Enhancing Retrieval and Re-ranking in RAG: A Case Study on Tax Law

Zaid Rustamov
NLP Engineer & Data Scientist
MegaSec LLC
Baku, Azerbaijan
zaid.r@megasec.ai

Mehdi Gasimzade
Junior AI Engineer
MegaSec LLC
Baku, Azerbaijan
mehdigasimzade09@gmail.com

Samir Rustamov
Program Director of BCSC
ADA University
Baku, Azerbaijan
srustamov@ada.edu.az

2025 IEEE 19th International Conference on Application of Information and Communication Technologies (AICT) | 979-8-3315-9342-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/AICT67988.2025.11268677

***Abstract*— This paper explores the effectiveness of various retrieval and re-ranking strategies within a Retrieval-Augmented Generation (RAG) framework, applied to the Azerbaijani Tax Code. We evaluated two sparse retrievers (BM25 and SPLADE) and three dense embedding models (BGE-m3, OpenAI's text-embedding-3-large, and MiniLM's all-MiniLM-L6-v2), comparing their performance across standard information retrieval metrics. Among individual retrievers, BGE-m3 achieved the highest recall of 0.49 at the top 100 retrieved documents but still missed over half of the relevant documents. To address this limitation, we implemented a hybrid retrieval strategy combining BM25 and BGE-m3, which improved recall to 0.60—a relative gain of 11%. Further, we applied cross-encoder re-ranking with the bge-reranker-base model, increasing NDCG from 0.39 to 0.44. These results highlight the importance of layered architecture that integrates both hybrid retrieval and re-ranking to enhance relevance, especially in regulation-heavy domains. Our findings offer practical insights into building robust and interpretable RAG systems for legal and structured text retrieval.**

Keywords*— *Retrieval-Augmented Generation (RAG); Information Retrieval; Re-ranking; Hybrid Retrieval; Sparse and Dense Embeddings; Cross-Encoder; Legal NLP; Azerbaijani Tax Code; Document Ranking; Question Answering

I. Introduction

Retrieval-Augmented Generation (RAG) has become a foundational technique in Natural Language Processing (NLP), combining the strengths of information retrieval and generative models [1]. RAG systems operate in a two-stage pipeline: first, a *retriever* identifies a set of candidate documents that are potentially relevant to a given query; then, a *re-ranker* refines these results to prioritize the most relevant ones [1]. This integration allows language models to generate responses grounded in external knowledge, rather than relying solely on internal parameters [3]. By incorporating retrieved evidence, RAG models significantly reduce hallucinations and improve factual accuracy [4, 5].

They also support knowledge-grounded generation [6] and can be adapted for personalized responses [7], making them suitable for a wide range of applications such as question answering [8], search engines [9], and knowledge-based generation tasks [10]. Recent advancements in dense retrieval, embedding models, and re-ranking techniques have further improved the effectiveness of RAG systems. These methods have demonstrated state-of-the-art performance across multiple NLP benchmarks [11], positioning RAG as a key approach in the development of reliable and knowledge-aware language systems.

Despite the growing popularity and effectiveness of RAG systems, building robust and high-performing pipelines remains a complex task. One of the major challenges lies in the retrieval stage, which significantly influences the quality of the generated response. Two critical issues in retrieval are *missing top-ranked documents* and *incomplete answers*. The first challenge occurs when the answer to a user query exists in the corpus but is ranked too low to be retrieved. For instance, if the relevant document appears at rank 15 but the system only retrieves the top 10 documents (i.e., K=10), the system will miss the correct information, leading to incomplete or incorrect responses. The second challenge involves generating incomplete answers, where the retrieved context contains all the necessary information, but the system outputs only a partial response. These retrieval-related pitfalls not only affect answer completeness but also hinder user trust and system reliability.

Addressing these issues requires both improved retrieval mechanisms and the integration of effective re-ranking strategies to surface the most relevant and diverse information. In this work, we aim to bridge this gap by systematically examining how different embedding strategies, chunking techniques, retrieval configurations, and re-ranking methods influence the overall retrieval quality and, ultimately, the final output relevance.

To support this investigation, we employ a domain-specific dataset composed of question–answer pairs published by The Tax Code of the Republic of Azerbaijan. In this dataset, citizens pose tax-related questions which are then addressed by experts, often accompanied by references to specific articles from The Tax Code. This setting provides a unique opportunity to benchmark retrieval performance: the referenced legal articles serve as a natural ground truth for evaluating whether the RAG system retrieves the correct supporting documents in response to a given question.

II. Literature Review

Retrieval-Augmented Generation (RAG) systems are widely used for enriching language generation with external knowledge. A crucial component of these systems is the information retrieval (IR) module, which ranks documents based on their relevance to a query. The quality of this retrieval largely depends on the embedding models employed. To enhance their performance, contrastive learning is often used, training on query–document triplets to help distinguish relevant content from irrelevant ones [12].

However, the performance of retrieval systems depends not only on the model architecture but also on how the data is preprocessed. The importance of data preprocessing has been emphasized by noting that ambiguous or underspecified

queries can degrade retrieval accuracy [13]. Similarly, it was demonstrated that enriching user queries through expansion or rephrasing can significantly improve retrieval precision [14]. Techniques such as keyword extraction, linguistic simplification, and large language model (LLM)-based expansions have been explored to drive more context and intent into queries.

Alongside improvements in queries, recent research has explored hybrid retrieval methods that aim to overcome the individual limitations of sparse and dense retrieval paradigms. Dense retrieval methods—such as those based on BERT [15] or SentenceTransformers [16] produce continuous vector embeddings that are well-suited for capturing deep semantic relationships. Nevertheless, they often fail to match documents containing exact keywords, proper nouns, or domain-specific phrases. On the other hand, sparse retrieval approaches such as BM25 [17] offer strong keyword matching and better interpretability, but they struggle with semantic generalization and recall in complex or multi-faceted queries [18].

To leverage the complementary strengths of these approaches, hybrid retrieval strategies have been proposed. These techniques combine the scores from dense and sparse retrieval models using ensemble-based methods like linear weighted fusion or Reciprocal Rank Fusion (RRF), improving both recall and precision [18]. Such methods aim to dynamically balance semantic understanding with exact term matching, which is particularly beneficial in specialized domains where both types of signals are crucial. A notable example of this approach is the hybrid retrieval strategy developed in [19]. Their work highlights the inherent limitations of RAG systems that rely solely on either dense or sparse retrieval techniques. In dense retrieval, while embeddings provide rich contextual understanding, they often fall short in matching critical exact terms. Sparse retrieval, conversely, captures specific keywords but lacks broader semantic awareness. To addressed these challenges, the following formula was propose a hybrid scoring mechanism defined as:

$$S_{total} = \alpha * S_{dense} + (1 - \alpha) * S_{sparse} \quad (1)$$

Here, S_{dense} and S_{sparse} represent the scores from dense and sparse retrieval systems respectively, while $\alpha \in [0,1]$ is a tunable hyperparameter that controls their contribution.

As retrieval models have advanced, so too have re-ranking strategies, which aim to reorder initially retrieved documents to better align with the user's query intent. Early approaches focused on pointwise and pairwise learning to rank models, but more recent developments have shifted toward listwise methods such as LambdaRank and ListNet [19], [20]. These listwise approaches consider the entire set of retrieved documents simultaneously, leading to more globally optimized rankings.

Transformer-based architectures and cross-encoders have become dominant in re-ranking tasks due to their ability to model intricate interactions between queries and candidate documents. These models evaluate relevance more precisely by jointly encoding both inputs. In response to the growing capabilities of large language models (LLMs), zero-shot and few-shot re-ranking methods have also emerged. Models like RankT5 and GPT-4 can effectively re-order documents without requiring task-specific fine-tuning, offering flexibility and strong generalization [21]. Furthermore, studies such as [22], [23] demonstrate that embedding models can be specifically fine-tuned or pretrained on Azerbaijani data across various NLP tasks. This not only enhances their performance on language-specific applications but also highlights that LLM-based re-ranking—especially when tailored to the language domain—is a viable and effective solution.

In the context of RAG, where the quality of the final generated output is highly dependent on the relevance of the supporting documents, re-ranking plays a pivotal role. Techniques such as self-consistency checks and noise filtering have been introduced to enhance the factual consistency of generated answers [24]. These methods operate on top of the re-ranked outputs, further refining the input before generation.

While previous research [19] has extensively explored hybrid retrieval and re-ranking in RAG systems, most of this work has been conducted on general-purpose benchmarks or English-language datasets. Our study uniquely applies a tiered approach; combining sparse and dense retrieval with subsequent cross-encoder re-ranking to a highly specialized, domain-specific corpus: the Azerbaijani Tax Code. This focus is critical because legislative texts present unique challenges, such as a reliance on internal cross-references and a high density of domain-specific terminology, which are not captured by standard benchmarks. By evaluating RAG effectiveness on this novel dataset, our work provides a first-of-its-kind analysis of how these techniques perform in a low-resource, legal context and offers valuable insights into the practical application of RAG for a domain where high factual accuracy is paramount.

III. Methodology

A. Data Gathering & Preprocessing

We collected domain-specific data from two primary sources: the Tax Code of the Republic of Azerbaijan and a publicly available Questions & Answers (QA) section published by the State Tax Service [25], [26]. The QA dataset consists of real inquiries from citizens alongside expert-provided answers, often citing specific legal articles. This allowed us to establish a grounded, reference-based benchmark for evaluating document retrieval quality in a real-world legal context.

Using Python web scraping libraries, we extracted both the full text of the Tax Code and 5,657 QA pairs. For each QA pair, we employed AI agents to extract key annotations: referenced legal articles under two categories—primary articles ("*Referenced Main Articles*") and supplementary legal sources ("*Referenced Sub Articles*"), which may include external regulations or secondary references. Since our focus is exclusively on the Tax Code, we filtered out QA pairs that referenced external legal sources (supplementary regulations under "*Referenced Sub Articles*") as these fall outside the scope of our corpus. After this filtering process, we retained 2,178 QA pairs that could be fully answered using the articles within the Tax Code alone. These references provide a gold-standard target to compare against documents retrieved by our RAG system. An illustration of the structured dataset is shown in Fig 1. The corpus contains a total of 240 unique articles, including main and sub-articles (e.g., "Article 214" and "Article 214-1").

Query	Response	Referenced Main Articles	Referenced Sub Articles
Salam. Kirayə mənzil müqaviləsi ilə obyekt kodu alda edə bilərəm? Əgər mümkündürsə, müqavilədə minumum nə qədər göstərə bilərik? Çünki həyat yoldaşım tərəfindən mənə icarə veriləcək deyə simvolik olmasını istəyiram.	Bildiririk ki, sorğunuzda qeyd olunan fəaliyyət Vergi Məcəlləsinin 13.2.37-ci maddəsinə əsasən sahibkarlıq fəaliyyəti hesab olunur və bu fəaliyyətlə məşğul olmaq istəyən fiziki şəxslər Vergi Məcəlləsinin 33-cü və 34-cü maddələrinə uyğun olaraq yaşadıqları yer üzrə vergi orqanına «Fiziki şəxsin sahibkarlıq uçotu haqqında ərizə»	13, 32, 33, 34, 124	13.2.37, 33.2, 124.4
Hörmətli Vergi Xidməti nümayəndələri, Azərbaycan Respublikası Nazirlər Kabinetinin 2024-cü il 22 noyabr tarixli 492 nömrəli Qərarı ilə nümayəndəlik xərclərinin, işçilərin mənzil və yemək xərclərinin, eləcə də əmək şəraiti zərərli, ağır olan və yeraltı işlərdə çalışan işçilərə verilən	Bildiririk ki, Vergi məcəlləsinin 108.1-ci maddəsinə əsasən bu fəsilə uyğun olaraq gəlirdən çıxılmayan xərclərdən başqa, gəlirin əldə edilməsi ilə bağlı olan bütün xərclər, həmçinin qanunla nəzərdə tutulmuş icbari ödənişlər gəlirdən çıxılır. Gəlirdən çıxılan xərclər	108, 109, 119	108.1, 119.2
İşçinin vəfatı ilə əlaqədar əmək müqaviləsinə xitam verildikdə vəfat edənin varəsələrinə ödənilən orta aylıq əmək haqqının 3 misli məbləğində ödənilən müavinatdan gəlir vergisi tutulur ?	Bildiririk ki, Əmək Məcəlləsinin 77-ci maddəsinin 7-ci bəndinə əsasən, işçinin vəfatı ilə əlaqədar əmək müqaviləsinə xitam verildikdə vəfat edənin varəsələrinə orta aylıq əməkhaqqının azı üç misli miqdarında müavinat ödənilir. Vergi Məcəlləsinin 102-ci və "Sosial sığorta haqqında"	102	
Şirkət balansında olan əsas vasaiti qalıq dəyərindən aşağı qiymətə satır. ƏDV hansı məbləğə hesablanmalıdır? Qalıq dəyəri və satış qiyməti arasında fərq xərc kimi tanına bilər mi?	Bildiririk ki, ƏDV məqsədləri üçün vergi tutulan əməliyyatın dəyəri vergi ödəyicisinin müştəridən və ya hər hansı digər şəxsdən aldığı, yaxud almağa hüququ olduğu haqqın ƏDV nəzərə alınmadan məbləği (yol vergisi istisna olmaqla, digər vergilər, rüsumlar və ya başqa yığımlar da daxil olmaqla) əsasında müəyyən edilir. Mallar	142, 143, 161	142.1, 142.2

Fig. 1. Structure of the QA Dataset with Extracted Article References

In parallel, we processed the full legal corpus of the Azerbaijani Tax Code. Due to the document's regulatory nature, which includes dense and interdependent clauses, proper chunking of the text became a critical preprocessing step. Chunking plays a vital role in balancing semantic completeness with retrieval granularity—too small chunks fragment contextual meaning, while overly large ones introduce irrelevant content [27].

To analyze the structure of the articles, we first measured their token distributions. The mean article length was approximately 1,500 tokens, with a median of 670 tokens. Given this variability, we adopted a fine-grained chunking strategy. The text was segmented into chunks of 300 tokens with a 60-token overlap, ensuring continuity and minimizing the risk of splitting sentences mid-thought. It is an important consideration in legal documents where precision and coherence are critical. The choice of a 300-token chunk size was a strategic decision to balance the preservation of semantic context with computational efficiency. A smaller chunk size might fragment sentences or lose key information, making it difficult for the embedding models to capture the full meaning of legal provision. Conversely, an overly large chunk could dilute the core relevance signal within the retrieved passage, and it risks exceeding the context window limitations of certain models, a phenomenon often referred to as "lost in the middle." The 300-token size was selected to ensure each chunk contained a cohesive legal concept while remaining concise enough for effective vector embedding and subsequent re-ranking.

B. Information Retrieval Framework

Following the completion of data collection and preprocessing, we proceeded to the retrieval stage by embedding both the questions and the chunked articles using a variety of models capable of generating either sparse or dense vector representations. For sparse embeddings, we selected two retrieval models: *BM25* and *SPLADE*. To represent dense vectors, we employed the *BGE-m3*, *text-embedding-3-large* from OpenAI, and *all-MiniLM-L6-v2* models.

Each question and document chunk was embedded using the same respective model to ensure vector compatibility. Given that the embeddings share the same dimensional space, we utilized cosine similarity as the distance metric to calculate the semantic closeness between the query and each document. For each question, we retrieved the top-k most similar chunks, with k set to 10, 50, and 100.

To systematically evaluate the retrieval effectiveness of each embedding model, we adopted the following standard information retrieval metrics:

- Recall (R)
- Precision (P)
- Hit Ratio (Hit)
- Normalized Discounted Cumulative Gain (NDCG)
- Mean Reciprocal Rank (MRR)
- Mean Average Precision (MAP)

The complete results for all models across different top-k values are presented in Table 1. Among the dense models, BGE-m3 exhibited a consistent performance advantage, outperforming other models across nearly all metrics and retrieval depths. Increasing the number of retrieved documents generally improves recall, confirming that a broader retrieval set enhances the likelihood of including relevant articles. However, even at k = 100, the best-performing models still fall short of ideal coverage. For instance, *BGE-m3*, *text-embedding-3-large and all-MiniLM-L6-v2* achieved a recall of only 0.49 - indicating that, on average, more than half of the relevant documents are still missing, even under generous retrieval settings.

TABLE I. RETRIEVAL PERFORMANCE OF SPARSE AND DENSE EMBEDDING MODELS ACROSS TOP K DOCUMENTS

Top K Ret. Docs	Metric	Sparse		Dense		
		BM25	Splade	BGE-m3	text-embed-ding-3-large	all-MiniLM-L6-v2
10	R	0.19	0.14	**0.25**	0.23	0.11
	P	0.09	0.07	**0.12**	0.11	0.06
	Hit	0.66	0.54	**0.82**	0.78	0.45
	NDCG	0.20	0.14	**0.29**	0.26	0.10
	MRR	0.41	0.26	**0.58**	0.52	0.18
	MAP	0.12	0.07	**0.18**	0.15	0.05
50	R	0.36	0.35	**0.41**	0.40	0.35
	P	0.04	0.04	**0.04**	0.04	0.04
	Hit	0.90	0.87	**0.96**	0.95	0.85
	NDCG	0.27	0.22	**0.36**	0.33	0.19
	MRR	0.43	0.29	**0.59**	0.53	0.21
	MAP	0.14	0.09	**0.20**	0.18	0.07
100	R	0.47	0.48	**0.49**	**0.49**	**0.49**
	P	0.02	0.02	**0.03**	**0.03**	**0.03**
	Hit	0.96	0.95	**0.98**	**0.98**	0.94
	NDCG	0.31	0.26	**0.39**	0.36	0.23
	MRR	0.43	0.29	**0.60**	0.54	0.21
	MAP	0.15	0.10	**0.21**	0.19	0.08

While recall indicates whether relevant documents are retrieved, it does not consider their position in the ranked list. In contrast, NDCG is a widely used metric in information

retrieval that evaluates not only the presence of relevant documents but also their ranking order. It rewards systems that retrieve highly relevant documents near the top of the result list and penalizes those that rank them lower. The general formula for NDCG at position k is defined as:

$$NDCG_k = \frac{DCG_k}{IDCG_k} \tag{2}$$

Where,

- DCG (Discounted Cumulative Gain) is calculated as:

$$DCG_k = \sum_{i=1}^{K} \frac{relevance\ score\ of\ the\ item\ at\ i}{\log_2(i+1)} \tag{3}$$

- IDCG (Ideal Discounted Cumulative Gain) represents the maximum possible DCG achievable by an ideal ranking.

By comparing DCG with IDCG, NDCG normalizes the score between 0 and 1, allowing for fair comparison across different queries. A higher NDCG score indicates that the system not only retrieved the correct documents but also ranked them effectively.

Despite BGE-m3 demonstrating strong recall, its NDCG performance still reflects the limitations of the retrieval step. As shown in Figure 2, even at k = 100, the NDCG score of *BGE-m3* peaked at only 0.39, highlighting that many of the relevant documents were ranked too low in the result list. This underscores the importance of re-ranking mechanisms, which can reorder retrieved documents to better reflect their true relevance, especially in applications like legal question answering.

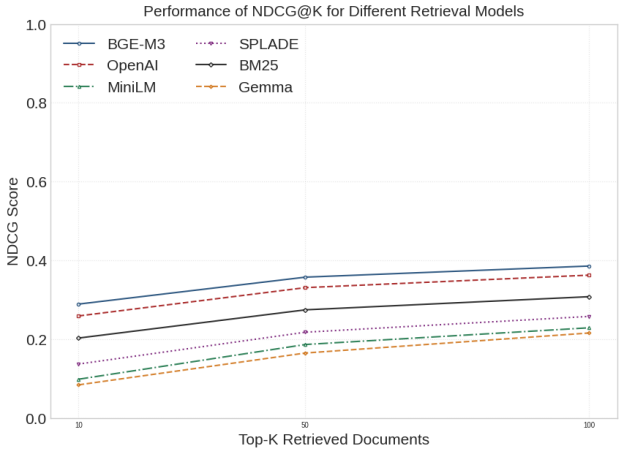

Fig. 2. NDCG Scores of Embedding Models Across Top-k Retrieval Sizes

The relatively low recall observed in the standalone retrieval phase—where even the best-performing model, *BGE-m3*, achieved a recall of only 0.49—indicates that over half of the relevant documents are not being retrieved. To address this issue, we implemented a hybrid retrieval approach, inspired by the method proposed in [25], which combines the strengths of both sparse and dense retrieval techniques. In this strategy, document scores are computed as a weighted combination of sparse and dense similarity scores.

For the sparse component, we selected *BM25*, as it consistently outperformed *SPLADE* in our earlier evaluations. For the dense embeddings, we experimented with both OpenAI's *text-embedding-3-large* and *BGE-m3*, given their relatively strong individual performances. We tested three different values for the weight parameter α: 0.3, 0.5, and 0.7, to analyze the sensitivity of performance to the balance between sparse and dense scores. The results showed consistent improvements in recall across all top-k values. The most successful combination was *BM25* + *BGE-m3*, which achieved a recall of 0.60, representing a significant improvement of 11% over the best individual model (*BGE-m3*, 0.49). These results, summarized in Table 2, demonstrate that hybrid retrieval strategies can effectively mitigate the recall limitations of retrieval systems.

TABLE II. RECALL SCORES OF HYBRID RETRIEVAL COMBINATIONS ACROSS TOP-K VALUES

Top K Ret Docs	Alpha (threshold)	Sparse Vector	Dense Vector	Recall
10	0.3	BM25	Text-embedding-3-large	0.26
			BGE-m3	0.27
	0.5	BM25	Text-embedding-3-large	0.27
			BGE-m3	0.28
	0.7	BM25	Text-embedding-3-large	0.27
			BGE-m3	**0.29**
50	0.3	BM25	Text-embedding-3-large	0.46
			BGE-m3	0.46
	0.5	BM25	Text-embedding-3-large	0.47
			BGE-m3	0.48
	0.7	**BM25**	Text-embedding-3-large	0.48
			BGE-m3	**0.49**
100	0.3	BM25	Text-embedding-3-large	0.58
			BGE-m3	0.57
	0.5	BM25	Text-embedding-3-large	0.57
			BGE-m3	0.59
	0.7	BM25	Text-embedding-3-large	0.58
			BGE-m3	**0.60**

C. *Re-ranking Methods*

Despite improvements in recall through hybrid retrieval, the ranking quality of retrieved documents are poor, as reflected by low NDCG and MRR scores. At k = 100, the highest NDCG score reached only 0.38 (see Table 1), indicating that even when relevant documents were successfully retrieved, they often appeared lower in the ranking. The decision to re-rank the top 100 retrieved documents represent an optimal balance between computational cost and retrieval performance. Re-ranking the entire corpus is unnecessary given that the initial retrieval stage (BM25 + BGE-m3) has already narrowed down the most probable candidates. Re-ranking fewer documents (e.g., top 10 or top 20) would risk excluding a relevant document that was correctly retrieved by the initial hybrid model but ranked lower in the top-k list. The Mean Reciprocal Rank

(MRR)—which evaluates how early the first relevant document appears in the ranked list was underwhelming. MRR is calculated as the average reciprocal rank of the first relevant result across all queries, using the formula:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \quad (4)$$

Where $rank_i$ represents the position of the first relevant document for query i.

In our baseline retrieval setting, the best MRR score was achieved by the BGE-m3 model at 0.60 (see Table 1), which remains relatively low. These findings underscore the critical need for re-ranking mechanisms to reorder the initially retrieved documents, ensuring that the most relevant ones are prioritized at the top of the list and improving the overall quality of the RAG system.

To enhance the quality of the retrieved document rankings, we employed cross-encoder models. Unlike bi-encoders that compute independent embeddings for queries and documents, cross-encoders are specifically designed to assess the relationship between input pairs directly. This architecture allows the model to capture fine-grained semantic interactions, making them highly suitable for precise relevance scoring in re-ranking tasks.

We selected three widely used cross-encoder models: *ms-marco-MiniLM-L6-v2*, *bge-reranker-base and bert-multilingual-passage-reranking-msmarco*. Re-ranking was applied to the top 100 retrieved documents from the initial retrieval step. This larger candidate set provides better recall coverage and allows us to evaluate the model's ability to prioritize the most relevant documents effectively.

The results are presented in Table 3. The *ms-marco-MiniLM-L6-v2* model demonstrated no effectiveness in both NDCG and MRR scores. In contrast, the *bge-reranker-base* model exhibited notable gains, especially in NDCG, where it improved the score from 0.39 to 0.44. However, it also performed a decrease in MRR, indicating a failure to bring the first relevant documents closer to the top of the list.

TABLE III. IMPACT OF CROSS-ENCODER RE-RANKING ON RETRIEVAL QUALITY

Retrieved by	**Metric**	**Before reranking**	**Ms-marco-MiniLM-L6-v2**	**Bert-multilingual-passage-reranking-msmarco**	**Bge-reranker-base**
BM25	NDCG	0.31	0.27	**0.36**	**0.39**
	MRR	0.43	0.16	0.36	0.37
Text-embedding-3-large	NDCG	0.36	0.32	**0.40**	**0.42**
	MRR	0.54	0.19	0.36	0.38
BGE-m3	NDCG	0.39	0.33	**0.43**	**0.44**
	MRR	0.60	0.20	0.37	0.39

D. Error Analysis & Discussion of Limitations

To provide a more nuanced understanding of our quantitative results, we conducted a qualitative analysis of our system's performance, focusing on specific examples where our hybrid retrieval and re-ranking strategies exhibited nuanced behaviors. For example, for the following query, *BGE-m3*, a dense model, successfully retrieved Articles 108 and 119 but failed to retrieve Article 109. While Article 119 was ranked highly in the results, Article 108 appeared much lower in the list. A qualitative analysis reveals two key reasons for these retrieval issues:

- Semantic Disconnection: Article 109 was not retrieved because it references other legal articles and is not a self-contained, complete article. This inherent lack of direct semantic content makes it difficult for a dense model like BGE-m3 to establish a strong vector similarity with the query. The model struggles to follow external references and cannot capture the full context of a fragmented legal provision.
- Re-ranking Failure: While Article 108 was correctly retrieved, its lower rank in the initial list presented a challenge for the re-ranker. The purpose of a re-ranking model, such as bge-reranker-base, is to fix these initial ranking issues and bring the most relevant documents to the forefront. However, it appears the re-ranker failed to sufficiently promote Article 108. The reason for this is that Article 108 provides general, rather than specific, information. The cross-encoder may have evaluated its overall relevance as lower than other documents.

These retrieval-related pitfalls not only affect answer completeness but also hinder user trust and system reliability, as noted in the introduction. They highlight the fundamental challenge that even advanced RAG systems face when dealing with complex, interdependent legal documents.

A potential solution to this problem would be to develop graph-based architecture for the legal codes. By explicitly mapping the relationships between articles that reference each other, a knowledge graph could provide a structured framework. This would allow the retrieval system to traverse these connections, ensuring that even articles with incomplete textual information are retrieved by following their links to other relevant documents. This approach could effectively bridge the semantic gaps that a pure vector-based system struggles with and provide a more robust and accurate retrieval mechanism.

Original Query:

Hörmətli Vergi Xidməti nümayəndələri, Azərbaycan Respublikası Nazirlər Kabinetinin 2024-cü il 22 noyabr tarixli 492 nömrəli Qərarı ilə nümayəndəlik xərclərinin, işçilərin mənzil və yemək xərclərinin, eləcə də əmək şəraiti zərərli, ağır olan və yeraltı işlərdə çalışan işçilərə verilən müalicə-profilaktik yeməklər, süd və ona bərabər tutulan digər məhsullar və vasitələrlə bağlı xərclərin vergitutma məqsədləri üçün gəlirdən çıxılması normaları və qaydaları müəyyən edilmişdir. Qərarda "nümayəndəlik xərcləri" anlayışı qeyd olunsa da, bu anlayışın dəqiq məzmun dairəsi ilə bağlı praktikada qeyri-müəyyənlik yaranır. Sualımız: Qərarda nəzərdə tutulan "nümayəndəlik xərcləri" anlayışına işçilərə verilən yemək pulu və mənzil xərcləri də daxildirmi? Yəni bu cür xərclər də "nümayəndəlik xərcləri" kimi qiymətləndirilib, vergitutma məqsədləri üçün yalnız bu xərclərin 50 faizi və illik gəlirin 1%-i həddində gəlirdən çıxıla bilərmi? Yoxsa bu xərclər nümayəndəlik xərcləri kateqoriyasına daxil edilməyərək ayrıca tam şəkildə gəlirdən çıxılan xərclər kimi qiymətləndirilir?

Translated:

Dear Representatives of the Tax Service. According to Decision No. 492 of the Cabinet of Ministers of the Republic of Azerbaijan, dated November 22, 2024, the norms and rules for deducting, for taxation purposes, expenses related to representation, employees' housing and meal costs, as well as expenses for medical and preventive meals, milk, and equivalent products and supplies provided to employees working under harmful, strenuous, or underground conditions, have been established. Although the Decision refers to the concept of "representation expenses," in practice there is uncertainty regarding the exact scope of this concept. Our question is: Does the concept of "representation expenses" as set out in the Decision also include meal allowances and housing expenses provided to employees? In other words, should such expenses be classified as "representation expenses" and therefore deductible for taxation purposes only within the limit of 50 percent of such expenses and 1 percent of annual income? Or should these expenses not be treated as representation expenses, but rather be considered as separate deductible expenses that can be fully deducted from income?

True Article in the Tax Code: {**108, 109, 119**}

Retrieved Documents by BGE-m3 Model:

{**119**, 149, 13, 90, 125, 125, 14-1, 13, 33, 159, 104, 114, 105, 124, 165, 104, 227, 14-1, 4, 124, 18, 228, 220, 4, 150, 166, 114, 23, 209, 59, 19, 149, 19, 130, 108, 99, 168, 161, 13, 96, 79, 197, 89, 125, 16, 102, 24, 42, 132, 199, 79, 14, 85, 2, 199, 164, 2, 125, 36, 164, 218, 172, 227, 31, 102, 33, 13, 102, 13, 13, 33, 174, 35, 55, 211, 13, 2, 58, 23, 53, 2, 159, 143, 14-1, 13, 102, 53, 14-1, 13, 13, 50, 13, 50-1, 58, 164, 90, 104, 83, **108**, 101}

[a.] Example of failed query

IV. Discussion of Results

Our experiments revealed meaningful distinctions in retrieval and ranking effectiveness across various sparse and dense retrievers. BGE-m3 consistently outperformed both traditional sparse models like BM25 and SPLADE, as well as other dense models such as OpenAI's text-embedding-3-large, and MiniLM's all-MiniLM-L6-v2 across all standard metrics, including Recall, NDCG, and MRR. Nevertheless, even this best-performing retriever only achieved a Recall of 0.49 at the top 100 documents, indicating that over half of the truly relevant documents were still being missed, underscoring a fundamental limitation in relying solely on initial retrieval. To address this, we explored a hybrid retrieval strategy, combining the keyword precision of BM25 with the contextual depth of BGE-m3 through a weighted scoring mechanism. This approach yielded a notable increase in Recall, improving it to 0.60 at the top 100 documents. This represented a significant relative gain of 11% over the best individual model and demonstrated the complementary nature of sparse and dense retrieval. Hybridization effectively mitigated the limitations of each model: sparse methods excelled at matching domain-specific terminology, while dense models better captured paraphrased or semantically rich queries. Despite this improvement, many relevant documents were still ranked too low, as reflected in relatively modest NDCG and MRR scores. For example, the highest NDCG for any single retriever peaked at just 0.39, indicating that even when relevant documents were retrieved, they often appeared too low in the list to be useful. We therefore applied cross-encoder re-ranking with the bge-reranker-base model to the top 100 retrieved candidates. This re-ranking step significantly improved the NDCG score from 0.39 to 0.44, highlighting its effectiveness in reordering the document list to prioritize relevance. However, it failed to raise the MRR score, suggesting a limitation in its ability to consistently promote the single most relevant document to the top position. Overall, the findings advocate for a layered retrieval architecture, where strong initial retrieval is enhanced through hybrid scoring and further refined through targeted re-ranking, resulting in a more robust and semantically aware RAG pipeline.

V. Conclusion

This study explored the effectiveness of various retrieval and re-ranking strategies within a Retrieval-Augmented Generation (RAG) framework applied to the Azerbaijani Tax Code. Through extensive evaluation, we demonstrated that while dense models like *BGE-m3* offered strong retrieval performance, hybrid methods combining sparse and dense signals significantly improved recall. Furthermore, incorporating cross-encoder re-ranking—particularly with the *BGE reranker* proved effective in surfacing relevant documents higher in the ranked list. These findings underscore the importance of a multi-stage retrieval pipeline that combines methods to enhance both coverage and ranking quality, especially in structured and domain-specific settings.

Acknowledgment

This research was conducted at the Center for Data Analytics Research of ADA University and the AI Laboratory of MegaSec LLC.

REFERENCES

[1] Ren, R., Qu, Y., Liu, J., Zhao, W. X., She, Q., Wu, H., ... & Wen, J. R. (2021). RocketQAv2: A Joint Training Method for Dense Passage Retrieval and Passage Re-ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

[2] Yates, A., Nogueira, R., & Lin, J. (2021, March). Pretrained transformers for text ranking: BERT and beyond. In Proceedings of the 14th ACM International Conference on web search and data mining (pp. 1154-1156).

[3] Izacard, G., & Grave, E. (2021, April). Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering. In *EACL 2021-16th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 874-880). Association for Computational Linguistics.

[4] Agrawal, G., Kumarage, T., Alghamdi, Z., & Liu, H. (2024, June). Can Knowledge Graphs Reduce Hallucinations in LLMs?: A Survey. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)* (pp. 3947-3960).

[5] Shuster, K., Poff, S., Chen, M., Kiela, D., & Weston, J. (2021). Retrieval Augmentation Reduces Hallucination in Conversation. *Findings of the Association for Computational Linguistics: EMNLP 2021*.

[6] Yih, S. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. In Conference on Neural Information Processing Systems, Vancouver, Canada.

[7] Salemi, A., Kallumadi, S., & Zamani, H. (2024, July). Optimization methods for personalizing large language models through retrieval augmentation. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 752-762).

[8] Liu, Z., Zhou, Y., Zhu, Y., Lian, J., Li, C., Dou, Z., ... & Nie, J. Y. (2024, May). Information retrieval meets large language models. In Companion Proceedings of the ACM Web Conference 2024 (pp. 1586-1589).

[9] Xiong, H., Bian, J., Li, Y., Li, X., Du, M., Wang, S., ... & Helal, S. (2024). When search engine services meet large language models: visions and challenges. IEEE Transactions on Services Computing.

[10] Long, X., Zeng, J., Meng, F., Ma, Z., Zhang, K., Zhou, B., & Zhou, J. (2024, March). Generative multi-modal knowledge retrieval with large language models. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 38, No. 17, pp. 18733-18741).

[11] Salemi, A., & Zamani, H. (2024, July). Evaluating retrieval quality in retrieval-augmented generation. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 2395-2400).

[12] Karpukhin, V., Oguz, B., Min, S., Lewis, P. S., Wu, L., Edunov, S., ... & Yih, W. T. (2020, November). Dense Passage Retrieval for Open-Domain Question Answering. In *EMNLP (1)* (pp. 6769-6781).

[13] Gao, Y., Xiong, Y., Wang, M., & Wang, H. (2024). Modular RAG: Transforming RAG Systems into LEGO-like Reconfigurable Frameworks. *CoRR*.

[14] Patel, C. (2024). Hypothetical Retrieval-Augmented Generation (Hypothetical RAG): Advancing AI for Enhanced Contextual Understanding and Creative Problem-Solving. Scientific Research Journal of Science, Engineering and Technology, 2(1), 1-4.

[15] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 4171-4186).

[16] Reimers, N., & Gurevych, I. (2019, November). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (pp. 3968-3977).

[17] Wang, S., Zhuang, S., & Zuccon, G. (2021, July). Bert-based dense retrievers require interpolation with bm25 for effective passage retrieval. In Proceedings of the 2021 ACM SIGIR international conference on theory of information retrieval (pp. 317-324).

[18] Sengupta, S., Heaton, C., Cui, S., Sarkar, S., & Mitra, P. (2024, December). Towards Efficient Methods in Medical Question Answering using Knowledge Graph Embeddings. In 2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) (pp. 5089-5096). IEEE.

[19] Burges, C. J. (2010). From ranknet to lambdarank to lambdamart: An overview. Learning, 11(23-581), 81.

[20] Liu, Y., Zhang, X., Zhu, X., Guan, Q., & Zhao, X. (2017). Listnet-based object proposals ranking. Neurocomputing, 267, 182-194.

[21] Zhuang, H., Qin, Z., Jagerman, R., Hui, K., Ma, J., Lu, J., ... & Bendersky, M. (2023, July). Rankt5: Fine-tuning t5 for text ranking with ranking losses. In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 2308-2313).

[22] Guliyev, N., Rustamov, Z., & Rustamov, S. (2024, April). Analysis of public sentiment in Azerbaijani news and social media. In Proceedings of the International Conference on Computing, Machine Learning and Data Science (pp. 1-6).

[23] Alizada, T., Suleymanov, U., & Rustamov, Z. (2024, September). Contextualized Word Embeddings in Azerbaijani Language. In 2024 IEEE 18th International Conference on Application of Information and Communication Technologies (AICT) (pp. 1-6). IEEE.

[24] Fang, F., Bai, Y., Ni, S., Yang, M., Chen, X., & Xu, R. (2024, August). Enhancing Noise Robustness of Retrieval-Augmented Language Models with Adaptive Adversarial Training. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 10028-10039).

[25] https://www.taxes.gov.az/az/page/ar-vergi-mecellesi

[26] https://www.taxes.gov.az/az/page/suallar-ve-cavablar

[27] Chu, Y., He, P., Li, H., Han, H., Yang, K., Xue, Y., ... & Tang, J. (2025). Enhancing LLM-Based Short Answer Grading with Retrieval-Augmented Generation. *CoRR*.