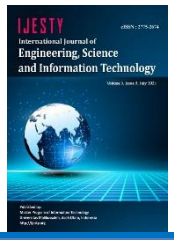
International Journal of Engineering, Science and InformationTechnology
Volume 5, No. 3 (2025) pp. 268-277
eISSN: 2775-2674
Website: http://ijesty.org/index.php/ijesty
DOI: https://doi.org/10.52088/ijesty.v5i3.951
Research Paper, Short Communication, Review, Technical Paper



Retrieval Augmented Generation-Based Chatbot for Prospective and Current University Students

Luluk Setiawati Hartono[1], Esther Irawati Setiawan[1*], Vrijraj Singh[2]

[1]Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Indonesia
[2]Agprop, India

**Corresponding author Email: esther@istts.ac.id*

The manuscript was received on 16 January 2025, revised on 28 February 2025, and accepted on 25 May 2025, date of publication 13 June 2025

Abstract

Universities utilize chatbots as assistants for users, especially prospective and current students, to access information and answer questions with relevant answers. This study introduces a new approach to an open-source model-based Q&A system using Gemma2-2b-it by combining Retrieval Augmented Generation (RAG) and Finetuning (FT) techniques. Previously, some studies have focused on only one approach, but this study will combine and compare both methods separately. Raw conversation data from WhatsApp, the main university website, and university PDF documents are used. The Retrieval Augmented Generation Assessment (RAGAS) framework will be used to evaluate the performance of the RAG model. In contrast, precision, recall, and similarity are used to assess the comparative performance of RAG and finetuning. The results of the RAGAS show that RAG using the base model is better than RAG using a finetuned model, which has 0.78 faithfulness, 0.64 answer relevancy, 0.81 context precision, and 0.68 context recall, so the overall RAGAS Score is 0.72. The comparison of precision and recall of finetuning are higher than those of using RAG, but the similarity score is not much different. Furthermore, the potential improvement for RAG of this study can be increased by adding a reranking process in the retrieved context, and finetuning of the embedding model can also be added to increase the retrieval process's performance. In addition, further experiments on various datasets and the challenge of overfitting in finetuning must be overcome so that the model can perform better generalization.

Keywords: *Natural Language Processing, Retrieval Augmented Generation, RAGAS, Finetuning, Chatbot.*

1. Introduction

A chatbot is an artificial intelligence that simplifies the interaction between services and users through conversation. The availability of information 24/7 will certainly affect user satisfaction, so chatbots are widely applied in many sectors, such as banking [1], e-commerce [2], health [3][4], and education [5][6][7]. In higher education, service providers sometimes must re-inform different users of the same information. Of course, this reduces the work efficiency of service providers with high service needs. So, they utilized chatbots as virtual assistants to help users, especially students and prospective students, to access information easily [8].

Chatbots started as rule-based systems that relied on pre-defined rules and scripts to interact with users [6]. On the other hand, large pre-trained language models perform well in various NLP tasks, including question-answering [9]. NLP aims to develop techniques that enable computers to understand natural human language [10]. LLM has developed well recently and has shown outstanding performance in many fields and applications [11]. However, they are not good at answering knowledge-related questions, such as details of a particular product or service in a company. Language models produce poor responses due to the lack of data during their training. RAG [12] emerged as a solution. The main components of RAG are the retriever and generator [13]. Retriever to retrieve information from external sources based on input queries. Large pre-trained language models act as generators to generate responses using information from documents retrieved by the retriever. Although RAG has limitations, such as the possible failure to recover the most relevant documents to a query and the potential failure of the generative model to incorporate the retrieved information into its response, RAG has established itself as a practical means for developing many LLM-based applications serving a variety of purposes [14]. RAG presents a more cost-effective alternative to the extensive training and refinement processes typically required for an LLM [15].

This study will create a chatbot for university operational activities, including new student admissions and some University information. The external data used are data from the university website and documents. The WhatsApp chat file .txt dataset is used to finetune the base model and add it to the knowledge base. This study's large pre-trained language model is Gemma-2-2b-it [16] and will be evaluated using RAGAS [17]. This study will also compare finetuning and RAG based on precision, recall, and cosine similarity.

2. Literature Review

This section will discuss the supporting theories used in this research and references from previous research.

2.1. Finetuning

Finetuning is taking a pre-trained machine learning model and training it on a specific dataset to adapt it for a particular application. The internal parameters are adjusted during the finetuning, and the weights are pre-trained on one particular task-oriented dataset to improve performance [18]. Finetuning is controlled by hyperparameters such as the learning rate, batch size, number of epochs, optimizer, etc. LoRA [19] and QLoRA [20] are used because they allow finetuning of large models in a more efficient and memory-saving manner. LoRA only changes a small number of parameters by adding low-rank matrices, making the process lighter and faster. QLoRA combines LoRA with model quantization into a smaller format, making it more memory-efficient and suitable for devices with limited resources without sacrificing significant performance.

2.2. Information Retrieval

The basis of RAG is a QA system that combines information retrieval and NLP [21]. Information retrieval is essential in searching and retrieving relevant information from external data sources as a context for generative models to produce accurate and informative responses. This process ensures that the model does not rely on innate knowledge but on current and specific data according to user needs. RAG information retrieval can be done through keyword-based search, embedding semantic search, or using vector databases. This is typically achieved by measuring the vector distance between the document and the question, combining traditional retrieval metrics with semantic understanding to improve the quality of generative output [15]. RAG searches for relevant information from databases using a retrieval technique and then combines the information as a context to produce more informative responses through the generation process.

2.3. Large Language Model

A large Language Model (LLM) is a transformer-based neural language model containing tens to hundreds of billions of parameters pre-trained on a large text corpus [22]. The transformer architecture consists of two primary mechanisms: an encoder, responsible for reading all input text simultaneously, and a decoder, which produces an output sequence in the form of predictions [23]. It traces the evolution of language models from early statistical and neural models to pre-trained language models and finally to LLMs such as GPT, LLaMA, and Gemma. It has emerging capabilities such as learning in context, following instructions, and multi-step reasoning. In the generation phase of the RAG system, the role of the LLM is crucial as a text or answer generator that utilizes information retrieved in the retrieval phase. After the retrieval phase provides relevant documents or information from external sources, the LLM uses the data as context to generate responses to the user's question.

2.4. Prior Research

Several previous studies have become an essential foundation in developing this research topic. Table 1 summarizes prior research and the methods proposed in this study. [8] introduces BARKPLUG V.2, a comprehensive chatbot that utilizes university data from the Mississippi State University website, which is crawled as an external data corpus and used in RAG. The pre-trained language model used in this study is GPT-3.5-turbo to generate the final response based on the context of the retrieved results and the prompt from the user. BARKPLUG V.2 was evaluated using RAGAS and produced a good performance with an average RAGAS score of 0.96.

The study [1] created a KemenkeuGPT chatbot using financial data from the Ministry of Finance of the Republic of Indonesia. In addition, the data is in the form of question-and-answer pairs related to Indonesian financial data collected from financial websites. The base model used is GPT-3.5-turbo, which has the highest performance after being compared with seven other large language models. This study uses several stages to improve the chatbot's performance, first using RAG with an accuracy of 42%, then adding prompt engineering with an accuracy of 60%, and finally using fine tuning with an accuracy of 61%. KemenkeuGPT was also evaluated using RAGAS and obtained scores for each criterion, correctness (0.44), faithfulness (0.73), precision (0.40), and recall (0.60), which are the highest when compared to 7 other LLMs. This study also uses human evaluation involving several experts in various fields of public finance expertise to assess the response of the Ministry of Finance and Public Finance.

Another study [6] creates a chatbot for students to access information on various topics such as admission, course selection, campus facilities, etc. They use Meta-llama/Llama-2-7b-chat-hf and Mistralai/Mistral-7B-Instruct-v0.2. External data is obtained from the university website, followed by data preprocessing. The chatbot retrieves the most relevant data from the university's comprehensive knowledge base, ensuring the responses remain up-to-date and appropriate, using the BLEU score for chatbot evaluation. The experimental results show that Llama-2-7b-chat-hf provides a viable solution to the challenge of delivering university-related information to students.

Table 1. Summary of Prior and Current Research

Prior Research	Dataset	Methods	Evaluation Metrics
S. Neupane et al. (2024)	42 campus resources	RAG using GPT-3.5-turbo base model	RAGAS Score and SUS
G. F. Febrian and G. Figueredo (2024)	Financial data, question-answer pair, and documents from scraping the Kemenkeu website	RAG using a finetuned GPT-3.5-turbo model	Accuracy and RAGAS Score
M. Ali Quidwai and A. Lagana (2024)	Collection of research papers related to multiple myeloma	RAG using Mistral Instruct 7B base model	Comparative Analysis and Benchmarking Framework by Expert

Prior Research	Dataset	Methods	Evaluation Metrics
A. Balaquer et al. (2024)	Document and question-answer pair about agriculture	Comparing RAG, finetuning, and a combination of RAG + finetuning using Llama2 and GPT-4 model	Accuracy
H. Soudani et al. (2024)	Long tail knowledge from Wikipedia and Wikidata	Comparing RAG, finetuning, and a combination of RAG + finetuning on multiple LMs	Accuracy
Current Research	Raw conversation WhatsApp file, University Website, and PDF supporting document	Comparing RAG, finetuning, and a combination of RAG + finetuning using the Gemma2 model	RAGAS Score, Precision, Recall, ROUGE, Cosine Similarity

Meanwhile, [24] evaluates the effectiveness of RAG and finetuning techniques in agriculture. The datasets used include large domain-specific datasets from three major producing countries in the form of documents and question-and-answer data. The methodology involves a flow of document collection, information extraction from PDFs, question generation using GPT-4, responses generated using RAG, and finetuning LLM with LoRA. The results show that RAG and finetuning significantly improve the accuracy and relevance of reactions, with finetuning producing more concise and precise outputs. At the same time, RAG excels in contextual significance and has a lower initial cost. The combination of both yields the highest accuracy, up to 74%.
The study initiated by [25] also examines the performances of finetuning and RAG, but this study handles less popular or low-frequency factual knowledge in question-answering tasks. The study uses three datasets focused on long-tail knowledge, namely PopQA, WitQA, and EntityQuestion (EQ), all derived from Wikipedia and Wikidata with varying levels of entity popularity. The method includes synthetic data generation for finetuning using prompt-based techniques, end-to-end QA generation, and various retrieval models such as BM25, DPR, and Contriever for RAG. Experiments are conducted on multiple LMs with different sizes and architectures, comparing full finetuning, PEFT, and other data augmentation strategies. The results show that RAG significantly outperforms finetuning in handling less popular knowledge, especially when combined with high-quality retrievers. PEFT is better able to maintain reasoning ability when used with RAG.

3. Methods

This section will give more information about the stage of this study. The first stage is dataset collection and data preprocessing. The second stage is data chunking and embedding before being inserted into the database vector. The third stage is finetuning the model using WhatsApp data, and finally, RAG is deployed with the prepared data from the vector database as external knowledge. The research flow can be seen in Figure 1.

3.1. Dataset

This study uses some sources of data, including .txt chat files from WhatsApp that answered questions about student admissions, non-degree programs, and university operations. The chat file .txt is obtained from the export menu in the WhatsApp application and collected 1078 conversations. In addition, the main university website and four PDF documents regarding academic guidelines are also used.

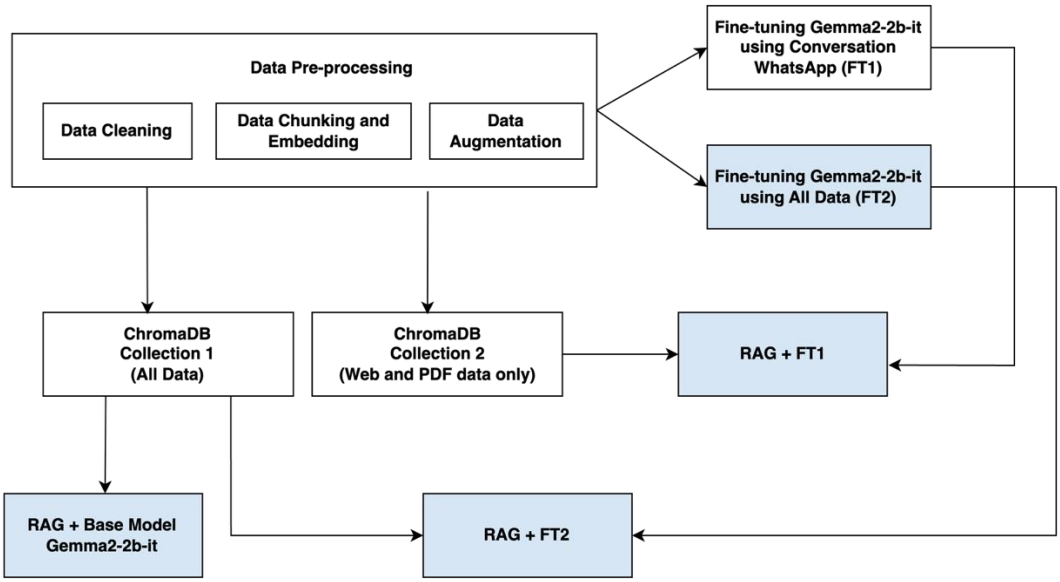


Fig 1. Research Flow

3.2. Data Preprocessing

In the preprocessing stage, all the data that has been collected will be processed. The process carried out includes data cleaning, data chunking, and data augmentation.

1. Data Cleaning
   In the WhatsApp dataset, words in the form of emails and mobile phone numbers will be masked with the prefixes {phone} and {email}. Then, the data will be cleaned by removing unnecessary sentences in the conversation, such as '<this message was deleted>,'

'<this message was edited," <{contact} is a contact>,' etc., and deleting empty rows. The clean chats will be merged if there are multiple lines with the same sender. Chats from the university will be labelled with "model," while the chats from the student or prospective student will be labelled with "user." Since the chat dataset is irregular, it is necessary to manually check with an expert whether the question and answer pairs are appropriate. If there is something that is not appropriate, the expert will shift to the relevant answer. After cleaning, we need to reorder the chat. Finally, merged data will be changed to the chat template received by the Gemma model, <start_of_turn> to indicate the beginning of the dialogue turn, and <end_of_turn> to indicate the end of the dialogue turn. Table 2 shows each example of data preprocessing performed on WhatsApp files. This formatted data will be used in the LLM finetuning process.

Table 2. Example of WhatsApp File Preprocessing

Preprocessing Step	Before	After
Masking personal information	**User:** Halo, selamat pagi **Model:** Halo, bisa dibantu isi nama, No. tlp/WA, email dan apa yang dapat kami bantu? **User:** Nama Lola, no tlp/wa 081xxxxx, email lolaxxxx@gmail.com **User:** Halo kak izin bertanya dulu boleh terkait kelas pascasarjana?	**User:** Halo, selamat pagi **Model:** Halo, bisa dibantu isi nama, No. tlp/WA, email dan apa yang dapat kami bantu? **User:** Nama Lola, no tlp/wa{phone}, email {email} **User:** Halo kak izin bertanya dulu boleh terkait kelas pascasarjana?
Merge conversation	**User:** Halo, selamat pagi **Model:** Halo, bisa dibantu isi nama, No. tlp/WA, email dan apa yang dapat kami bantu? **User:** Nama Lola, no tlp/wa{phone}, email {email} **User:** Halo kak izin bertanya dulu boleh terkait kelas pascasarjana?	**User:** Halo, selamat pagi **Model:** Halo, bisa dibantu isi nama, No. tlp/WA, email dan apa yang dapat kami bantu? **User:** Nama Lola, no tlp/wa{phone}, email {email}. Halo kak izin bertanya dulu boleh terkait kelas pascasarjana?
Apply chat template	**User:** Halo, selamat pagi **Model:** Halo, bisa dibantu isi nama, No. tlp/WA, email dan apa yang dapat kami bantu? **User:** Nama Lola, no tlp/wa{phone}, email {email}. Halo kak izin bertanya dulu boleh terkait kelas pascasarjana?	**<start_of_turn>user** Halo, selamat pagi**<end_of_turn>** **<start_of_turn>model** Halo, bisa dibantu isi nama, No. tlp/WA, email dan apa yang dapat kami bantu**?<end_of_turn>** **<start_of_turn>user** Nama Lola, no tlp/wa{phone}, email {email}. Halo kak izin bertanya dulu boleh terkait kelas pascasarjana?**<end_of_turn>**

2. Data Chunking and Embedding

The chunking stage will be treated on the website and in PDFs. The main website of this university will be scraped, and all content on every URL on the website will be taken. Meanwhile, PyPDFLoader will extract the text from the PDF documents. All data from the website and PDF will be divided into several parts before being inserted into the database vector. The mechanism used to split the content into chunks is RecursiveCharacterTextSplitter, which efficiently divides and manages text data based on character limits. If a chunk exceeds the limit, it is recursively divided into smaller, meaningful units, such as sentences. The chunk will be embedded using two embedding models, namely, paraphrase-multilingual-mpnet-base-v2 (PMM), which supports multilingual, and LazarusNLP/all-indo-e5-small-v4 (LZR), which is an embedding model in Indonesian, and the result of this embedding will be inserted together with the chunk into the chroma vector database. Both embedding models will also be viewed and compared regarding RAG performance. The chunk size used is 1000 with a chunk overlap of 100. Chunk size 1000 ensures the information in each chunk is large enough but still well managed. It allows the model to process data without exceeding the existing token limit. If the chunk size is too large, the information contained may be too much and risk being truncated, while if it is too small, the model may lose broader context.

Meanwhile, overlap 100 is used to maintain the continuity of the context between chunks. With overlap, the model can capture the relationship between parts of the text on the chunk boundary so that information at the end of the previous chunk can still be connected to the next part. If the overlap is too significant, the model may face redundancy. If the overlap is too small, the model may miss meaningful connections between chunks, losing important context.

3. Data Augmentation

The data augmentation process in this study was carried out by utilizing various data sources to enrich and expand the variety of training datasets. Chat conversation data often has less than optimal quality for direct use in model training because it contains non-standard language, abbreviations, typos, and incomplete or ambiguous context. [26] the synthetic datasets generated using LLM are promising, so in this study, several types of augmentation will be carried out using LLM. First, informal and natural WhatsApp conversation data was used as a basis for augmentation data in conversation paraphrases to produce sentence variations that maintain the original meaning but with different structures and word choices. Second, content from websites and PDFs was processed to build question-and-answer pairs (QA Pairs) based on relevant content. QA pairs from websites and PDFs were used to evaluate and compare RAG and finetuning. This approach increases the amount of training data. It improves the quality and diversity of data representation so that the trained model is expected to understand broader contexts and diverse language styles in natural language

processing tasks. We also augment the conversation data stored in the chroma database, changing the conversation data into contextual text to help RAG use contextual retrieval. Table 3 shows examples of each of the data augmentations performed. By performing augmentation, the trained model will be able to understand variations in conversational language better and provide more stable performance.

Table 3. Example of Data Augmentation

Augmentation Process	Original	Augmented
Paraphrase Conversation Data	**User:** Brp lama S2 untk kuliahny? **Model:** Untuk kuliahnya S2 selama 2 tahun ya	**User:** Berapa lama durasi studi S2? **Model:** Untuk kuliahnya S2 selama 2 tahun ya
QA Pairs Website and PDF	Jenis Beasiswa Kerjasama Mitra 7.000.000 Instansi telah melakukan perjanjian kerjasama dalam bentuk penandatanganan kerjasama	**Question:** Siapa saja yang berhak untuk mendapatkan beasiswa kerjasama mitra? **Answer:** Dapat diperoleh oleh instansi yang telah melakukan kerjasama
Contextual Data	**User:** saya mau tanya untuk S1 psikologi **Model:** mohon maaf untuk S1 psikologi belum ada ya	Mahasiswa ingin tahu informasi mengenai program S1 Psikologi. Namun, informasi yang diberikan adalah bahwa program S1 Psikologi belum tersedia. **User:** saya mau tanya untuk S1 psikologi **Model:** mohon maaf untuk S1 psikologi belum ada ya

3.3. Finetuning LLM

The model finetuning process in this study was carried out using SFTTrainner combined with the QLoRA (Quantized Low-Rank Adaptation) technique to enable efficient finetuning of large models with lower memory usage without sacrificing performance. This also considers the use of Google Colab as a training platform that provides access to high-performance GPUs with limited memory capacity used in this study. QLoRA is an extended version of LoRA that works by quantizing the precision of the weight parameters to 4-bit precision. This method significantly reduces the memory footprint, making it possible to run LLM models on less powerful hardware. With QLoRA, the training process can run optimally in this environment while maintaining the quality of model adaptation to new data. The training process divides 80% of the total data into 80% training data and 20% validation data. At the same time, 20% of the data is used for testing. The hyperparameter used is using a learning rate of 1e-3 for two epochs. In this study, two finetuning scenarios will be run. The first is finetuning the model with a pure dataset from WhatsApp conversations (FT1), and the second is finetuning with all data, both WhatsApp conversations and question-answering pairs from the web and pdf (FT2), which will be used for the comparing process.

3.4. RAG Workflow

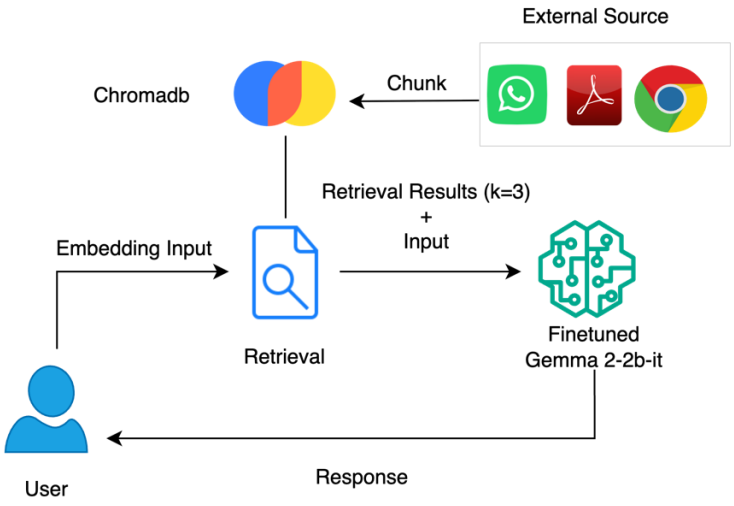


Fig 2. RAG Workflow

As can be seen in Figure 2, the first step is to convert the source document into a vector representation through embedding. This study breaks documents that have gone through the preprocessing stage into chunks to maintain context granularity. Each chunk is then embedded using the selected embedding models above. The resulting embedding vectors are then stored in a vector database, ChromaDB,

which allows fast and efficient searching based on cosine distance. The user question input is also transformed into an embedding using the same model to ensure consistent vector space. Then, a search is performed to find the k-most relevant document chunks based on similarity. A total of 3 retrieved contexts, together with the question, will be used by the finetuned Gemma2 model to generate a response. The prompt, which consists of the retrieval contexts and the user's question, is then processed by a finetuned Gemma2 model to create a response. At this stage, the generative model takes the information captured by the retrieval stage and formulates a coherent and contextually appropriate answer.

3.5. Evaluation Metrics

This section will discuss the metrics used in this study. The experimental results can be analyzed comprehensively by understanding the evaluation metrics used. Three types of evaluations will be carried out: LLM Finetuning evaluation, RAG evaluation, and comparing RAG and Finetuning performance.

1. Finetuning LLM Evaluation
   Performance evaluation of a finetuned LLM is crucial to ensure the model can produce relevant and accurate output according to the desired task. In this study, the assessment was carried out using the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metric. The measures count the number of overlapping units such as n-grams, word sequences, and word pairs between reference and generated responses [27]. The higher the ROUGE score obtained, the better the text quality produced by the model, reflecting the content in ground truth. In addition, the evaluation process also considers the training loss and validation loss values during training to identify potential overfitting and ensure model generalization to new data. With a combination of quantitative metrics and loss analysis, this finetuning evaluation aims to select the best hyperparameter configuration that produces optimal performance for the model.

2. RAG Evaluation
   Es et al. [17] propose RAGAS, a framework that introduces a set of metrics to evaluate RAG flows on retrieval and generation metrics, such as context or answer relevance. In this study, we consider the following metrics: (i) Faithfulness (F), (ii) Context Recall (CR), (iii) Context Precision (CP), and (iv) Answer Relevance (AR).

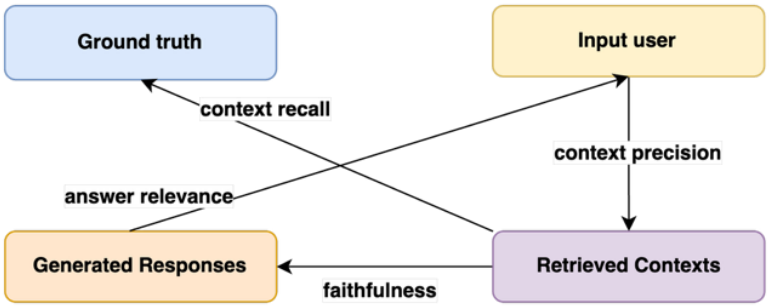


Fig 3. RAGAS Framework

Based on Figure 3, faithfulness assesses the extent to which the generated answer is consistent and matches the information in the retrieved context, thus ensuring that the model does not generate false or biased information. Context recall assesses how completely the retrieval system retrieves the relevant context. These two context metrics greatly influence the quality of the input provided in the generation phase, as an accurate and complete context will improve the quality of the generated answer. Context precision measures the accuracy of the retrieved context, that is, how much of the information retrieved from the database is relevant to the user's question. Finally, answer relevance evaluates the overall suitability of the final answer to the user's question, reflecting the success of the integration stages. This criterion is used cosine as an absolute component [28]. The interrelationship of these four criteria is crucial in RAG because optimal retrieval performance will provide a strong foundation for the generation phase, thereby improving the system's overall quality.

3. RAG and Finetuning Comparison
   This study also compares the performance between the RAG method and the finetuning model. There are four scenarios proposed, as shown in Figure 4.
   a. Scenario 1 - Finetuning Gemma2 model using all datasets, including raw and augmented data (FT2)
   b. Scenario 2 – RAG using the base model as a generation model and using all datasets as external knowledge
   c. Scenario 3 - RAG using a model that has been finetuned with chat data only (FT1) while web and pdf data are in external knowledge
   d. Scenario 4 - RAG using the finetuned model in scenario 1 (FT2) and using all data as external knowledge.

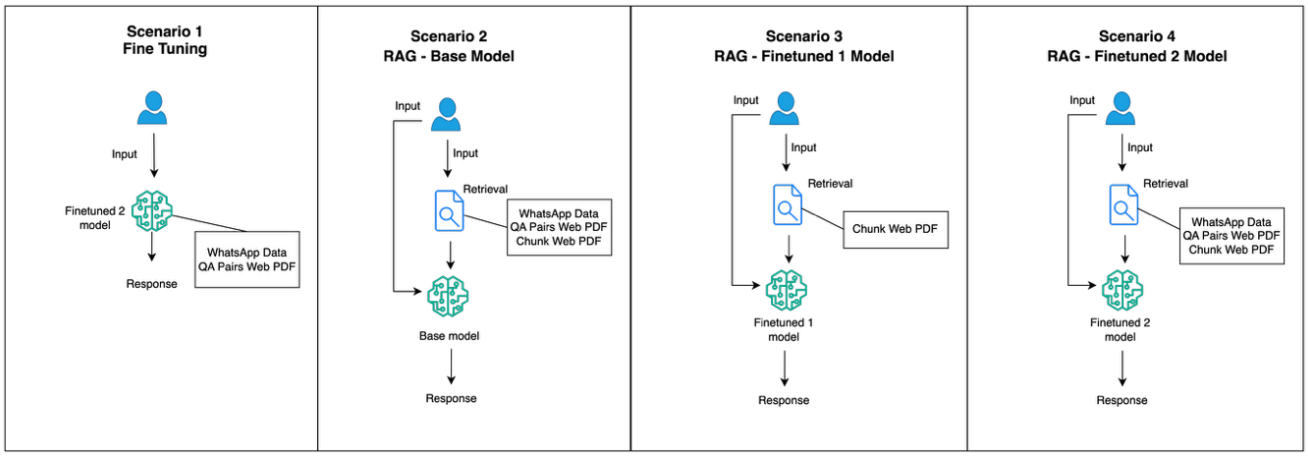


Fig 4. Finetuning and RAG Scenarios Comparison

In scenario 1, inference will be performed using the FT2 model, where the model has been trained on all data, including conversation, web, and PDF data. This is done so that the data owned by the finetuning and RAG is balanced when comparing performance. Scenario 2 will perform inference using the base model where there is external knowledge in the form of all existing datasets. Scenario 3 will use the FT1 model, which is a model that has been trained on the conversation dataset only, while web and pdf data are external knowledge. This is done to see if the retrieval phase can provide context that matches the questions not in the FT1 training data and if the FT1 model can use the retrieved context to produce a response. The last scenario uses the FT2 model combined with complete external knowledge. ROUGE Score and cosine similarity will be used to see the performance of each scenario.

4. Results and Discussion

This section presents the results and discusses experiments conducted to test the effectiveness of the various approaches in this study.

4.1. Finetuned LLM Performance

From 2 types of finetuning that have been processed, FT1 with conversation training data obtained a training loss of 0.9 and a validation loss of 1.2, while FT2 trained on all data obtained a training loss of 0.9 and validation loss of 1. This means the FT1 and FT2 models are slightly overfitting because they are good in training data but not so good in validation data. Then, the model was tested with test data, and an average ROUGE score of 0.36 for FT1 and 0.45 for FT2 was obtained. The model must be retrained with all training, validation, and test data for RAG use. The retraining results obtained a training loss of 1.3 for FT1 and 1.2 for FT2.

On the other hand, we tested FT2 with 300 data, which is the same data test used in RAG. The results of this data evaluation will be compared with other RAG data evaluations. The experiment results obtained ROUGE score as follows: ROUGE1 score of 0.36, ROUGE2 score of 0.24, and ROUGE-L of 0.35. The precision and recall values from the results of this experiment are both 0.43. We also measured cosine similarity and obtained 0.51, indicating that although the generated text does not always match word-for-word, its meaning is close enough to the reference so that the model can understand and convey the gist of information well.

4.2. RAG Performance

The evaluation of RAG performance using RAGAS shows that the metric used provides a clear picture of the strengths and weaknesses of the model in generating answers. Table 4 shows that the scenario with the LazarusNLP model embedding in scenario 2 has a high RAGAS score of 0.72 compared to other scenarios. The high faithfulness value indicates that the base model can respond fairly accurately to the available context. The base model with different embedding models also obtained the second rank of the faithfulness score. In all finetuned model scenarios, they have very low faithfulness, indicating that the retrieval phase failed to provide sufficiently relevant or accurate information, so the model generation stage could not process the data correctly.

Meanwhile, scenario 2 obtained the highest context recall in the LazarusNLP embedding model, scoring 0.69. This means that the model can extract vital information from the context or supporting document available so that the answer completely covers the source's key elements. The context precision of the results above shows a score > 0.35. The high context precision value of 0.81 in scenarios 2 and 4 in the LazarusNLP means that most of the model's information is relevant and accurate for the question asked, so the model does not include unnecessary or misleading data in its answers. Last, answer relevance > 0.34, and the highest is 0.65. This means that the answer generated is sufficient to answer the core of the question precisely according to what is requested by the user. This component reflects the model's effectiveness in providing responses that suit the user's needs.

Table 4. RAGAS Score Results

Embedding Model	Scenario	RAGAS Criteria				RAGAS Score
		F	CR	CP	AR	
PMM	Scenario 2	0.76	0.53	0.65	0.56	0.62
	Scenario 3	0.04	0.28	0.36	0.44	0.28
	Scenario 4	0.09	0.54	0.65	0.36	0.41
LZR	Scenario 2	0.78	0.69	0.81	0.65	0.72

Embedding Model	Scenario	RAGAS Criteria				RAGAS Score
		F	CR	CP	AR	
	Scenario 3	0.08	0.30	0.45	0.38	0.30
	Scenario 4	0.11	0.68	0.81	0.34	0.48

From the results above, it can be concluded that RAG using a base model is superior to a finetuned model because a base model that has not been finetuned tends to have good generalization capabilities. After all, it has been trained on comprehensive and diverse data. So, when combined with dynamically relevant document retrieval, the base model can use additional context from the retrieved documents without being limited by the bias of specific finetuning data.

4.3. Comparison of RAG and Finetuning

The evaluation results show that RAG does not always provide better results than finetuning. Although RAG can potentially improve answers' relevance by utilizing external information through the retrieval stage, the results obtained from the finetuning are more consistent in producing accurate and relevant answers. This is because finetuning allows the model to learn deeply about the specific data provided without depending on the quality and relevance of the database, which may not always be optimal in particular contexts.

One of the main factors influencing these results is RAG's reliance on the quality and relevance of external data. RAG incorporates information from external sources through the retrieval phase before the generation process, so the data available in the database greatly influences the answer quality. In this experiment, although the retrieval phase successfully retrieved relevant information, the quality of the information was not always well integrated into the generation phase, which reduced the model's effectiveness in providing accurate and contextual answers to the question.

Table 5. The Score of Finetuning and RAG Comparison Results

Embedding Model	Scenario	Score					
		ROUGE1	ROUGE2	ROUGE-L	Precision	Recall	Cosine Similarity
-	Scenario 1	0.36	0.24	0.35	0.43	0.43	0.51
PMM	Scenario 2	0.28	0.16	0.25	0.23	0.23	0.63
	Scenario 3	0.29	0.18	0.27	0.25	0.25	0.58
	Scenario 4	0.15	0.06	0.13	0.12	0.12	0.48
LZR	Scenario 2	0.32	0.19	0.29	0.26	0.26	0.59
	Scenario 3	0.28	0.16	0.26	0.24	0.24	0.50
	Scenario 4	0.15	0.07	0.13	0.11	0.11	0.40

Table 5 shows finetuning in scenario one results better than other RAG scenarios in precision and recall. Finetuning allows the model to learn deeply from a more focused and specific dataset without relying on the quality and relevance of external data. Finetuning is more efficient and faster because it only involves one stage, generation, without additional processes such as retrieval. In the performance evaluation, the finetuned model showed a ROUGE1 score of 0.36 and a precision and recall score of 0.43. These values indicate that the finetuning model can produce answers that are pretty similar to the reference, with a relatively high level of information accuracy, and it can cover most of the critical information from the training data. In contrast, when comparing the reference data with its generated responses, the RAG model obtained a lower score than finetuning in rouge score, precision, and recall. A comparison of metrics between scenarios can be seen in Figure 5.

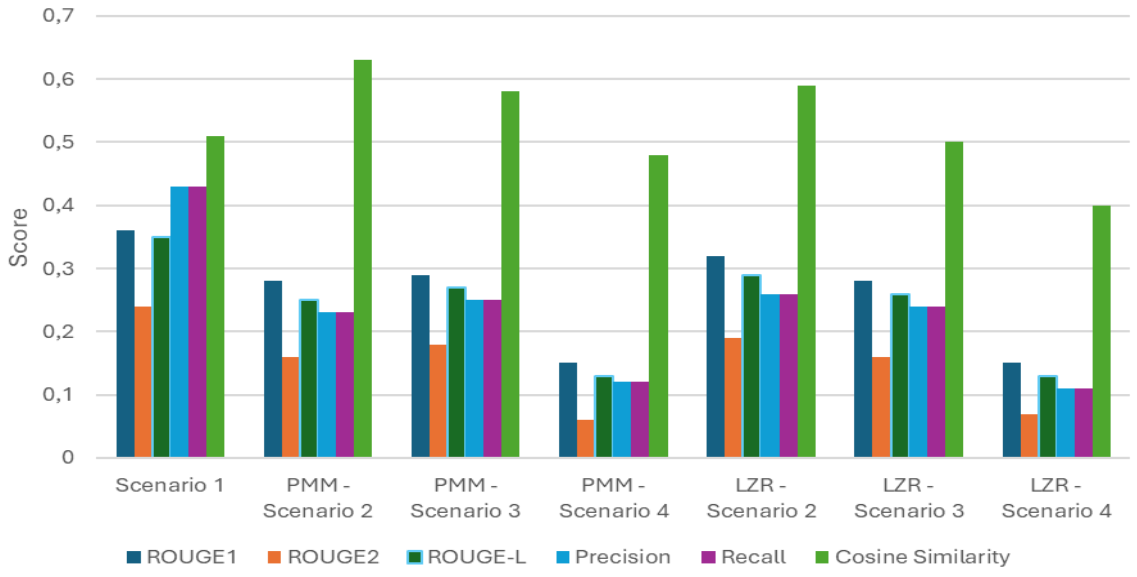


Fig 5. Comparison of Finetuning and RAG on Gemma2-2b-it based on scenarios

Table 6. Standard Deviation of Finetuning and RAG Comparison Results

Embedding Model	Scenario	Standard Deviation					
		ROUGE1	ROUGE2	ROUGE-L	Precision	Recall	Cosine Similarity
-	Scenario 1	0.32	0.33	0.32	0.36	0.33	0.29
PMM	Scenario 2	0.26	0.24	0.25	0.24	0.29	0.24
	Scenario 3	0.27	0.26	0.27	0.26	0.33	0.23
	Scenario 4	0.20	0.17	0.19	0.19	0.25	0.21
LZR	Scenario 2	0.26	0.24	0.25	0.25	0.29	0.22
	Scenario 3	0.26	0.25	0.26	0.26	0.32	0.23
	Scenario 4	0.20	0.19	0.20	0.20	0.26	0.21

In this study, the standard deviation of each scenario was also measured. It can be seen in Table 6 that in the finetuning model (scenario 1), all criteria except ROUGE2 are below the mean value, while the other models are almost above the mean value. This means that the model in scenario 1 is more consistent in producing answers than other models. However, with a standard deviation value that is not far from the average, the model shows large fluctuations in its performance when faced with varying data. These results are also influenced by the limitation of the dataset used, which causes the model to experience overfitting. WhatsApp conversation data is less than ideal for use as a dataset in these models, primarily due to the irregularity and limited structure of the conversations. WhatsApp conversations are often informal, with more casual language, abbreviations, and even non-standard slang, affecting the model's ability to understand context well. Additionally, WhatsApp conversations frequently contain typos, incomplete sentences, and unstructured discussions with topics that change suddenly, making them more difficult to extract into organized and relevant data for tasks such as automated Q&A. These factors make WhatsApp conversation data noisier and less structured to be applied in models that require a more consistent and organized dataset.
Based on the discussion above, integration between RAG and real-time finetuning has the potential to be implemented. It allows the model to adjust to new data or conversation patterns dynamically. The model can continuously learn from direct feedback or changes in the conversation, thereby reducing fluctuations or high standard deviations that may arise from the instability of WhatsApp conversation. This combination will improve the model's consistency and accuracy, reducing significant variations in results.

5. Conclusion

Based on the evaluation results that have been carried out, it can be concluded that RAG using a base model shows better performance compared to RAG using a finetuned model because the base model has a broader generalization capability and is flexible in utilizing external contexts that are dynamically taken through the retrieval process. The base model is not tied to a particular finetuning data pattern or bias, so that it can adapt to various questions and contexts. Thus, RAG based on a base model can provide more relevant and accurate responses by combining the strengths of pre-trained models and real-time contexts without the risk of overfitting.
However, when compared between RAG and finetuning, the finetuning model remains superior in producing precise answers and following training data in a specific domain. Finetuning allows the model to optimize its performance on a particular task or dataset to provide more focused and consistent output. Therefore, the choice between finetuning and RAG must consider the application's needs. This research makes a practical contribution by offering an approach that can be implemented in small to medium-sized universities despite resource constraints without the need for high-tech infrastructure. The proposed open-source model solution enables educational institutions with limited budgets to improve the efficiency of data-driven academic services. The developed model and methods have the potential to be widely replicated, developed, and produced, thus opening up opportunities for educational technology to be applied in a broader range of academic institutions.
For further research, it is recommended to use a more qualified and representative conversation dataset to improve model performance, considering that the current chat dataset has limitations in terms of language standardization and completeness of context. In addition, exploring more adaptive finetuning techniques and finetuning the embedding model can be an essential step in producing a more accurate vector representation that suits the characteristics of specific conversation data. In the retrieval phase, the application of reranking techniques also needs to be explored to improve the relevance and quality of the context retrieved, as well as trying various other retrieval methods, such as hybrid retrieval, that can provide more optimal context search results and support a better answer generation process.

References

[1] G. F. Febrian and G. Figueredo, “KemenkeuGPT: Leveraging a Large Language Model on Indonesia’s Government Financial Data and Regulations to Enhance Decision Making,” Jul. 2024, [Online]. Available: http://arxiv.org/abs/2407.21459
[2] B. A. T. de Freitas and R. de Alencar Lotufo, “Retail-GPT: leveraging Retrieval Augmented Generation (RAG) for building E-commerce Chat Assistants,” 2024. [Online]. Available: https://arxiv.org/abs/2408.08925
[3] M. Quidwai and A. Lagana, *A RAG Chatbot for Precision Medicine of Multiple Myeloma*. 2024. doi: 10.1101/2024.03.14.24304293.
[4] A. Sen, D. R. Satya Ranjan Dash, D. R. Manas Ranjan Pradhan, and S. Parida, “HEALTHCARE QUESTION ANSWERING SYSTEM IN BENGALI-A PROPOSED MODEL,” *J Theor Appl Inf Technol*, vol. 30, no. 12, 2024.
[5] H. Dinh and T. K. Tran, “EduChat: An AI-Based Chatbot for University-Related Information Using a Hybrid Approach,” *Applied Sciences*, vol. 13, no. 22, p. 12446, Nov. 2023, doi: 10.3390/app132212446.

[6] U. H. Khan, M. H. Khan, and R. Ali, "Large Language Model based Educational Virtual Assistant using RAG Framework," *Procedia Comput Sci*, vol. 252, pp. 905–911, 2025, doi: 10.1016/j.procs.2025.01.051.
[7] T. A. I. T. Baha, M. E. L. Hajji, Y. Es-Saady, and H. Fadili, "Towards highly adaptive Edu-Chatbot," in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 397–403. doi: 10.1016/j.procs.2021.12.260.
[8] S. Neupane *et al.*, "From Questions to Insightful Answers: Building an Informed Chatbot for University Resources," May 2024, [Online]. Available: http://arxiv.org/abs/2405.08120
[9] Z. Jiang *et al.*, "Active Retrieval Augmented Generation," May 2023, [Online]. Available: http://arxiv.org/abs/2305.06983
[10] Q. Zaman, S. Safwandi, and F. Fajriana, "Supporting Application Fast Learning of Kitab Kuning for Santri' Ula Using Natural Language Processing Methods," *International Journal of Engineering, Science and Information Technology*, vol. 5, no. 1, pp. 278–289, Jan. 2025, doi: 10.52088/ijesty.v5i1.713.
[11] S. Sharma *et al.*, "Retrieval Augmented Generation for Domain-specific Question Answering," Apr. 2024, [Online]. Available: http://arxiv.org/abs/2404.14760
[12] P. Lewis *et al.*, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," May 2020, [Online]. Available: http://arxiv.org/abs/2005.11401
[13] H. Yu, A. Gan, K. Zhang, S. Tong, Q. Liu, and Z. Liu, "Evaluation of Retrieval-Augmented Generation: A Survey," May 2024, [Online]. Available: http://arxiv.org/abs/2405.07437
[14] J. Swacha and M. Gracel, "Retrieval-Augmented Generation (RAG) Chatbots for Education: A Survey of Applications," Apr. 01, 2025, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/app15084234.
[15] Y. Huang and J. Huang, "The Survey of Retrieval-Augmented Text Generation in Large Language Models," Apr. 2024, [Online]. Available: http://arxiv.org/abs/2404.10981
[16] Gemma Team *et al.*, "Gemma 2: Improving Open Language Models at a Practical Size," Jul. 2024, [Online]. Available: http://arxiv.org/abs/2408.00118
[17] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "RAGAS: Automated Evaluation of Retrieval Augmented Generation," Sep. 2023, [Online]. Available: http://arxiv.org/abs/2309.15217
[18] M. R. J, K. VM, H. Warrier, and Y. Gupta, "Fine Tuning LLM for Enterprise: Practical Guidelines and Recommendations," 2024. [Online]. Available: https://arxiv.org/abs/2404.10779
[19] E. J. Hu *et al.*, "LoRA: Low-Rank Adaptation of Large Language Models," Jun. 2021, [Online]. Available: http://arxiv.org/abs/2106.09685
[20] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient Finetuning of Quantized LLMs," May 2023, [Online]. Available: http://arxiv.org/abs/2305.14314
[21] E. I. Setiawan, J. Santoso, and Gunawan, "Answer Ranking with Weighted Scores in Indonesian Hybrid Restricted Domain Question Answering System," in *3rd 2021 East Indonesia Conference on Computer and Information Technology, EIConCIT 2021*, Institute of Electrical and Electronics Engineers Inc., Apr. 2021, pp. 456–461. doi: 10.1109/EIConCIT50028.2021.9431915.
[22] S. Minaee *et al.*, "Large Language Models: A Survey," Feb. 2024, [Online]. Available: http://arxiv.org/abs/2402.06196
[23] Z. Fitra Ramadhan and A. Benny Mutiara, "Sentiment Analysis of Honkai: Star Rail Indonesian Language Reviews on Google Play Store Using Bidirectional Encoder Representations from Transformers Method," *International Journal of Engineering, Science and Information Technology*, vol. 3, no. 3, pp. 1–6, Sep. 2023, doi: 10.52088/ijesty.v3i3.462.
[24] A. Balaguer *et al.*, "RAG vs Finetuning: Pipelines, Tradeoffs, and a Case Study on Agriculture," Jan. 2024, [Online]. Available: http://arxiv.org/abs/2401.08406
[25] H. Soudani, E. Kanoulas, and F. Hasibi, "Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge," Mar. 2024, doi: 10.1145/3673791.3698415.
[26] P. Sutanto, J. Santoso, E. I. Setiawan, and A. P. Wibawa, "LLM Distillation for Efficient Few-Shot Multiple Choice Question Answering," 2024. [Online]. Available: https://arxiv.org/abs/2412.09807
[27] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," in *Text Summarization Branches Out*, Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: https://aclanthology.org/W04-1013/
[28] S. Roychowdhury, S. Soman, H. G. Ranjani, N. Gunda, V. Chhabra, and S. K. Bala, "Evaluation of RAG Metrics for Question Answering in the Telecom Domain," Jul. 2024, [Online]. Available: http://arxiv.org/abs/2407.12873