Mugla Journal of Science and Technology

RETRIEVAL-AUGMENTED GENERATION IN TURKISH NATURAL LANGUAGE UNDERSTANDING: A COMPARATIVE STUDY OF LARGE LANGUAGE MODELS

Ercan ATAGÜN*, Department of Computer Engineering, Düzce University, TURKEY, ercanatagun@duzce.edu.tr
(https://orcid.org/0000-0001-5196-5732)
Merve GÜLLÜ, R&D Department, TT Mobile Communication Services Inc., TURKEY, merve.gullu@turktelekom.com.tr
(https://orcid.org/0000-0001-7442-1332)
Serdar BİROĞUL, Department of Computer Engineering, Düzce University, TURKEY, serdarbirogul@duzce.edu.tr
(https://orcid.org/0000-0003-4966-5970)
Necaattin BARIŞÇI, Department of Computer Engineering, Gazi University, TURKEY, nbarisci@gazi.edu.tr
(https://orcid.org/0000-0002-8762-5091)

Received: 09.09.2025, Accepted: 03.11.2025
Research Article
*Corresponding author
DOI: 10.22531/muglajsci.1781095

Abstract

Large Language Models (LLMs) have markedly progressed natural language processing. Nevertheless, owing to the restricted availability of training data, they may prove insufficient in generating current and precise information, particularly for low-resource languages. The Retrieval-Augmented Generation (RAG) methodology, designed to resolve this challenge, improves the precision and dependability of models' outputs by leveraging external information sources. This study comparatively evaluated four distinct LLMs (Qwen-14B, Gemma3-12B, LLaMA3.1-8B, and DeepSeek-R1-14B) within the RAG framework using a Turkish question-answer dataset. Experimental results demonstrate the RAG methodology markedly enhances information precision, response uniformity, and contextual relevance in Turkish question-answering systems. Moreover, the LLaMA3.1-8B model had the best equitable performance regarding precision and recall. The findings illustrate the relevance of RAG-based applications for Turkish and offer significant insights for advancing knowledge-assisted generation methods. This study addresses a significant gap in the literature by illustrating the viability of RAG-based systems in morphologically rich and low-resource languages, including Turkish. It serves as a foundational reference for subsequent Turkish natural language processing research.

Keywords: Large language models , Retrieval-augmented generation, Turkish RAG

TÜRKÇE DOĞAL DİL ANLAMA BECERISINDE GERİ ÇAĞIRMA-ARTIRILMIŞ ÜRETIM: BÜYÜK DİL MODELLERİNİN KARŞILAŞTIRMALI BİR ÇALIŞMASI

Özet

Büyük Dil Modelleri (BDL), doğal dil işlemeyi önemli ölçüde ilerletmiştir. Bununla birlikte, eğitim verilerinin sınırlı erişilebilirliği nedeniyle, özellikle düşük kaynaklı diller için güncel ve kesin bilgi üretmede yetersiz kalabilirler. Bu zorluğu çözmek için tasarlanan Geri Alma-Artırılmış Üretim (GAÜ) metodolojisi, harici bilgi kaynaklarından yararlanarak modellerin çıktılarının kesinliğini ve güvenilirliğini artırır. Bu çalışmada, Türkçe soru-cevap veri kümesi kullanılarak GAÜ çerçevesi içinde dört farklı BDL (Qwen-14B, Gemma3-12B, LLaMA3.1-8B ve DeepSeek-R1-14B) karşılaştırmalı olarak değerlendirilmiştir. Deneysel sonuçlar, GAÜ metodolojisinin Türkçe soru cevap sistemlerinde bilgi kesinliğini, yanıt tekdüzeliğini ve bağlamsal alaka düzeyini önemli ölçüde artırdığını göstermektedir. Ayrıca, LLaMA3.1-8B modeli kesinlik ve geri çağırma konusunda en iyi performansa sahipti. Bulgular, GAÜ tabanlı uygulamaların Türkçe için önemini ortaya koymakta ve bilgi destekli üretim yöntemlerinin geliştirilmesi için önemli bilgiler sunmaktadır. Bu çalışma, Türkçe de dahil olmak üzere morfolojik olarak zengin ve düşük kaynaklı dillerde GAÜ tabanlı sistemlerin uygulanabilirliğini göstererek literatürdeki önemli bir boşluğu doldurmaktadır. Ayrıca, sonraki Türkçe doğal dil işleme araştırmaları için temel bir referans görevi görmektedir.

Anahtar Kelimeler: Büyük dil modelleri, Geri alma-artırılmış üretim, Türkçe GAÜ

Cite

Atagün, E., Güllü, M., Biroğul, D., Barışçı, N., (2025). "Retrieval-Augmented Generation In Turkish Natural Language Understanding: A Comparative Study Of Large Language Models", Mugla Journal of Science and Technology, 11(2), 56-65.

1. Introduction

Large language models (LLMs) have pioneered developments in Natural Language Processing (NLP) in recent years [1]. With billions of parameters, these models are trained on large-scale text data and demonstrate superior performance in understanding

and generating human language [1-3]. In question-answering systems in particular, LLMs have transformed how we access information with their ability to provide contextual and consistent answers to users' questions on various topics [4]. Thus, even complex questions can be answered naturally and fluently, as if consulting an expert.

Despite this strong performance, LLMs also have inherent limitations [5-6]. Models only store information present in their training data in their parameters, and this information may become outdated over time or prove insufficient for specific topics [6-7]. As a result, LLM-based systems can sometimes generate outdated or misleading (hallucinatory) responses [5-7]. The Retrieval-Augmented Generation (RAG) approach, developed to overcome this problem, enhances response generation by allowing the model to leverage an external knowledge source[8]. RAG is based on the principle that a language model retrieves relevant information from external databases or document collections before generating a response. This allows the model to generate responses based on its own static knowledge base and up-to-date and reliable information from external sources. The key advantages of the RAG approach are that it fills potential knowledge gaps in LLMs, increases the accuracy of responses, and enables the model to provide more context-aware answers to the user.

Although RAG and similar knowledge-assisted generation techniques have been extensively studied in high-resource languages (especially English), such applications remain limited in low-resource languages like Turkish [9-11]. Due to its rich morphological structure and agglutinative grammar, Turkish presents additional challenges in NLP tasks [12]. This situation affects the performance of LLMs on Turkish and complicates the development of systems that utilize external knowledge retrieval. Indeed, the scarcity of data sources containing comprehensive and up-to-date information in Turkish is one of the main factors limiting the application of the RAG approach in this language. Therefore, researching the use of RAG in Turkish question-answering systems is important to generate solutions to the language structure's challenges and obtain answers based on up-to-date information in a low-resource language environment.

The main objective of this study is to evaluate the effectiveness of the RAG approach in LLM-based question-answering systems in the Turkish language. The contributions of this study can be summarized as follows:

- A comparative performance analysis of four different LLMs (Qwen-14B, Gemma3-12B, LLaMA3.1-8B, DeepSeek-R1-14B) using RAG was conducted on a Turkish question-answer dataset.
- Improvements in information accuracy, response consistency, and contextual appropriateness provided by RAG in low-resource languages such as Turkish have been demonstrated.
- The findings provide important insights for the development of Artificial Intelligence-supported information access systems for low-resource languages in the digital age.

This study fills an important gap in the literature by demonstrating the applicability of RAG-based approaches for Turkish, given the limited availability of large-scale datasets in Turkish and the fact that existing LLMs are predominantly trained on English-centric data. It contributes critically to Turkish NLP, both academically and practically.

The article consists of six sections. The second section presents the literature on Turkish NLP and knowledge-assisted generation approaches. The third section explains the proposed methodological framework and the dataset used. The fourth section addresses the experimental setup. The fifth section presents the findings and their detailed analysis. The final section summarizes the overall results and discusses possible directions for future research.

2. Related Works

The RAG approach, first introduced by Lewis et al. [13], outperformed parameterized models in knowledge-intensive NLP tasks, particularly open-domain QA. By leveraging external sources, RAG mitigates the memory limits of pre-trained models. Comprehensive surveys, such as Gupta et al. [14], reviewed RAG's evolution, architectural variants, and applications like QA and summarization, highlighting its ability to improve reliability and reduce hallucinations. Sharma et al. [15] demonstrated that fine-tuning retrievers enhanced Adobe QA, while Shi et al. [16] improved Vicuna-7B with medical knowledge in MKRAG. Alan et al. [17] showed RAG reduced misinformation in religious QA.

In Turkish research, Bikmaz et al. [18] achieved higher QA accuracy by fine-tuning retrieval on Turkish data. Yüksel et al. [19] evaluated 40+ LLMs on 10,032 Turkish MMLU questions, documenting model strengths and weaknesses. Kesgin et al. [20] introduced cosmosGPT, showing small Turkish monolingual models can rival larger multilingual ones.

Recent LLMs bring distinct advantages to RAG: Qwen-14B [21] excelled in multilingual QA, Gemma 3 [22] offered strong performance at small scales with 140-language support and 128k context, and LLaMA 3.1 [23] scaled up to 405B parameters, rivaling GPT-4. DeepSeek-R1 [24], optimized for reasoning, excelled in logical inference. Vake et al. [25] addressed style mismatch in retrieval with Hypothetical Prompt Embeddings, boosting precision (42%) and recall (45%).

Overall, these studies confirm RAG's central role in enhancing QA by enabling better information utilization. For resource-limited languages like Turkish, adapting retrievers and training with local data significantly improves system accuracy and reliability.

3. Materials and Methods

3.1. Dataset and Preprocessing

The dataset used in this study was obtained from question-and-answer content and related documents published on TÜBİTAK's official website. The operations performed on the data have been detailed to ensure the transparency of the research.

- Source Identification: TÜBİTAK is an institution that offers various support programs to scientists, researchers, and industrial organizations in many national and international fields. It was selected due to its wide range of topics and appeal to different segments. The frequently asked questions, project calls, guidelines, and documents on support programs available on the TÜBİTAK website [26] were evaluated within this scope.
- Data Collection: Data from the identified source was collected manually. The question-and-answer sections were processed directly as text-based data. Documents in PDF and similar formats were converted using text extraction techniques.
- Preprocessing: A series of preprocessing steps was applied to prepare the data for use with language models:
    - Removal of special and unwanted characters
    - Whitespace normalization
    - Word-level tokenization for both training and BLEU score calculation
    - Preservation of Turkish-specific characters (e.g., "ü, ö, ğ, ş, ı") to maintain linguistic integrity

3.2. Retrieval-Augmented Generation (RAG)

RAG is an artificial intelligence approach that integrates the information retrieval and text generation stages[13]. Fundamentally, it aims to overcome the limited context window of LLMs, customized knowledge culture, and lack of up-to-date information[27-28]. RAG incorporates external knowledge bases into the information production process by retrieving relevant content from them to answer user queries, rather than relying solely on information learned within its parameters. This enables it to provide contextually appropriate, more reliable, and domain-specific information[28].

Figure 1 depicts a standard RAG workflow. A user query is normalized and embedded; the query embedding interfaces with a Vector Index to retrieve top-k passages via a Retriever and, optionally, a Reranker. (ii) Retrieved evidence is merged into a Prompt Template & Context Builder that conditions the LLM Generator to produce a grounded answer. (iii) Post-processing handles formatting, citation stitching, or guardrails. Upstream, Documents, Databases, and APIs are ingested through Chunking & Cleaning and a Document Embedder to populate the vector index. Dashed connections indicate optional feedback loops that log user signals and cache intermediate artifacts to improve retrieval quality and latency over time.

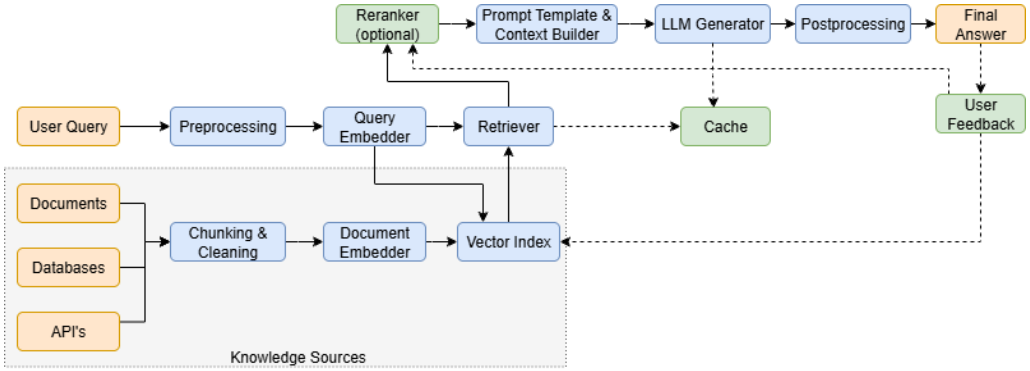

Figure 1. Overview of the RAG architecture. The runtime pipeline enriches the user query with retrieved evidence from a vector index populated via an ingestion pipeline over heterogeneous knowledge sources (dashed cluster). Optional components (dashed arrows) such as caching and feedback improve efficiency and quality.

3.3. Embedding

In this study, the paraphrase-multilingual-MiniLM-L12-v2 sentence-transformer model[29] is used for the embedding model. This model was chosen because it supports multiple languages, including Turkish, and is optimized for semantic search tasks. This embedding model ensures consistency in cosine similarity calculations by representing both query and document texts in the same vector space[30]. After the embeddings are created, they are compared using cosine similarity.

3.4. Facebook AI Similarity Search (FAISS)

Facebook AI Similarity Search (FAISS) is a library optimized for similarity searches and dense clustering operations in high-dimensional vector spaces [31]. It is widely used for storing and searching embedding vectors, particularly in NLP problems. Its specific design to efficiently perform similarity searches on large-scale

datasets with dense vectors makes it highly useful for problems like RAG[32].

In this study, the FAISS library was used in the retrieval phase. Searches are performed using the vector database's L2 (Euclidean) distance metric. This allows for the exact comparison of embedding vectors. The system provides memory optimization by indexing the embeddings of chunks generated by the sentence-transformer model in float32 format.

4. Large Language Models

Large Language Models (LLMs) are neural architectures trained on massive text corpora to capture linguistic patterns, semantic relations, and contextual dependencies [5]. With their large parameter sizes, they perform diverse NLP tasks, such as translation, summarization, and QA, with human-like fluency [33]. Beyond generation, they act as foundational models adaptable through fine-tuning or integration with mechanisms like RAG [27].

In this study, four LLMs were integrated into the RAG framework for Turkish question answering: Qwen-14B, known for strong multilingual NLU/NLG and long-context consistency [21,34]; Gemma3-12B, optimized for efficiency, low latency, and ethical reliability in research settings [22,35-36]; LLaMA3.1-8B, an open-source model balancing performance with lower hardware demands, effective in QA, summarization, and classification [23, 37]; and DeepSeek-R1-14B, designed with deep optimizations for high accuracy in complex, knowledge-oriented tasks and broad adaptability [24,38].

4.1. Evaluation Metrics

Both classical information retrieval metrics and metrics specific to NLG and RAG are used to evaluate in this systems. This section explains these metrics and provides their formulas.

1.) Information Retrieval Metrics:

a.) Precision@K

Measures how many of the top K results are relevant[39]:

$$Precision@K = \frac{|\{relevant\ documents\ in\ the\ top\ K\ results\}|}{K} \quad (1)$$

b.) Recall@K

Measures how many of all relevant documents are found in the top K results:

$$Recall@K = \frac{|\{Measures\ how\ many\ all\ relevant\ documents\ are\ found\ in\ the\ top\ K\ results\}|}{|\{all\ relevant\ documents\}|} \quad (2)$$

c.) Mean Reciprocal Rank (MRR)

Calculates the average of the reciprocals of the rank positions of the first correct answers for each query:

$$MRR = \frac{1}{|Q|} + \sum_{q=1}^{|Q|} \left(\frac{1}{rank_q}\right) \quad (3)$$

Here, $rank_q$ denotes the rank of the first correct answer for query q.

d.) Normalized Discounted Cumulative Gain (NDCG)

A ranking metric that accounts for multiple levels of relevance. First, the following is computed:

$$DCG@ = a_0 + \sum_{i=1}^{K} \left(\frac{2^{rel_i}}{log_2(i+1)}\right) \quad (4)$$

Then it is normalized as:

$$NDCG@K = \left(\frac{DCG@K}{IDCG@K}\right) \quad (5)$$

where rel_i denotes the relevance grade of the i-th document, and IDCG represents the ideal ranking.

2.) RAG-Specific Metrics

a.) Context Precision Score

In RAG systems, context precision measures the extent to which the retriever prioritizes relevant passages within the retrieved context[40].

$$\text{ContextPrec@}K = \frac{1}{K} a_0 + \sum_{i=1}^{K} I(i), \quad I(i) = \begin{cases} 1 & \text{if the i} - \text{th context is relevant,} \\ 0 & otherwise. \end{cases} \quad (6)$$

b.) Context Recall

Measures how much of all the necessary relevant information for the answer is present in the retrieved context[40]:

$$ContextRecall@K = \frac{|\{relevant\ info\ in\ top\ K\}|}{|\{all\ relevant\ info\}|} \quad (7)$$

c.) Faithfulness

Evaluates the consistency of the generated answer with the information contained in the source documents. It is typically scored through human or LLM-based assessments, relying more on quality evaluation rather than a mathematical formula[39].

3.) Natural Language Generation (NLG) Metrics

a.)BLEU: Calculated using the geometric mean of n-gram precisions combined with a brevity penalty[41]:

$$BLEU = BP\ .\ exp + \sum_{n=1}^{N} w_n\ lnp_n \quad (8)$$

Here, p_n is the n-gram precision, w_n is the weight coefficient, and BP denotes the brevity penalty.

b.) ROUGE-N

Measures n-gram overlap between the reference and the model output[42]:

$$ROUGE-N = \frac{\sum_{ref} \sum_{g \in \text{n-gram (ref)}} \min(count_{sys}g,\ count_{ref}(g))}{\sum_{ref} \sum_{g \in \text{n-gram (ref)}} count_{ref}(g)} \quad (9)$$

c.) Exact Match (EM)

Represents the proportion of predictions that exactly match the reference answers[43]:

$$EM = \frac{|\{exactly\ matching\ predictions\}|}{total\ number\ of\ examples} \quad (10)$$

d.) Accuracy

The ratio of correct predictions to the total number of predictions:

$$Accuracy = \frac{number\ of\ correct\ predictions}{total\ number\ of\ predictions} \quad (11)$$

5. Experimental Study

A multi-layered experimental environment was designed for a comprehensive RAG system performance evaluation, as summarized in Figure 1. In the hardware infrastructure, a CUDA-compatible GPU architecture was used to support the computationally intensive operations of deep learning models, and high VRAM capacity was utilized to run four different LLMs (Qwen 14B, Gemma3 12B, LLaMA3.1 8B, DeepSeek-R1 14B) simultaneously. Software-wise, a Python 3.10.14-based framework was adopted, and the OLLaMA inference server was integrated for model management and API services. This study adopts a specialized query-response matching strategy. Retrieved chunks are stored for each query and evaluated using a multi-metric approach. Along with information retrieval metrics such as Precision@K, Recall@K, MRR, and NDCG, RAG-specific metrics (context precision, context recall, faithfulness) are used. Thus, it aims to optimize by analyzing the quality of information retrieval and the accuracy of the generated responses.

Important parameters in RAG problems [44] are chunk size [45] and overlap [46]. Chunk size refers to the maximum number of tokens/words each piece will contain when dividing a document into semantically and functionally meaningful units [45]. Overlap refers to the number of tokens/words shared between consecutive chunks and is a redundancy mechanism [45] used to minimize information loss [46].

6. Results and Analysis

This section reports the performance of LLMs within the proposed RAG framework, focusing on Precision, Recall, F1, and ROUGE scores. As shown in Table 1, the LLaMA3.1:8b model achieved the best overall balance, with precision (0.891), F1 (0.746), and the highest ROUGE scores. Gemma3:12b followed, with F1 (0.735), and ROUGE-1 (0.720), confirming its strong performance. In contrast, Qwen:14b performed considerably worse, with precision 0.561, recall 0.539, F1 0.522, and low ROUGE values, especially ROUGE-2 (0.369). Finally, DeepSeek-R1:14b showed asymmetric behavior, achieving high recall (0.824) but very low precision (0.272), indicating good data capture but weak accuracy.

Table 1. RAG Model Text Quality Metrics

Model	Precision	Recall	F1 Score	ROUGE-1	ROUGE-2	ROUGE-L
Qwen:14b	0.561	0.539	0.522	0.498	0.369	0.465
Gemma3:12b	0.858	0.664	0.735	0.720	0.548	0.622
LLaMA3.1:8b	**0.891**	0.664	**0.746**	**0.747**	**0.680**	**0.723**
DeepSeek-r1:14b	0.272	**0.824**	0.404	0.307	0.245	0.299

Table 2. Rag Model Retrieval Performance And Efficiency

Model	Recall@K	NDCG@K	BLEU Score	Avg Response Time
Qwen:14b	0.335	0.993	0.261	6.10s
Gemma3:12b	0.335	0.993	0.420	5.79s
LLaMA3.1:8b	0.335	0.993	**0.570**	4.02s
DeepSeek-r1:14b	0.335	0.993	0.136	18.54s

Table 3. Rag Model Semantıc Quality Assessment

Model	Context Relevancy	Faithfulness	Answer Augmentation Acc
Qwen:14b	0.712	0.730	0.850
Gemma3:12b	0.712	0.901	0.915
LLaMA3.1:8b	0.712	**0.963**	0.818
DeepSeek-r1:14b	0.712	0.389	0.793

Table 4. Qwen:14b Performance Metrics Across Different Chunk Size Configurations

Chunk Config	Ans.	F1 Score	R1	BLEU	Faith.	Sim.	Time
Small (32/8)	0.588	0.588	0.538	0.249	0.778	0.893	10.51
Medium (64/16)	0.377	0.377	0.317	0.093	0.536	0.869	**6.77**
Large (128/32)	0.371	0.371	0.339	0.202	0.536	0.858	12.95
XLarge (1024/64)	0.270	0.270	0.235	0.023	0.606	0.774	11.78
Small (64/8)	0.780	0.780	0.768	0.421	0.814	0.910	13.20
Medium (128/16)	0.235	0.235	0.198	0.029	0.556	0.737	7.56
Medium (256/32)	**0.843**	**0.843**	**0.871**	**0.620**	**0.904**	**0.936**	10.85
Large (512/64)	0.216	0.216	0.190	0.022	0.455	0.718	11.76

Table 5 LLaMA3.1:8B Performance Metrics Across Different Chunk Size Configurations

Chunk Config	Ans.	F1 Score	R1	BLEU	Faith.	Sim.	Time
Small (32/8)	0.392	0.392	0.333	0.018	0.954	0.792	5.45
Medium (64/16)	0.043	0.043	0.036	0.000	0.800	0.107	**2.45**
Large (128/32)	0.423	0.423	0.361	0.021	**0.954**	0.794	2.94
XLarge (1024/64)	0.294	0.294	0.282	0.134	0.953	0.883	4.78
Small (64/8)	0.676	0.676	0.682	0.366	**0.954**	0.804	3.50
Medium (128/16)	0.410	0.410	0.426	0.178	0.946	0.757	3.77
Medium (256/32)	**0.911**	**0.911**	**0.926**	**0.620**	**0.954**	**0.937**	4.41
Large (512/64)	0.233	0.233	0.222	0.007	**0.954**	0.549	3.81

Table 6 Gemma3:12B Performance Metrics Across Different Chunk Size Configurations

Chunk Config	Ans.	F1 Score	R1	BLEU	Faith.	Sim.	Time
Small (32/8)	0.526	0.526	0.478	0.061	0.938	0.862	8.62
Medium (64/16)	0.548	0.548	0.521	0.076	0.810	0.892	4.21
Large (128/32)	0.636	0.636	0.667	0.214	0.840	0.927	4.67
XLarge (1024/64)	0.214	0.214	0.196	0.033	0.953	0.635	7.54
Small (64/8)	0.615	0.615	0.650	0.128	0.833	0.913	4.67
Medium (128/16)	0.481	0.481	0.413	0.049	0.846	0.724	**3.30**
Medium (256/32)	**0.950**	**0.950**	**0.949**	**0.807**	**0.974**	**0.944**	6.06
Large (512/64)	0.536	0.536	0.485	0.091	0.800	0.789	4.77

TABLE 7. DeepSeek-R1:14B Performance Metrics Across Different Chunk Size Configurations

Chunk Config	Ans.	F1 Score	R1	BLEU	Faith.	Sim.	Time
Small (32/8)	0.393	0.393	0.299	0.133	0.379	0.722	15.82
Medium (64/16)	0.515	0.515	0.423	**0.240**	0.506	**0.772**	**14.65**
Large (128/32)	0.459	0.459	0.376	0.204	0.394	0.663	17.47
XLarge (1024/64)	0.123	0.123	0.085	0.016	0.324	0.688	35.45
Small (64/8)	**0.522**	**0.522**	0.357	0.154	**0.612**	0.755	21.75
Medium (128/16)	0.268	0.268	0.225	0.079	0.398	0.742	17.90
Medium (256/32)	0.519	0.519	**0.444**	0.202	0.496	0.548	18.55
Large (512/64)	0.166	0.166	0.142	0.027	0.305	0.466	25.70

Table 2 shows the model's retrieval metrics. All models used in this study performed similarly with Recall@K 0.335 and NDCG@K 0.993. This indicates that the evaluated retrieval infrastructure is similar. The NDCG@K value of 0.993 indicates that the models rank highly. However, the Recall@K value of 0.335 indicates that approximately one-third of the relevant documents were successfully retrieved during retrieval, suggesting that improvement is needed in this area. The LLaMA3.1:8b model achieved a BLEU score of 0.570, making it the highest-performing model in terms of text generation. This demonstrates that the model's capacity to produce semantically appropriate and grammatically correct responses in Turkish is superior to that of other models. The Gema3:12b model ranked second in the table with a BLEU value of 0.420, while the Qwen:14b model ranked third with 0.261. The DeepSeek-R1:14b model achieved the lowest value in the table with a BLEU score of 0.136 and was found to be inadequate in terms of Turkish text generation. When the models are ranked in response time, the LLaMA3.1:8b model showed the fastest performance with 4.02 seconds. Gemma3:12b ranked second in the table with 5.79 seconds, while the Qwen:14b model produced responses in 6.10 seconds. The DeepSeek R1:14b model performed significantly slower than others, taking 18.54 seconds.

Table 3 compares the performance of different LLM-based RAG models according to semantic quality metrics. The Context Relevancy metric shows a consistent value of 0.712 across all models. This indicates that the choice of retriever or context does not distinguish the models, and context retrieval occurs under equal conditions. Regarding Faithfulness (response consistency), the LLaMA3.1:8b model provides the highest reliability with a value of 0.963. This indicates that the responses generated by the model are highly consistent with the source documents and that the hallucination rate remains low. In contrast, the DeepSeek-R1:14b model achieved a low score of 0.389. This value suggests that the model is inadequate in generating responses based on source documents and shows a greater tendency toward hallucination.

In the Augmentation Accuracy metric, the Gemma3:12b model stands out with a score of 0.915. This result shows that the model accurately reflects the information obtained from the context in its responses. Despite its high faithfulness score, the LLaMA3.1:8b model lags slightly behind with an augmentation accuracy value of 0.818. This indicates that while the model maintains consistency in its responses, it may experience partial limitations in utilizing additional information.

Table 3 summarizes that the LLaMA3.1:8b model provides the highest response reliability with a faithfulness score of 0.963, while the DeepSeek-r1:14b model shows the lowest performance in this metric with a score of 0.389. Table 4 results show that the performance of the Qwen:14b model is susceptible to chunk size and overlap rate. Specifically, the 256/32 configuration achieved the highest performance across all metrics, with values of Answer Correctness (0.843), F1 Score (0.843), ROUGE-1 (0.871), BLEU (0.620), Faithfulness (0.904), and Semantic Similarity (0.936). This finding shows that medium-sized chunk sizes (256 tokens) and controlled overlap rates (32 tokens) provide optimal results in accuracy and consistency, ensuring sufficient context coverage while maintaining information density. In contrast, minimal chunk configurations led to information gaps due to limited context, while extensive chunk configurations caused information dilution in the model's attention mechanism, resulting in low ROUGE-1 and BLEU scores. Therefore, this analysis reveals that carefully optimizing chunk configuration in RAG-based systems is critical in improving response quality.

Table 5 results reveal that the performance of the LLaMA3.1:8b model varies significantly under different chunk size and overlap configurations. Specifically, the 256/32 configuration exhibits the highest performance metrics: Answer Correctness (0.911), F1 (0.911), ROUGE-1 (0.926), BLEU (0.620), Faithfulness (0.954), and Semantic Similarity (0.937) values, achieving the highest success across all metrics and demonstrating that it is the most efficient configuration for the model in terms of both accuracy and source consistency. While small chunk sizes (32/8, 64/8) produced partially reasonable results, the scope of information remained limited, and large chunk sizes (512/64, 1024/64) caused a decline in metrics by distributing information density in the model's attention mechanism. This situation reveals that the combination of a medium-sized chunk (256 tokens) and appropriate overlap (32 tokens) in the LLaMA3.1:8b model maximizes both response accuracy and semantic similarity while maintaining contextual adequacy, thus establishing chunk configuration as a critical optimization parameter in RAG-based applications.

Table 6 shows that the Gemma3:12b model exhibits performance sensitive to different chunk sizes and overlap configurations. Specifically, the 256/32 configuration achieves the highest performance with Answer Correctness (0.950), F1 (0.950), ROUGE-1 (0.949), BLEU (0.807), Faithfulness (0.974), and Semantic Similarity (0.944), demonstrating that it is the most efficient configuration for both answer accuracy and source consistency. Similarly, the 128/32 configuration also demonstrated high performance (Answer Correctness: 0.937, BLEU:0.765), showing that medium-sized chunk sizes yield strong results. In contrast, very small chunk sizes (32/8, 64/16) achieved lower success due to limited information coverage, while very large chunk sizes (1024/64, 512/64) produced low ROUGE-1 and BLEU scores because the model could not effectively process the context density. These findings indicate that medium-sized chunks (especially 256/32) provide optimal information extraction for Gemma3:12b, thus demonstrating that chunk configuration is a critical optimization parameter

in RAG-based systems.

Table 7 results show that the DeepSeek-r1:14b model generally exhibits limited performance under different chunk size and overlap configurations. The highest success was achieved with the 64/8 (Small) configuration (Answer Correctness: 0.522, F1: 0.522, ROUGE-1: 0.357, BLEU: 0.154, Faithfulness: 0.612, Semantic Similarity: 0.755), indicating that the model works more efficiently with relatively small chunks. Although partially balanced results are seen in medium-scale configurations (64/16, 128/32, 256/32), the performance metrics have not reached the level of other models such as Gemma3:12b or LLaMA3.1:8b. Particularly in XLarge (1024/64) and Large (512/64) configurations, the significantly low Answer Correctness and ROUGE-1 scores indicate that the model cannot effectively process long contexts and disperses information density. Furthermore, the fact that faithfulness values generally remain below 0.5 reveals that the model's responses have severe limitations regarding consistency with source documents. These results indicate that smaller chunk sizes are relatively more suitable for DeepSeek-r1:14b.

However, the model's overall performance remains weak compared to other LLMs. It should be carefully evaluated

for reliability in RAG scenarios. When evaluating RAG using these experimental results, chunk optimization demonstrates a critical impact across four different architectures (Qwen:14b, LLaMA3.1:8b, Gemma3:12b, DeepSeek-R1:14b). Qwen, LLaMA3.1, and Gemma3 demonstrated remarkably consistent and optimal performance in the 256/32 chunk configuration. Figure 2 further illustrates the comparison between faithfulness and answer correctness across these architectures, highlighting how standard models maintain both factual consistency and accuracy more effectively than reasoning-focused architectures.

However, the DeepSeek-R1:14b reasoning model deviated from the other models, achieving optimal performance of 0.525 in a 64/16 configuration, while also experiencing low accuracy scores of 0.305-0.612 and high computational costs of 14.65-35. 45 seconds, revealing fundamental challenges for the Turkish RAG problem. These findings provide critical insights for future hybrid RAG-reasoning system developments, demonstrating that architecture independent optimization strategies are effective for standard models. However, specialized architectures require model-specific RAG design approaches.

7. Discussion

The findings reveal that Turkish's unique morphology, rich inflectional system, and complex syntax significantly impact RAG systems. All models perform similarly in terms of contextual relevance and semantic similarity, indicating that the information retrieval process is largely standardized. However, the observed differences in text generation and faithfulness indicate that these depend on language-specific processing capabilities.

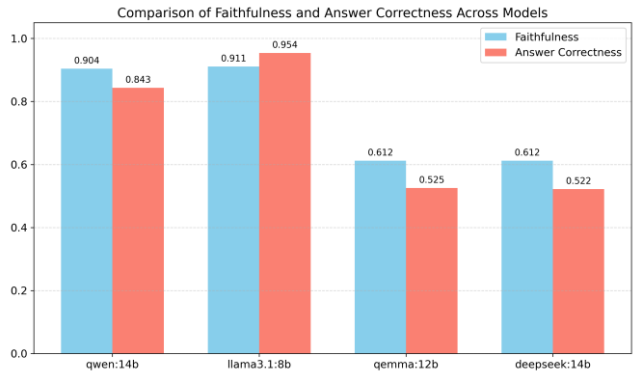

Figure 2. Overview of the comparison between faithfulness and answer correctness metrics across different models.

This analysis reveals that, to achieve the best performance in Turkish RAG systems, model selection must be strategically made according to the requirements of the application context, and that language-specific optimizations play a critical role. By examining the findings, optimal model selection strategies for different application scenarios can be determined. For example, the LLaMA3.1:8B model stands out in reliability focused applications due to its high accuracy performance. This model is the most suitable choice in areas such as academic research, legal consulting, and medical information, where minimizing the risk of hallucination is critical. In scenarios requiring balanced performance, the Gemma3:12B model stands out for its balance between reliability and information augmentation accuracy. The Qwen:14B model offers a combination of moderate accuracy and high augmentation accuracy, making it a suitable option for general-purpose RAG applications. The relatively low faithfulness performance of the DeepSeek-R1:14B model limits its usability in Turkish RAG systems. These findings highlight the necessity of selecting models for Turkish RAG systems based on the requirements of the application context, emphasizing the importance of strategically evaluating the balance between reliability and accuracy.

8. Results and Future Work

This study contributes to the literature by comprehensively evaluating the performance of LLMs in RAG systems on Turkish regulations from the TÜBİTAK institution. Four LLMs—LLaMA3.1:8B, Gemma3:12B, Qwen:14B, and DeepSeek-R1:14B—were systematically compared, providing experimental evidence for model selection and optimization strategies in Turkish NLP. The analysis employed a multidimensional evaluation framework, combining quantitative metrics (precision, recall, F1, ROUGE) with qualitative criteria (faithfulness, augmentation accuracy, computational efficiency).

The findings highlight the impact of Turkish morphology and syntax on model performance, revealing language-specific challenges and opportunities. Retrieval performance proved largely stable across models, as indicated by similar contextual relevance and semantic similarity scores, suggesting that standardized embeddings can reliably capture the Turkish semantic space. In contrast, substantial variations in BLEU and ROUGE scores confirmed that text generation quality is highly sensitive to model architecture. These results emphasize the need for component-specific optimization, particularly in the generation phase.

Overall, the study provides evidence-based guidance for application-oriented model selection in Turkish RAG systems and establishes a benchmark for future research. While LLaMA3.1:8B and Qwen:14B achieved strong results, limitations observed in other models underscore the need for further improvements. Future research should focus on larger, more diverse datasets to enhance generalization, and on developing multimodal RAG systems that integrate textual, visual, and auditory data. Such approaches promise richer, contextually consistent responses and more effective solutions for complex, multidimensional queries, thereby advancing the reliability and applicability of AI-powered information extraction in Turkish contexts.

9. References

[1] Zhou, H., Hu, C., Yuan, Y., Cui, Y., Jin, Y., Chen, C., Wu, H., Yuan, D., Jiang, L., Wu, D., Liu, X., Zhang, J., Wang, X., and Liu, J., "Large language model (LLM) for telecommunications: A comprehensive survey on principles, key techniques, and opportunities", IEEE Communications Surveys & Tutorials, Vol. 27, No. 3, 1955-2005, 2025.

[2] Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z., and Zhang, Y., "A survey on large language model (LLM) security and privacy: The good, the bad, and the ugly", High-Confidence Computing, Vol. 4, No. 2, 100211, 2024.

[3] Li, X., Wang, S., Zeng, S., Wu, Y., and Yang, Y., "A survey on LLM-based multi-agent systems: Workflow, infrastructure, and challenges", Vicinagearth, Vol. 1, No. 1, 9, 2024.

[4] Yue, M., "A survey of large language model agents for question answering", arXiv preprint arXiv:2503.19213, 2025.

[5] Gao, M., Hu, X., Yin, X., Ruan, J., Pu, X., and Wan, X., "LLM-based NLG evaluation: Current status and challenges", Computational Linguistics, 1-27, 2025.

[6] Laskar, M. T. R., Alqahtani, S., Bari, M. S., Rahman, M., Khan, M. A. M., Khan, H., Jahan, I., Bhuiyan, A., Tan, C. W., Parvez, M. R., and others, "A systematic survey and critical review on evaluating large language models: Challenges, limitations, and recommendations", arXiv preprint arXiv:2407.04069, 2024.

[7] Matarazzo, A., and Torlone, R., "A survey on large language models with some insights on their capabilities and limitations", arXiv preprint arXiv:2501.04040, 2025.

[8] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Kuksa, P., Minervini, P., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D., "Retrieval-augmented generation for knowledge-intensive NLP tasks", Advances in Neural Information Processing Systems (NeurIPS), 2020.

[9] Guţu, B. M., and Popescu, N., "Exploring data analysis methods in generative models: From fine-tuning to RAG implementation", Computers, Vol. 13, No. 12, 327, 2024.

[10] Zhong, T., Yang, Z., Liu, Z., Zhang, R., Liu, Y., Sun, H., Pan, Y., Li, Y., Zhou, Y., Jiang, H., and others, "Opportunities and challenges of large language models for low-resource languages in humanities research", arXiv preprint arXiv:2412.04497, 2024.

[11] Joshua, C., Banerjee, A., Kaplan, M. A., Willie, A., Ria, P., and Kalluri, W. A., "Efficient multi-lingual LLM deployment for low-resource languages", 2024.

[12] Cekinel, R. F., Karagoz, P., and Coltekin, C., "Cross-lingual learning vs. low-resource fine-tuning: A case study with fact-checking in Turkish", 2024. [Online]. Available: https://arxiv.org/abs/2403.00411

[13] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., and others, "Retrieval-augmented generation for knowledge-intensive NLP tasks", Advances in Neural Information Processing Systems, Vol. 33, 9459-9474, 2020.

[14] Gupta, S., Ranjan, R., and Singh, S. N., "A comprehensive survey of retrieval-augmented generation (RAG): Evolution, current landscape and future directions", 2024. [Online]. Available: https://arxiv.org/abs/2410.12837

[15] Sharma, S., Yoon, D. S., Dernoncourt, F., Sultania, D., Bagga, K., Zhang, M., Bui, T., and Kotte, V., "Retrieval augmented generation for domain-specific question answering", arXiv preprint arXiv:2404.14760, 2024.

[16] Shi, Y., Xu, S., Yang, T., Liu, Z., Liu, T., Li, X., and Liu, N., "MKRAG: Medical knowledge retrieval augmented generation for medical question answering", AMIA Annual Symposium Proceedings, Vol. 2024, 1011, 2025.

[17] Alan, A. Y., Karaarslan, E., and Aydın, Ö., "Improving LLM reliability with RAG in religious question-answering: MufassirQAS", Turkish Journal of Engineering, Vol. 9, No. 3, 544-559, 2025.

[18] Bikmaz, E., Briman, M., and Arslan, S., "Bridging the language gap in RAG: A case study on Turkish retrieval and generation", Researcher, Vol. 5, No. 1, 38-49, 2025.

[19] Yüksel, A., Köksal, A., Şenel, L. K., Korhonen, A., and Schütze, H., "TurkishMMLU: Measuring massive multitask language understanding in Turkish", arXiv preprint arXiv:2407.12402, 2024.

[20] Kesgin, H. T., Yuce, M. K., Dogan, E., Uzun, M. E., Uz, A., Seyrek, H. E., Zeer, A., and Amasyali, M. F., "Introducing cosmosGPT: Monolingual training for Turkish language models", 2024 International Conference on Innovations in Intelligent Systems and Applications (INISTA), IEEE, 1-6, 2024.

[21] Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., and others, "Qwen technical report", arXiv preprint arXiv:2309.16609, 2023.

[22] Google AI Blog, "Gemma 3: Google's next LLM", 2025. [Online]. Available: https://ai.googleblog.com/2025/03/introducing-Gemma-3-our-most-capable-models.html

[23] IBM News (Think), "Meta releases new LLaMA 3.1 models, including highly anticipated 405B parameter variant", 2024. [Online]. Available: https://www.ibm.com/think/news/meta-releases-LLaMA-3-1-models-405b-parameter-variant

[24] DeepSeek-AI, "DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning", arXiv preprint arXiv:2501.12948, 2025.

[25] Vake, D., Vičič, J., and Tošić, A., "Bridging the question-answer gap in retrieval-augmented generation: Hypothetical prompt embeddings", IEEE Access, 2025.

[26] TÜBİTAK, "Türkiye Bilimsel ve Teknolojik Araştırma Kurumu resmi web sitesi", 2025. [Online]. Available: https://www.tubitak.gov.tr. Accessed: 07.09.2025.

[27] Mao, K., Liu, Z., Qian, H., Mo, F., Deng, C., and Dou, Z., "RAG-Studio: Towards in-domain adaptation of retrieval augmented generation through self-alignment", Findings of the Association for Computational Linguistics: EMNLP 2024, 725-735, 2024.

[28] Arslan, M., Ghanem, H., Munawar, S., and Cruz, C., "A survey on RAG with LLMs", Procedia Computer Science, Vol. 246, 3781-3790, 2024.

[29] Reimers, N., and Gurevych, I., "paraphrase-multilingual-MiniLM-L12-v2", 2025. [Online]. Available: https://huggingface.co/sentence-

transformers/paraphrase-multilingual-MiniLM-L12-v2
[30] "Sentence-BERT: Sentence embeddings using Siamese BERT-networks", Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019.
[31] Facebook AI Research, "Faiss: A library for efficient similarity search and clustering of dense vectors", 2025. [Online]. Available: https://github.com/facebookresearch/faiss
[32] Johnson, J., Douze, M., and Jégou, H., "Billion-scale similarity search with GPUs", IEEE Transactions on Big Data, 2019.
[33] Ahmed, M. A., "From text to understanding the inner text: LLMs and translation accuracy and fluency", International Journal of Language and Literary Studies, Vol. 7, No. 2, 139-156, 2025.
[34] Qwen Team (Alibaba Cloud), "Qwen-14B (base model)", 2025. [Online]. Available: https://huggingface.co/Qwen/Qwen-14B
[35] Kopanov, K., and Atanasova, T., "A comparative pattern analysis of Qwen 2.5 and Gemma 3 text generation", WSEAS Transactions on Information Science and Applications, Vol. 22, 604-615, 2025.
[36] Kamath, A., Ferret, J., Pathak, S., Vieillard, N., Merhej, R., Perrin, S., Matejovicova, T., Ramé, A., Rivière, M., Rouillard, L., and others, "Gemma 3 technical report", CoRR, 2025.
[37] Kassianik, P., Saglam, B., Chen, A., Nelson, B., Vellore, A., Aufiero, M., Burch, F., Kedia, D., Zohary, A., Weerawardhena, S., and others, "LLaMA-3.1-FoundationAI-SecurityLLM-Base-8B technical report", arXiv preprint arXiv:2504.21039, 2025.
[38] DeepSeek Research Team, "DeepSeek R1-14B (model)", 2025. [Online]. Available: https://huggingface.co/DeepSeek-ai/DeepSeek-R1-Distill-Qwen-14B
[39] Yu, H., Gan, A., Zhang, K., Tong, S., Liu, Q., and Liu, Z., "Evaluation of retrieval-augmented generation: A survey", CCF Conference on Big Data, Springer, 102-120, 2024.
[40] Ammar, A., Koubaa, A., Nacar, O., and Boulila, W., "Optimizing retrieval-augmented generation: Analysis of hyperparameter impact on performance and efficiency", arXiv preprint arXiv:2505.08445, 2025.
[41] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J., "BLEU: A method for automatic evaluation of machine translation", Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, 311-318, 2002.
[42] Lin, C.-Y., "ROUGE: A package for automatic evaluation of summaries", Text Summarization Branches Out, 74-81, 2004.
[43] Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P., "SQuAD: 100,000+ questions for machine comprehension of text", arXiv preprint arXiv:1606.05250, 2016.
[44] Şakar, T., and Emekci, H., "Maximizing RAG efficiency: A comparative analysis of RAG methods", Natural Language Processing, Vol. 31, No. 1, 1-25, 2025.
[45] Hladěna, J., Šteflovič, K., Čech, P., Štekerová, K., and Žváčková, A., "The effect of chunk size on the RAG performance", Computer Science Online Conference, Springer, 317-326, 2025.
[46] Stäbler, M., Turnbull, S., Müller, T., Langdon, C., Marx-Goméz, J., and Köster, F., "The impact of chunking strategies on domain-specific information retrieval in RAG systems", 2025 IEEE International Conference on Omni-layer Intelligent Systems (COINS), IEEE, 1-6, 2025.