SoICT Hackathon 2024 - Legal Document Retrieval

Tung Le Thanh, Phuoc Duong-Huy Chu, Duy Hoang Nguyen, Duong Mai
University of Information Technology (UIT), Ho Chi Minh City, Vietnam
23521740@gm.uit.edu.vn, 23521229@gm.uit.edu.vn,
22520329@gm.uit.edu.vn, 22520302@gm.uit.edu.vn

Abstract

This research addresses Vietnamese Legal Text Retrieval, focusing on improving the efficiency and accuracy of retrieving relevant legal documents from large-scale corpora. Utilizing data from the SoICT Hackathon 2024 - Legal Document Retrieval competition, the proposed pipeline leverages a chunk-based data processing approach to enhance document segmentation and retrieval granularity. The system integrates bi-encoder models and BM25 for initial retrieval and employs cross-encoder models for reranking to deliver high-quality results. Evaluated using MRR@10, the approach demonstrates notable performance improvements, providing a robust solution for managing and accessing Vietnamese legal information.

1 Introduction

Legal document retrieval is a fundamental component of legal research, enabling professionals to efficiently access relevant documents, such as case law and statutes. However, the complexity of legal language, with its domain-specific terminology and contextual nuances, poses significant challenges for traditional retrieval systems. This is especially true in the context of Vietnamese legal texts, where linguistic diversity and the rapid expansion of legal corpora further complicate retrieval tasks. Addressing these challenges requires innovative solutions that combine the strengths of traditional information retrieval with advanced neural network models.

Recent research has focused on improving the accuracy and efficiency of legal document retrieval systems. Kim et al. (2015) demonstrated the value of combining information retrieval with textual entailment for legal question answering, emphasizing the importance of understanding the relationships between legal queries and documents (Kim et al., 2015). Building on this, Rabelo et al. (2018) explored the application of pairwise paragraph similarity for case law retrieval and textual entailment, offering a framework for addressing challenges related to statute law and case law retrieval (Rabelo et al., 2018). Furthermore, Shao et al. (2023) proposed an intent taxonomy for legal case retrieval, improving semantic understanding and retrieval accuracy by better categorizing legal queries (Shao et al., 2023).

In parallel, Nguyen et al. (2022) highlighted the effectiveness of attentive deep neural networks for legal document retrieval, demonstrating the superiority of attention mechanisms in handling long and complex legal documents. Their study found that models using attention, such as Attentive CNN and Paraformer, significantly outperformed traditional methods in retrieval tasks across different languages and datasets, including the COLIEE competition dataset (Nguyen et al., 2022). These advancements suggest that combining semantic understanding with attention mechanisms can lead to more efficient and accurate legal document retrieval systems.

This paper proposes a hybrid pipeline for Vietnamese legal document retrieval, which integrates traditional methods like BM25 for initial ranking with bi-encoder models that capture semantic similarity between queries and documents. Additionally, a reranking model refines the results by incorporating deeper contextual understanding. This approach aims to enhance both the precision and relevance of retrieved legal texts, addressing the specific challenges posed by Vietnamese legal corpora and contributing to the development of intelligent tools for legal professionals.

2 Related Work

Sparse vector representations have been a cornerstone of traditional information retrieval (IR) systems. BM25 (Robertson and Zaragoza, 2009), based on term frequency and inverse document frequency, is widely used for ranking documents. Robertson et al. extended BM25 to handle structured documents by applying field-specific weights (Robertson et al., 2009), which improves retrieval precision in domains like legal document retrieval.

Dense retrieval methods, such as Dense Passage Retrieval (DPR) (Karpukhin et al., 2020), have gained attention due to their ability to capture semantic similarity. DPR employs dual encoders to map both queries and documents into dense low-dimensional vectors, enhancing retrieval performance, especially in cases of low term overlap. Luan et al. (2021) investigate the limitations of fixed-length dense encodings and propose a hybrid model combining sparse and dense representations for improved retrieval of long documents. Their model outperforms BM25 in large-scale retrieval tasks, demonstrating the potential of hybrid systems in practical applications.

Hybrid retrieval approaches combine the strengths of both sparse and dense models. These systems typically use traditional sparse methods like BM25 for initial document ranking, followed by reranking using dense models. Gao et al. (2021) introduce CLEAR, a retrieval model that complements BM25 with semantic matching from a neural embedding model. CLEAR improves the efficiency and accuracy of retrieval systems by capturing semantic information that lexical models fail to detect, making it particularly useful for domains such as legal document retrieval.

Multilingual and Multi-Functionality Models: Recent advancements like M3-Embedding (Chen et al.) and E5 (Wang et al.) have significantly improved IR, especially for multilingual legal document retrieval. M3-Embedding supports over 100 languages and can process long documents (up to 8192 tokens), while handling both dense and sparse embeddings. Its self-knowledge distillation technique integrates relevance scores from different retrieval methods, enhancing performance across languages and document granularities. E5, trained using weakly-supervised contrastive learning, excels in both zero-shot and fine-tuned settings, surpassing traditional models like BM25 in multilingual tasks.

These methods highlight the ongoing shift in information retrieval, particularly in legal contexts, where combining semantic understanding with traditional precision can lead to significant improvements in both relevance and efficiency.

3 Methodology

Our system for the Legal Document Retrieval track adopts a hybrid retrieval approach, integrating sparse (BM25) and dense (bge-m3, e5-large multilingual) search models. The methodology consists of preprocessing, fine-tuning the dense retrieval model, hybrid retrieval, and reranking. Below, we detail the components of our system in their proper sequence.

3.1 System Architecture Overview

The architecture of system is shown in Figure 1. The process includes the following stages:

1. Corpus Preprocessing and Chunking: Preparing and chunking the provided corpus into smaller segments to efficiently handle long documents.
2. Fine-Tuning Dense Models: Fine-tuning the bge-m3 model using hard negatives generated from BM25 to improve its ability to retrieve relevant documents.
3. Hybrid Retrieval: Combining sparse (BM25) and dense (bge-m3, e5) retrieval methods for robust candidate selection.
4. Re-Ranking: Using a cross-encoder to rerank top candidate documents for final submission.

3.2 Fine-Tuning Bge-m3 with Hard Negatives

Our approach to improving retrieval performance centered on fine-tuning the BGE-M3 model using hard negatives generated from BM25. The choice of BM25 for hard negative generation was motivated by its proven effectiveness in Information Retrieval tasks (Robertson et al., 2009) and its successful application in training dense retrievers, as demonstrated by (Gao et al., 2021).

In our hard negative generation process, we configured BM25 to retrieve the top-20 documents for each query. From these results, we selected one positive example and used the remaining 19 documents as hard negatives. This approach helps the model learn to distinguish between truly relevant documents and those that appear superficially similar but are actually irrelevant to the query (Zhan et al., 2021).

This approach of using BM25-generated static negatives was chosen over dynamic negative sampling as it provides consistent, challenging negative examples during the training process. The fine-tuning process was designed to enhance the dense model's retrieval capabilities while leveraging BM25's strong lexical matching properties

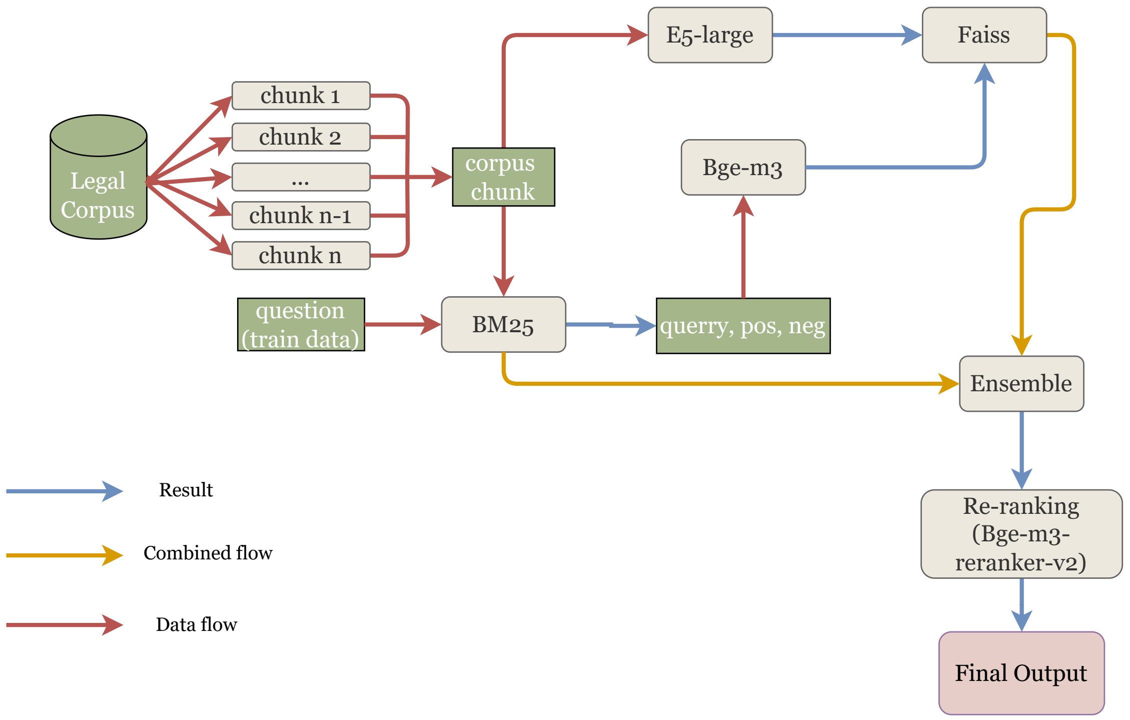

Figure 1: Architecture of our system.

(Gao et al., 2021). Using in-batch training (Mao et al., 2020), we set the hyperparameters as follows: a training group size of 2 (1 positive, 1 hard negative), a batch size of 8 distributed across dual T4 GPUs, 18 epochs, and 14 random negatives per batch for improved generalization.

The Okapi BM25 score for a query-document pair can be calculated as:

$$\mathrm{BM25}(q,d) = \sum_{i=1}^{n} \mathrm{IDF}(i) \cdot \frac{\mathrm{TF}(i,d)\cdot(k_1+1)}{\mathrm{TF}(i,d)+k_1\cdot\left(1-b+b\cdot\frac{|d|}{\mathrm{avgdl}}\right)}$$

Thus, the BM25 score combines the term frequency with the inverse document frequency to determine the relevance of a document to a query. This formula was used in the hybrid retrieval approach for hard negative generation, allowing us to enhance the training of the BGE-M3 model.

3.3 Hybrid Retrieval: Combining Okapi BM25, Bge-m3, and E5

In our hybrid retrieval strategy, we combined the strengths of sparse and dense models. Following empirical testing, we set the top-k parameter to 1500, selecting the top 1500 candidates from the dense models. For Sparse-Dense Integration, Okapi BM25 was first used to select the top 20 candidates, and then these were refined by multiplying with the ensembled dense scores, which were calculated as:

$$\text{ensemble} = \left(w_{\text{bge-m3}} \cdot \mathrm{S}_{\text{bge-m3}} + w_{\text{e5}} \cdot \mathrm{S}_{\text{e5}}\right) \cdot \mathrm{S}_{\text{BM25}}.$$

Finally, the top 100 candidates were selected based on this integrated ranking. The ensemble method is inspired by the approach described in (Do et al., 2023), where a similar combination of retrieval models was used for question answering in the context of legal regulations.

3.4 Re-Ranking with Cross-Encoder

To further improve the quality of results, a cross-encoder was used for reranking. This step involved candidate selection, where the top-100 results from the hybrid retrieval phase were reranked, and cross-encoder architecture, which jointly processes query-document pairs to capture fine-grained semantic relevance. For the re-ranking process, we utilized the `bge-reranker-v2-m3` model, which was fine-tuned to better capture the relevance between query-document pairs. The reranked results were sorted by relevance scores, ensuring that the most relevant documents were prioritized for final submission.

As shown in Figure 2, the cross-encoder architecture allows for more precise ranking by evaluating the query-document pairs in a more semantically aware manner, improving retrieval accuracy. This approach has been proven effective in (Nogueira and Cho, 2020), where a BERT-based model was utilized for reranking to enhance retrieval performance.

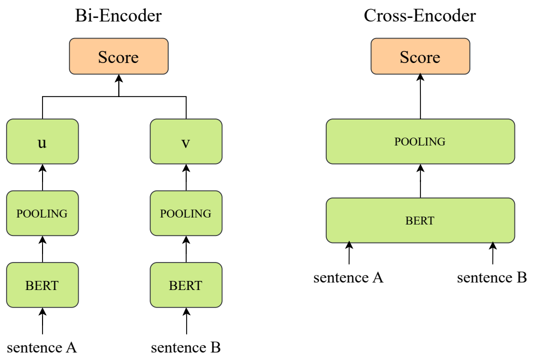

Figure 2: Bi-Encoder and Cross-Encoder Comparison.

4 Experiments

4.1 Dataset and Preprocessing Phase

The preprocessing process begins by splitting the dataset from the competition into training, validation, and test sets using a 70:20:10 ratio. The training sets, which include both questions and their corresponding answers, are used for creating the chunked data. The test set, however, is reserved for evaluating the performance of the bi-encoder and reranking models.

Next, the corpus data which include the answers of legal questions is transformed into a structured JSON format, where each unique identifier (inforid) is mapped to its corresponding text content. This structured format facilitates efficient handling of answer data during the chunking process.

This text data is then chunked into smaller, semantically coherent pieces, each capped at a maximum of 400 words. Each chunk is assigned a unique identifier (chunkid) to ensure traceability and efficient retrieval.

A legal dictionary is subsequently created, mapping each chunkid to its respective chunk text. This dictionary acts as a quick reference during the retrieval and ranking phases.

Finally, questions from the training and validation sets are paired with their corresponding answer chunks based on the cid field. The result is a structured dataset containing question-answer pairs, where answers include both the chunkid and the associated text. This structured format is essential for training and fine-tuning retrieval models.

By organizing the data and dedicating the test set to bi-encoder and reranker evaluation, this preprocessing pipeline ensures a robust foundation for both training and performance assessment.

	Unique CIDs	**Unique QIDs**
Train	53072	83219
Validation	20004	23777
Test	10878	11889

Table 1: Overview of the datasets

4.2 Experimental environment

For our experiments, we used pre-trained models available on Hugging Face. Specifically, we utilized the bge-m3 and multilingual e5-large models for dense retrieval, and the bge-reranker-v2-m3 model for reranking. For fine-tuning bge-m3, we cloned the FlagEmbedding repository from GitHub and followed the provided guidelines for fine-tuning. The training process was monitored and logged using Wandb for effective tracking of metrics and training progress. All experiments were conducted on Kaggle, utilizing two NVIDIA T4 GPUs. The fine-tuning process took approximately 70 hours to complete under this setup.

4.3 Evaluation metrics

For evaluation, we employed the MRR@10 metric, as specified by the SOICT Legal Document Retrieval competition guidelines. Mean Reciprocal Rank (MRR) is a common metric for evaluating ranking systems, and MRR@10 specifically considers the top 10 ranked results. It is defined as:

$$\text{MRR@10} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i}$$

where Q is the set of all queries, and rank_i is the position of the first relevant document in the ranked list for query i, provided it appears within the top 10 results. If no relevant document is found within the top 10, the reciprocal rank for that query is considered zero. MRR@10 values range from 0 to 1, with higher values indicating better retrieval performance.

We divided our evaluation into two phases: Phase 1: bfore Fine-tuning: Using the pre-trained models without additional fine-tuning. Phase 2: after Fine-tuning: Incorporating the fine-tuned bgem3 model to assess the impact of training with BM25-generated hard negatives.

This two-phase evaluation allowed us to measure the effectiveness of our fine-tuning approach in improving retrieval performance.

4.4 Results

As mentioned earlier, we evaluated our system in two phases on both the training and validation datasets. This approach allows for a comprehensive comparison of the results across different methods. Below, we present the results for each phase in detail.

Pretrained Models (MMR10 Scores)		
Model	**Val (MMR10)**	**Test (MMR10)**
BM25	0.6139	0.6052
Bge	0.6660	0.6681
e5	0.6612	0.6576
Bge+e5	0.6804	0.6811
Bge+bm25	0.6534	0.6497
Bge+e5+bm25	0.6493	0.6597
After Fine-Tuned Bge-m3 (MMR10 Scores)		
Model	**Val (MMR10)**	**Test (MMR10)**
Bge	0.5317	0.5555
Bge+e5+bm25	0.7417	0.7193
Bge+bm25	0.7421	0.7348

Table 2: The result First-Retrieval (MMR10 scores)

Reranking (MMR10 Scores)		
First-Retrieval by	**Val (MMR10)**	**Test (MMR10)**
Bge-pretrained + e5	0.7150	0.7155
Bge-tuned+bm25	0.7708	0.7746

Table 3: The result after reranking (MMR10 scores)

Before fine-tuning, the combination of the Bi-Encoder model (Bge) with BM25 did not yield the expected improvement. In fact, the results from the combination of Bge and BM25 were even worse than using the models individually. This is likely due to the fact that combining BM25 with a pre-trained dense retrieval model like Bge might introduce noise or redundant retrieval signals, leading to a less effective overall model.

However, after fine-tuning the Bge model (fine-tuned to Bge-m3), the results showed significant improvement. Specifically, when combining the fine-tuned Bge model with BM25, the MMR10 score jumped to 0.7348 on the test set, which was a notable enhancement from the pretrained Bge+BM25 combination (0.6497). Furthermore, the reranking process also demonstrated substantial gains, with the Bge-tuned+bm25 model achieving 0.7746 on the test set, showcasing how fine-tuning can effectively leverage the information from both the pretrained and traditional retrieval methods, improving the overall ranking and retrieval quality.

4.5 Analyzing Suboptimal Fine-Tuning of BGE Model

In this subsection, we analyze the reasons behind the suboptimal performance of the fine-tuned BGE model. A major contributing factor was the resource constraints we faced during the training phase, particularly with the train group size. Due to limited computational resources, we set the train group size to 2, which significantly restricted the model's ability to learn effectively from hard negatives.

The importance of train group size in dense retrieval model training is well-documented. For instance, the paper *Dense Passage Retrieval for Open-Domain Question Answering* (Karpukhin et al., 2020) demonstrates that increasing the train group size generally improves model performance. A larger train group size provides more hard negatives in each batch, allowing the model to better distinguish between relevant and irrelevant passages. In contrast, a smaller train group size limits this diversity, reducing the model's capacity to learn effectively.

Ideally, the train group size should match the number of hard negatives, which is typically around 20 in dense retrieval settings. This alignment is crucial because, during batch training, all non-positive passages in the same batch are treated as negatives. For example, a batch might consist of

$$[[q, p, n_1, n_2], [q', p', n'_1, n'_2]],$$

where the negatives for the positive passage p include n_1, n_2, p', n'_1, n'_2, and similarly, the negatives for the positive passage p' include n'_1, n'_2, p, n_1, n_2. A smaller train group size, such as 2, limits the availability of these hard negatives, thereby reducing the model's ability to generalize effectively.

Moreover, (Zhan et al., 2020) points out that the BM25 Neg method, which uses top-ranked BM25 documents as negatives during training, can introduce optimization bias. Specifically, this bias arises because the model is trained to avoid documents that share overlapping query terms, even when those documents might be relevant. This can harm the model's ability to accurately identify relevance in dense retrieval tasks. In our case, the small train group size exacerbated these challenges, leading to a drop in performance when combining BGE and BM25 compared to non-fine-tuned models.

This analysis suggests that while BM25 scores provide useful guidance during training, using BM25 top documents as negative examples can be detrimental, especially when the train group size is small. A larger train group size, ideally matching the number of hard negatives, could help mitigate this issue and improve the model's overall performance.

5 Conclusion

In our system, we developed a hybrid retrieval system that combines BM25 and fine-tuned bge-m3 with negatives from BM25 to enhance performance and ensemble score with e5 and BM25. Our approach achieved Top 8 on the private test leaderboard of the SOICT Legal Document Retrieval competition, demonstrating the effectiveness of the system. This combination allowed us to capture both lexical and semantic features, resulting in improved retrieval accuracy. While our system performs well, future work could explore optimizing the hybrid strategy further and applying it to other complex retrieval domains to generalize its effectiveness.

Limitations

Despite achieving Top 8 on the private test leaderboard, our system has several limitations that present opportunities for future improvement. A primary constraint was the limited computational resources available for training. We relied on a T4 GPU on Kaggle, which restricted both the training duration and the complexity of our experiments. As a result, we could only fine-tune our model for a limited number of epochs, potentially leaving room for further performance gains with more extensive training. Additionally, our document segmentation relied on sentence-based chunking, which, while straightforward and efficient, may not fully capture the semantic structure of the documents. We investigated agentic chunking, a more advanced method where segmentation is guided by classifications from large language models (LLMs). However, this approach was deemed infeasible due to its high computational cost and the financial burden of utilizing LLM APIs. These limitations highlight the need for more efficient resource management and the exploration of advanced chunking methods in future work to further improve system performance.

References

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding: Multilingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation.

Phuc-Tinh Pham Do, Duy-Ngoc Dinh Cao, Khanh Quoc Tran, and Kiet Van Nguyen. 2023. R2gqa: Retriever-reader-generator question answering system to support students understanding legal regulations in higher education. *Journal of Information Technology*, XX(YY):ZZ–ZZ.

Luyu Gao, Zhuyun Dai, Tongfei Chen, Zhen Fan, Benjamin Van Durme, and Jamie Callan. 2021. Complementing lexical retrieval with semantic residual embedding.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-Tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1751–1760.

Mi-Young Kim et al. 2015. Evaluation of legal question answering. In *Proceedings of the International Conference on Legal Information Retrieval*, Location. Publisher.

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, dense, and attentional representations for text retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 755–764.

Jiaxin Mao, Jingtao Zhan, Yiqun Liu, Min Zhang, and Shaoping Ma. 2020. Repbert: Contextualized text embeddings for first-stage retrieval. *arXiv preprint arXiv:2006.15498*.

Ha-Thanh Nguyen, Manh-Kien Phi, Xuan-Bach Ngo, Vu Tran, Le-Minh Nguyen, and Minh-Phuong Tu. 2022. Attentive deep neural networks for legal document retrieval. *Artificial Intelligence and Law (to be published)*. Preprint version available at arXiv:2212.13899, DOI: 10.48550/arXiv.2212.13899.

Rodrigo Nogueira and Kyunghyun Cho. 2020. Passage reranking with bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 741–750. ACM.

Juliano Rabelo, Mi-Young Kim, Housam Babiker, Randy Goebel, and Nawshad Farruque. 2018. Legal information extraction and entailment for statute law and case law. In *Proceedings of the COLIEE 2018*, Location. Publisher.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Stephen Robertson, Hugo Zaragoza, and Michael Taylor. 2009. Simple bm25 extension to multiple weighted fields. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 42–49. ACM.

Liang Shao et al. 2023. An intent taxonomy of legal case retrieval. In *Proceedings of the COLIEE 2023*, Location. Publisher.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. E5: Text embeddings by weakly-supervised contrastive pre-training.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2021. Optimizing dense retrieval model training with hard negatives.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2020. Learning to retrieve: How to train a dense retrieval model effectively and efficiently.