SoICT Hackathon 2024 - Legal Document Retrieval

Anonymous ACL submission

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 integrates bi-encoder models and BM25 for initial retrieval and cross-encoder models for re-ranking 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 plays a crucial role in supporting legal professionals by providing quick and accurate access to relevant legal texts. In the context of Vietnamese law, the complexity and volume of legal documents pose significant challenges for traditional information retrieval systems. Developing an effective retrieval system that can handle the nuances of the Vietnamese legal language while delivering precise and relevant results is essential.

This study aims to address the challenges of Vietnamese Legal Text Retrieval by leveraging advanced natural language processing (NLP) techniques. Using the dataset provided by the SoICT Hackathon 2024 - Legal Document Retrieval competition, we propose a robust pipeline that integrates state-of-the-art models for document retrieval and ranking. The pipeline is designed to process large-scale legal corpora efficiently, ensuring relevance and precision in response to user queries.

The primary objective of this work is to enhance retrieval performance by combining the strengths of bi-encoder models for embedding generation and cross-encoder models for re-ranking. This combination allows for the retrieval of the most relevant legal documents, tailored to the semantic and contextual understanding of the queries.

The proposed approach is evaluated using Mean Reciprocal Rank (MRR@10), a widely recognized metric in information retrieval. Experimental results demonstrate that our system achieves notable performance, offering a reliable and efficient solution for Vietnamese legal text retrieval. This work contributes to advancing the accessibility of legal information and supports the development of intelligent tools for legal professionals in Vietnam.

2 Related Work

2.1 Pipeline of Zalo AI Legal Text Retrieval Challenge 2021

In the domain of legal text retrieval, the pipeline approach introduced in the Zalo AI Challenge 2021 for Legal Text Retrieval serves as a notable reference. The pipeline employs a multi-stage retrieval process integrating traditional and modern methods to address the challenges of retrieving relevant legal documents. Specifically, it combines the BM25 retrieval algorithm with fine-tuned language models, such as PhoBERT and viBERT, to effectively capture semantic nuances in legal texts. Furthermore, the system generates and refines negative pairs during training to enhance the performance of the Sentence Transformer, enabling it to better distinguish relevant from irrelevant documents. An ensemble strategy further consolidates scores from multiple methods to produce final predictions, leveraging the strengths of various retrieval and ranking mechanisms. However, a notable limitation of this pipeline is the absence of a reranker in its architecture. While the system utilizes fine-tuned Sentence Transformers for scoring and ranking, it lacks a dedicated reranking stage to refine the results further. This omission restricts its ability to fully leverage contextual nuances in the retrieved documents, potentially impacting accuracy

for complex queries where deeper semantic understanding is required.

2.2 Enhancing Retrieval Tasks with BM25

BM25 is a highly effective ranking function widely used in information retrieval tasks. Estimates the relevance of documents to a query by considering the frequency of the term, the inverse frequency of the document, and the normalization of the document length. The function's parameters enable fine-tuning to balance term saturation and document length effects, making it adaptable to various datasets.

Robertson et al. extended BM25 (**Robertson et al., 2004**) to handle multiple weighted fields, allowing better retrieval from structured documents. By combining term frequencies across fields before applying the saturation function, this approach preserves the nonlinear properties of term frequencies and improves the interpretability of field weights, leading to significant performance gains.

In a more recent application, Nguyen et al. (**Thi-Hong Nguyen et al., 2022**) used BM25 as the initial stage in a two-stage question-answering system. Here, BM25 efficiently retrieves relevant answer passages, addressing issues such as lexical gaps and sequence length limitations. This preprocessing ensures that transformer-based models, such as SBERT, operate on a refined set of passages, improving overall system performance.

Building on the foundation of the Zalo AI pipeline, our system introduces a refined corpus processing method that divides the dataset into, smaller manageable chunks, each linked to specific question-answer pairs and their identifiers. This approach allows for more precise embeddings using multilingual models, such as Multilingual E5 and BGE-M3, while integrating BM25 into the bi-encoder stage to enhance lexical and semantic representation. The system employs FAISS for efficient vector-based storage and retrieval, supporting an ensemble search over the top-k retrieved documents. To address the limitation of the Zalo AI pipeline's lack of a reranker, our system incorporates a fine-tuned bge-m3-rerank-v2 model in the final stage. This reranker plays a critical role in refining the results by leveraging contextual information from the retrieved documents. It enables the system to re-evaluate and reorder the top-k candidates based on deeper semantic understanding, ensuring that the most relevant results are ranked higher. This addition significantly enhances retrieval accuracy, particularly for complex legal queries that require nuanced interpretation beyond lexical similarity. This pipeline improves retrieval accuracy, particularly for complex legal queries, by effectively integrating granular processing with advanced ranking techniques.

The evolution from processing entire documents to a chunk-based approach, leveraging multilingual embedding models, positions our system as a significant advancement over prior work in legal text retrieval. This refined methodology ensures higher retrieval relevance by aligning closely with the structural and linguistic nuances of the legal domain.

3 System Architecture

Several studies often rely on BM25 for document retrieval due to its strength in capturing lexical similarity. However, we enhance this approach by combining bi-encoder, cross-encoder, and BM25 search to address the legal document retrieval task effectively. First, the provided corpus is preprocessed using chunking techniques. The system then proceeds in two stages. In the first stage, we fine-tune the bge-m3 bi-encoder model in two rounds. In Round one, BM25 retrieves relevant documents, which are used as training data to fine-tune bge-m3. The fine-tuned model is then applied with FAISS to search for the top-k relevant documents. These results are reused to fine-tune bge-m3 further in Round two, refining its semantic understanding. After Round 2, FAISS retrieves the final top-k candidates. In parallel, the chunked corpus is processed using the e5-large model to encode documents. FAISS is used to search for top-k results based on semantic similarity. Both sets of top-k results from the fine-tuned bge-m3 and e5-large are combined with the BM25 search results. Finally, the combined results are passed through a cross-encoder for re-ranking, where we select the top-10 documents based on relevance scores for the final prediction. Our system illustrated in Figure 1 combination of BM25, bi-encoder models, and cross-encoder re-ranking effectively balances lexical and semantic retrieval for improved accuracy.

3.1 Train Bge-m3 with two round

To improve the quality of document retrieval, we adopt a two-round fine-tuning strategy for the bge-m3 model. The process leverages BM25 search

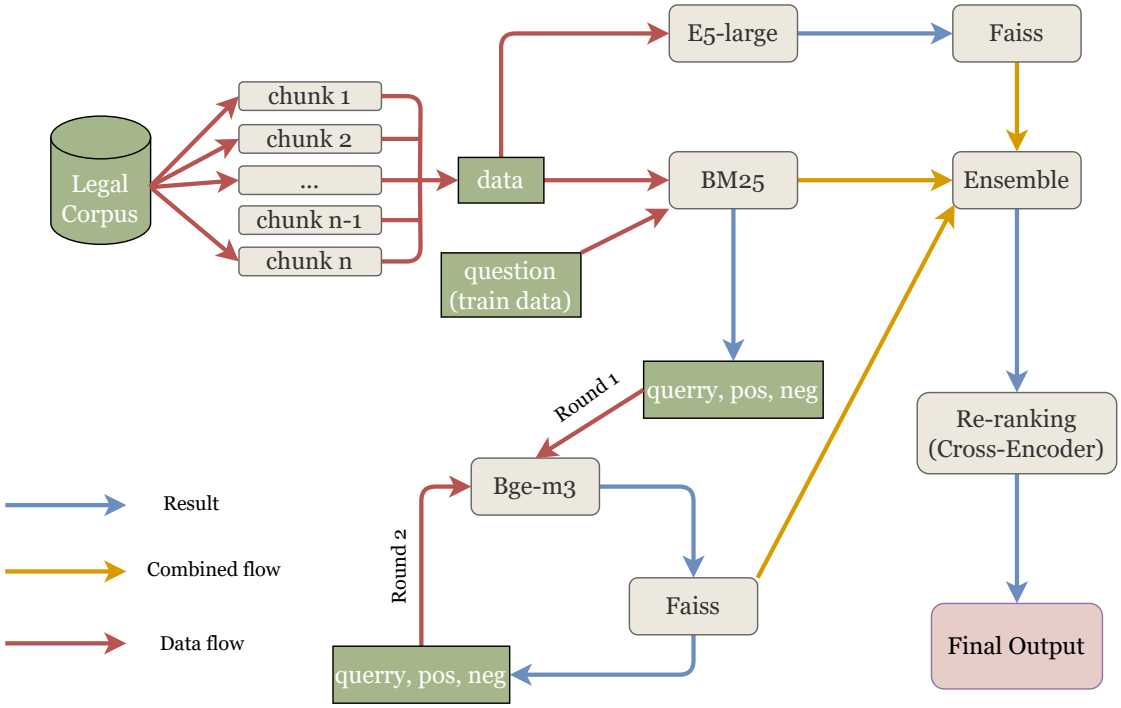

Figure 1: Architecture of our system.

results and FAISS-based retrieval to iteratively enhance the model's ability to understand semantic relationships between queries and legal documents.

Round 1 begins with the chunked corpus obtained through preprocessing. We first use BM25 to retrieve a set of relevant documents for each query. BM25 operates as a lexical-based retriever that identifies passages with strong token-level matches to the query. These retrieval results are treated as training data, where positive examples are the most relevant documents retrieved by BM25, and negative examples are documents with low or no relevance scores. This data is then used to fine-tune the bge-m3 model, aligning its semantic embedding space more closely with the specific task of legal document retrieval.

Once the model is fine-tuned in Round 1, the updated bge-m3 embeddings are indexed using FAISS, a highly efficient library for fast similarity search. FAISS enables us to search for the top-k relevant documents in the semantic space. These top-k results are then used to construct a new set of training data for Round 2, where we further refine the model. Specifically, the positive examples are the top-ranked documents returned by FAISS, while the hard negatives are lower-ranked documents that still have semantic similarity but are not directly relevant to the query. By incorporating these hard negatives, Round 2 fine-tuning pushes the model to better discriminate between truly relevant documents and near-relevant ones.

After completing Round 2, the final fine-tuned bge-m3 model is once again indexed with FAISS. This final index is used to retrieve the top-k relevant documents for each query, which are later combined with results from other components of the system.

The two-round fine-tuning process is critical for adapting bge-m3 to the specific characteristics of the legal document retrieval task. BM25 ensures a strong lexical foundation in Round 1, while FAISS-driven retrieval in Round 2 progressively enhances the model's semantic understanding by introducing hard negatives and positives. This iterative approach helps the bge-m3 model achieve robust performance in capturing both lexical and semantic relevance within the legal corpus.

The resulting embeddings and top-k retrieval results from bge-m3 serve as a core component of our system, complementing the results obtained from BM25 and the e5-large model. Further stages of the pipeline, including ensemble methods and re-ranking with the cross-encoder, refine these results to achieve the final predictions.

3.2 Ensemble E5 and Bge-m3 with BM25

To improve retrieval performance, we combine results from BM25, the fine-tuned bge-m3 bi-encoder, and the e5-large model in an ensemble framework. Each component contributes uniquely: BM25 captures lexical similarity, while bge-m3 and e5-large focus on semantic relationships. The ensemble ensures a balance between precision and recall. With BM25 search we retrieves the **top-20** documents ranked by lexical relevance. BM25 assigns scores based on word overlap and term frequency (TF-IDF) using the formula:

$$\text{score} = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{f_i \cdot (k_1 + 1)}{f_i + k_1 \cdot \left(1 - b + b \cdot \frac{\ell_d}{\text{avgdl}}\right)}$$

where: q is the query, d is the document, and q_i represents query terms. f_i is the term frequency of q_i in document d. $IDF(q_i)$ is the inverse document frequency of q_i. ℓ_d is the length of the document d, and avgdl is the average document length. The hyperparameters k_1 and b control the saturation and document length normalization, respectively, with $k_1 = 1.2$ and $b = 0.75$. BM25 retrieves a candidate set of documents by ranking them based on lexical relevance to the query.

As described in Section **??**, the bi-encoder bge-m3 model is fine-tuned in two rounds. After fine-tuning, the chunked corpus is indexed using FAISS, and **top-k** documents are retrieved for each query.

Given a query q and document d, the relevance score is calculated as the (**Xia et al., 2015**) between their embeddings:

$$\text{cosine similarity} = \frac{\mathbf{E}_q \cdot \mathbf{E}_d}{\|\mathbf{E}_q\| \|\mathbf{E}_d\|},$$

where $\mathbf{E}_q$ and $\mathbf{E}_d$ are the embeddings of the query q and document d, respectively. The **top-50** most relevant documents are selected based on their scores. As we said, our system divided in two blocks: the other one with e5-large Model. The e5-large model generates sentence embeddings for the chunked corpus. These embeddings are also indexed in FAISS for efficient semantic search. For each query q, we retrieve the **top-50** documents based on cosine similarity. After get scores, we continue to emsemble the score of bge-m3, e5 and BM25 together to make a hybrid search. The final ensemble result is obtained by merging and re-ranking the outputs of BM25, fine-tuned bge-m3, and e5-large. Specifically, we retrieve the top-k documents from BM25 and the top-k documents each from bge-m3 and e5-large using FAISS. The results are combined by taking the union of the document sets while ensuring no duplicates. To assign an ensemble score to each document, we combined the scores from bge-m3 and e5-large. Then we multiply with BM25 score. The ensemble score for a document d with respect to a query q is calculated as:

$$\text{ensemble} = \left(w_{\text{bge}} \cdot \text{score}_{\text{bge}} + w_{\text{e5}} \cdot \text{score}_{\text{e5}}\right) \cdot \text{score}_{\text{BM25}}$$

where w_{bge} and w_{e5} are the weights assigned to the bge-m3 and e5 scores, respectively. This approach ensures a balanced combination of lexical and semantic retrieval models, leveraging the strengths of each to improve overall ranking quality. The top-k results from the ensemble are passed to the next stage for re-ranking using the cross-encoder.

3.3 Re-Ranking

After obtaining the candidate document sets from BM25, fine-tuned bge-m3, and e5-large, we apply a re-ranking step using bge-reranker-v2-m3 (cross-encoder model) to improve the final ranking. Specifically, we utilize a pre-trained cross-encoder to evaluate the semantic relevance of each document d with respect to the query q. In this step, the model takes the query and document as input pairs and outputs a single relevance score. The re-ranking model is implemented using the cross-encoder architecture, which processes the input pair jointly rather than independently, enabling it to capture fine-grained semantic interactions.

We limit the re-ranking process to the top-10 documents obtained from the ensemble phase to balance computational efficiency and performance. The final prediction is obtained by sorting the top-10 candidate documents based on their re-ranked scores. This approach ensures that the most relevant documents are prioritized by leveraging the power of the cross-encoder's joint representation of query-document pairs.

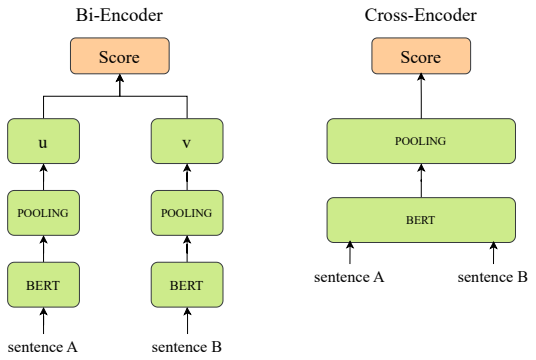

Figure 2: Bi-encoder and Cross-encoder

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 and training set are 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

...

4.3 Evaluation metrics

...

4.4 Results

...

5 Conclusion

...

Limitations

ACL 2023 requires all submissions to have a section titled "Limitations", for discussing the limitations of the paper as a complement to the discussion of strengths in the main text. This section should occur after the conclusion, but before the references. It will not count towards the page limit. The discussion of limitations is mandatory. Papers without a limitation section will be desk-rejected without review.

While we are open to different types of limitations, just mentioning that a set of results have been shown for English only probably does not reflect what we expect. Mentioning that the method works mostly for languages with limited morphology, like English, is a much better alternative. In addition, limitations such as low scalability to long text, the requirement of large GPU resources, or other things that inspire crucial further investigation are welcome.

References

Stephen Robertson, Hugo Zaragoza, and Michael Taylor. 2004. Simple bm25 extension to multiple weighted fields. In *Proceedings of the CIKM*.

Nhung Thi-Hong Nguyen, Phuong Phan-Dieu Ha, Luan Thanh Nguyen, Kiet Van Nguyen, and Ngan Luu-Thuy Nguyen. 2022. Spbertqa: A two-stage question answering system based on sentence transformers for medical texts. *arXiv e-prints*, pages arXiv–2206.

Peipei Xia, Li Zhang, and Fanzhang Li. 2015. Learning similarity with cosine similarity ensemble. *Information Sciences*, 307:39–52.