IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 24 January 2025, accepted 11 February 2025, date of publication 14 February 2025, date of current version 20 February 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3542334

RESEARCH ARTICLE

Unveiling the Power of Large Language Models: A Comparative Study of Retrieval-Augmented Generation, Fine-Tuning, and Their Synergistic Fusion for Enhanced Performance

GÜLSÜM BUDAKOGLU AND HAKAN EMEKCI
Graduate School, Applied Data Science, TED University, 06420 Ankara, Türkiye
Corresponding author: Gülsüm Budakoglu (gulsum.budakoglu@tedu.edu.tr)

ABSTRACT Large-language model optimization for a particular application is crucial and challenging in natural language processing. This study compares two salient techniques for retrieve-augmented generation (RAG) and fine-tuning along with a new hybrid method that combines both. In this study, we investigate the effectiveness of various methods using the Stanford Question Answering Dataset (SQuAD), Microsoft Machine Reading Comprehension (MS MARCO) and SQL CREATE TABLE statements. RAG is used because it enriches the model responses with external data without much computational load during the inference. Fine-tuning updates the model parameters to improve the contextual accuracy. Our hybrid model balances the accuracy and efficiency of the two techniques. While fine-tuning entails semantic precision, RAG is more resource efficient. The hybrid approach while it may not offer surpassing results over fine-tuning-offers a balanced solution in scenarios where the application demands both efficiency and accuracy. These findings represent the trade-off involved in LLM optimization and offers a scope for further studies and practical applications.

INDEX TERMS Large language models (LLMs), retrieval-augmented generation (RAG), fine-tuning, hybrid models, performance optimization.

I. INTRODUCTION

Interest in large language models (LLMs) is on the rise, a fact that indicates a technological turning point and thus heralds new uses and demands from developers and institutions to tap into the potential of such advanced systems. In most cases, their performance remains far from expectations when pre-trained LLMs are applied directly. These limitations have been documented in the literature, as represented by A Primer in BERTology [1], in which the operational subtleties and limitations of BERT are elaborated as a seminal LLM. Furthermore, the work that Language Models are Few-Shot Learners [2] in "Advances in Neural Information Processing Systems" illuminates GPT-3's optimization hurdles and posits strategies for refining LLMs, thereby underpinning the criticality of enhancement methodologies. This gap between the current capabilities and the desired performance highlights the urgent need for robust optimization methods. The choice between implementing the Retrieval-Augmented Generation (RAG) framework and proceeding with model fine-tuning is crucial for advancing LLM applications. This paper discusses the optimization methods used for LLMs to perform better in special tasks. It weighs the merits of RAG against fine-tuning methods. The RAG algorithm boosts LLMs by incorporating external data, whereas fine-tuning refines their efficiency on particular tasks through targeted data training. This study also considered the potential of hybrid models that merge the contextual advantage of the RAG score with the task-specific accuracy of fine-tuning. We evaluated these strategies on the Stanford Question Answering Dataset (SQuAD) [3], Microsoft Machine Reading Comprehension (MS MARCO) [4], and SQL CREATE

The associate editor coordinating the review of this manuscript and approving it for publication was Shadi Alawneh.

TABLE statements [5] to determine the most effective approach for advancing LLM applications, aiming to develop an optimized enhancement pipeline for these sophisticated models. With the aim of identifying the most effective approach for enhancing LLMs in terms of both accuracy and efficiency. The study also considered a hybrid model that combines the strengths of both RAG and fine-tuning to balance computational efficiency with high semantic accuracy.

II. RELATED WORK

The advent of Large Language Models (LLMs) has marked a revolutionary shift in the field of natural language processing, providing unprecedented capabilities for generating human-like text, understanding language nuances, and performing a myriad of linguistic tasks. These models, such as generative pre-trained transformers (GPTs) and their successors, have demonstrated their potential in various applications, ranging, from chatbots and content generation to complex problem-solving and decision-making processes. However, despite their impressive abilities, LLMs have limitations. The direct deployment of pre-trained LLMs often results in performance that falls short of the requirements for specific tasks or contexts. This has led to the exploration of methods for enhancing the performance of LLMs. with particular focus on Retrieval-Augmented Generation (RAG) and fine-tuning [6], [7], [8].

The need for performance enhancement of LLMs arises from several factors. First, the static nature of pre-trained models means that they do not adapt to new information or specific domain knowledge post deployment. This introduces the prospect of outputs that may be outdated or irrelevant to the existing context. While LLMs can have bias erasing from systemic sources in their training data, this leaves many applications that are valid across different situations. Finally, the process of training these models and running them at scale is colossally expensive; methods need to be developed that do this more efficiently or else it would require one not to sacrifice performance or to retrain it too often [9], [10].

The two of the most striking approaches to counteract these issues have been retrieval-augmented generation and fine-tuning. RAG aims to enhance the performance of LLMs by embedding a retrieval mechanism into the model; thus, the model can dynamically access knowledge bases. This method gives more updated information and facts to the responses of the model, which increases the level of accuracy of the outputs of the model. On the other hand, fine- tuning consists of extra training of a pre-trained LLM on a smaller dataset of the task at hand. This procedure changes the model's parameters and further tunes them to be closer to the specific needs of the target application, thereby improving the model's performance within that task [11], [12].

RAG represents an important advance of the state-of-the-art for LLMs, leveraging strengths from both advanced retrieval systems and generative models. This method improves model reliability and interpretability while reducing inaccuracies and data misrepresentations, and is particularly useful in dynamic domains where access to the most recent information plays a vital role. On the other hand, fine-tuning enables LLM to be adjusted for tasks and enhances its ability to build contextually relevant, accurate responses. Although both methods offer considerable benefits, they also involve a host of challenges and limitations, including integrating external knowledge and the potential for catastrophic forgetting during fine-tuning [13]. Additionally, fine-tuning embedding models represents a widely adopted and effective strategy for enhancing the capacity of these models. An insightful study by the work that "What Happens to Embeddings During Fine-tuning?" [14] demonstrated the potential of fine-tuning. This study reveals that despite the substantial modifications introduced through this process, it does not result in catastrophic forgetting of linguistic phenomena. This finding underscores the robustness and power of fine-tuning as a mechanism for the model improvement. However, much of the existing understanding of fine-tuning stems largely from empirical observations of the model behavior. Notably, fine-tuned transformers show state-of-the-art performance, but can also learn shallow shortcuts, heuristics, and biases, as evidenced in works that are "Annotation Artifacts in Natural Language Inference Data" and "What do you learn from context?" [15], [16], [17]. The complexities of such behavior require a nuanced look into the process and trade-offs that arise with fine-tuning.

Several studies have investigated the complexity of fine-tuning. Initial investigations into fine-tuned encoders showed remarkable performance on benchmark suites such as GLUE [18], with surprising sample efficiency. However, deeper behavioral analyses using challenge sets [16], [17] have revealed limitations in generalizing out-of-domain data and across syntactic perturbations. This emphasizes the need for a nuanced understanding the impact of fine-tuning on the model generalization. In addition, fine-tuning pre-trained contextual word embedding models, exemplified by BERT [19], has become the established norm for downstream tasks.

Another study which "To Tune or Not to Tune?" [20] conducted an exhaustive examination, scrutinizing the impacts of fine-tuning on diagnostic classifiers across different layers. Also, "What do you learn from context?" [17] primarily centered on correlating representations with fMRI data and delved into fine-tuning using representational similarity analysis (RSA). Collectively, these studies contribute to a more nuanced understanding of the implications and hurdles associated with fine-tuning the embedding models.

The advent of Large Language Models (LLMs) has heralded a new era in natural language processing, offering unparalleled capabilities for generating human-like text. Among the techniques for augmenting LLMs, fine-tuning and retrieval-augmented generation (RAG) stand out for their distinct approaches to enhance model performance. Fine-tuning, as discussed in studies such as EW-Tune and

full parameter fine-tuning for LLMs with limited resources, involves adjusting the weight of a pre-trained model on a specific dataset to improve its performance on related tasks, highlighting the balance between model adaptability and computational efficiency [21], [22].

On the other hand, RAG introduces a novel paradigm by combining the retrieval of relevant documents from a vast corpus with the generative prowess of LLMs, thereby enabling the model to produce responses informed by external knowledge, as exemplified by ''Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks'' [9]. This further enriches the model output in terms of increasing the range of information derived from it and at the same time answering the question of how to keep the model's knowledge up to date. The combination of these techniques triggers a series of interesting questions regarding the relative effectiveness, efficiency, and domain applicability, suggesting that future research may explore the optimal methods of LLM augmentation.

The research that is ''Task-aware Retrieval with Instruction'' [23] explained a pioneering undertaking termed ''retrieval with instructions,'' which aimed explicitly to explain a user's search intent. This novel approach requires equipment with a natural language elucidation (instruction) with the search query Thereby mandating retrieval systems to discern documents is not only correspond to the query but also resonate with the equipment instructions. Another study ''Unsupervised Dense Information Retrieval with Contrastive Learning'' [24] explored dense retrievers in information retrieval. This supports their limitations in new applications that lack training data. By employing contrastive learning, this study achieved robust unsupervised performance across various retrieval scenarios. The approach also proved effective in multilingual retrievals with limited training data, because it shows notable cross lingual transferability, even between different scripts. The findings of this study expand our understanding of dense retrievers and demonstrate their adaptability and effectiveness in scenarios where training data are scarce, highlighting their potential for diverse retrieval applications. Another important study on RAG is the ERAGent paper [25]. ERAGent is a more matured form of RAG technology that reveals how strategic enhancements improve performance and user experience in tandem. Equipped with strong personalization capabilities, this agent is a practical solution for real-world systems to make their responses user-centric. Capable of incremental learning through an Experiential Learner module, ERAGent continuously adapts to evolving knowledge and user interactions to remains relevant and efficient in its applications. This work provides insights into overcoming some of the existing limitations in retrieval-augmented systems, and thus lays the foundation for future research in personalized AI applications.

A comparison between the fine-tuning and RAG methodologies for the augmentation of LLMs brings into view a spectrum of different strategies, each with a fit to specific requirements and constraints. Fine-tuning is also explained in more detail at such works as ''The Janus Interface'' and ''LLM-Adapters,'' while such a method increases model biases and privacy risks yet allows for an adaptable and resource-efficient way of specializing models for niche applications [26], [27]. These points to several discriminating trade-offs between model customization and the ethical concerns of amplifying preexisting data biases. Moreover, the full potentiality of RAG is demonstrated in the foundational paper, not only in benchmarking efforts but also in automated evaluation frameworks such as RAGAS and ARES, for critical assessment of its noise handling capability, integration of information from disparate documents, and reconstruction of irrelevant data, thereby enhancing model reliability and accuracy [6], [9], [28].

Both approaches reflect the shifting landscape of augmenting LLMs, and the touchstone between the fine-tuning approach and the RAG approaches is in the desired balance between computational efficiency, ethical use of AI, and the need for updated information retrieval with a high degree of accuracy. Continuous development tools such as Federated Scope-LLM and ASPEN indicate a future in which fine-tuning and RAG will also become increasingly efficient and less resource-intensive, thus finding broader applications in areas ranging from medical research to ustomer service [11], [29]. This dichotomy not only underscores the versatility of LLMs but also highlights the critical importance of continuous innovations in methodologies for their augmentation, ensuring that they remain effective tools in the expanding domain of artificial intelligence. Another important comparative study [30] compared the effectiveness of fine-tuning and RAG in the context of developing AI-driven knowledge-based systems. They also explored the combination of NLP and information retrieval techniques to enhance performance. In addition, Blended RAG [31] focused on improving the retrieval accuracy of RAG systems by integrating semantic searches and hybrid query-based retrievers, providing a comparative analysis with standard RAG setups. Another study [32] offers a comprehensive comparison of fine-tuning and RAG methods, particularly for less popular knowledge areas. This study assessed the advantages of combining these methods to enhance performance.

Our approach builds upon insights from previous studies like comparing Retrieval-Augmented Generation (RAG) and fine-tuning, such as the pipeline and tradeoffs discussed in 'RAG vs Fine-tuning: Pipelines, Tradeoffs, and a Case Study on Agriculture' [33].'' This study serves as a valuable reference for understanding the tradeoffs between RAG and fine-tuning, particularly in real-world scenarios that require domain-specific knowledge. Another study is ''Fine-Tuning or Retrieval? Comparing Knowledge Injection in LLMs'' [10]. According to their study, RAG is stronger in the injection of real-time knowledge, particularly in applications with very high updating frequencies or dynamic

knowledge databases. Fine-tuning is much worse when encoding new information, and it can only achieve certain benefits through heavy engineering such as presenting facts in various ways. Additionally, the paper ''Fine-Tuning or Fine-Failing? Debunking Performance Myths in Large Language Models'' [34]. critically reviewed the fine-tuning of LLMs within RAG pipelines to improve the accuracy and contextual understanding across diverse domains. While fine-tuning is often credited with enhancing performance for domain-specific queries in standalone applications of LLMs, this research has come up with a different result when applied to RAG systems, where fine-tuned models underperform compared to their baseline variants. These findings challenge the assumption that fine-tuning universally improves model performance by underlining domain-specific tasks in which careful validation and testing are performed. The present work has shown that fine-tuned LLM integration in RAG pipelines is not without its pitfalls and calls for a much more rigorous framework of evaluation with respect to optimality. In addition,the paper ''Should We Fine-Tune or RAG? Evaluating Different Techniques to Adapt LLMs for Dialogue'' [35] presents an investigation of adaptation methods across dialogue types for response generation with the Llama-2 and Mistral models. This study compares in-context learning, fine-tuning, and knowledge integration via RAG and gold knowledge. It is concluded that no single best approach exists, since effectiveness varies across models and dialogue types, and that human evaluations complement automated metrics in obtaining accurate assessments.

The size and detailed characteristics of the dataset to be used will be discussed in the following section. This analysis provided a broad overview of the data used in this study.

III. DATASET

Each model was evaluated on three datasets: the Stanford Question Answering Dataset (SQuAD) [3], Microsoft Machine Reading Comprehension (MS MARCO) [4], and SQL CREATE TABLE statements [5]. The selection was deliberately diverse to ensure that the model performance was comprehensively tested and allows the derivation of robust and generalizable findings across different data types and tasks.

The Stanford Question Answering Dataset (SQuAD) was a data asset for this research. The dataset is an articulated form aimed at developing reading comprehension models. SQuAD is a collection of context; question; and answer triplets developed by crowd workers and extracted from a wide spectrum of Wikipedia articles. Each triplet in the dataset has an associated answer that is, textually represented as a segment or span extracted from a related passage. A distinct aspect of the SQuAD is the inclusion of unanswerable questions. These questions introduce an advanced degree of complexity and simulate real-world scenarios, in which definitive answers may not be available.

The Stanford Question Answering Dataset (SQuAD) is import for machine comprehension. It was designed to evaluate the ability of a model to understand and process human language through the lens of Wikipedia articles. This dataset contains over 100,000 questions crafted by crowd workers based on a selection of Wikipedia articles, with each question's answer being a text segment extracted directly from the corresponding passage. In addition, the creation of the SQuAD was motivated by the necessity of a large-scale dataset. It also maintains high quality while being sufficiently substantial to train data-intensive models for natural language processing (NLP) tasks.

The structure of the SQuAD involves an important three-stage process. It begins with the curation of passages from top-ranked Wikipedia articles using internal PageRank scores. This is followed by the crowdsourcing of question-answer pairs and concludes with the collection of additional answers for a robust evaluation. The dataset contained unique questions associated with multiple context columns. Each context column included different question and answer pairs. This revised sentence conveys that each unique question has multiple context columns, which contain different pairs of questions and answers. This methodological approach ensures wide coverage of topics and a rich variety of question types and answer formats. These formats range from information and definitions to complex reasoning questions. The architecture of the dataset necessitates that models have not only linguistic understanding but also the capability to perform inferential reasoning based on the context provided by the scenarios. A single context may appear multiple times within the dataset, and for each instance in the same context, there may be different questions with corresponding answers. An example row illustrating the structure of the dataset is in Figure 1.

{ "Title": "Architecture",
"Context": "Architecturally, the school has a Catholic character. Atop the Main Building's gold dome is a golden statue of the Virgin Mary. Immediately in front of the Main Building and facing it, is a copper statue of Christ with arms upraised with the legend 'Venite Ad Me Omnes'. Next to the Main Building is the Basilica of the Sacred Heart. Immediately behind the basilica is the Grotto, a Marian place of prayer and reflection. It is a replica of the grotto at Lourdes, France where the Virgin Mary reputedly appeared
to Saint Bernadette Soubirous in 1858.",
"Question":"Which artist created the mural on the Theodore M. Hesburgh Library?",
"Answer": { "text": ["Millard Sheets"], "answer start": [344] } }

FIGURE 1. Example of dataset's structure.

However, MS MARCO is among the most influential datasets created to challenge the performance of models in reading comprehension and answering questions. It consisted of 1,010,916 anonymized questions from Bing's search query logs, each with human-generated answers, 182,669 fully rewritten human answers, and 8,841,823 passages based on 3,563,535 web documents through Bing to produce a natural language answer.

What sets the MS MARCO dataset apart, however, is that each question can have multiple answers or no answer at all, which makes the tasks it supports both complex and realistic. This dataset allows three different tasks with increasing difficulty, namely: i) whether an answer to a question is available in a given set of context passages along

with synthesizing a human-like answer, and ii) generating a well-formed answer ranking a set of retrieved passages given a question. It differs from other publicly available datasets on machine reading comprehension in terms of scale, origin, and diversity. Therefore, this dataset is a valuable benchmark for measuring the capability of models to handle realistic, complex, and diverse question-answering scenarios. An example row illustrating the dataset's structure of MS MARCO is shown in Figure 2.

{"passage": "Since 2007, the RBA's outstanding reputation has been affected by the 'Secrecy' or NPA scandal. These RBA subsidiaries were involved in bribing overseas officials so that Australia might win lucrative note-printing contracts. The assets of the bank include the gold and foreign exchange reserves of Australia, which is estimated to have a net worth of A$101 billion. Nearly 94% of the RBA's employees work at its headquarters in Sydney, New South Wales and at the Business Resumption Site.,...."
"query": "what is rbi",
"answer": "Results-Based Accountability is a disciplined way of thinking and taking action that communities can use to improve the lives of children, youth, families, adults and the community as a whole."}

FIGURE 2. Example of dataset's structure.

The SQL CREATE TABLE Statements dataset has a different format compared to all other datasets presented here because it is related to structured data. It comprises 78,577 examples of natural language queries, SQL CREATE TABLE statements that correspond to these queries, and SQL queries answering questions using the CREATE TABLE statement as the context.

This dataset is designed with text-to-SQL LLMs in mind to avoid common issues observed in the models trained with standard text-to-SQL datasets, such as the hallucination of column and table names. Explicit table names, column names, and their respective data types can often be copied directly from the CREATE TABLE statements of different DBMSs. This provides a better context to the models for grounding, and the dataset offers only the CREATE TABLE statement and, not necessarily the actual rows of data. This saves tokens and avoids the exposure of private, sensitive, or proprietary information; hence, it is a practical and efficient resource for training and evaluating models. An example row illustrating the structure of the dataset's shown in Figure 3.

{"context": "CREATE TABLE city (Status VAR CHAR)"
"Question": "How many different statuses do cities have?",
"answer": "SELECT COUNT(DISTINCT Status) FROM city"}

FIGURE 3. Example of dataset's structure.

The datasets employed in this study were partitioned into three distinct sets: training, validation, and test to facilitate the development and evaluation of the model. Specifically, the datasets were distributed as follows and are shown in Figure 4.

- Fine-tuning: 10,000 context-question-answer pairs are designated for training, while 2,500 pairs are allocated for both validation and testing.
- Retriever-Augmented Generation (RAG): This approach utilizes a vector database derived from 10,000 pairs, with an additional 2,500 pairs set aside for testing. There is no validation dataset for the RAG because we do not have a training part in the RAG.
- RAG with Fine-Tuning: This combined strategy also divides 10,000 pairs for training, 2,500 for validation, and 2,500 for testing.

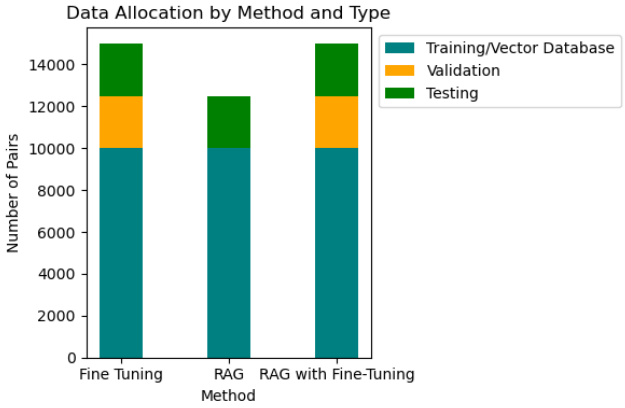

FIGURE 4. Size of data used for Fine-Tuning, RAG, and Fine-Tuning + RAG.

Owing to the large size of the dataset and the limitations of available hardware resources, the training, testing, and validation of results is challenging. Therefore, a subset of 12,500 unique context- question-answer triplets was been selected from the complete datasets for this study. For this study, the sample dataset feature topics were chosen randomly, ensuring a variety of subjects in the training, testing, and validation datasets. Additionally, all topics follow the same structural format. This dataset was uniformly applied to fine-tuning, Retrieval-Augmented Generation (RAG), and hybrid methods. In the following section presents a detailed discussion of the experiments conducted.

IV. EXPERIMENTS

This study conducted an empirical analysis to determine the most effective method for enhancing the performance of large language models (LLMs). It compares three strategies: retrieval- augmented generation (RAG) framework, traditional fine-tuning, and a combination of both. These strategies were evaluated using three distinct experimental setups, as shown in Figure 5. The ''BAAI/bge-small-en'' pre-trained embedding model (BAAI/Bge-Small-En · Hugging Face, n.d.) serves as the base model for embeddings, where the GPT-3.5 Turbo model was used to generate responses to user queries.

A. EXPERIMENT 1: FINE-TUNING

Approach 1 begins with the base LM, which undergoes fine-tuning before being applied to a language model for linguistics (LLM) fine-tuning process. This sequential fine-tuning aims to enhance the ability of the LM to generate responses to user queries, emphasizing linguistic refinement and contextual accuracy. There is a process for the fine-tuning

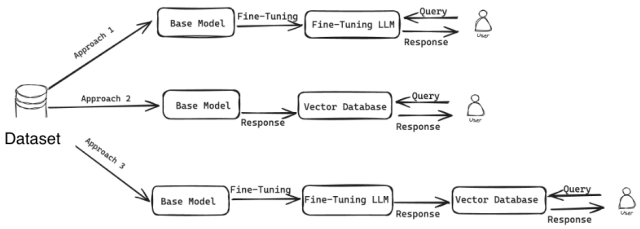

FIGURE 5. Experimental Setup. Approach 1: Fine-Tuning; Approach 2: RAG; Approach 3: Fine-Tuning + RAG.

part and there are details about the fine-tuning process step by step shown in Figure 6.

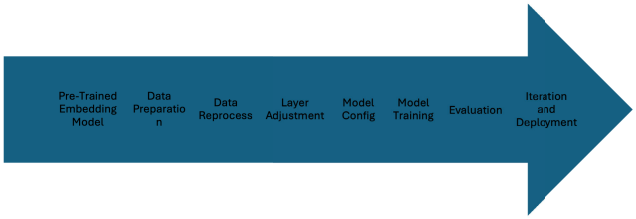

FIGURE 6. Fine-tuning process.

1) SELECTION OF AN INITIAL PRE-TRAINING MODEL

In this research aimed at improving semantic search capabilities, the choice fell on the "BAAI/bge-small-en" (BAAI/Bge-Small-En · Hugging Face, n.d.) pre-trained embedding model as a base model. This demonstrates the efficiency in processing and understanding natural language. This model was initially pre-trained on a comprehensive dataset encompassing a wide range of topics. It can be adept at grasping the subtleties of the language necessary for an effective semantic search. Moreover, the "BAAI/bge-small-en" embedding model was selected because of its exceptional balance between performance and computational efficiency. Its architecture, designed to handle complex NLP tasks, has shown superior performance in retrieval-augmented generation systems, such as the project's focus on semantic searches. Furthermore, its proven capability to process queries and return relevant results with high accuracy makes it an ideal candidate for fine-tuning.

2) LAYER ADJUSTMENT AND MODEL CONFIGURATION

In the domain of natural language processing, the application of a full fine-tuning embedding model to LLMs involves comprehensive layer adjustments to tailor the entire model towards the specific linguistic characteristics of a given task. This approach is especially useful when transitioning from the granular level of word embeddings, typical of transformer models such as BERT and RoBERTa [19], [36] to the more nuanced demands of sentence-level comprehension required for tasks such as semantic search. To overcome this complexity, our methodology leverages the advancements introduced in [37] through Sentence-BERT (SBERT). SBERT refines transformer architecture using Siamese and triplet networks. This architecture supports the efficient generation of sentence embeddings that can be rapidly compared using cosine similarity for semantic congruence. To integrate SBERT within our framework, we specifically adjust the pooling operations of the model to enhance the representation of sentence embeddings. We ensured that they encapsulated a comprehensive semantic spectrum suitable for our targeted semantic search task within an extensive digital library corpus. Our empirical evaluations indicate that these adjustments yield substantial improvements in retrieval accuracy and relevance. Our evaluations also underscored the efficacy of full fine-tuning in optimizing sentence-level semantic performance. Furthermore, the training parameters were selected to harmonize the computational efficiency with memory constraints. This is critical for deployment in environments with limited computational resources. A batch size of 16 represented a good balance. It allows reasonable data because it manages the memory overhead. The model was trained on the dataset for five epochs to ensure sufficient exposure to linguistic variations in the corpus. Warm-up steps were computed using the epoch size, which gradually increased the learning rate in the initial phase of training to facilitate smoother convergence. Regular evaluation every 50 steps of training was performed, as instructed by the evaluation step parameters. This gave us insight into the model's learning trajectory and thus enabled quick changes to be made to optimize the training regimen. The described configuration of the model and its parameters highlight our commitment to precision and efficiency in adapting LLMs to specialized tasks. This provides a strong foundation for both research and practical applications in natural language processing.

3) MODEL TRAINING

The embedding model is fine-tuned in a systematic process involving careful tuning of parameters and monitoring of the model's performance. In this project, fine-tuning begins by initializing weights from the preselected "BAAI/bge-small-en" embedding model. This was designed to provide a contextually rich embedding. The training data consists of labeled examples curated specifically to represent a wide range of linguistic constructs seen in semantic search queries. In preparation for training, the data were pre-processed to arrange the model's input requirements. It is essential to ensure that each input token is accompanied by the appropriate contextual tags and segment identifiers. Throughout the training process, the model is exposed to the data in batches of 16 samples that batch size allows for sufficient granularity in weight adjustments without exceeding our computational capacity. We designated five epochs for the model to learn iteratively from the data. This method offers a duration that balances the need for thorough learning with the risks associated with overfitting from excessive exposure. Moreover, the warm-up steps accounted for 10% of the

total number of training iterations. Gradually acclimating the model to the learning process reduces the likelihood of destabilizing the training dynamics. Performance evaluation was systematically combined with the training regimen at intervals of 50 steps. During each step, the model predictions were compared against a validation set to assess the accuracy, loss, and other relevant metrics. These evaluations serve as checkpoints to observe the progress of the model and inform any necessary adjustments in real-time. These metrics also ensure that the training remains on course toward optimal performance.

4) PERFORMANCE EVALUATION

In the assessment of our embedding models for information retrieval, key performance indicators such as the Mean Reciprocal Rank (MRR), Recall, and Normalized Discounted Cumulative Gain (NDCG) are computed to provide a multifaceted view of model effectiveness. These evaluation metrics harness both cosine similarity and dot product scoring functions to calculate the similarity scores between query and document embeddings. By processing these scores, The evaluator ranked the documents, enabling it to deduce performance measures across various dimensions. The precision of this approach lies not only in the conventional metrics of accuracy, precision, and recall but also in nuanced insights. They also offer into the embedding model's ability to distribute and retrieve relevant information. Moreover, the evaluator's design allows for the adjustment of parameters, such as corpus chunk size and evaluation cut-offs. This design provides researchers with the flexibility to refine the evaluation process to suit the contours of their specific datasets and investigative goals.

B. EXPERIMENT 2: RAG

Approach 2 adopts a direct strategy, in which the base LM processes queries using a pre-established vector database, bypassing additional fine-tuning to provide immediate responses. The structure and details of RAG process are shown in Figure 7.

FIGURE 7. RAG process.

1) CONSTRUCTING VECTOR DATABASE IN RAG

Vector databases engineered for high-efficiency retrieval operations on vector data represent an innovative solution to the retrieval challenges of RAG models. The storage of data in dense vector formats is key to RAG models. Vector databases such as Pinecone, Faiss, and Chrome are optimally positioned to augment the efficiency of knowledge retrieval processes. These databases index the vector representations of knowledge documents. They enable rapid similarity searches to identify the most pertinent documents for any given query. The integration of vector databases into Retrieval-Augmented Generation (RAG) models offers substantial advantages. They enhance knowledge retrieval. These databases are intrinsically designed for vector data, which removes the inefficiencies endemic to the conversion between vector formats and traditional database records. Compared with normal databases, vector databases have faster GPU systems. Approximate search techniques such as locality-sensitive hashing enable swift and accurate approximate nearest-neighbor searches. This search technique is a vital capability given the data size that RAG models contend with. Vector databases support dynamic updating, which permits the insertion and alteration of vector data in real time. Scalability is an important feature of cloud-native vector databases because it facilitates searches across an extensive range of vectors. In this study, we incorporated the Chroma DB (Chroma, n.d.), which is a distinguished open-source vector database. It is commonly used to improve the efficacy of our Retrieval-Augmented Generation (RAG) models. Chroma DB is an open-source embedding database optimized for Large Language Models (LLMs). It offers a suite of functionalities aimed at facilitating the integration of knowledge, facts, and skills into LLM applications. The Chroma DB distinguishes itself by providing a robust infrastructure for storing embeddings and their corresponding metadata, embedding documents and queries, and executing searches across embeddings. This database system is uniquely designed to support the development of LLM applications by making external knowledge sources seamlessly accessible and searchable. A notable feature of Chroma DB that sets it apart from other vector databases, is its operational flexibility. It can be configured to run in memory or a client/server architecture (currently in the alpha stage). It offers users the choice between hosting the database locally on a device or deploying it in a cloud environment for remote access. This skill ensures that Chroma DB can be tailored to suit a range of deployment scenarios, from standalone applications requiring rapid, local access to embeddings to distributed applications that benefit from cloud-based scalability and accessibility. Moreover, mismatched or fragmented context from the retrieved data sometimes generates nonsensical or completely incorrect outputs. Our retrieval mechanism seeks to prevent this by calculating a set of similarity scores between the query and the returned data. Using a threshold higher than the average for these scores would exclude the returns that do not pass that threshold.

2) CONSTRUCTING CONTEXTUAL INTERACTIONS AND KNOWLEDGE SYNTHESIS IN RAG

In the advanced landscape of Natural Language Processing, the integration of Retrieval- Augmented Generation (RAG) into AI systems heralds a transformative approach to user interaction. Such systems, when empowered by RAG technology, demonstrate a heightened accuracy for interpreting and responding to user prompts through contextually aware

actions. In the context of our research, this interactive prowess is exemplified by a validation dataset. It serves as a base for simulating user queries linked to explicit contexts. It provides a robust framework for testing the AI's reactive capabilities. The core of our RAG application is the retrieval mechanism. Also, a vector database is populated with nuanced data. As queries are submitted, the database activates its retrieval function, pinpointing and extracting the relevant information from the dataset. This process ensures that the user's query is met with an informed and context-rich response. The model then undergoes a process of Knowledge Transmutation, beginning with the Concatenation for Contextualization which means melding the user's query with the pertinent data retrieved from the database. This integrative step is crucial for constructing an informed context that sets the stage for the subsequent Generative Synthesis phase. Leveraging the synthesized context, the Generative AI model that is GPT 3.5-Turbo for this study embarks on producing responses. It crafts answers that are not merely reactions to the query but are thoughtful reflections of the comprehensively understood context. Furthermore, The performance of retrieval-augmented generation methods inherently relies on the quality of the external knowledge base. If this knowledge base is incomplete, outdated, or biased, it might result in inaccuracies in the generated outputs and limit the generalizability of the approach. However, in this study, we used a static dataset instead of dynamic data, which mitigates this issue since the dataset remains consistent and controlled.

C. EXPERIMENT 3: COMBINING FINE-TUNING AND RAG

Approach 3 presents a hybrid model combining the methods of Approaches 1 and 2; after the base LM is fine-tuned, similar to Approach 1, it utilizes a vector database for query response generation, similar to Approach 2. This approach is designed to balance the linguistic sophistication of fine-tuning with the responsiveness of a vector database. The integration of Retrieval-Augmented Generation (RAG) and fine-tuning methodologies offers a robust approach to enhancing the performance of Language Models (LLMs). Our strategy is based on the synergistic combination of these two techniques that utilize the fine-tuned embedding model to serve both as a semantic search mechanism within the RAG framework and to produce more contextually relevant embeddings that improve the overall generation quality. As the user interacts with the system, their queries are transformed by the fine-tuned embedding model into vector representations. These vectors, when queried against the database, ensure that the retrieval phase of the RAG system is fed with the most pertinent and semantically relevant context. This enriched context then serves as a foundation upon which the generative model is fine-tuned. That system allows to production of responses that are not just accurate but also contextually nuanced. The research adopts a symbiotic workflow wherein the embedding models operate for both documents and queries. It ensures a consistent and enriched representation throughout the system. This consistency is pivotal for the generative model. It is also iteratively fine-tuned in a feedback loop with the outputs of the RAG. Such an iterative process is instrumental for the continuous refinement of the model, facilitating an evolutionary understanding of context and relevance. The integration of RAG and fine-tuning methodologies thus reflects a committed effort to optimize and customize the model for specific task requirements. The iterative refinement, anchored in the use of fine-tuned embeddings, represents a strategic alignment of retrieval accuracy and generative precision. Ultimately, this integration forms the base on which the superior performance of the LLM is built, enabling it to generate responses that are not only precise but also deeply attuned to the intricacies of the user's input. This methodology stands as a key to the research's commitment to elevating the operational capabilities of LLMs. It demonstrates a significant leap forward in the field of language understanding and response generation.

Moreover, aligning the retrieved context with the model's learned knowledge, managing retrieval latency, and optimizing the synergy between retrieval embeddings and fine-tuned representations were challenging when combining the retrieval-augmented generation with fine-tuning (RAG). This hybrid approach is very precise and fast as fine-tuning is used for domain adaptation with RAG, dynamically providing the information that is missing or changing thus reducing the need for regular retraining. The trade-off is set up in the best way, thus selective retrieval, tuning the retrieval depth, and refining the response fusion mechanisms are exercised to make sure that the speed is kept without losing accuracy.

In the forthcoming section, this paper will examine the evaluation metrics employed to assess the performance outcomes of the experimental results.

V. EVALUATION METRICS

In evaluating text generation models, a range of metrics is used to characterize different aspects of the generated text, including linguistic, syntactic, and semantic dimensions. These metrics are crucial in the verification of the quality and applicability of the models for various tasks, including translation, summarization, and general text generation. To evaluate the linguistic and syntactic integrity of the outputs, well-established metrics, including Cosine Similarity, Bert Score, BLEU (Bilingual Evaluation Understudy), ROUGE (Recall-Oriented Understudy for Gisting Evaluation), and METEOR (Metric for Evaluation of Translation with Explicit ordering), are utilized. Moreover, to ensure the content's relevance and accuracy, metrics specifically designed to assess the precision and factual correctness of the generated text are adopted. These are: Context Precision, Faithfulness, Answer Relevancy, and Answer Correctness. These altogether provide information on the level to which text is contextually appropriate and devoid of inaccuracies or a lack of relevancy. Additionally, this study incorporates

resource utilization metrics such as RAM and GPU usage. These metrics provide insights into the computational efficiency and resource demands of text generation models, which are vital for their scalability and practical deployment. Table 1 illustrates a taxonomy of the metrics utilized in this study, categorizing them by type and detailing their respective applications.

- **Linguistic Quality Metrics**:
  ROUGE, BLEU, BertScore(precision, recall,F1), Cosine Similarity
- **Contextual Relevance and Accuracy Metrics**:
  Context Precision, Faithfulness, Answer Relevancy, and Answer Correctness
- **Resource Utilization Metrics**:
  RAM, GPU, CPU usage

A. EVALUATION METRICS BASED ON LINGUISTIC QUALITY

Evaluating Retrieval-Augmented Generation (RAG) and fine-tuned Large Language Models (LLMs) is vital for measuring their performance and identifying improvement areas. Metrics such as BertScore, BLEU, METEOR, and ROUGE are pivotal, each providing insights into different aspects of model output.

The BLEU score is a widely used metric in machine translation tasks due to its simplicity and effectiveness in assessing the quality of machine-generated translations compared to reference translations. Its ease of calculation and interpretation makes it a popular choice for evaluating translation models. However, BLEU does have its weaknesses. It is heavily dependent on n-grams, which may not always relate to the overall meaning or fluency of the translation. Also, it can't help but penalize translations for length when the translations are longer than the reference translations. On the other hand, the ROUGE score is commonly used in text summarization to objectively assess the quality of machine-generated summaries by comparing them to reference summaries. It measures the overlap of n-grams, capturing key content effectively, and is flexible enough to accommodate different n-gram lengths based on task requirements. Moreover, the METEOR score combines the BLEU and METEOR scores. It evaluates machine translations by comparing them to human translations, considering both accuracy and fluency, as well as the word order.

This paper employs these metrics for a detailed comparison of RAG and fine-tuning methods, highlighting their effectiveness in various language tasks. In the BertScore paper [38] BERTScore marked a significant advance in automatic text evaluation by leveraging renormalized vectors from BERT embeddings to assess token-level similarity. Unlike traditional metrics focused on surface form, the BERTScore emphasizes semantic content, aligns more closely with human judgment, and offers a nuanced assessment of text quality. The BERTScore is commonly used in evaluating text for tasks such as summarization, translation, and data-to-text generation. The BERTScore provides precision, recall, and F1 scores. The following are the equations for the BERT precision, recall, and F1-scores.

Moreover, cosine similarity is a metric used to measure how similar two vectors are, irrespective of their size. Mathematically, it calculates the cosine of the angle between two vectors projected in a multidimensional space. This approach is particularly useful in various fields, such as information retrieval, text analysis, and machine learning, where it helps to assess the similarity between documents, search engine results, or data patterns. It is commonly used to evaluate language model performance. For example, one of the studies [39] described various similarity-based models, which include cosine similarity scores for recommendation systems.

First, the range of the metrics is different: precision, recall, and F1 score range from 0 to 1 for the BertScore, BLEU, METEOR, and ROUGE algorithms, where higher scores mean better translation quality or effectiveness of summarization. Cosine similarity falls within a range from -1 to 1, where 1 means perfect similarity between vectors quality is highly important for document comparison tasks. Taken together, these metrics provide a way of comprehensively determining textual accuracy and semantic alignment. These are essential tools in computational linguistics and document analysis.

To summarize and compare these metrics:

- BertScore evaluates token-level similarity. It uses BERT embeddings, focusing on semantic content. It also arranges closely with human judgment and provides precision, recall, and F1 metrics.
- BLUE measures the n-gram overlap between generated text and reference text, serving as a proxy for translation quality.
- METEOR considers synonyms and stemming to integrate semantic analysis, enhancing evaluation with linguistic nuances.
- ROUGE evaluates linguistic quality and effectiveness in summarization, providing a comprehensive assessment.
- Cosine similarity measures how similar two vectors are by calculating the cosine of the angle between them, which is useful in comparing documents and data patterns across various fields.

Also, each metric has its unique advantages and disadvantages:

- BERTScore excels in semantic similarity but is computationally demanding.
- BLEU is simple and widely accepted but lacks semantic nuance.
- METEOR provides linguistic richness but is complex.
- ROUGE offers a balanced evaluation for summarization but focuses on lexical matches.
- Cosine Similarity is versatile and robust but ignores word order and requires high-quality embeddings.

For a comprehensive evaluation of language model performance, it's beneficial to use a combination of these metrics, balancing their strengths and addressing their weaknesses. This multifaceted approach will give a more holistic understanding of the models' capabilities.

The application of these metrics in a comparative study illuminates the relative strengths and weaknesses of RAG and fine-tuning LLM approaches. It also provide critical insights into their operational efficacious and guiding future enhancements in model architecture and training methodologies.

B. EVALUATION METRIC BASED ON CONTEXTUAL RELEVANCE AND ACCURACY

When traditional metrics such as BLEU or ROUGE fail to capture the nuanced performance of LLMs in specific contexts, metrics such as Context Precision, Faithfulness, Answer Relevancy, and Answer Correctness become essential. They reveal the complexity of NLP evaluations, especially for applications such as question answering and regulatory compliance. They point out the need for metrics that extend beyond lexical similarity to semantic equivalence. Quite an appropriate approach when placed in a setting where retrieval-augmented generation combines with the instruction following models, where much emphasis needs to be given to semantic and factual integrity in the generated text. Our study will introduce a suite of specialized metrics, RAGAs [28] to evaluate the efficacy and faithfulness of LLM-generated responses along various dimensions, ranging from relevance to factual accuracy. These dimensions afford a more fine-grained look at LLM performance in generating contextually correct answers. Here, the generated answer is said to be faithful when its claims are inferable from the given context. This is determined by the set of claims from the answer and cross-checking each claim with the given context. The formula to calculate the faithfulness score will be the number of claims inferred divided by:

$$\text{Faithfulness score} = \frac{|\text{Inferred claims}|}{|\text{Total claims}|} \tag{1}$$

On the other hand, The Answer Relevancy is defined as the mean cosine similarity of the original question to several artificial questions, which were generated (reverse engineered) based on the answer:

$$\text{Answer Relevancy} = \frac{1}{N}\sum_{i=1}^{N} \cos(\mathrm{E_{gi}}, \mathrm{E_0}) \tag{2}$$

where:

- $\mathrm{E_{gi}}$ is the embedding of the generated question i.
- $\mathrm{E_0}$ is the embedding of the original question.
- N is the number of generated questions, which is 3 defaults.

Moreover, Context Precision assesses whether all relevant items from the ground truth in the contexts are ranked higher. Ideally, these relevant chunks should appear at the top ranks. This metric utilizes the question, ground truth, and the contexts for computation.

$$\text{Context Precision@K} = \frac{\sum_{k=1}^{K} \mathrm{P@k} \times \mathcal{V}_k}{\text{Total relevant items in top K}} \tag{3}$$

where:

- K is the total number of chunks in contexts and $\mathcal{V}_k \in \{0, 1\}$ is the relevance indicator at rank k.

To summarize and compare these metrics:

- Context Precision measures how well the generated text aligns with the context of the input, which is crucial for applications requiring high relevance to specific contexts.
- Faithfulness assesses the extent to which the generated text accurately represents the information in the source data, which is important for maintaining integrity in the output.
- Answer Relevancy evaluates how relevant the generated answers are to the questions posed, which is essential for question-answering systems.
- Answer Correctness checks the factual accuracy of answers, which is significant in environments such as regulatory compliance where factual correctness is paramount.

Each metric offers unique insights into language model performance:

- Context Precision focuses on relevance to the input context but can be subjective.
- Faithfulness ensures accurate representation of source data but is complex to evaluate.
- Answer Relevancy enhances QA systems' pertinence but involves subjective assessment.
- Answer Correctness ensures factual accuracy but relies on access to reliable data.

Using a combination of these metrics provides a comprehensive evaluation framework. It also provides balance between their strengths and their weaknesses for a well-rounded assessment of language model performance.

C. EVALUATION METRICS BASED ON RESOURCE UTILIZATION

Large language models have brought many changes in the field of natural language processing. These models, though computationally intensive, suffer from resource consumption and efficiency problems. This chapter evaluates the computational resources required for Retrieval-Augmented Generation, fine-tuning, and the combination of both methods, which are considered fundamental in improving the performance of LLMs.

Several key metrics are to be considered in the comparative analysis of RAG, fine-tuning, and their combined approaches, such as RAM usage and processing time. These two parameters will be indicative of the computational resource requirements when deploying improved LLMs. RAM usage indicates the memory efficiency required for operation in

resource-constrained environments and denotes the requisite memory allocation during both training and inference. Besides that, the time required for processing reveals how many training cycles could be run, and inference tasks could be performed. Indeed, this forms a critical metric for assessing the scalability and practical feasibility of LLMs. Jointly, they provide a complete picture of the computational footprint to improve and deploy LLMs, with a careful balance between resource usage and model performance in this regard. High resource consumption shall be an expression used when the RAM needed exceeds 64GB, which is the most common high-end limit of a common desktop computer, or when training might take more than a few days with high-end GPUs. Examples of such tasks are those requiring specialized hardware, such as multi-GPU solutions or TPUs, and at the same time, large computational time is needed. These benchmarks help contextualize the computational demands of various approaches.

As we delve deeper into the application of these metrics, the contrasting strategies of RAG and fine-tuning manifest in their unique approaches to addressing the challenges of computational efficiency and model utility. RAG stands out for enhancing precision in expansive models, which is particularly beneficial for contextually significant data, as exemplified by farm data analysis. Its appeal lies in the minimal initial investment needed to generate embeddings–vector representations of data–making it a cost-effective strategy. However, it is pivotal to acknowledge the potential for increased prompt size due to larger input token sizes, alongside a tendency for outputs to be more expansive and less manageable. Conversely, fine-tuning excels in producing concise, targeted outputs conducive to summarization. This method proves paramount in acquiring domain-specific capabilities, such as refining crop yield forecasts or advancing irrigation timing in response to meteorological conditions. However, the upfront expenditure for fine-tuning novel datasets is considerable because of the intensive process of model adjustment. Furthermore, fine-tuning demands a restrained input token size, rendering it a more streamlined approach for processing voluminous datasets. The subsequent section contains results and discussion centered around the evaluation metrics focused on natural language generation, contextual accuracy, and computational resources. It presents an analysis that identifies which methodologies excel or underperform across these distinct metrics.

VI. RESULTS AND DISCUSSION

This section provides an error analysis using examples from the test dataset and discusses various metrics: quality metrics, contextual relevance and accuracy metrics, and resource utilization metrics.

In the first example in Figure 8, we present the first example from the test dataset from SQuAD. Notably, the fine-tuned and fine-tuned + RAG models consistently avoid responses such as "I don't know" or "I don't have that information." In contrast, the RAG model generates these types of responses multiple times, as shown in Figure 8. The fine-tuning process effectively teaches the model the domain knowledge during training, enabling it to understand and respond to topics more accurately. Since the RAG model lacks a dedicated training phase, it may perform poorly on certain examples.


```
[
  {
    "Query": "What percentage of the food served at Notre Dame is locally grown?",
    "Answer":"40%",
    "Context": "The University of Notre Dame has made being a sustainability leader an integral part of its mission, creating the Office of Sustainability in 2008 to achieve number of goals in the areas of power generation, design and construction, waste reduction, procurement, food services, transportation, and water. As of 2012 four building construction projects were pursuing LEED-Certified status and three were pursuing LEED Silver. Notre Dame's dining services sources 40% of its food locally and offers sustainably caught seafood as well as many organic, fair-trade, and vegan options.",
    "RAG Result": "I don't have that information.",
    "Fine-Tuning Result": "The University of Notre Dame sources 40% of its food locally.",
    "Fine-Tuning + RAG Result": "Notre Dame's dining services sources 40% of its food locally."
  }
]
```


FIGURE 8. Error analysis example 1.

Additionally, the RAG model can produce incorrect answers. The second example in Figure 9 provides an example where the RAG model gives a wrong answer, despite the query and context being quite clear. This issue arises because the RAG system lacks a training phase, preventing the model from adequately learning the topic.


```
[
  {
    "Query": "How many big trends are involved in how much the position of dogs has changed in human civilization?",
    "Answer": "two",
    "Context": "There have been two major trends in the changing status of pet dogs. The first has been the 'commodification' of the dog, shaping it to conform to human expectations of personality and behavior. The second has been the broadening of the concept of the family and the home to include dogs-as-dogs within everyday routines and practices.",
    "RAG Answer": "There are three big trends involved in how much the position of dogs has changed in human civilization.",
    "Fine-Tuning Answer": "There have been two major trends in the changing status of pet dogs: the 'commodification' of the dog and the broadening of the concept of the family and the home to include dogs-as-dogs within everyday routines and practices.",
    "Fine-Tuning + RAG Answer": "There have been two major trends in the changing status of pet dogs in human civilization."
  }
]
```


FIGURE 9. Error analysis example 2.

On the other hand, in the test dataset, we observe that the fine-tuning and fine-tuning + RAG models tend to provide longer answers compared to the RAG model. The following example in the Figure 10 illustrates this situation with a specific example. The increased complexity of the models after the training phase can lead to longer, more detailed responses. However, this complexity can also introduce the problem of hallucination, where the model generates information that is not present in the input data.

To systematically assess the efficacy of various language model training techniques, we conducted a comparative analysis of model performances employing three distinct strategies: fine-tuning, retrieval-augmented generation (RAG), and a hybrid approach that combines both, specifically targeting the Stanford Question Answering Dataset (SQuAD),

```
[
  {
    "Query": "What is the general opinion of Whitehead in most philosophical schools?",
    "Answer": Whitehead has not been recognized as particularly influential within the most
dominant philosophical schools.",
    "Context": Deleuze's and Latour's opinions, however, are minority ones, as Whitehead has
not been recognized as particularly influential within the most dominant philosophical schools. It
is impossible to say exactly why Whitehead's influence has not been more widespread, but it may
be partly due to his metaphysical ideas seeming somewhat counter-intuitive (such as his assertion
that matter is an abstraction), or his inclusion of theistic elements in his philosophy, or the
perception of metaphysics itself as pass\u00e9, or simply the sheer difficulty and density of his
prose.",
    "RAG Answer": Whitehead has not been recognized as particularly influential within the
most dominant philosophical schools.",
    "Fine-Tuning Answer": Whitehead's influence has not been widespread in dominant
philosophical schools, possibly due to his counter-intuitive metaphysical ideas, inclusion of
theistic elements, and the perception of metaphysics as pass\u00e9. However, Whitehead believed
metaphysical investigations are essential for progress in both science and philosophy.",
    "Fine-Tuning + RAG Answer": Whitehead's influence has not been particularly widespread
within the most dominant philosophical schools. This could be due to various reasons such as his
metaphysical ideas being seen as counter-intuitive, the inclusion of theistic elements in his
philosophy, the perception of metaphysics as pass\u00e9, or the difficulty and density of his prose."
  }
]
```


FIGURE 10. Error analysis example 3.

MS MARCO and SQL CREATE TABLE statements. We utilized linguistic quality metrics, namely, BLEU, ROUGE, and METEOR, to measure the outcomes.

Our analytical results are comprehensively presented in Table 1, employing three critical evaluation metrics: ROUGE, BLEU, and METEOR scores. These results illustrate that, in general, fine-tuning tends to obtain higher ROUGE and METEOR scores compared to RAG on SQuAD and MS MARCO, which indicates better recall and alignment with the reference texts. The BLEU scores remain low and relatively comparable across methods, reflecting difficulties in capturing precise word sequences. The combination of fine-tuning and RAG has mixed results: there is a slight improvement in ROUGE, while BLEU and METEOR are reduced, possibly pointing to potential trade-offs in output consistency. For the SQL Statements, all metrics are considerably higher; both fine-tuning and the combined approach work similarly well, thereby underlining the advantage of structured data for accurate output generation. These differences illustrate the word-sequence-based nature of these metrics, which do not consider semantic meaning yet serve as useful means to estimate the quality of surface aspects of the text that is why they do not need the generated text to be perfectly parallel to the reference text but still provide quantitative insight into performance. In addition, as can be seen from our results, the SQL statement dataset yields higher ROUGE, METEOR, and BLEU scores. Most of these metrics judge the order and sequence of the tokens rather than the meaning of the tokens. While they are not designed for assessing semantic equivalence, we include them here to highlight their capability of measuring performance in structurally oriented datasets. For clarity, the range of metric scores is set between 0 and 1.

Another metric focused on linguistic quality is BertScore, which evaluates text using precision, recall, and the F1 score. In Table 2, the BERT scores, including the precision, recall, and F1 scores, are tabulated to illustrate the comparative performance of the distinct language model training methods

TABLE 1. ROUGE, BLEU, and METEOR scores for fine-tuning, RAG, and their combination.

Experiments	ROUGE	BLEU	METEOR
SQuAD Fine-Tuning	0.25	*0.06*	*0.44*
SQuAD RAG	0.17	*0.06*	0.27
SQuAD Fine-Tuning + RAG	*0.26*	0.04	0.16
MS MARCO Fine-Tuning	0.20	*0.06*	*0.23*
MS MARCO RAG	0.14	0.03	0.17
MS MARCO Fine-Tuning + RAG	*0.22*	0.05	0.21
SQL Statements Fine-Tuning	*0.78*	*0.49*	*0.48*
SQL Statements RAG	0.62	0.16	0.40
SQL Statements Fine-Tuning + RAG	0.75	0.42	*0.48*

applied to my three datasets. BertScore, employing BERT embeddings, assesses the semantic similarity of model outputs through these metrics. The results show that the hybrid model, which combines fine-tuning and RAG, has the highest recall for all datasets while keeping very strong precision and F1 scores. The reason is simple: the hybrid approach leverages the retrieval capabilities of RAG for better contextual grounding while fine-tuning refines the outputs.

Another important parameter is the cosine similarity score. In the evaluation of text generation models, the comparative analysis of cosine similarity scores offers insight into the effectiveness of different methodologies. The results are shown in Table 3. Results are indicative of the strength brought about by fine-tuning with RAG in generating semantically aligned outputs. Generally, the hybrid approach performs best on all datasets, especially on SQL Statements, with a score of 0.89, indicative of a stronger alignment between the generated and reference embeddings as an indication of the ability of the hybrid model in capturing more nuanced semantic relationships. The improvement in cosine similarity scores has its roots in the complementary strengths of fine-tuning and RAG. Fine-tuning ensures that model outputs are refined and contextually accurate, while RAG enhances semantic richness by incorporating relevant retrievals. This synergy results in embeddings that are not only contextually relevant but also semantically coherent, demonstrating the hybrid model's robustness in understanding and generating complex text.

Furthermore, outcomes derived from metrics that concentrate on contextual relevance and accuracy, such as

TABLE 2. BERT scores (Precision, recall, and f1 scores) for each model.

Experiments	Precision	Recall	F1
SQuAD Fine-Tuning	0.84	0.89	*0.87*
SQuAD RAG	0.84	0.88	0.86
SQuAD Fine-Tuning + RAG	*0.85*	*0.90*	*0.87*
MS MARCO Fine-Tuning	0.84	0.84	*0.83*
MS MARCO RAG	0.84	0.84	*0.83*
MS MARCO Fine-Tuning + RAG	*0.85*	*0.86*	*0.83*
SQL Statements Fine-Tuning	0.90	0.92	*0.91*
SQL Statements RAG	0.86	0.89	0.88
SQL Statements Fine-Tuning + RAG	*0.91*	*0.94*	0.90

TABLE 3. Cosine similarity score for each model.

Experiments	Cosine Similarity Score
SQuAD Fine-Tuning	0.47
SQuAD RAG	0.49
SQuAD Fine-Tuning +RAG	*0.54*
MS MARCO Fine-Tuning	0.56
MS MARCO RAG	0.55
MS MARCO Fine-Tuning +RAG	*0.58*
SQL Statements Fine-Tuning	0.87
SQL Statements RAG	0.83
SQL Statements Fine-Tuning +RAG	*0.89*

answer relevancy, context precision, faithfulness, and answer correctness, are imperative and are given in Table 4. The hybrid model has always performed well, especially on metrics that balance retrieval quality and output precision. On SQuAD, the hybrid model achieves excellent scores across all metrics, notably improved by the individual methods in context precision of 0.89 and answer correctness of 0.82, hence good at returning accurate answers with relevant context. On MS MARCO, the hybrid model yields moderate improvement over strong baselines in answer relevancy at 0.89 and correctness at 0.54; besides that, it outperforms RAG significantly on faithfulness and is competitive with fine-tuning. The structured nature of the data in SQL Statements leads to strong performance across all models, but the hybrid approach offers a good balance, improving faithfulness to 0.77 and answer correctness to 0.80 over RAG and fine-tuning in isolation. This shows that the hybrid approach indeed combines the retrieval strengths of RAG with refinement from fine-tuning and results in more contextually accurate and faithful response generation, especially for complex or structured datasets. For clarity, the range of metric scores, set between 0 and 1.

TABLE 4. Answer relevancy, context precision, faithfulness, and answer correctness scores for fine-tuning, RAG, and their combination.

Experiments	Answer Relevancy	Context Precision	Faithfulness	Answer Correctness
SQuAD Fine-Tuning	*0.94*	0.88	0.88	0.74
SQuAD RAG	0.77	0.57	0.82	0.50
SQuAD Fine-Tuning + RAG	0.91	*0.89*	*0.89*	*0.82*
MS MARCO Fine-Tuning	0.85	*0.58*	*0.88*	0.48
MS MARCO RAG	0.71	0.32	0.59	0.53
MS MARCO Fine-Tuning + RAG	*0.89*	0.51	0.65	*0.54*
SQL Statements Fine-Tuning	*0.84*	0.29	0.75	0.70
SQL Statements RAG	0.81	0.18	0.45	0.57
SQL Statements Fine-Tuning + RAG	0.83	*0.23*	*0.77*	*0.80*

On the other hand, according to the computational resources, for the initial CPU and memory expenditures, the RAG demonstrates a minimal footprint, making it a cost-efficient choice at the outset. When considering the initial data costs, RAG again proves economical, relying on low-cost related documents, whereas fine-tuning and the combination approach necessitate high-cost labeled data, with the hybrid method bearing the cumulative costs of both high-cost related documents and labeled data. In terms of training, a RAG offers a significant advantage by incurring no costs, in stark contrast to fine-tuning and the combined method, which both require high investment. Finally, the vector database cost is substantial for the RAG and Fine-Tuning+RAG approaches, implying greater resource allocation for data storage and retrieval, while fine-tuning

alone carries no such cost. The table below highlights the trade-offs between initial investment, ongoing operational costs, and the underlying infrastructure requirements for each approach, offering a clear financial and logistical perspective for decision-making in LLM deployment. Table 5 shows the models' resource utilizations.

TABLE 5. Computing resources for each model.

	RAG	Fine-Tuning	Fine-Tuning + RAG
Initial CPU - Memory Cost	Minimal	High	High
Initial Data Cost	Low - related documents	High - labeled data	High - related documents and labeled data
Training Cost	No cost	High	High
Vector Database Cost	High	No cost	High

This work investigates in detail the language model training methodologies-fine-tuning, retrieval-augmented generation, and their hybrid integration on three datasets: SQuAD, MS MARCO, and SQL CREATE TABLE statements. These results indicate that fine-tuning always results in higher ROUGE and METEOR scores on SQuAD and MS MARCO, reflecting better recall and alignment with reference texts. However, BLEU scores are uniformly low across methods, highlighting difficulties in reproducing exact word sequences. SQL Statements, due to the structured nature of the data, engender scores that are much higher on all metrics, hence the importance of data format in facilitating the generation of accurate and contextually relevant text. Although these metrics are not intended to assess semantic equivalence, we include them to emphasize their utility in evaluating performance for structurally oriented datasets.

Semantic evaluation metrics of BertScore and cosine similarity further reiterate the strong points of the hybrid approach: the hybrid model achieved the highest recall while retaining high precision and F1 score, leveraging both the contextual grounding of RAG and refinement capabilities afforded by fine-tuning. These findings are further reinforced by cosine similarity: the hybrid approach outperformed all other systems across all the datasets; especially on SQL Statements, it achieved a very high score of 0.89, indicating robust semantic coherence and alignment of generated and reference embeddings.

Metrics related to contextual relevance and faithfulness, such as answer relevancy, context precision, faithfulness, and answer correctness, confirm the superiority of the hybrid model in producing responses that are both accurate and contextually appropriate. This was particularly evident in the SQL Statements, with the hybrid approach effectively balancing retrieval quality and precision in that regard. Such results thus show the robustness of the hybrid model, especially towards complex or structured datasets, as it combines the strengths of retrieval and fine-tuning.

Apart from performance metrics, the paper gives insight into the usage of computational resources, making this quite practical for deployment. Among all the methods compared, RAG leads the way in terms of cost-efficiency regarding initial CPU and memory requirements, whereas the hybrid approach costs higher due to the demand for labeled data and vector databases. This trade-off analysis balances performance and operational costs, providing actionable guidance to researchers and practitioners on how to choose methodologies considering task complexity, dataset structure, and resource constraints.

VII. CONCLUSION

This work presents a systematic evaluation of fine-tuning, retrieval-augmented generation, and their hybrid integration on a wide variety of datasets: SQuAD, MS MARCO, and SQL CREATE TABLE Statements. Applying a range of linguistic, semantic, and contextual evaluation metrics, this study unravels some key insights into the strengths, weaknesses, and trade-offs of these training methodologies.

Although fine-tuning is very computationally expensive, it is good for unstructured tasks, with much better recall and semantic alignment to the reference texts, as demonstrated by the high ROUGE and METEOR scores on the SQuAD and MS MARCO datasets. However, the BLEU scores are uniformly low across the methods, reflecting the difficulties in reproducing exact word sequences. SQL statements produce, due to the structured nature of the data, much higher scores on all metrics and, hence, the importance of data format in facilitating the generation of text with accuracy and contextual relevance. Note that these metrics are not aimed at semantic equivalence, although we include them because they tend to be useful for conducting performance evaluation in structurally oriented datasets. Semantic evaluations metrics such as BertScore and cosine similarity further reiterate the strength points of the hybrid approach: the hybrid model achieved the highest recall while retaining high precision and F1 score, leveraging both the contextual grounding of RAG and refinement capabilities afforded by fine-tuning. This is further confirmed by cosine similarity: the hybrid approach outperformed all other systems on all datasets; it indeed scored very high on SQL Statements, which indicates robust semantic coherence and alignment of generated and reference embeddings.

Metrics involving the dimension of contextual faithfulness, including answer relevancy, context precision, faithfulness, and correctness of the answer, show that the hybrid model ensure response accuracy and is contextually relevant. This was particularly demonstrated within the SQL statements, as it was able to appropriately balance retrieval quality and precision. These results reveal the robustness of the hybrid model, particularly for complex or structured datasets, by focusing on the strengths of both retrieval and fine-tuning. In addition to performance metrics, this study also

considers computational resource usage. RAG, however, is cost-effective among the methods compared, especially with regard to CPU and memory costs, and the hybrid model is very expensive considering that it also involves labeled data and vector databases. Thus, the present trade-off analysis attempts to equilibrate between performance and operational cost, and provides practical hints for the choice of methodologies depending on task complexity, dataset structure, and resource constraints.

Our results highlight that methodological choices need to be compatible with application-specific requirements, whether one prioritizes computational efficiency, linguistic precision, or contextual adaptability. Future work should focus on refining hybrid methodologies, exploring adaptive training frameworks, and optimizing resource utilization to extend the practical applicability of LLMs to a wider range of real-world challenges.

REFERENCES

[1] A. Rogers, O. Kovaleva, and A. Rumshisky, "A primer in BERTology: What we know about how BERT works," *Trans. Assoc. Comput. Linguistics*, vol. 8, pp. 842–866, Dec. 2020.

[2] T. B. Brown et al., "Language models are few-shot learners," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 1877–1901.

[3] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ questions for machine comprehension of text," 2016, *arXiv:1606.05250*.

[4] P. Bajaj, D. Campos, N. Craswell, L. Deng, J. Gao, X. Liu, R. Majumder, A. McNamara, B. Mitra, T. Nguyen, M. Rosenberg, X. Song, A. Stoica, S. Tiwary, and T. Wang, "MS MARCO: A human generated machine reading comprehension dataset," 2016, *arXiv:1611.09268*.

[5] (May 12, 2024). *b-mc2/sql-create-context · Datasets at Hugging Face*. Accessed: Jan. 11, 2025. [Online]. Available: https://huggingface.co/datasets/b-mc2/sql-create-context

[6] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking large language models in retrieval-augmented generation," in *Proc. AAAI Conf. Artif. Intell.*, Mar. 2024, vol. 38, no. 16, pp. 17754–17762.

[7] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, Q. Guo, M. Wang, and H. Wang, "Retrieval-augmented generation for large language models: A survey," 2023, *arXiv:2312.10997*.

[8] Z. Shao, Y. Gong, Y. Shen, M. Huang, N. Duan, and W. Chen, "Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy," 2023, *arXiv:2305.15294*.

[9] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-T. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 9459–9474.

[10] O. Ovadia, M. Brief, M. Mishaeli, and O. Elisha, "Fine-tuning or retrieval? Comparing knowledge injection in LLMs," 2023, *arXiv:2312.05934*.

[11] Z. Ye, D. Li, J. Tian, T. Lan, J. Zuo, L. Duan, H. Lu, Y. Jiang, J. Sha, K. Zhang, and M. Tang, "ASPEN: High-throughput LoRA fine-tuning of large language models with a single GPU," 2023, *arXiv:2312.02515*.

[12] O. Ram, Y. Levine, I. Dalmedigos, D. Muhlgay, A. Shashua, K. Leyton-Brown, and Y. Shoham, "In-context retrieval-augmented language models," *Trans. Assoc. Comput. Linguistics*, vol. 11, pp. 1316–1331, Nov. 2023.

[13] H. A. Alawwad, A. Alhothali, U. Naseem, A. Alkhathlan, and A. Jamal, "Enhancing textbook question answering task with large language models and retrieval augmented generation," 2024, *arXiv:2402.05128*.

[14] A. Merchant, E. Rahimtoroghi, E. Pavlick, and I. Tenney, "What happens to BERT embeddings during fine-tuning?" 2020, *arXiv:2004.14448*.

[15] S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. R. Bowman, and N. A. Smith, "Annotation artifacts in natural language inference data," 2018, *arXiv:1803.02324*.

[16] A. Poliak, J. Naradowsky, A. Haldar, R. Rudinger, and B. Van Durme, "Hypothesis only baselines in natural language inference," 2018, *arXiv:1805.01042*.

[17] I. Tenney, P. Xia, B. Chen, A. Wang, A. Poliak, R. T. McCoy, N. Kim, B. Van Durme, S. R. Bowman, D. Das, and E. Pavlick, "What do you learn from context? Probing for sentence structure in contextualized word representations," 2019, *arXiv:1905.06316*.

[18] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "GLUE: A multi-task benchmark and analysis platform for natural language understanding," 2018, *arXiv:1804.07461*.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[20] M. E. Peters, S. Ruder, and N. A. Smith, "To tune or not to tune? Adapting pretrained representations to diverse tasks," in *Proc. 4th Workshop Represent. Learn. NLP*, 2019, pp. 7–14.

[21] R. Behnia, M. R. Ebrahimi, J. Pacheco, and B. Padmanabhan, "EW-tune: A framework for privately fine-tuning large language models with differential privacy," in *Proc. IEEE Int. Conf. Data Mining Workshops (ICDMW)*, Nov. 2022, pp. 560–566.

[22] K. Lv, Y. Yang, T. Liu, Q. Gao, Q. Guo, and X. Qiu, "Full parameter fine-tuning for large language models with limited resources," 2023, *arXiv:2306.09782*.

[23] A. Asai, T. Schick, P. Lewis, X. Chen, G. Izacard, S. Riedel, H. Hajishirzi, and W.-T. Yih, "Task-aware retrieval with instructions," 2022, *arXiv:2211.09260*.

[24] G. Izacard, M. Caron, L. Hosseini, S. Riedel, P. Bojanowski, A. Joulin, and E. Grave, "Unsupervised dense information retrieval with contrastive learning," 2021, *arXiv:2112.09118*.

[25] Y. Shi, X. Zi, Z. Shi, H. Zhang, Q. Wu, and M. Xu, "ERAGent: Enhancing retrieval-augmented language models with improved accuracy, efficiency, and personalization," 2024, *arXiv:2405.06683*.

[26] C. S. Chan, H. Kong, and G. Liang, "A comparative study of faithfulness metrics for model interpretability methods," 2022, *arXiv:2204.05514*.

[27] Z. Hu, L. Wang, Y. Lan, W. Xu, E.-P. Lim, L. Bing, X. Xu, S. Poria, and R. K.-W. Lee, "LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models," 2023, *arXiv:2304.01933*.

[28] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "RAGAS: Automated evaluation of retrieval augmented generation," 2023, *arXiv:2309.15217*.

[29] W. Kuang, B. Qian, Z. Li, D. Chen, D. Gao, X. Pan, Y. Xie, Y. Li, B. Ding, and J. Zhou, "FederatedScope-LLM: A comprehensive package for fine-tuning large language models in federated learning," in *Proc. 30th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, 2024, pp. 5260–5271.

[30] R. Lakatos, P. Pollner, A. Hajdu, and T. Joó, "Investigating the performance of retrieval-augmented generation and domain-specific fine-tuning for the development of AI-driven knowledge-based systems," *Mach. Learn. Knowl. Extraction*, vol. 7, no. 1, p. 15, Feb. 2025.

[31] K. Sawarkar, A. Mangal, and S. R. Solanki, "Blended RAG: Improving RAG (retriever-augmented generation) accuracy with semantic search and hybrid query-based retrievers," 2024, *arXiv:2404.07220*.

[32] H. Soudani, E. Kanoulas, and F. Hasibi, "Fine tuning vs. retrieval augmented generation for less popular knowledge," in *Proc. Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. Asia Pacific Region*, 2024, pp. 12–22.

[33] A. Balaguer, V. Benara, R. L. D. F. Cunha, R. D. M. E. Filho, T. Hendry, D. Holstein, J. Marsman, N. Mecklenburg, S. Malvar, L. O. Nunes, R. Padilha, M. Sharp, B. Silva, S. Sharma, V. Aski, and R. Chandra, "RAG vs fine-tuning: Pipelines, tradeoffs, and a case study on agriculture," 2024, *arXiv:2401.08406*.

[34] S. Barnett, Z. Brannelly, S. Kurniawan, and S. Wong, "Fine-tuning or fine-failing? Debunking performance myths in large language models," 2024, *arXiv:2406.11201*.

[35] S. Alghisi, M. Rizzoli, G. Roccabruna, S. M. Mousavi, and G. Riccardi, "Should we fine-tune or RAG? Evaluating different techniques to adapt LLMs for dialogue," 2024, *arXiv:2406.06399*.

[36] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*.

[37] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," 2019, *arXiv:1908.10084*.

[38] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating text generation with BERT," 2019, *arXiv:1904.09675*.

[39] S. C. Mana and T. Sasipraba, "Research on cosine similarity and Pearson correlation based recommendation models," *J. Phys., Conf.*, vol. 1770, no. 1, Mar. 2021, Art. no. 012014.

GÜLSÜM BUDAKOGLU received the bachelor's degree in mathematics from Middle East Technical University, Ankara, Türkiye, in 2020, and the master's degree in applied data science from TED University, Ankara, in 2024.

She has been a Data Scientist, since 2020. She has developed fine-tuned state-of-the-art language models, among others for text generation and language understanding, achieving model performance improvements. She develops superior text generation and works on generative AI models, vector search, and databases, while adhering to all privacy regulations. She has experience in using machine learning techniques to facilitate processes. Her research interests include large language models, machine learning, and data science applied to real-world problems.

HAKAN EMEKCI received the B.Sc. degree in computer engineering from Middle East Technical University (METU), Ankara, Türkiye, in 2011, the M.Sc. degree from London Business School, and the Ph.D. degree from Hacettepe University, Ankara.

He is currently a Faculty Member with TED University, specializing in artificial intelligence and machine learning. He is the Founder of NavigaAI, a company focused on AI-driven solutions. He is actively involved in educational technology development, including the Robomentor chatbot project, which serves over 5000 users during the university preference period. He has led impactful projects in collaboration with the European Bank for Reconstruction and Development (EBRD) and the United Nations, where his work has supported AI-driven solutions for social and economic development initiatives. Beyond his academic and professional achievements, he has dedicated to bridging advanced AI methodologies with practical applications that address real-world challenges, advancing both the technology and its accessibility. He continues to mentor the next generation of data scientists, focusing on ethical AI practices and innovative problem-solving approaches. He has published several papers, including a recent work on optimizing retrieval-augmented generation (RAG) efficiency using large language models. He is the author of *Data Mining With R*. His research interests include natural language processing, machine learning applications, and information retrieval systems.