations, and MongoDB provides persistent storage for historical interactions. The fine-tuned LLM generates relevant responses, and Docker containerization ensures scalability. Data flows via APIs between these components, enabling efficient, reliable interactions.

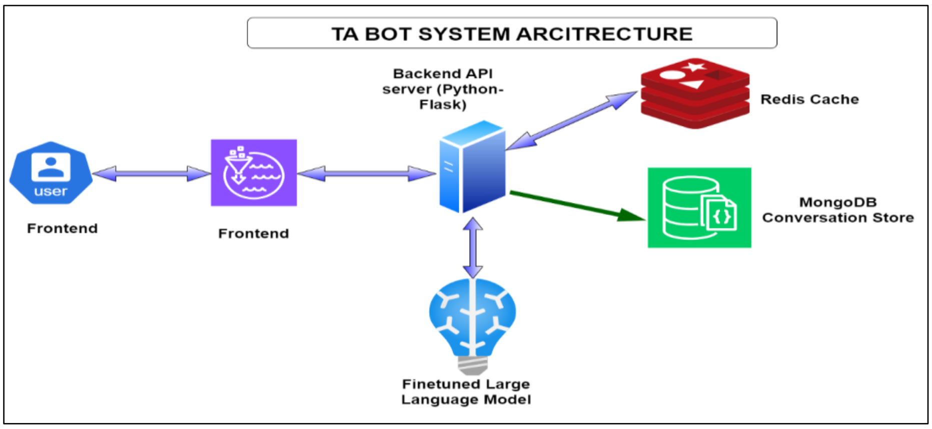

Figure 2: TA BOT System Architecture

3.5 Data Collection and Evaluation Approach

The TA BOT, which is available for public access at ***http://chat.ucsc.cmb.ac.lk***, was shared among students who followed software engineering courses at the University of Colombo School of Computing. Later, a feedback survey was carried out to assess usability, functionality, and system impact on student learning. Questions were dynamically adapted based on user interactions, ensuring relevance. Distribution methods included LMS course pages, WhatsApp groups, and direct integration in the TA BOT interface (https://chat.ucsc.cmb.ac.lk/). A mixed-methods evaluation, combining quantitative metrics (close-ended questions) and qualitative insights (open-ended responses), helped assess technical performance, user satisfaction, and educational impact.

The qualitative analysis focused on themes such as user interface, system responsiveness, and potential improvements. Suggestions ranged from UI design tweaks to expanding TA BOT functionalities to support images and diagrams. By combining both types of feedback, a comprehensive evaluation highlighted areas for further system development to enhance student learning support.

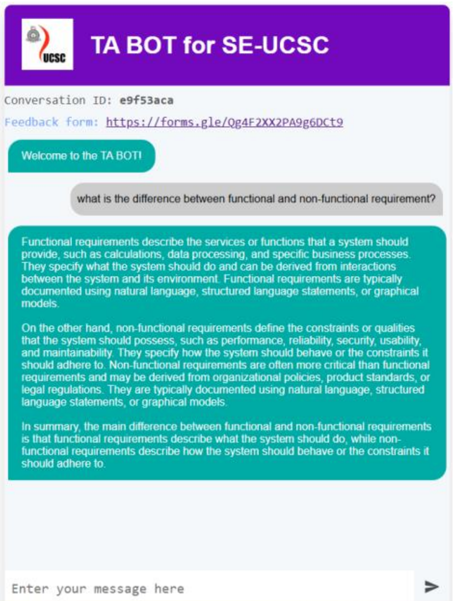

Figure 3: TA BOT Implementation – **http://chat.ucsc.cmb.ac.lk**

4 Data Analysis, Results and Discussion

4.1 Comparison Between Generic ChatGPT and TA BOT

The comparison between generic ChatGPT and the TA BOT reveals that the TA BOT offers more focused, contextually relevant responses specifically tailored to the UCSC software engineering courses. Unlike ChatGPT, which gives broad, generalized answers that lack alignment with course-specific learning goals, the TA BOT provides concise, syllabus-aligned definitions, improving student comprehension. This effectiveness stems from the TA BOT's fine-tuning using Retrieval-Augmented Generation (RAG) techniques, which integrate domain-specific data, making it superior for targeted educational support [7, 10].

4.2 Data Analysis

The evaluation involved 34 valid responses, where 91.1% of respondents could successfully start the TA BOT. Among them, 93.3% did not face technical issues, and 62.1% found the responses meaningful. However, two users reported issues like an ellipsis without meaningful content. Additionally, 93.1% agreed that the TA BOT is a valuable educational tool, with several respondents providing suggestions for improvement, such as enhancing response accuracy and expanding the system’s features. For those unable to access the TA BOT, browser compatibility and connection type were possible factors affecting accessibility. Feedback covered themes like UI design, responsiveness, feature enhancements, and educational impact, with suggestions focusing on improving usability and expanding interaction capabilities.

4.3 Discussion

Different large language models (LLMs) used in TA BOTs exhibit varying behaviors, leading to differences in the quality and relevance of responses. This variability arises from unique training datasets, model architectures, and fine-tuning methodologies, highlighting the importance of careful model selection and ongoing evaluation [3, 14]. The need for standardization and optimization is essential to ensure effective educational support, thereby enabling TA BOTs to align with specific course objectives and enhance the overall learning experience [6].

5 Conclusion and Future Work

5.1 Conclusion

The study developed and evaluated a Generative AI-based Teaching Assistant BOT specifically for the UCSC software engineering course, demonstrating superior performance compared to generic models like ChatGPT in providing focused, contextually relevant responses. Fine-tuned using a simplified Retrieval-Augmented Generation technique and domain-specific data, the TA BOT effectively enhanced students' understanding of core concepts while minimizing confusion. User feedback was positive, with 93.1% acknowledging the TA BOT's value, though areas for improvement in response accuracy and features were noted. Future work will include enhancing the user interface, integrating speech capabilities, improving backend efficiency through multithreading, and expanding applicability to other subjects and platforms such as Moodle LMS forums, aiming to offer more versatile and personalized educational support.

Acknowledgments. This study was funded by the AI in Education research group (*http://bit.ly/ai4e*) of the University of Colombo School of Computing.

6 References

1. Anderson, T. ed: The theory and practice of online learning. AU Press, Edmonton (2008).

2. Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. J. Mach. Learn. Res. 13, null, 281–305 (2012).

3. Brown, T.B. et al.: Language models are few-shot learners. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. pp. 1877–1901 Curran Associates Inc., Red Hook, NY, USA (2020).

4. Essel, H.B. et al.: The impact of a virtual teaching assistant (chatbot) on students' learning in Ghanaian higher education. Int J Educ Technol High Educ. 19, 1, 57 (2022). https://doi.org/10.1186/s41239-022-00362-6.

5. Goel, A.K., Polepeddi, L.: Jill Watson: A Virtual Teaching Assistant for Online Education. (2016).

6. Holmes, W. et al.: Artificial Intelligence In Education.

7. Howard, J., Ruder, S.: Universal Language Model Fine-tuning for Text Classification. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 328–339 Association for Computational Linguistics, Melbourne, Australia (2018). https://doi.org/10.18653/v1/P18-1031.

8. Lewis, P. et al.: Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. pp. 9459–9474 Curran Associates Inc., Red Hook, NY, USA (2020).

9. Mahroof, A. et al.: An AI based Chatbot to Self-Learn and Self-Assess Performance in Ordinary Level Chemistry. In: 2020 2nd International Conference on Advancements in Computing (ICAC). pp. 216–221 (2020). https://doi.org/10.1109/ICAC51239.2020.9357131.

10. Moore, M.G., Anderson, W.G. eds: Handbook of distance education. L. Erlbaum Associates, Mahwah, N.J (2003).

11. Petrovska, O. et al.: Incorporating Generative AI into Software Development Education. In: Proceedings of the 8th Conference on Computing Education Practice. pp. 37–40 ACM, Durham United Kingdom (2024). https://doi.org/10.1145/3633053.3633057.

12. Preiksaitis, C., Rose, C.: Opportunities, Challenges, and Future Directions of Generative Artificial Intelligence in Medical Education: Scoping Review. JMIR Med Educ. 9, e48785 (2023). https://doi.org/10.2196/48785.

13. Sommerville, I.: Software engineering. Pearson, Boston Columbus Indianapolis New York San Francisco Hoboken Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montreal Toronto Delhi Mexico City São Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo (2016).

14. Ziegler, D.M. et al.: Fine-Tuning Language Models from Human Preferences, http://arxiv.org/abs/1909.08593, (2020). https://doi.org/10.48550/arXiv.1909.08593.

15. LlamaIndex, Data Framework for LLM Applications, https://www.llamaindex.ai/, last accessed 2024/09/19.