Journal of
Information Systems Engineering
and Business Intelligence

Vol.11, No.3, October 2025
Available online at: http://e-journal.unair.ac.id/index.php/JISEBI

Academic Guidebook Chatbot: Performance Comparison of Fine-Tuned Mistral 7B and LlaMA-2 7B

Davied Indra Rachman[1)*], Agus Subhan Akbar [2)], Alzena Dona Sabilla[3)]

[1)2)3)]*Department of Information System, Faculty of Science and Technology, Universitas Islam Nahdlatul Ulama Jepara, Jawa Tengah, Indonesia*

[1)]211250000399@unisnu.ac.id, [2)]agussa@unisnu.ac.id, [3)]alzena.dona@unisnu.ac.id

Abstract

Background: Chatbot is recently ranked as the main technological solution due to the high demand for fast and efficient information retrieval. Therefore, this study was carried out to develop a local document-based chatbot that can answer questions related to the contents of PDF documents using open-source AI models such as Mistral 7B and LLaMA-2 7B. Although these models were effective at processing natural language, a major challenge was observed in the tendency to generate hallucinated answers, characterized by having inaccuracies and being out of context.
Objective: This study aims to reduce hallucinatory responses from chatbot models by making their responses more precise and accurate through fine-tuning. The performance of fine-tuned models (Mistral 7B and LLaMA-2 7B) was also compared.
Methods: Fine-tuning of the two models was performed using domain-specific datasets taken from Academic Guidebook. This process was conducted to improve models ability to understand and answer questions relevant to Academic Guidebook context. Performance was evaluated using METEOR Score to measure literal agreement and BERTScore to assess meaning agreement. In addition, response time was measured to assess efficiency, while chatbot system was developed using Streamlit and LangChain for real-time interaction.
Results: Fine-tuned Mistral 7B model achieved the highest METEOR value of 0.40 and F1 of 0.78 based on BERTScore results. Regarding efficiency, fine-tuned Mistral 7B showed a faster response time than LLaMA-2. Meanwhile, the non-fine-tuned Mistral 7B and LLaMA-2 7B showed a longer response time than fine-tuned Mistral 7B and LLaMA-2 7B.
Conclusion: The results showed that the enhancements significantly improved the performance of large language models in specific tasks, reduced hallucinations, and enhanced response quality

Keywords: Chatbot, Large Language Model, Mistral 7B, LLaMA-2 7B, METEOR Score

Article history: Received 29 November 2024, first decision 18 April 2025, accepted 14 August 2025, available online 28 October 2025

I. INTRODUCTION

Academic Guidebook is a primary reference comprising study guidelines, prevailing regulations, academic policies, and available students' services. Furthermore, this document plays a crucial role in decision-making processes, such as course selection, study leave, and the comprehension of rights and responsibilities. In practice, students frequently encounter challenges when seeking essential information in Academic Guidebook. These occur because the conventional methods used, such as reading the entire document or asking administrative staff directly, are proven to be inefficient and time-consuming. The length of document and the amount of information are the main causes of the difficulty [1][2]. Therefore, students urgently need a solution that is faster and more efficient than traditional methods.

Some previous studies have explored domain-specific methods. In the medical domain, [3] MEDITRON was introduced as a large language model with 70 billion parameters designed for medical analysis, while [4] MedAlpaca was developed through fine-tuning for comparable applications. Additionally, [5] MedDoc-bot was designed as an intelligent chatbot that enabled users to upload PDF documents of medical guidelines and retrieve answers from four local models, namely MEDITRON, MedAlpaca, Mistral, and LLaMA-2. These models are evaluated on their comprehension of pediatric hypertension guidelines from ESC [5]. Similar methods have been applied beyond the medical field, including in finance. Phogat et. al. proposed smaller model refinements to improve question answering over financial documents, suggesting that targeted refinements could improve performance in non-medical settings [6]. Other studies introduced CONVFINQA, a dataset designed to investigate numerical reasoning chains in financial conversations, focusing on the challenges of modeling complex reasoning in real-world scenarios [7]. Meanwhile,

* Corresponding author

ISSN 2443-2555 (online) 2598-6333 (print)
doi: http://dx.doi.org/10.20473/jisebi.11.3.383-392

EduChat [8], a large-scale chatbot system built on a language model for intelligent education, reflecting the growing interest in using AI-powered conversational agents for academic engagement and learning, such as question answering, essay grading, and emotional support, with promising results.

Although many studies have successfully used Large Language Models (LLMs) to answer questions in a variety of fields, no similar method has particularly focused on meeting students' information needs based on Academic Guidebook. There is important information in this document that students frequently struggle to quickly and effectively access. Therefore, the development of document-based chatbot that can understand and react to questions in line with the content of Academic Guidebook is made possible. The ability of chatbot to work with local documents has recently been developed, enabling users to obtain information straight from uploaded PDF files [9]. This facilitates information retrieval from lengthy and intricate documents.

According to the background above, this study presents a document-based chatbot implemented with two open-source AI, Mistral 7B and LLaMA-2 7B, delivering answers to user queries about PDF documents. The selection of model was based on strong performance, open source, and suitability for use in a resource-constrained environment. This chatbot aids students in gathering information from Academic Guidebook, avoiding the time-consuming process of manual searching. A major weakness of current artificial intelligence (AI) models is generation of large inaccurate information or hallucination data [10]. Therefore, fine-tuning is adopted as a solution to this problem in open-source AI models, as training with carefully selected data enhances accuracy and significantly reduces the likelihood of hallucinations. [11].

The majority of LLMs available are not specifically designed or adapted to any particular domain. This shows the need for fine-tuning with some domain-specific data to understand general language and answer questions in a particular context. This current study will perform this fine-tuning using Unsloth, which is a training optimization framework that has not been used by previous investigations [3]–[8]. However, the usage has recently started because of memory efficiency as well as speed of training [12]–[16]. The framework makes training more memory efficient, increases the speed of training, and works with 8-bit quantization, accelerating the process of fine-tuning [17].

The purpose of this study is to improve the relevance and semantic quality of answers from open-source AI models by fine-tuning. Specifically, Mistral 7B and LLaMA-2 7B from HuggingFace are selected because of a good mixture of performance and model size. These models use compressed versions in GGUF format, optimized for local inference, to keep size and performance in check. Streamlit V1.34 also allows users to interact with model in real-time, ask questions, and get answers [18]. To combine and organize large language models, LangChain is used to handle requests from local [19]. The quality of each generated response will be evaluated using two distinct metrics, namely METEOR Score and BERTScore. METEOR assesses precision, recall, and a word order penalty, while BERTScore evaluates the semantic similarity between sentences. The application of these two metrics offers a comprehensive assessment of the response quality produced by model following fine-tuning procedure.

II. METHODS

A. AI Models

Publicly available models such as Mistral 7B have drawn wide interest in natural language processing due to their strong performance in text comprehension, reasoning, and code generation [20]. This study used Mistral 7B Instruct v0.2, an improved version of v0.1, converted into GGUF format by LLaMA C++ community and distributed through HuggingFace [21]. Model was pre-quantized to 4-bit, reducing memory use and improving processing efficiency while maintaining accuracy, showing suitability for CPU- and GPU-limited environments [22][23].

LLaMA-2 is a large language model available in versions with 7 billion and 70 billion parameters [24]. This model facilitates various language-related tasks, including text comprehension, text generation, and question answering. With the range of parameter sizes, LLaMA-2 shows robust capabilities in managing intricate linguistic structures and delivering accurate and contextually appropriate responses across diverse applications, such as chatbot and document retrieval systems. The comparison of models used is shown in Table 1.

TABLE 1
LLMS THAT ARE USED FOR FINE-TUNING

Name	HuggingFace Repository	Parameters	Format	Precision
Mistral	unsloth/mistral-7b-instruct-v0.2-bnb-4bit	7B	GGUF	4-bit
LLaMA-2	unsloth/llama-2-7b-bnb-4bit	7B	GGUF	4-bit

This study will compare the performance of fine-tuned Mistral 7B against non-fine-tuned counterpart. To ensure a fair comparison, LLaMA-2 7B is subjected fine-tuning process before evaluation.

B. Dataset

This study applied a dataset from Academic Guidebook [25] to train and test language models using PDF conversational tool interface. The dataset consisted of 228 validated question-answer pairs, verified by relevant bureau officials, with 158 used for training and 70 for testing. These questions were categorized into three groups based on chapter content, namely (B1) Academic Regulations, which comprised queries regarding curriculum, study duration, credit load, academic evaluation, and graduation. (B2) Campus Conduct comprises questions about the rights, regulations, and prohibitions for students as well as faculty on campus. Meanwhile, (B3) Student Organizations contained questions related to organizational activities, membership, and the rights and responsibilities of students in organizations.

The categorization was performed manually, referencing the structure and content of the guidebook. Each question was identified based on topic and grouped into B1, B2, and B3. The test dataset was also divided into these three categories. The results were highly dependent on this classification, as model accuracy would be evaluated based on performance in answering questions in each group. Consequently, this division was crucial for analyzing model comprehension of the specific domains in the various information types contained in Academic Guidebook.

C. Hardware and Software

The user interface was built using Streamlit V1.34, which allowed for quick and easy creation of interactive interfaces. AI models were trained using Google Colab to simplify processing through high-performance servers [26]. In this chatbot, document processing was performed using Langchain, which was supported by LLMs [27]. METEOR Score was used to evaluate the quality of the responses generated by chatbot to ensure that the generated text was contextually appropriate and highly accurate [28]. BERTScore was also used to assess the semantic similarity between model responses and the reference answers. Compared to traditional metrics that rely on exact word matching, BERTScore was used to measure similarity between meanings, ensuring that chatbot answers were closely correlated with the intended context [29].

D. Chatbot Development Process

As shown in Fig. 1, stages in chatbot development start with the process, which represents the same sequence of steps applied to both Mistral 7B and LLaMA-2 7B models, ensuring a consistent and fair comparison of their performances. These stages include application system design, dataset preparation, model training, testing, and performance measurement evaluation is carried out using METEOR Score and BERTScore.

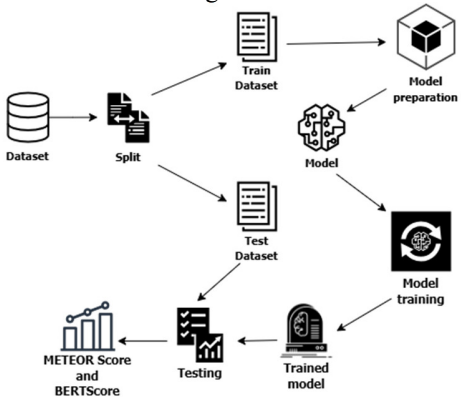

Fig. 1 The process flow for developing and evaluating chatbot

During the inference stage, AI model was run using CTransformers library with a temperature of 0.7. This value was selected to achieve a balance between consistency and creativity in generating more natural and coherent responses. Holtzman et al. suggested that a temperature in this range avoided excessively deterministic or random output [30]. The decoding strategy used was a sampling-based method, which allowed model to generate a variety of responses relevant to the context of the question. To control the output, the maximum limit of tokens that each response can generate was set to 1048 tokens, while the context length was set to 2048 tokens, to ensure the responses remain in the scope of model understanding of document content.

E. Preprocessing and Data Embedding

The system initially preprocessed PDF document using LangChain, which obtained important information from the document and broke the text into smaller parts. These parts were changed into number sequences, saved in a vector

database (DB). Subsequently, FAISS library was used to store number sequences in the vector database, which allowed for quick searching of the meaning of the information taken from PDF document [31].

TABLE 2
SAMPLE QUESTION-ANSWER PAIRS FROM THE DATASET DERIVED FROM ACADEMIC GUIDEBOOK.

Question	Answer
Explain the awards and recognition rights for students at UNISNU Jepara (*Jelaskan hak penghargaan bagi mahasiswa di UNISNU Jepara*)	Every student is entitled to receive awards from the University, Faculty, or other departments based on their achievements (*Setiap mahasiswa berhak untuk mendapatkan penghargaan Universitas, Fakultas dan atau bagian lainnya sesuai dengan prestasi yang diraih*)
Explain the forms of awards and recognition for students at UNISNU Jepara (*Jelaskan bentuk penghargaan bagi mahasiswa di UNISNU Jepara*)	Awards for students can include certificates, prizes, tuition waivers, priority for scholarships, and other benefits (*Bentuk penghargaan bagi mahasiswa dapat berupa piagam, hadiah, pembebasan uang kuliah, dan prioritas untuk mendapatkan beasiswa, serta fasilitas lainnya*)
Explain the scholarship rights provided by UNISNU Jepara (*Jelaskan hak beasiswa yang diberikan oleh UNISNU Jepara*)	The University provides scholarships to students who excel in both academic and non-academic fields (*Universitas memberikan beasiswa kepada mahasiswa yang mempunyai prestasi di bidang akademik dan bidang non-akademik*)
What is the name of the degree for the Electrical Engineering Study Program? (*Apa nama gelar untuk Program Studi Teknik Elektro*)	Electrical Engineering Study Program Degree: Bachelor of Engineering (*Program Studi Teknik Elektro Gelar : Sarjana Teknik*)

As shown in Table 2, examples of question-answer pairs from the entire dataset are taken from Chapter 3 of Academic Guidebook, which discusses the manners of campus life. The collected dataset has been converted into a JSON file and uploaded to HuggingFace.

The process of tokenization, made easy by SFTTrainer tool in TRL library from HuggingFace, greatly simplifies the supervised fine-tuning process by keeping the format of the training data consistent and ensuring efficient processing [32]. Structured text is converted into numerical tokens, maintaining the same input format for subsequent processing steps.

After tokenization, the data are changed into numbers as high-dimensional vectors by the embedding part of model. This stage helps model learn how words and meanings are connected, showing how dense embeddings improve the structure and effectiveness of large language model [33].

F. Fine-Tuning

Fine-tuning method was used to train two open-source models, LLaMA-2 7B and Mistral 7B. To ensure a fair comparison, this study used the same datasets and experimental setups. Google Colaboratory was selected as the training platform because of the accessible GPU T4 runtime, which delivered sufficient performance for medium-scale training tasks, although each session was limited to about 4 hours. To further enhance speed and memory efficiency, the Unsloth framework was used [13]. Designed for resource-constrained environments such as free Colab sessions, Unsloth enables fine-tuning of large language model by optimizing memory management and supporting 4-bit and 8-bit quantization.

Low-Rank Adaptation (LoRA) was applied during fine-tuning to reduce the number of trainable parameters without compromising model quality. This method lowers memory use and computational cost, enabling efficient fine-tuning in resource-limited environments such as Colab. Training was carried out with SFTTrainer module from the HuggingFace TRL library, which streamlined the process by integrating training configurations, dataset loading, and tokenizer setup. The dataset followed the Alpaca-style format, consisting of instruction, input, and output fields. In this study, the input field was left blank when the instruction provided sufficient context, but was used when additional information was required. An example of the prompt format is provided in Table 3.

TABLE 3
EXAMPLE OF ALPACA-STYLE PROMPT FORMAT USED IN FINE-TUNING

Instruction	Input	Output
Explain how students obtain a Student Identity Card (KTM) at UNISNU Jepara (*Jelaskan bagaimana mahasiswa mendapatkan Kartu Tanda Mahasiswa (KTM) di Unisnu Jepara*)	(optional)	The Student Affairs Division issues Identity Card for new students after they receive their Student Identification Number (NIM) (*Mekanisme penerbitan kartu tanda mahasiswa bagi mahasiswa baru diterbitkan oleh Bagian kemahasiswaan, dibuat setelah mahasiswa mendapatkan NIM*).
What is the name of the degree for the Islamic Economics Study Program? (*Apa nama gelar untuk Program Studi Ekonomi Islam*)	(optional)	Islamic Economics Study Program: Bachelor of Economics (*Program Studi Ekonomi Islam Gelar: Sarjana Ekonomi*)

To avoid overloading memory constraints, the batch size was set small, at 2 per GPU, with gradient accumulation every 4 steps. This method helped to simulate large batches without draining memory. Training was performed for 50 epochs with a warm-up phase during the initial 30 steps. A learning rate of 2e-4 was used with a linear scheduler and an 8-bit version of AdamW optimization. Mixed precision was also applied, using bf16 when available, or fp16 as an alternative. A seed of 3407 was used to ensure reproducible results. During training, the process was logged every step of the way, and the results were stored in a local directory. The detailed training configuration for both models is summarized in Table 4.

TABLE 4
FINE-TUNING CONFIGURATION FOR MISTRAL 7B AND LLAMA-2 7B

Model	Quantization	Load Memory	Batch Size	Epoch	Learning Rate	Precision(bf16/fp16)	Fine-tune Memory
Mistral 7B	8-bit (AdamW optimizer)	4.5 GB	2 (4x acc step)	50	2e-4	bf16	1.461 GB
LLaMA-2 7B	8-bit (AdamW optimizer)	3.826 GB	2 (4x acc step)	50	2e-4	bf16	0.651 GB

G. Performance Evaluation

The test was performed by presenting questions from the dataset to fine-tuned model. The generated responses were also assessed using two distinct metrics, namely METEOR Score and BERTScore. Each of these metrics offers unique strengths in evaluating response quality from complementary perspectives, specifically literal correspondence and semantic similarity, respectively.

1) METEOR Score

METEOR Score is used to evaluate the quality of the generated text, considering precision, recall, and a penalty for fragmentation [28]. Specifically, METEOR Score is calculated for each candidate response against its corresponding reference, ranging from 0 to 1 [34]. It also calculates precision as the proportion of unigrams in the machine translation that are also present in the reference translation. Recall is defined as the proportion of unigrams in the reference translation that are found in the machine translation. Subsequently, FMean score is calculated according to Equation (1) using the harmonic mean of precision (P) and recall (R), with a greater emphasis placed on recall.

$$FMean = \frac{10PR}{R + 9P} \quad (1)$$

Penalty is used to capture word order mismatches between the machine answer and the reference. The calculation is based on the number of sequentially mapped word groups (chunks) in the machine answer compared to the reference [35]. As more word groups are out of order, penalty will significant affect METEOR's final score. The formula used is Equation (2):

$$Penalty = 0.5 * \left(\frac{chunks}{unigrams\ matched}\right)^3 \quad (2)$$

METEOR score is calculated according to Equation (3):

$$Score = FMean * (1 - Penalty) \quad (3)$$

METEOR is selected as the evaluation metric in this study because of the ability to balance precision and recall, which is crucial for assessing the quality of short, context-sensitive responses generated in question-answering tasks. The higher correlation with human judgment ensures that the evaluation is in line with how humans perceive response quality [35].

2) BERTScore

BERTScore is used to measure how similar the meaning of model answers is to the reference answers, not just when the words are the same [29]. This method uses language model like BERT, where each sentence is turned into

a list of numbers (a vector). With the vector, BERTScore can calculate how similar the meaning between words is despite different formations. Furthermore, BERTScore calculates three main components namely, precision, recall, and F1 score.

The precision measures how many words in model answer are similar in meaning to the words in the reference. However, the recall looks at how many words from the reference are successfully recognized in the answer. F1 score is calculated as the harmonic mean of precision (P) and recall (R), which is formulated as in Equation (4):

$$F_{BERT} = 2\frac{P_{BERT} \cdot R_{BERT}}{P_{BERT} + R_{BERT}} \quad (4)$$

The use of BERTScore is important because answers from LLM model do not always have identical wording to the reference, despite having correct content. Therefore, BERTScore provides an evaluation that is closer to how humans understand answers.

III. Results

Training dataset was used to train AI model with the help of Unsloth in the Google Colab system. During the process, the training dataset was uploaded to HuggingFace to be assessed by Unsloth. Subsequently, the trained AI model could be uploaded back to HuggingFace, consuming approximately 3-4 hours.

After training process, the prepared PDF documents were input into the system. When the system runs, the document is processed into vector form using embedding. Subsequently, the system was given questions from the testing dataset. The answers generated would be based on the prepared source documents. Fig. 2 indicates the increased ability of model to respond after fine-tuning process.

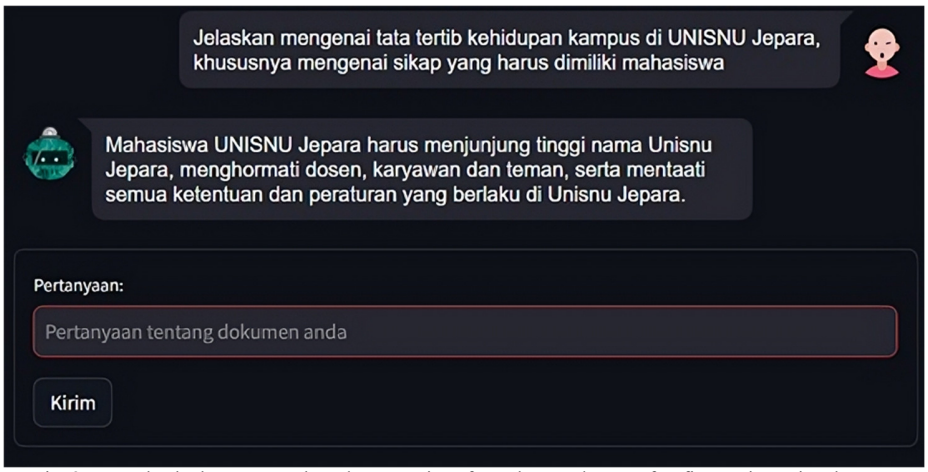

Fig. 2 Example chatbot answers based on questions from the test dataset after fine-tuning Mistral 7B

To evaluate the learning progress of both models during fine-tuning, training loss was tracked at each stage. The results showed distinct learning curves for Mistral 7B and LLaMA-2 7B, as presented in Fig. 3.

At the beginning of training, Mistral 7B showed a relatively high loss value. In the first 15 to 20 stages, there was a very aggressive decline, where the loss value quickly dropped below 1.0, recording 0.8777 at step 15. After this phase, the loss value continued to decrease substantially and reached a range below 0.1. For most of the training duration, the training loss was very stable and in the low range, namely between 0.04 and 0.01. However, some minor spikes occurred, such as at stage 39 (1.3703), 248 (0.0490), and 799 (0.0484), without affecting the overall decreasing trend. The average loss value was close to 0.02–0.01 at the end of training, indicating that Mistral 7B was able to learn and adapt to the dataset very effectively and efficiently

Initially, the training loss value of LLaMA-2 7B was in a relatively high range, but experienced a significant and consistent decrease in the first 50 to 100 steps until it reached a value below 1.0. After the initial decline stage, the loss value tends to be stable for most of the training duration. Although the downward trend is generally positive, there are several significant spikes in the loss value, such as at step 39 (4.3955), step 134 (4.4281), and step 723 (4.5518). These spikes are caused by anomalous data batches, tokenization errors, or fluctuations in the learning rate settings. Several loss values of 0.0000 were found due to empty batches or interference in the logging process. This model

showed efficient learning performance, with an average loss value approaching 0.5–0.6 at the end of training (step 950). The results showed that LLaMA-2 7B successfully absorbed patterns from the data stably despite minor fluctuations.

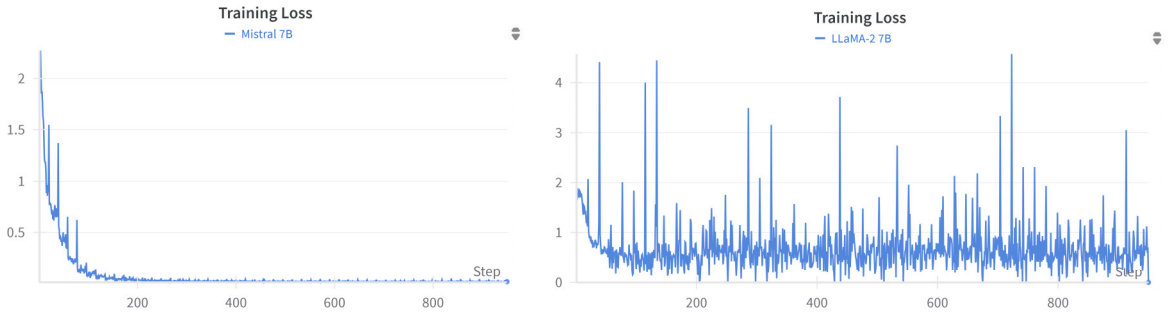

Fig. 3 Training loss progress on Mistral 7B and LLaMA-2 7

The response times produced by fine-tuned Mistral 7B showed a competitive level of efficiency using LLaMA-2, with an average of 3.76 and 3.80 minutes, respectively. In group B3, Mistral 7B achieved a response time of 3.47 minutes, which was slightly faster than LLaMA-2 7B at 3.67 minutes. These results showed differences in computational efficiency due to variations in model architecture and inference approaches. Similarly, a previous study on LLMs emphasized that design options, including attention mechanisms and inference optimization, could affect computational performance [36].

As shown in Fig. 4, fine-tuned Mistral 7B could produce a faster response than non-fine-tuned Mistral 7B. Additionally, fine-tuned Mistral 7B was superior to LLaMA-2 7B in terms of response time. In group B2, LLaMA-2 7B was superior to others by recording a response time of 3.48 minutes, followed by fine-tuned Mistral 7B, non-fine-tuned Mistral 7B, and non-fine-tuned LLaMA-2 7B at 3.66, 4.60, and 4.44, respectively.

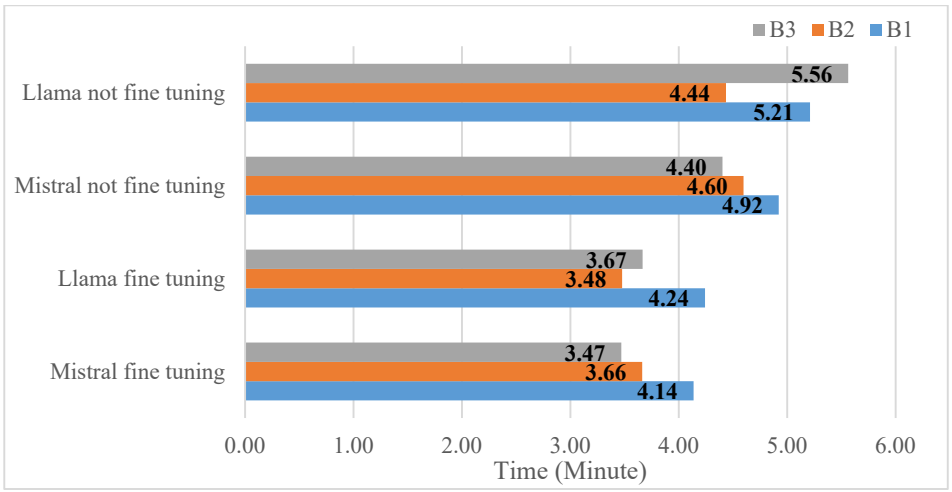

Fig. 4 Average response

Results from the appropriate fine-tuning process are shown by the fact that the reaction speed generated by fine-tuned Mistral 7B is superior to non-fine-tuned. In line with the increase in processing speed, the refined model also produces better quality responses that are more contextually aware and precise. Therefore, fine-tuned Mistral 7B is highly appropriate for use in circumstances where dependability and effectiveness are necessary.

In comparison to the other two models, fine-tuned Mistral is the most optimal. As shown in Table 5, fine-tuning not only increases response accuracy but also decreases hallucinations and accelerates response times. Alongside impressive outcomes, performance can still be maximized with more enhancements and adjustments, particularly in terms of cutting response times even more without sacrificing response relevance and accuracy.

IV. DISCUSSION

This study presents an understanding of how to improve the effectiveness of LLMs for domain-specific chatbot. The improvement process had a considerable effect on model performance. For example, the enhanced Mistral 7B obtained an average METEOR Score [28] of 0.36, average BERTScore [29] of 0.77, compared to only 0.10 and 0.66, respectively, in non-fine-tuned counterpart. A similar trend was also observed in LLaMA-2 7B, where fine-tuning led to METEOR Score improvement from 0.16 to 0.27 and BERTScore from 0.68 to 0.72. These results show how important the refinement process is in terms of improving accuracy and relevance in context, particularly when dealing with specific questions.

TABLE 5
METEOR SCORE, BERT SCORE, AND TIME COMPARISON TABLE ON FINE-TUNING MISTRAL 7B, FINE-TUNING LLAMA-2 7B AND NON-FINE-TUNING MISTRAL 7B, NON-FINE-TUNING LLAMA-2 7B

Group	not-fine-tuned LLaMA-2 7B			not-fine-tuned Mistral 7B			fine-tuned LLaMA-2 7B			fine-tuned Mistral 7B		
	Meteor Score	F1 Score	Time (minute)	Meteor Score	F1 Score	Time (minute)	Meteor Score	F1 Score	Time (minute)	Meteor Score	F1 Score	Time (minute)
B1	0,16	0,68	5,21	0,10	0,66	4,92	0,27	0,72	4,24	**0,34**	**0,75**	**4,14**
B2	0,08	0,63	4,44	0,08	0,64	4,60	0,30	0,74	**3,48**	**0,40**	**0,78**	3,66
B3	0,12	0,67	5,56	0,11	0,66	4,40	0,27	0,73	3,67	**0,34**	**0,77**	**3,47**

The success of fine-tuning method is attributed to several key strategies that correlate with both the computational constraints and task-specific objectives. The use of Unsloth and Google Colab's T4 GPU enabled memory-efficient training through 4-bit and 8-bit quantization, while LoRA reduced the number of trainable parameters, making the process lighter and faster. Furthermore, SFTTrainer module simplified the supervised training pipeline and ensured consistency. The Alpaca-style prompt format provided clear instruction-output pairs, assisting model in better understanding domain-specific queries. A memory-aware configuration, including small batch size with gradient accumulation and mixed-precision training, also contributed to training stability.

TABLE 6
SAMPLE MODEL RESPONSES EVALUATED BY RELEVANCE (METEOR) AND SEMANTIC CORRELATION (BERTSCORE)

Model	Question	Model Response	METEOR	BERTScore
Mistral 7B (FT)	What is the name of the degree for the Electrical Engineering study program? (*Apa nama gelar untuk Program Studi Teknik Elektro*)	The title of the undergraduate degree in the Electrical Engineering Study Program is Bachelor of Engineering (*Nama gelar sarjana Program Studi Teknik Elektro adalah Sarjana Teknik*)	0,81	0,83
Mistral 7B (FT)	How many credits are students allowed to take at UNISNU Jepara? (*Berapa jumlah sks yang boleh diambil oleh mahasiswa di Unisnu Jepara*)	The maximum number of credits programmed for new, transfer, and lateral students in the first academic year is 24 (twenty-four) credits (*Jumlah beban sks yang diprogramkan bagi mahasiswa baru, pindahan dan lateral jalur pada tahun akademik pertama maksimal 24 (dua puluh empat) sks*)	0,04	0,54
LLaMA-2 7B (FT)	Explain the introduction to the Student Activity Unit of the Faculty of Economics and Business at UNISNU Jepara (*Jelaskan mengenai pengenalan Unit Kegiatan Kemahasiswaan Fakultas Ekonomi dan Bisnis di Unisnu Jepara*)	The Faculty of Economics and Business's Student Activity Units (UKK) are: 1. USB. 2. Mapala. 3. Student Press Institute (*UKK Fakultas Ekonomi dan Bisnis yaitu: 1. USB. 2. Mapala. 3. Lembaga Pers Mahasiswa*)	0,88	0,94
LLaMA-2 7B (FT)	How many credits are students allowed to take at UNISNU Jepara? (*Berapa jumlah sks yang boleh diambil oleh mahasiswa di Unisnu Jepara*)	New students at UNISNU Jepara are subject to the following provisions: f. The number of credits allowed for students with academic leave status is subject to the exception of the provisions regarding the number of credits allowed for students in SIAKAD (*Mahasiswa baru Unisnu Jepara berdasarkan ketentuan sebagaimana berikut :f. Jumlah sks yang boleh diambil oleh mahasiswa dikecualikan ketentuan tentang jumlah sks yang boleh diambil mahasiswa di SIAKAD dengan status cuti akademik*)	0,02	0,54

Examples of responses from the models are shown in Table 6. Further analysis was performed using METEOR Score to measure relevance and BERTScore to assess semantic correlation. Responses that scored high on both metrics were considered to show good understanding and strong relevance to the lesson. Meanwhile, responses with low

scores on both metrics tended to be off-topic or semantically inappropriate. Several examples showed an association between the two metrics, such as responses with adequate semantic correlation but were less relevant, or vice versa. Fine-tuned Mistral 7B and LLaMA-2 7B showed faster response times than the non-fine-tuned Mistral 7B [20] and LLaMA-2 7B [24]. This result showed that there were variations in the computational design capable of affecting the processing speed. Fine-tuning process changed model's response behavior.

A key limitation of this study is the limited dataset from Academic Guidebook, which restricts result applicability. Expanding the dataset and improving response speed without reducing accuracy should be priorities for future work [25], to strengthen chatbot performance and usability in resource-constrained environments.

V. Conclusions

In conclusion, this study examines how Mistral 7B and LLaMA-2 7B process Academic Guidebook and address hallucinations in inaccurate responses. The results show that fine-tuning significantly improves performance, with fine-tuned Mistral 7B outperforming both non-fine-tuned Mistral and LLaMA 7B. Higher accuracy and faster responses are observed in Academic Guidebook domain, showing the value of fine-tuning for enhancing LLMs in domain-specific chatbot applications. Due to the accuracy and efficiency, fine-tuned Mistral 7B is well-suited for academic environments and resource-limited settings. However, this study is constrained by a limited dataset, showing the need for future investigation to expand data sources concerning Academic Guidebook to improve response time and maintain accuracy.

Author Contributions: *Davied Indra Rachman*: Conceptualization, Methodology, Software, Investigation, Writing - Original Draft, Writing - Review & Editing. *Agus Subhan Akbar*: Conceptualization, Methodology, Review, Supervision. *Alzena Dona Sabilla*: Review, Supervision.

All authors have read and agreed to the published version of the manuscript.

Funding: This research received no specific grant from any funding agency.

Conflicts of Interest: The authors declare no conflict of interest.

Data Availability: The data that supports the findings of this study are available upon request from the authors.

Informed Consent: There were no human subjects.

Institutional Review Board Statement: Not applicable.

Animal Subjects: There were no animal subjects.

ORCID:
Davied Indra Rachman: https://orcid.org/0009-0005-1711-0788
Agus Subhan Akbar: https://orcid.org/0000-0002-6011-7011
Alzena Dona Sabilla: https://orcid.org/0009-0005-3048-5787

References

[1] Z. Chen *et al.*, "MEDITRON-70B: Scaling Medical Pretraining for Large Language Models," pp. 1–38, 2023, [Online]. Available: http://arxiv.org/abs/2311.16079

[2] T. Han *et al.*, "MedAlpaca -- An Open-Source Collection of Medical Conversational AI Models and Training Data," pp. 1–9, 2023, [Online]. Available: http://arxiv.org/abs/2304.08247

[3] M. Y. Jabarulla, S. Oeltze-Jafra, P. Beerbaum, and T. Uden, "MedDoc-Bot: A Chat Tool for Comparative Analysis of Large Language Models in the Context of the Pediatric Hypertension Guideline," 2024, [Online]. Available: https://github.com/yaseen28/MedDoc-Bot

[4] H. Abdelazim, M. Tharwat, and A. Mohamed, "Semantic Embeddings for Arabic Retrieval Augmented Generation (ARAG)," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 11, pp. 1328–1334, 2023, doi: 10.14569/IJACSA.2023.01411135.

[5] K. Muludi, K. M. Fitria, J. Triloka, and S. -, "Retrieval-Augmented Generation Approach: Document Question Answering using Large Language Model," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 3, pp. 776–785, 2024, doi: 10.14569/IJACSA.2024.0150379.

[6] S. Verma, K. Tran, Y. Ali, and G. Min, "Reducing llm hallucinations using epistemic neural networks," *arXiv Prepr. arXiv2312.15576*, 2023.

[7] R. A. *et al.*, "Automating Machine Learning Model Development: An OperationalML Approach with PyCARET and Streamlit," in *2023 Innovations in Power and Advanced Computing Technologies (i-PACT)*, 2023, pp. 1–6. doi: 10.1109/i-PACT58649.2023.10434389.

[8] O. Topsakal and T. C. Akinci, "Creating large language model applications utilizing langchain: A primer on developing llm apps fast," in *International Conference on Applied Engineering and Natural Sciences*, 2023, vol. 1, no. 1, pp. 1050–1056.

[9] A. Q. Jiang *et al.*, "Mistral 7B," pp. 1–9, 2023, [Online]. Available: http://arxiv.org/abs/2310.06825

[10] M. Nagel, M. Fournarakis, R. A. Amjad, Y. Bondarenko, M. Van Baalen, and T. Blankevoort, "A white paper on neural network quantization," *arXiv Prepr. arXiv2106.08295*, 2021.

[11] L. Chen and P. Lou, "Clipping-Based Post Training 8-Bit Quantization of Convolution Neural Networks for Object Detection," *Appl. Sci.*, vol. 12, no. 23, p. 12405, 2022.

[12] Z. Zhang *et al.*, "Exploring the potential of flexible 8-bit format: Design and algorithm," *arXiv Prepr. arXiv2310.13513*, 2023.

[13] H. Touvron *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv Prepr. arXiv2307.09288*, 2023.

[14] U. JEPARA, "Pedoman Akademik TA. 2021/2022," in *UNISNU JEPARA*, Mulyadi, K. Sa'diyah, H. Amalia, A. Riyadi, P. Nugroho, A. Tyanto, A. Zamroni, A. Maulana, and Miftahurrohman, Eds. Jepara: Universitas Islam Nahdlatul Ulama Jepara, 2021, pp. 52–60. [Online]. Available: https://drive.google.com/file/d/1guq08eoE0cx9rtQ_7B865MZqkXsWJucF/view

[15] R. Gelar Guntara, "Pemanfaatan Google Colab Untuk Aplikasi Pendeteksian Masker Wajah Menggunakan Algoritma Deep Learning YOLOv7," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 1, pp. 55–60, 2023, doi: 10.47233/jteksis.v5i1.750.

[16] T. T. Tin, S. Y. Xuan, W. M. Ee, L. K. Tiung, and A. Aitizaz, "Interactive ChatBot for PDF Content Conversation Using an LLM Language Model," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 9, 2024, doi: 10.14569/IJACSA.2024.01509105.

[17] B. Hendrickx, "Meteor," *Sp. Explor. Humanit. a Hist. Encycl. Vol. 1-2*, vol. 1–2, no. June, pp. 344–346, 2010, doi: 10.1145/2567940.

[18] M. Douze *et al.*, "The Faiss library," 2024, [Online]. Available: http://arxiv.org/abs/2401.08281

[19] M. Amin, "Development of a Music Education Framework Using Large Language Models (LLMs)," 2024.

[20] D.-M. Petroșanu, A. Pîrjan, and A. Tăbușcă, "Tracing the Influence of Large Language Models across the Most Impactful Scientific Works," *Electronics*, vol. 12, no. 24, p. 4957, 2023.

[21] D. Lin, Y. Wen, W. Wang, and Y. Su, "Enhanced Sentiment Intensity Regression Through LoRA Fine-Tuning on Llama 3," *IEEE Access*, vol. 12, pp. 108072–108087, 2024, doi: 10.1109/ACCESS.2024.3438353.

[22] D. A. Hameed, T. A. Faisal, A. M. Alshaykha, G. T. Hasan, and H. A. Ali, "Automatic evaluating of Russian-Arabic machine translation quality using METEOR method," *AIP Conf. Proc.*, vol. 2386, no. 1, p. 40036, 2022, doi: 10.1063/5.0067018.

[23] S. Banerjee and A. Lavie, "Meteor: an automatic metric for MT evaluation with high levels of correlation with human judgments," *Proc. ACL-WMT*, pp. 65–72, 2004.

[24] S. Minaee *et al.*, "Large language models: A survey," *arXiv Prepr. arXiv2402.06196*, 2024.