Fine-Tuning Large Language Model (LLM) to Answer Basic Questions for Prospective New Students at Syiah Kuala University Using the Retrieval-Augmented Generation (RAG) Method

2024 Ninth International Conference on Informatics and Computing (ICIC) | 979-8-3315-1760-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICIC64337.2024.10956296

Hary Rachmat
Department of Informatics
Universitas Syiah Kuala
Aceh, Indonesia
haryrachmat10@gmail.com

Hammam Riza
Research Center for Artificial Intelligence and Cyber Security
National Research and Innovation Agency (BRIN)
Jakarta, Indonesia
hammam.riza@brin.go.id

Taufik Fuadi Abidin*
Department of Informatics
Universitas Syiah Kuala
Aceh, Indonesia
taufik.abidin@usk.ac.id
**corresponding author*

***Abstract*— USK Mistral 7B is a large language model designed to answer basic admission questions at Universitas Syiah Kuala (USK). The model was fine-tuned using the open-source model of Mistral 7B using collected data from admissions and lectures at the university. The QLoRA and RAG techniques were used to train the model and retrieve relevant information from external data sources. The results were evaluated using the ROUGE score. Responses were generated with a score of >0.5 on ten out of 46 questions with the RAG method, and testing with the fine-tuning method was carried out on 20 questions and resulted in responses with a score of 1.0 from all questions asked. The performance of USK Mistral 7B shows its potential as an effective tool in helping students querying information about admission and lectures at USK.**

***Keywords*— Large Language Model, Fine-tuning, RAG**

I. INTRODUCTION

With the rapid advancement of artificial intelligence (AI), large language models (LLMs) have revolutionized natural language processing (NLP), thus allowing computers to interact with text and language. One example of an LLM is Generative Pretrained Transformer 4 (GPT-4), which has been extensively trained on large amounts of text data, allowing GPT-4 to perform text analysis across multiple domains. LLM can be applied to various fields such as health, education, law, and other fields by conducting pre-training using large amounts of data in these fields.

ChatGPT, one of the most popular chatbot applications today, has been the subject of various studies exploring its capabilities [1]. These studies have demonstrated its potential in the medical field to aid clinical documentation [2], in the banking sector to classify texts [3] and in the field of law for determining potential violations. These examples highlight the diverse and practical applications of large language models like ChatGPT [4].

In addition to closed-source LLMs such as GPT-4, some studies use open-source ones like those conducted by Huang et al. [5] by adapting Llama's LLM to the legal domain to assist lawyers in preparing technical reports. Bhatti et al. [6] fine-tuning the LLM Llama 70B named "SM70" to address a wide range of complex medical questions and clinical decision-making. Zhao et al. [7] fine-tuning the LLM Llama 7B named "Ophtha-LLaMA2" to help diagnose eye diseases that will provide decision support for doctors. Barandoni et al. [8] conducted a comparative analysis of LLMs to extract the needs of travel customers from TripAdvisor posts by Leveraging a variety of models, including open-source models such as Mistral 7B and closedsource models such as GPT-4 and Gemini. The results highlight the efficacy of open-source LLMs, especially the Mistral 7B, in achieving comparable performance to closed-source models. Research on chatbots as virtual assistants has been conducted by Jonatan and Igor [9] to improve the efficiency of service to customers. The research that has been presented has shown the potential of LLM to help work in various fields.

Based on data.usk.ac.id website [10], the number of students registering for Syiah Kuala University (USK) is increasing yearly. To provide information such as registration details, tuition fees, and other information, USK provides a website where students can find information about matters related to the university. When there is information that is not yet available on the website, prospective students can ask questions directly through social media, such as direct messages (DM) through the Instagram application, and also meet university staff directly at the Integrated Service Unit (ULT) or Public Relations (HUMAS) section. This study will examine the application of LLM as a virtual assistant as a chatbot to provide interactive information related to academic administration at USK.

II. RESEARCH METHODS

A. Dataset Conditions

The dataset contains 231 data samples related to information about the lecture system and new student admissions at USK. The data is then preprocessed from the dataset obtained for fine-tuning the Mistral 7B model here using as many as 20 Q&A datasets, converted to Alpaca style format, and then stored in a format with csv extension. A total of 231 datasets are also created in the form of statements and saved in a format with extensions pdf extensions are then saved into the huggingface repository, which will later be used in the next stage, namely the fine-tuning model [11]. Data in pdf format will be used in the RAG method [12] to manage the data more efficiently.

B. *Quantization and Model Training*

In Fig. 1, the fine-tuning process with the Mistral 7B model is divided into three stages, which can be seen as follows:

Fig. 1. Fine-tuning process on the Mistral Model 7B

1) Model Quantization and Data Preparation

The fine-tuning process on the Mistral 7B model is trained with the help of the transformer library to quantize the Mistral 7B, and the BitsAndBytesConfig library interface is used to quantize the Mistral 7B USK model. The purpose of this BitsAndBytes library is to reduce the precision value of the floating point in the weight of the USK Mistral Model 7B, which is processed by quantization from the precision with a high value to a lower value. The Mistral 7B USK model weights are converted into int4 format through a quantization layer and stored in the GPU. The primary computing process is performed on the CUDA, which will reduce memory usage and improve the model's efficiency. This process makes it possible to fine-tune larger LLMs on consumer-grade GPUs. The fine-tuned model is stored in the hugging face repository.

2) Pre-trained Model

Before the training, an analysis of well-known LLMs from the current LLM pool was conducted to select the most suitable model for this research. Evaluation of LLMs mainly considers several key aspects such as adaptability to specific domains, compatibility in academic standards, bilingual language skills, availability of models on open source, efficiency in parameters, cost, and licensing considerations on the model. This consideration also looks at several evaluation metrics tested across various benchmarks.

3) Parameter-Efficient Fine-Tuning (PEFT)

Considering dataset availability, the goal of reducing the cost of the training process, and the potential for failure risk, the researchers chose the Progressive Layer Freezing and Fine-tuning (PEFT) method [13] to refine the Mistral 7B model. PEFT selectively reduces a small number of parameters on additional models. This way, model training can significantly reduce computational and memory storage costs. It will enable efficient adaptation of pre-trained LLMs in various application domains.

In this study, a low-level adjustment method (QLoRA) was explicitly used [14] to improve large language models. The Lora method [15] involves freezing pre-trained weights on the original model and creating a new version on the matrix with lower rank values for layers and query values. This lower-rated matrix has values on significantly fewer parameters than the original model, allowing for adjustment of memory usage on GPUs with smaller storage sizes. The advantage of this method can be seen in the ability of various LoRA adapters to repurpose native LLMs, thereby reducing the total memory usage required when providing answer text in use cases in tasks on specific domains and in various cases to be applied. Unlike LoRA, the QLoRA method represents an iteration of the LoRA method that will save more memory storage. The QLoRA method takes LoRA one step further by measuring the value or weight on a LoRA adapter with a smaller matrix value to a lower precision value (e.g., the model weight value becomes 4-bit, and not the value on a model with an initial value of 8-bit). This approach can reduce the memory size and requirements of model storage. In the QLoRA method, The pre-trained model is used on GPU memory with a quantized 4-bit weight value, different from the 8-bit model with the LoRA method. Although the bit precision value decreases, the QLoRA method is able to maintain the same effectiveness as the LoRA method. So, this study uses a model refinement method using the QLoRA technique. The fine-tuning method for LLM training uses the Unsloth library and Huggingface's TRL library [16]. The Unsloth library can make finetuning LLMs 2x faster.

Table I shows the hyperparameter settings used in fine-tuning using the Unsloth and Huggingface's TRL libraries. It summarizes the configuration for creating a FastLanguageModel.from_pretrained instance using the unsloth component, with configurations like maximum sequence length, data type, and 4-bit loading [17].

TABLE I. FASTLANGUAGEMODEL.FROM_PRETRAINED CONFIGURATION

Parameter	Value
model_name	mistralai/Mistral-7B-v0.1
max_seq_length	2048
dtype	None
load_in_4bit	True

In Table II, The FastLanguageModel object provides get_peft_model attribute settings, which can be set parameters for customization, such as setting the number of attention heads, target module, dropout rate, LoRa alpha, and more. Using checkpointing gradients as well as other better techniques demonstrates unslotting's ability to maximize model performance [17].

TABLE II. FASTLANGUAGEMODEL.GET_PEFT_MODEL CONFIGURATION

Parameter	Value
r	16
target_modules	["q_proj", "k_proj", "v_proj", "o_proj", "gate_proj", "up_proj", "down_proj",],
lora_alpha	16
lora_dropout	0
bias	None
use_gradient_checkpointing	"unsloth"
random_state	3407
use_rslora	False
loftq_config	None

Next, as summarized in Tables III and IV, is to initialize the supervised fine-tuning trainer, which helps the fine-tuning process. It includes initializing the model and the dataset that will be refined, the tokenizer, learning speed, maximum steps, weight reduction, and optimization [17].

4) USK Mistral 7B Approach Model with RAG Method

At this stage, the approach was carried out on the Mistral 7B USK using the RAG method [18]. This method aims to overcome the limitations of generative AI in finding information outside the training corpus to avoid the Mistral 7B USK model producing hallucinations in answering the given questions.

TABLE III. SFTTRAINER CONFIGURATION

Parameter	Value
model	model
tokenizer	tokenizer
train_dataset	dataset
dataset_text_field	"text"
max_seq_length	2048
dataset_num_proc	2
packing	False

TABLE IV. TRAININGARGUMENTS CONFIGURATION [17]

Parameter	Value
per_device_train_batch_size	2
gradient_accumulation_steps	4
warmup_steps	5
max_steps	60
learning_rate	2e-4
fp16	not is_bfloat16_supported()
bf16	is_bfloat16_supported()
logging_steps	1
optim	"adamw_8bit"
weight_decay	0.01
lr_scheduler_type	"linear"
seed	3407
output_dir	"outputs"

In the RAG method, the data is an external document containing information on the USK academic system and is stored in PDF format. The information data set is then embedded to convert text into vectors stored in a database vector. The database vector used in this study is FAISS. The model to be used at this stage is a model that has been quantized into GPT-Generated Unified Format (GGUF) [19] so that it is possible to use the CPU when running the LLM by moving some of its layers to the GPU so that it can accelerate the model's performance in generating text. The query process on the model in retrieving the most relevant context of the user command with the RAG method is shown in Fig. 2.

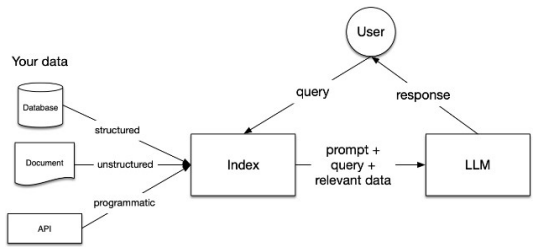

Fig. 2. The RAG pipeline, during the query phase, takes the most relevant context from the user's commands, passing them to the Large Language Model [20].

5) Model Performance Evaluation

The metric used to evaluate the LLM's performance in this study is the Recall-Oriented Understudy for Gisting Evaluation (Rouge) Metric. Rouge metrics are used to evaluate models on NLP tasks so that they can compare the text summaries generated by the model with the summaries in the references.

The rouge metric is an evaluation metric commonly used in NLP tasks to compare computer-generated text summaries with reference summaries (generated by humans) [21]. Rouge is mainly used to assess the answer tasks of the model. The value of the rouge metric ranges from 0 to 1, where 1 is the highest score, indicates that the computer-generated and reference summaries are highly similar. Rouge-1, rouge-2, and rouge-L will provide a comparison of two summaries with different details [22].

1. Rouge-1

Rouge-1 measures the accuracy of unigrams (single words) that overlaps between the text generated by the model and the reference text (Man-made).

$$Rouge-1(Recall) = \frac{unigram\ matches}{unigram\ in\ reference} \quad (1)$$

$$Rouge-1(Precision) = \frac{unigram\ matches}{unigram\ in\ output} \quad (2)$$

$$Rouge-1(F1) = 2x\ \frac{Precision\ x\ Recall}{Precision + Recall} \quad (3)$$

2. Rouge-2

Rouge-2 measure the accuracy of overlapping bigrams between the text generated by the model and the reference text (man-made). The rouge-2 formula is the same as rouge-1, but the words used are bigrams, not unigrams. Bigrams compensate for the problem of the position of the word rouge-1 to some extent.

3. Rouge-L

Unlike rouge-1 and rouge-2, rouge-L does not look into unigrams or bigrams but conforms to LCS (Longest Common Subsequence) or the longest word sequence in the reference text and the text generated by the model.

$$Rouge-L(Recall) = \frac{Length\ of\ LCS}{unigram\ in\ reference} \quad (4)$$

$$Rouge-L(Precision) = \frac{Length\ of\ LCS}{unigram\ in\ output} \quad (5)$$

$$Rouge-L(F1) = 2x\ \frac{Precision\ x\ Recall}{Precision + Recall} \quad (6)$$

III. RESULTS AND DISCUSSION

The output of this study is a web-based chatbot application utilizing the LLM Mistral 7B, which can be used as an alternative tool to help prospective new students get information. This chatbot was developed using the USK dataset, which summarizes LLM texts to generate information about the lecture system and new student admissions at USK. Several stages were carried out in utilizing and developing

LLM, such as collecting data related to lecture system information and new student admissions at USK. This data was collected during the preprocessing stage by converting the raw data into csv and pdf formats. After the preprocessing stage, fine-tuning was done on the Mistral 7B model. RAG was carried out using the concept of embedding to make it easier to manage data. The RAG aims to overcome generative AI's limitations because whenever a question requires information outside the LLM training corpus, it will result in hallucinations, inaccuracies, or distortions in the generated text. The next stage is to test and evaluate the generated text using the Rouge method to compare the text generated by the model with the summary of the provided references. The last stage is creating a UI or web interface accessible to users..

A. Testing Results and Evaluation of Inference Results

At this stage, the chatbot was tested by asking questions. The resulting text inference results were then calculated using the Rouge score method. Using the Rouge score, responses from the chatbot can be evaluated to see quantitative similarities between references and responses generated by the chatbot. The results of the rouge calculation score on the model are shown in Table V.

TABLE V. SCORE VALUE OF ROUGE

Method	Number of Questions	ROUGE Score		
		R-1	R-2	R-L
Fine-tuning	20/20	1.0	1.0	1.0
RAG	15/56	>0.5	>0.5	>0.5

B. Rouge Score Categories

The value of the Rouge Score category varies depending on the summary task and metrics. The category's rouge-1 score is very good, with a score of around 0.5. A score above 0.5 is considered good, and 0.4 to 0.5 is categorized as moderate. On rouge-2 with a score above 0.4 is categorized as good, and 0.2 to 0.4 is categorized as moderate.

The score on the Rouge-L Score is categorized as good at around 0.4, and is categorized as low with a score of around 0.3 to 0.4. Although the Rouge score helps in assessing the quality of the response, it does not assess the semantic or syntactic quality so it requires assessment with other metrics and human evaluation for a thorough assessment [23].

TABLE VI. ROUGE METRIC VALUE CATEGORY [23]

ROUGE Metric	Excellent	Good	Moderate
ROUGE-1	0.5+	>0.5	0.4-0.5
ROUGE-2	-	>0.4	0.2-0.4
ROUGE-L	-	~0.4	0.3-0.4

C. Calculating Resource Evaluation

The researcher conducted a time test on the Mistral 7B model in this study, with fine-tuning and inference testing. Researchers used a single NVIDIA Tesla T4 GPU graphics card in Google Colab for inference and fine-tuning. Based on the fine-tuning time test and using 20 datasets, as shown in Table VII, the researcher concluded that the fine-tuning approach with the unsloth library is effective and efficient and requires minimal time. The inference time test shows that the USK Mistral 7B does not require excessive computing power during the inference process, making it an energy-efficient and efficient system to respond to user inquiries only takes 5-6 minutes to respond to user inquiries. Hence, this model will be an efficient system for handling information related to USK's academic administration in the future.

TABLE VII. TIME COUNT ON MODEL WHILE FINE-TUNING AND RUNNING RAG

Model	Fine-tuning *time* (hour)	RAG Time (minute)
USK Mistral 7B	2	5-6

D. Result Analysis

1) Training Data Problems

The factors that affect LLM so that it produces hallucinatory answers are based on the nature of the model training data. LLMs, such as the Mistral 7B have undergone extensive unattended training using large and diverse datasets from a variety of sources. Ensuring factual truth in data is a challenge. When the model learns to generate a response in the form of text, it can find and replicate factual inaccuracies in the training data. This learning can lead to scenarios where the model cannot distinguish between truth and fiction so that the model produces outputs that do not correspond to facts or logical reasoning. At first, LLMs are trained using a set of data sourced from the internet, this information can be biased or incorrect data. This misinformation can extend to the model's output, as the model lacks the ability to distinguish between accurate and inaccurate data.

2) Reduces Hallucinations

Efforts to reduce hallucinations are essential to maintain the reliability, quality and functionality of LLMs. This primary method of identifying and mitigating errors involves a combination of other metrics and in-depth human evaluation. Among them are such as.

- Metrics to assess linguistic qualities such as metrics ROUGE and BLEU.
- Metrics to assess the validity of a piece of content, which are IE-based, QA-based, and NLI-based.
- FactScore is used in checking the accuracy level of facts.

3) Retrieval-Augmented Generation (RAG) Method

Innovative approaches such as SelfCheckGPT can identify hallucinations by evaluating the consistency of the various responses that the model provides in answering similar questions. In addition, methods such as *chain-of-thought prompting* and *Retrieval-Augmented Generation* (RAG) are continuously developed to optimize the model's proficiency in generating appropriate and relevant answers.

4) The Influence of GPUs in LLM Implementation

GPUs play an essential role in running LLMs. Dedicated GPUs with high VRAM can significantly accelerate the computation required by the model. In this study, the GPU used is the "NVIDIA Tesla T4 GPU," available on Google Colab for free. The results of testing with this GPU using the RAG method take 4-5 minutes to generate responses to the questions.

IV. Conclusion

Fine-tuning the Mistral 7B model to USK Mistral 7B requires enormous amounts of data to produce a better LLM in answering questions related to the lecture system and new student admissions at USK. It takes a long time, approximately 2 hours, to get a fine-tuning model with a dataset of 20 question-and-answer data.

The RAG method overcomes the limitations of generative AI when it requires information outside the LLM training corpus. This method will avoid LLMs that generate inaccurate text, hallucinations, or distortions when answering given questions. This RAG method can generate answers faster because it uses external data. The RAG method allows the model to avoid limitations on generative AI models. The response generated by the RAG method can produce a pretty good answer based on the rouge score that has been tested.

Based on open-source engineering metrics and assessment of computing resources, USK Mistral 7B has the potential to be applied because it can produce a response with a rouge score >5 with low energy consumption.

LLMs, especially Mistral 7B, have great potential in their application in various fields, such as academic and administrative services. Our study shows that with a small amount of data related to academics and administration at USK for training, USK Mistral 7B can respond well to various questions. The performance of LLMs, such as Mistral 7B, highlights their ability as a powerful tool to assist students in obtaining information.

In the future, this model can also be applied to various fields, such as implementation in faculties with different information and government offices that provide services to the community, such as licensing services. Further research can also be done by testing on other newer models with more advanced features. By exploring this, comparisons with other models are obtained to find the best model for various tasks.

References

[1] S. Mohamadi, G. Mujtaba, N. Le, G. Doretto, and D. A. Adjeroh, "ChatGPT in the Age of Generative AI and Large Language Models: A Concise Survey," pp. 1–60, 2023, [Online]. Available: http://arxiv.org/abs/2307.04251.

[2] H. P. Baker, E. Dwyer, S. Kalidoss, K. Hynes, J. Wolf, and J. A. Strelzow, "ChatGPT's Ability to Assist with Clinical Documentation: A Randomized Controlled Trial," *J. Am. Acad. Orthop. Surg.*, vol. 32, no. 3, pp. 123–129, Feb. 2024, doi: 10.5435/JAAOS-D-23-00474.

[3] L. Loukas, I. Stogiannidis, P. Malakasiotis, and S. Vassos, "Breaking the Bank with ChatGPT: Few-Shot Text Classification for Finance," *FinNLP-Muffin 2023 - Jt. Work. 5th Financ. Technol. Nat. Lang. Process. 2nd Multimodal AI Financ. Forecast. conjunction with IJCAI 2023 - Proc.*, pp. 74–80, 2023.

[4] A. Trozze, T. Davies, and B. Kleinberg, "Large Language Models in Cryptocurrency Securities Cases: Can ChatGPT Replace Lawyers?," pp. 1–49, 2023, [Online]. Available: http://arxiv.org/abs/2308.06032.

[5] Q. Huang, M. Tao, C. Zhang, and Z. An, "Lawyer LLaMA: Enhancing LLMs with Legal Knowledge," 2023.

[6] A. Bhatti, S. Parmar, and S. Lee, "SM70: A Large Language Model for Medical Devices," no. 1, pp. 1–5, 2023.

[7] H. Zhao *et al.*, "Ophtha-LLaMA2: A Large Language Model for Ophthalmology," *ArXiv [Preprint]*, pp. 1–19, 2023, [Online]. Available: http://arxiv.org/abs/2312.04906.

[8] S. Barandoni, F. Chiarello, L. Cascone, and S. Puccio, "Automating Customer Needs Analysis : A Comparative Study of Large Language Models in the Travel Industry," 2024.

[9] V. Jonatan and and A.-A. Igor, "Creation of a Chatbot Based on Natural Language Processing for Whatsapp," *J. Database Manag.*, vol. 3, no. 4, pp. 39–53, 2023, doi: 10.14810/elelij.2023.12402.

[10] data.usk.ac.id, "Data Mahasiswa Daftar Ulang," data.usk.ac.id. Accessed: Apr. 13, 2024. [Online]. Available: https://data.usk.ac.id/mahasiswa-daftar.

[11] T. Fatyanosa, "Fine-Tuning Pre-Trained Transformer-based Language Model," medium.com. Accessed: Jan. 25, 2024. [Online]. Available: https://fatyanosa.medium.com/fine-tuning-pre-trained-transformer-based-language-model-c542af0e7fc1.

[12] R. Saman, "In-Depth Guide to Retrieval-Augmented Generation (RAG) Workflow: From Concepts to Implementation," medium.com. Accessed: Oct. 19, 2024. [Online]. Available: https://medium.com/@saman.rahbar/in-depth-guide-to-retrieval-augmented-generation-rag-workflow-from-concepts-to-implementation-762b412a6d76.

[13] L. Xu, H. Xie, S.-Z. J. Qin, X. Tao, and F. L. Wang, "Parameter-Efficient Fine-Tuning Methods for Pretrained Language Models: A Critical Review and Assessment," pp. 1–20, 2023, [Online]. Available: http://arxiv.org/abs/2312.12148.

[14] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLORA: Efficient Finetuning of Quantized LLMs," *Adv. Neural Inf. Process. Syst.*, vol. 36, 2023.

[15] E. Hu *et al.*, "Lora: Low-Rank Adaptation of Large Language Models," *ICLR 2022 - 10th Int. Conf. Learn. Represent.*, pp. 1–26, 2022.

[16] D. Han-Chen, "Make LLM Fine-tuning 2x faster with Unsloth and ? TRL," https://huggingface.co/blog/unsloth-trl. Accessed: Jun. 30, 2024. [Online]. Available: https://huggingface.co/blog/unsloth-trl.

[17] K. V., "Fine-Tuning Large Language Models with Unsloth," medium.com. Accessed: Jul. 30, 2024. [Online]. Available: https://medium.com/@kushalvala/fine-tuning-large-language-models-with-unsloth-380216a76108.

[18] P. Lewis *et al.*, "Retrieval-augmented generation for knowledge-intensive NLP tasks," *Adv. Neural Inf. Process. Syst.*, vol. 2020-Decem, 2020.

[19] M. Grootendorst, "Which Quantization Method is Right for You? (GPTQ vs. GGUF vs. AWQ)," maartengrootendorst.com. Accessed: Feb. 29, 2024. [Online]. Available: https://www.maartengrootendorst.com/blog/quantization/.

[20] docs.llamaindex.ai, "High-Level Concepts," docs.llamaindex.ai. Accessed: Jul. 28, 2024. [Online]. Available: https://docs.llamaindex.ai/en/v0.10.17/getting_started/concepts.html.

[21] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries Chin-Yew," pp. 74–81, 2004, doi: 10.1253/jcj.34.1213.

[22] M. U. Amanat, "LLM evaluation with Rouge," medium.com. Accessed: Apr. 24, 2024. [Online]. Available: https://medium.com/@MUmarAmanat/llm-evaluation-with-rouge-0ebf6cf2aed4.

[23] S. M. Walker II, "What is the ROUGE Score (Recall-Oriented Understudy for Gisting Evaluation)?," klu.ai. Accessed: Jul. 30, 2024. [Online]. Available: https://klu.ai/glossary/rouge-score.