Optimizing Multi-Stage Language Models for Effective Japanese Legal Document Retrieval

Trung Quang Hoang
VJ Technologies, Da Nang City, Vietnam
trung.quang@vj-tech.jp (corresponding author)

Hoang Le Trung
VJ Technologies, Da Nang City, Vietnam
trunghoang04122002@gmail.com

Phuc Nguyen Van Hoang
VJ Technologies, Da Nang City, Vietnam
phucnvh2310@gmail.com

Hieu Quang Huu
AJ Technologies, Nagoya City, Japan | VJ Technologies, Da Nang City, Vietnam
hieuquang@aj-tech.jp

Received: 13 May 2025 | Revised: 24 June 2025, 12 July 2025, and 17 July 2025 | Accepted: 1 August 2025

 DOI: https://doi.org/10.48084/etasr.12111

ABSTRACT

Efficient text retrieval is critical for applications such as legal document analysis, especially within specialized domains such as Japanese legal systems. Existing methods often underperform in these scenarios: conventional BM25-based systems fail to capture nuanced legal expressions and formal sentence structures common in Japanese case law, resulting in low recall for relevant precedents. Consequently, tailored solutions are required. This study proposes a novel two-phase retrieval pipeline that replaces the sparse components (e.g., BM25+, TF–IDF) found in hybrid architectures, such as CoCondenser, instead operating end-to-end with only dense language models. This pipeline applies progressive fine-tuning—beginning with masked language model pretraining and followed by contrastive learning with hard negative mining—to iteratively improve accuracy on legal-domain queries. To ensure transparency and clear comparison, this study assesses two variants: an LM-only version (using both off-the-shelf and fine-tuned models) and a hybrid version that reintegrates BM25+, allowing for quantifying the impact of sparse components on retrieval performance. On a Japanese legal dataset, the proposed approach achieved state-of-the-art performance, yielding a 5.74% improvement in Recall@10 and an 11% gain in nDCG@10 over the strongest baseline, while remaining competitive on the MS-MARCO benchmark. To further enhance robustness and adaptability, an ensemble model integrated multiple retrieval strategies, yielding superior outcomes across diverse tasks. This work sets new standards for text retrieval in both domain-specific and general contexts, offering a comprehensive solution for handling complex queries in legal and multilingual environments.

Keywords-two-phase; text retrieval; ensemble

I. INTRODUCTION

In the field of text retrieval, traditional methods, such as sparse retrieval, have played a foundational role for decades. These approaches, including TF-IDF [1], BM25 [2], and BM25+ [3], rely on the frequency and direct relevance of terms between queries and documents. However, they face significant limitations when addressing semantic matching due to differences in meaning and expression between the query and the document. This issue is also evident in academic retrieval systems, where the increasing volume and complexity of the literature challenge the effectiveness of existing retrieval methods. Recent works have highlighted how conventional retrieval fails to incorporate citation networks and deeper semantic signals, leading to mismatches in user intent [4, 5]. Despite these challenges, BM25+ remains a strong baseline due to its ability to effectively balance the weight of important keywords and their frequency [6]. Retrieval methods are categorized into sparse, dense, generative, and contrastive-learning-based approaches.

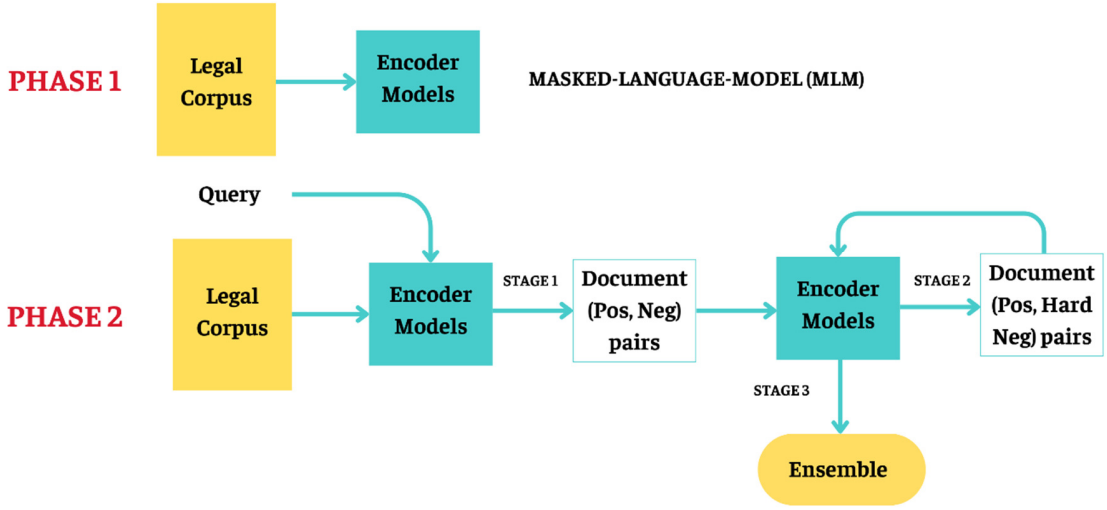

Fig. 1. An overview of the proposed two-phase text retrieval framework. In Phase 1, the model is pretrained using the Masked Language Model (MLM) task to establish a general contextual understanding of the dataset, creating a strong foundation for subsequent training. Phase 2 consists of three stages. In Stage 1, the encoder model, fine-tuned during Phase 1, is used to retrieve the most relevant documents. Among these, truly relevant documents (labeled as positive) are identified based on human annotations, whereas documents mistakenly considered relevant are labeled as negative. In Stage 2, both positive and negative documents are input into encoder models, such as BERT or RoBERTa, to further refine the model's ability to differentiate between relevant and irrelevant documents. Unlike traditional approaches, this method replaces sparse retrieval techniques with Language Models (LMs) to improve performance. In Stage 3, hard negative examples generated from the fine-tuned model in Stage 2 are used for additional training to enhance the model's capacity to address more challenging cases. The process concludes with an ensemble step that combines multiple models or techniques to leverage their individual strengths. This integration minimizes errors, improves accuracy, and enhances the stability of retrieval outcomes, resulting in superior overall performance.

Dense retrieval encodes both queries and documents in vectors within a continuous semantic space, enabling more accurate semantic comparisons. Pre-training enhancements such as CoCondenser [7] integrate term frequency and document length signals from BM25+ into the language model pre-training stage, condensing cooccurrence patterns before downstream fine-tuning. Recent studies, such as [8, 9], have shown that dense retrieval substantially improves the ability to retrieve relevant documents by alleviating the issue of lexical mismatch. Within dense retrieval, two widely adopted techniques are the cross-encoder and the bi-encoder. The cross-encoder processes both the query and the document simultaneously, generating a combined representation for enhanced accuracy, though at a high computational cost. Conversely, the bi-encoder independently encodes queries and documents, offering faster processing and greater scalability, particularly when working with large datasets.

More recently, generative retrieval models, such as DSI-QG [10] and GENRET [11], have introduced an innovative paradigm. These systems generate document identifiers (docids) or relevant document content directly from the query. By leveraging the power of Large Language Models (LLMs), these approaches enhance retrieval accuracy, especially in scenarios where traditional retrieval methods struggle. DSI QG [10] and GENRET [11] provide strong empirical baselines and illustrate the rapid pace of innovation in generative retrieval.

Contrastive learning has become a widely adopted technique in neural information retrieval, allowing models to learn semantically meaningful representations by aligning positive query-document pairs while separating negatives in the embedding space. Notable implementations include SimCSE [12], CoCondenser [7], and GENRET-CL [11], which apply contrastive pretraining objectives to enhance sentence and document encoders. Recent studies, such as CoCondenser [7] and ANCE [13], show that incorporating negative contrastive sampling into pretrained transformers significantly improves performance on general-domain retrieval benchmarks. SimCSE [12], ANCE [13], and GENRET-CL [11] underscore the need to balance cutting-edge innovation with methodological rigor. However, these methods have been predominantly developed and evaluated on English-language datasets, and thus do not directly address the distinct linguistic and structural challenges present in languages such as Japanese, which feature complex character sets, agglutinative morphology, and domain-specific terminologies such as those found in legal texts.

Despite these advances, legal domain retrieval remains underexplored, particularly for non-Latin scripts. Recent works in legal retrieval (e.g., LEXTREME [14], LeCaRDv2 [15]) focus on formal statutes and case law but are still predominantly English-centric or rely on heavy LLM inference. Moreover, few studies address the high morphological variation and precise terminology found in Japanese legal texts, nor do they provide efficient, low-latency solutions suitable for real-time querying.

Thus, the specific gap this study addresses is twofold: (i) the lack of tailored retrieval methods for Japanese legal documents, which feature complex kanji and domain-specific constructs, and (ii) the need for a fast and lightweight pipeline that uses pretrained LMs rather than large LLMs, ensuring both high throughput and accuracy. Unlike prior methods that depend on LLMs and incur substantial inference overhead, this approach maintains sub-second response times, making it practical for interactive legal search applications. In summary, the contributions of this study are as follows.

- Development of innovative LM-only pipelines: Inspired by CoCondenser [7] but divergent in completely removing BM25+, this study proposes four LM-based pipelines, LMS Round 1, LMS Round 2, LMS (Finetuned MLM) Round 1, LMS (Finetuned MLM) Round 2, that progressively fine-tune an encoder on Japanese legal data, including passage splitting and sliding-window tokenization to handle long documents.
- Ensemble integration: These pipelines are expanded by incorporating an ensemble of three dense representation models (paraphrase-multilingual-mpnet-base-v2, distiluse-base-multilingual-cased-v1, and LMS Round 2) to further boost robustness and adaptability across tasks.
- Comprehensive experimentation: Extensive experiments are conducted to validate the effectiveness of the proposed approach, showcasing improvements on Japanese legal benchmarks (e.g., 3.64% gain in Recall@5 over DSI) and general benchmarks such as MS-MARCO.

In summary, this study introduces a suite of LM-only retrieval pipelines tailored for Japanese legal contexts, provides a clear explanation of each phase and stage, details the proposed approach to handling long documents and multi-term queries, and demonstrates superior performance both in-domain and on standard benchmarks.

II. PRELIMINARIES

A. Masked Language Model (MLM)

This is a training technique where certain tokens (words or characters) in a sentence are masked (hidden) by replacing them with a special token, such as [MASK]. The task of the model is to predict the masked tokens based on their context (i.e., the surrounding words).

B. Contrastive Learning

Contrastive learning works by learning a representation that maps similar data points closer in the embedding space and dissimilar data points farther apart. This is achieved by optimizing a contrastive loss function, which encourages the model to reduce the distance between similar data points while increasing the distance between dissimilar points. In the context of learning representations for text retrieval, the data points could be queries and documents, aiming to learn a representation that maps related queries and documents closer together while mapping unrelated ones farther apart. The general form of contrastive loss is as follows:

$$L\big(W,(Y,\mathrm{X}_1,\mathrm{X}_2)^i\big) = (1-Y)L_S\big(D_W^i\big) + YL_D\big(D_W^i\big) \qquad (1)$$

where Y determines whether the two data points $(\mathrm{X}_1,\mathrm{X}_2)$ are similar $(Y=0)$ or dissimilar $(Y=1)$, L_S represents the loss function for similar data points, L_D represents the loss function for dissimilar data points, and D_W is a generic similarity (or dissimilarity) metric between two points X_1 and X_2, introduced in [16]. For example, one can use Euclidean distance:

$$D_W(\mathrm{X}_1,\mathrm{X}_2) = \|G_W(\mathrm{X}_1) - G_W(\mathrm{X}_2)\|_2 \qquad (2)$$

Here, G represents the mapping function (typically a neural network). This is an Euclidean distance (L2 norm), but other distance metrics such as Manhattan distance, Cosine similarity, etc., can also be used. In this implementation, cosine similarity was chosen (normalized to [0,1]) to measure similarity between embeddings. Thus, a cosine-based distance was defined as:

$$D_W^{(cos)}(\mathrm{X}_1,\mathrm{X}_2) = 1 - \cos\big(G_W(\mathrm{X}_1), G_W(\mathrm{X}_2)\big)$$
$$= 1 - \frac{G_W(\mathrm{X}_1)*G_W(\mathrm{X}_2)}{\|G_W(\mathrm{X}_1)\|*\|G_W(\mathrm{X}_2)\|} \qquad (2')$$

Equation (1) resembles the cross-entropy loss structurally. However, while cross-entropy loss operates over class probabilities for classification tasks, the contrastive loss focuses on data point distances in the learned embedding space. This makes contrastive loss particularly suited for tasks like face verification, where the goal is to learn representations without explicitly modeling class distributions. The exact contrastive loss proposed in [16] is as follows:

$$L(W,(Y,\mathrm{X}_1,\mathrm{X}_2)^i) =$$
$$(1-Y)\tfrac{1}{2}(D_W)^2 + Y\tfrac{1}{2}\max\{0, m - D_W\}^2 \qquad (3)$$

Replacing D_W with $D_W^{(cos)}$ yields the final loss:

$$L(W,(Y,\mathrm{X}_1,\mathrm{X}_2)^i) = (1-Y)\tfrac{1}{2}(D_W^{(cos)})^2$$
$$+Y\tfrac{1}{2}\max\{0, m - D_W^{(cos)}\}^2 \qquad (3')$$

where:

- L_S (loss for similar data points): If two data points are labeled as similar, minimize the cosine similarity distance $D_W^{(cos)}$ between them.
- L_D (loss for dissimilar data points): If two data points are labeled as dissimilar, maximize the cosine similarity distance $D_W^{(cos)}$ up to a margin m.

The goal of contrastive learning is to ensure that for each class/group of similar data points, the intra-class distance is minimized while the inter-class distance is maximized. This is illustrated in Figures 2 and 3. For any given anchor point (shown as a blue dot), it is ensured that:

- Black points (representing similar examples) lie within the margin boundary m, indicating that they are close to the anchor in the embedding space and should be pulled even closer during training.
- White points (representing dissimilar examples) lie outside the margin boundary m, meaning they are sufficiently dissimilar to the anchor, and the model is trained to maintain or increase this separation.

When adding a new data point, the nearest-neighbor algorithm is used to determine its similarity/dissimilarity based on whether it lies within the margin m. The contrastive loss ensures that:

- If $D_W^{(cos)} \geq m$ for dissimilar data points, the penalty becomes zero since $\max\{0, m - D_W^{(cos)}\} = 0$. This avoids unnecessarily pushing dissimilar points farther than required.

- For similar data points, $D_W^{(cos)}$ is minimized, ensuring that they are closer in the embedding space.

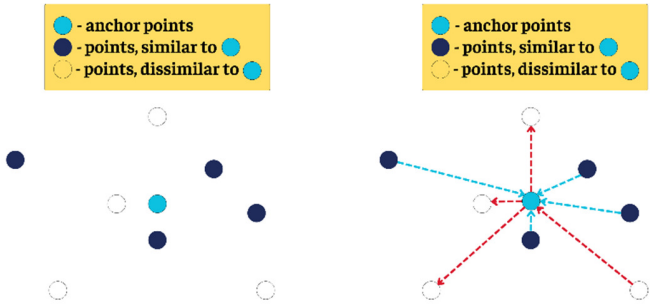

Fig. 2. Illustration of contrastive learning: Given an anchor point (blue), similar points (black) are pulled closer, while dissimilar points (white) are pushed farther away.

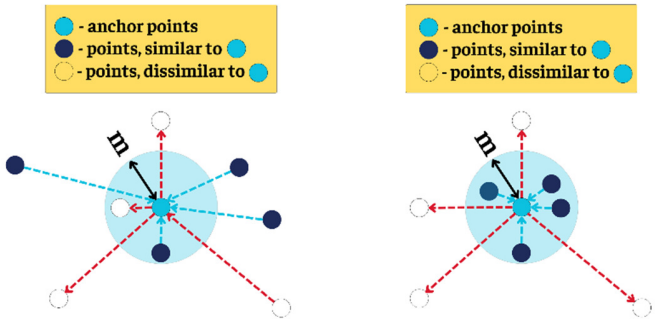

Fig. 3. Illustration of decision boundaries using contrastive loss.

In this implementation, Euclidean distance was replaced with a cosine-based distance $D_W^{(cos)} \in [0,1]$ defined in (2′) and applied directly in the margin-based contrastive loss given by (3′). The margin was set to $m = 0.5$, which is the midpoint of the cosine similarity range $[0,1]$. This choice follows the rationale of [16], where the margin is placed at the middle of the energy function range.

Contrastive loss is used to train the encoder model, as shown in Figure 4, where the input consists of a query (chunk) and documents (legal), which are processed independently by two separate encoder models. The outputs are pooled to generate two vectors representing the query and the documents. These vectors are then compared using cosine similarity, and the process is optimized with contrastive loss, as detailed in Algorithm 1.

Algorithm 1: Contrastive Loss with Cosine Similarity
Input: $chunk$, $legal$, $model$, margin $m = 0.5$, $score$ $(similar = 1 \mid dissimilar = 0)$
Output: $loss$
Step 1: Generate vector embeddings:
$chunk\_embedding \leftarrow model.embedding(chunk)$
$legal\_embedding \leftarrow model.embedding(legal)$
Step 2: Compute cosine similarity:
$distances \leftarrow 1 - cosine\_similarity(chunk\_embedding,\ legal\_embedding)$
Step 3: Define the margin: $m \leftarrow 0.5$
Step 4: Calculate loss function:
$loss \leftarrow 0.5 \cdot (score \cdot distances^2 + (1 - score) \cdot ReLU(m - distances)^2)$
return $loss$

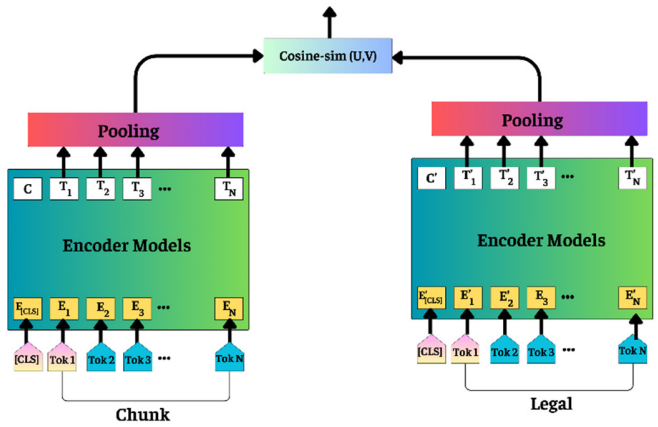

Fig. 4. Illustration of sentence similarity calculation, dual encoder models process text chunks and legal passages, and applying pooling and cosine similarity for semantic matching.

III. THE PROPOSED APPROACH

This study develops a set of novel two-phase pipelines to improve text retrieval performance, particularly in the legal domain. It should be noted that this approach is not a traditional retrieve-then-rerank framework; Instead, both retrieval and ranking are handled by a single, shared encoder trained end-to-end.

The first approach builds upon the CoCondenser pipeline, which combines symbolic retrieval (BM25+) with LM-based encoding. This setup treats the original CoCondenser as BM25+ (Round 1), which is then extended by introducing BM25+ (Round 2), where the encoder trained in Round 1 is reused to mine hard negatives, which are subsequently used to further fine-tune the model. This iterative training strategy aims to improve retrieval accuracy by focusing the model on more challenging, semantically close distractors.

The second approach introduces a family of LM-only pipelines, which eliminate the use of sparse retrieval methods such as BM25 or BM25+. Unlike the CoCondenser [7], which combines BM25+ with pre-trained LMs in a hybrid framework, this approach replaces BM25+ entirely and introduces a multi-stage contrastive learning process that progressively refines the encoder through masked language modeling, positive-negative sampling, and hard negative mining—all within a dense retrieval setup. This design enables fully neural retrieval without symbolic filtering.

The proposed methods of the second approach are described as follows.

A. LMS (Round 1): Pretraining and Fine-Tuning in Two Phases

The first pipeline consists of two distinct phases.

1) Phase 1

The encoder model is initialized with pretrained weights and further enhanced using an MLM objective. This step helps the encoder better understand contextual relationships within the corpus.

2) Phase 2

- Stage 1: Using the pretrained language model from Phase 1, the system generates document pairs (positive and negative) based on the query and the legal corpus.
- Stage 2: The encoder model, fine-tuned on data generated in Stage 1, is used to distinguish between relevant (positive) and irrelevant (negative) documents.

B. LMS (Round 2): Refinement with Stage 3 for Hard Negatives

This pipeline builds on LMS (Round 1) by introducing an additional stage:

- Stage 3: The encoder model is further refined by retraining on hard negative document pairs generated in Stage 2. This step enhances the encoder's ability to handle difficult or borderline cases, resulting in improved document discrimination.

C. LMS (Finetuned MLM) Round 1: Leveraging Finetuned MLM for Enhanced Retrieval

In this pipeline, the MLM finetuned in Phase 1 is directly integrated into the retrieval process.

- Phase 2:
  - Stage 1: The encoder, finetuned with the MLM objective in Phase 1, generates top-ranked documents with the highest relevance to the query.
  - Stage 2: The encoder model, further trained on contrastive loss using the data generated in Stage 1, distinguishes relevant documents more effectively.

D. LMS (Finetuned MLM) Round 2: Refining Finetuned Models for Greater Precision

This pipeline extends LMS (Finetuned MLM) Round 1 by incorporating Stage 3.

- Stage 3: Similar to LMS (Round 2), the encoder is re-trained on hard negative document pairs generated in Stage 2. This additional training step sharpens the encoder's ability to distinguish between closely similar and dissimilar documents, achieving the best possible performance.

These pipelines collectively demonstrate the effectiveness of fully replacing sparse retrieval methods with LMs for superior text retrieval performance. Iterative refinement through stages allows the model to handle increasingly challenging data distributions while maintaining contextual relevance.

IV. EXTENSIONS

A. Ensemble Methodology

An ensemble approach is adopted to further enhance the performance of the proposed retrieval framework, which combines the outputs of three models: paraphrase-multilingual-mpnet-base-v2, distiluse-basemultilingual-cased-v1, and the LMS (Finetuned MLM) Round 2. Each model captures unique aspects of the retrieval task, and combining their outputs allows the framework to leverage their complementary strengths. The ensemble score is calculated using:

$$Score = \alpha \cdot score_{model1} + \beta \cdot score_{model2} + \theta \cdot score_{model3} \quad (4)$$

where $\alpha + \beta + \theta$ and the $Score$ represent the Recall@3 metric used for evaluation in Figure 5. The coefficients α, β and θ serve as the weights assigned to the contributions of each model.

B. Weight Optimization Process

To identify the optimal set of weights, a systematic grid search was performed across possible values of α and β, adhering to the constraint $\alpha + \beta + \theta = 1$. The search process involved:

1. Varying α incrementally from 0 to 1 in steps of 0.05.
2. For each value of α, varying β from 0 to $1 - \alpha$ in steps of 0.05.
3. Calculating θ as $\theta = 1 - (\alpha + \beta)$ to satisfy the total weight constraint.

This process generated a comprehensive set of valid weight combinations, systematically covering the space of possible distributions. Each combination was evaluated based on the ensemble's Recall@3 performance.

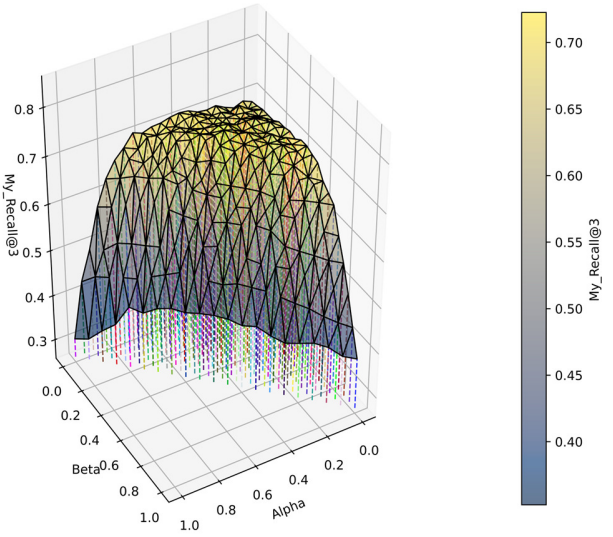

Fig. 5. Visualization of the grid search process, depicting Recall@3 scores across different weight combinations.

C. Results and Observations

The grid search identified the optimal weight combination as $\alpha = 0.3$, $\beta = 0.25$, and $\theta = 0.45$, resulting in a final Recall@3 score of 0.72. This configuration demonstrates the effectiveness of the ensemble approach, significantly improving performance over individual models. The relatively larger weight assigned to the LMS (Finetuned MLM) Round 2 ($\theta = 0.45$) highlights its critical role in the ensemble, benefiting from task-specific tuning. Figure 5 visualizes the grid search process, showing the progression of Recall@3 scores across various weight combinations. This figure highlights regions of high performance and illustrates how the ensemble's effectiveness depends on the weight distribution.

D. Example Weight Combinations

Table I presents sample weight combinations from the grid search, including the optimal set, to illustrate the systematic evaluation process. This demonstrates the robustness of the ensemble method, as even suboptimal configurations yield competitive Recall@3 scores.

TABLE I. A WIDE RANGE OF WEIGHT COMBINATIONS DERIVED FROM GRID SEARCH FOR OPTIMIZING ENSEMBLE PERFORMANCE ON JAPANESE LEGAL DATASET.

α	β	θ	R@3	R@5	R@10	R@20	R@50	R@100	R@200	MAP@10	MRR@10	nDCG@10
0.3	0.25	0.45	**0.72**	0.77	**0.85**	**0.87**	**0.93**	0.94	**0.99**	**0.71**	**0.91**	**0.78**
0.4	0.3	0.3	0.68	0.74	0.84	0.86	0.91	0.95	0.98	0.66	0.86	0.75
0.2	0.5	0.3	0.68	0.73	0.82	0.86	0.90	0.95	0.98	0.66	0.87	0.74
0.5	0.2	0.3	0.68	0.73	0.83	0.86	0.91	0.95	0.98	0.65	0.85	0.74
0.25	0.35	0.4	0.71	**0.78**	0.84	0.86	0.91	0.95	0.98	0.69	0.89	0.77
0.3	0.3	0.4	0.71	0.77	0.84	0.86	0.90	0.95	0.98	0.69	0.89	0.77
0.2	0.3	0.5	0.71	0.76	0.84	0.85	0.91	**0.96**	0.98	0.70	0.88	0.77
0.4	0.4	0.2	0.63	0.67	0.79	0.85	0.91	0.95	0.96	0.61	0.82	0.70

V. EXPERIMENTS

To evaluate the effectiveness of the proposed retrieval pipeline, experiments on Japanese legal datasets derived from [17] and a split of the MS MARCO passage dataset [18] were conducted. The proposed method, inspired by CoCondenser [7], leverages advanced LMs to enhance retrieval efficiency and accuracy, particularly within the Japanese legal domain. These experiments validate the robustness of the proposed approach, demonstrating its strong performance on both domain-specific and standard benchmarks.

A. Legal Japanese Retrieval Dataset

A Japanese legal retrieval dataset derived from [17] was employed, specifically constructed for dense retrieval tasks in the legal domain. The dataset comprises 3,259 training examples, 73 validation examples, and 130 test examples. Each example includes a legal query and its corresponding relevant document, forming a positive pair. Negative examples are generated either randomly or through BM25-based sampling, depending on the training stage. The dataset is built from publicly available legal case documents released by Japanese courts. However, the curated version used in this study is not publicly released due to licensing restrictions and legal sensitivities surrounding the redistribution of processed legal texts. All documents have undergone preprocessing to remove sensitive information such as personal names, locations, and identifiers, in accordance with relevant regulations. The dataset was prepared exclusively for academic research purposes. Further details on the labeling methodology and document processing pipeline are available in [17].

Standard metrics were used to evaluate retrieval performance: Recall, MRR@10, MAP@10, and nDCG@10. These metrics collectively assess both ranking precision and retrieval coverage within this domain-specific corpus. Recent research on multi-objective evaluation [19] highlights the importance of balancing diverse relevance dimensions, reinforcing the choice to report multiple metrics rather than relying on a single indicator. Table II highlights the superior performance of the proposed methods compared to sparse, dense, and generative retrieval approaches on Japanese legal datasets.

Sparse Retrieval, such as TF-IDF and BM25+, relies on keyword frequency-based matching. These methods often struggle to capture the semantic relationships necessary for effective retrieval in complex datasets. Consequently, their performance is suboptimal across all metrics. For example, at Recall@200, TF-IDF achieves 83.76, while BM25+ lags at 81.45. By contrast, LMS (Finetuned MLM) Round 2 achieves a significantly higher Recall@200 of 97.46, surpassing TF-IDF and BM25+ by 13.7 and 16.01 points, respectively. This disparity becomes even more pronounced at lower recall levels. At Recall@3, TF-IDF and BM25+ achieve 42.82 and 37.82, respectively, whereas the proposed BM25+ (Round 2) reaches 65.38, outperforming TF-IDF by 22.56 points and BM25+ by 27.56 points. These results illustrate the fundamental limitations of sparse retrieval in handling nuanced queries and emphasize the importance of leveraging language models for semantic understanding.

Dense retrieval leverages LMs to encode semantic representations of queries and documents. This category includes models such as Siamese [20], Two-Towers Siamese [21], GC-DPR [22], and CoCondenser [7]. Although dense methods outperform sparse retrieval approaches, they are consistently outperformed by the proposed methods. For example, at Recall@10, CoCondenser achieves 70.89, outperforming other dense retrieval baselines such as Siamese (62.16) and GC-DPR (53.05). However, LMS Round 2 further improves on CoCondenser with a Recall@10 of 78.73, demonstrating a 7.84-point lead. This advantage persists across higher recall levels, with the proposed method achieving 94.23 at Recall@200 compared to CoCondenser's 92.92, marking a 1.31-point improvement.

TABLE II. PERFORMANCE OF THE PROPOSED METHOD ON THE JAPANESE LEGAL DATASET, HIGHLIGHTING SIGNIFICANT ADVANCEMENTS IN RETRIEVAL EFFECTIVENESS OVER BASELINE APPROACHES AND EXISTING TECHNIQUES

Method	R@3	R@5	R@10	R@20	R@50	R@100	R@200	MAP@10	MRR@10	nDCG@10
Sparse Retrieval										
TF-IDF [1]	42.82	46.18	55.11	63.19	73.29	78.07	83.76	39.17	72.40	46.52
BM25+ [3]	37.82	41.09	50.59	57.10	65.84	73.51	81.45	29.76	58.11	38.29
Dense Retrieval										
Siamese [20]	46.79	53.56	62.16	67.23	80.66	86.64	92.08	39.61	53.64	49.37
Two-Towers Siamese [21]	47.82	55.63	62.23	70.12	80.29	85.74	91.99	41.76	57.74	51.28
GC-DPR [22]	42.56	46.42	53.05	63.10	76.18	82.49	87.96	37.47	75.98	46.38
CoCondenser [7]	60.77	62.02	70.89	77.10	82.26	87.34	92.92	57.11	73.44	65.01
3SRM [23]	39.87	42.38	50.13	59.95	76.85	86.76	90.08	34.88	48.16	42.15
BM25+ (Round 2) (Proposed)	65.38	68.33	75.91	78.93	**87.99**	92.36	96.71	62.16	**85.27**	69.69
LMS (Round 1) (Proposed)	62.95	67.90	73.52	76.81	86.47	90.40	96.60	57.88	79.41	66.23
LMS (Round 2) (Proposed)	**69.10**	72.39	78.73	80.27	87.48	92.33	94.23	64.99	84.98	72.51
LMS (Finetuned MLM) Round 1 (Proposed)	64.87	67.32	74.91	78.02	86.86	**94.03**	96.73	61.56	84.30	69.10
LMS (Finetuned MLM) Round 2 (Proposed)	68.72	**72.50**	**79.53**	**82.57**	85.67	91.58	**97.46**	65.42	82.37	**72.65**
Generative Retrieval										
DSI [24]	66.03	68.86	75.21	80.34	86.38	89.19	92.90	**66.06**	68.16	65.39
DSI-QG [10]	15.75	23.29	28.77	39.49	56.85	68.72	84.24	13.63	13.64	16.75

The improvements can be attributed to the two-phase training pipeline, particularly Phase 2 Stage 3, where hard negative examples are incorporated into the training process. This refinement allows the proposed methods to address challenging cases and ambiguous document-query relationships, which dense retrieval methods often fail to resolve.

Generative Retrieval methods, such as DSI [24] and DSI-QG [10], attempt to directly model query-document relationships by generating document representations. Although DSI achieves competitive results at lower recall levels (e.g., Recall@3 of 66.03), its performance declines for higher recall levels. For instance, at Recall@200, DSI achieves 92.90, falling behind the proposed LMS (Finetuned MLM) Round 2 by 4.56 points (97.46 vs. 92.90).

DSI-QG, which incorporates query generation to address documents without associated queries, struggles significantly with the linguistic complexities of Japanese and produces suboptimal results. It achieves only 28.77 at Recall@10, far below even Sparse Retrieval methods such as BM25+ (50.59) and the proposed methods, such as LMS (Round 1), which achieved 73.52. This gap highlights the challenges generative methods face in creating high-quality semantic representations for non-English datasets.

The proposed methods consistently outperform baselines across all metrics. Among them, LMS (Finetuned MLM) Round 2 achieves the highest performance, setting new benchmarks in Recall@5 (72.50), Recall@10 (79.53), and Recall@200 (97.46).

- Comparison with Sparse Retrieval: Compared to BM25+, the proposed method improves Recall@20 by 25.47 points (82.57 vs. 57.10) and Recall@200 by 16.01 points (97.46 vs. 81.45).
- Comparison with Dense Retrieval: At Recall@50, the proposed method achieves a 3.41-point improvement over CoCondenser (85.67 vs. 82.26) and a 5.38-point improvement over Two-Towers Siamese (85.67 vs. 80.29).
- Comparison with Generative Retrieval: The proposed method achieves significantly higher Recall@5, Recall@10 compared to DSI (72.50 vs. 68.86 and 79.53 vs. 75.21, respectively).

The superior performance of the proposed method can be attributed to Phase 2 Stage 3, where fine-tuned models are trained on hard negative examples. For example, LMS (Finetuned MLM) Round 1, which does not include Stage 3, achieves a Recall@10 of 74.91, while LMS (Finetuned MLM) Round 2 achieves 79.53, demonstrating a 4.62-point improvement.

These results underscore the robustness of the proposed retrieval pipeline. By replacing traditional sparse retrieval techniques with advanced LMs and incorporating multi-stage training, the proposed methods achieve state-of-the-art performance across all metrics. These improvements highlight the scalability and effectiveness of the proposed approach, especially in the Japanese legal domain.

B. Benchmark Datasets Retrieval

The MS MARCO passage dataset originally includes approximately 1.01 million rows. This study employed a subset of 17,132 rows, divided into 15,270 for training, 1,000 for validation, and 862 for testing. The corpus contained approximately 134,000 documents. This split was designed to ensure a manageable yet representative subset for experimentation. Stage 2 training involved regular negatives, while Stage 3 focused on hard negatives, as detailed in the experimental setup.

Table III illustrates the effectiveness of the proposed retrieval methods across various evaluation metrics on the training split of the MS MARCO passage dataset. The results underscore the superiority of the proposed approaches, CoCondenser [7] and BM25+ (Round 2), over baseline sparse and generative retrieval techniques. Sparse Retrieval performed moderately well but was inherently constrained by its reliance on exact lexical matches. For instance, BM25+ achieved a Recall@3 of 28.9 and MAP@10 of 24.5, while its nDCG@10 was higher at 30.9, indicating that its ranking quality benefits from a focus on top results. Similarly, TF-IDF showed strong performance in Recall@100 with a value of 83.6, but lagged in Recall@3 (26.9) and MAP@10 (23.2). In particular, BM25+ achieved the highest MRR@10 among the sparse methods at 49.3, surpassing the proposed methods in this metric. This suggests that the lexical matching mechanisms of BM25+ can still be effective in identifying highly relevant documents for ranking. However, these methods struggle to fully capture semantic nuances, which limit their ability to perform consistently across a range of retrieval metrics in complex scenarios.

TABLE III. EVALUATION OF THE PROPOSED METHOD ON THE MS MARCO BENCHMARK, SHOWCASING SUBSTANTIAL IMPROVEMENTS ACROSS RETRIEVAL METRICS COMPARED TO BASELINE APPROACHES AND EXISTING METHODS

Model/Metrics	R@3	R@100	MRR@10	MAP@10	nDCG@10
Sparse Retrieval					
TF-IDF	26.9	83.6	43.8	23.2	30.6
BM25+	28.9	73.5	49.3	24.5	30.9
Dense Retrieval					
CoCondenser [7]	29.2	73.5	25.9	24.6	30.0
BM25+ (Round 2) (Proposed)	34.5	84.1	30.2	29.0	36.1
Generative Retrieval					
DSI	14.3	40.9	13.7	13.6	15.5

In contrast, the proposed methods leverage Dense Retrieval techniques to overcome these limitations and deliver significant improvements across most metrics. CoCondenser [7], which omits hard negative training in Phase 2 Stage 3, achieved a Recall@3 of 29.2 and MAP@10 of 24.6, closely rivaling the performance of BM25+. In particular, when hard negative training is incorporated, BM25+ (Round 2) outperforms BM25+ in Recall@3 (34.5 vs. 28.9), MAP@10 (29.0 vs. 24.5), and nDCG@10 (36.1 vs. 30.9). These results highlight the robustness of the proposed methods, particularly in metrics focused on ranking quality.

Generative Retrieval performed poorly across all metrics, achieving only a Recall@3 of 14.3 and nDCG@10 of 15.5. These results emphasize the inherent limitations of generative approaches in retrieval tasks. Reliance on generative language modeling often lacks the precision and relevance required for effective ranking. For instance, compared to BM25+ (Round 2), which achieved a Recall@3 of 34.5 and nDCG@10 of 36.1, DSI's performance fell significantly short. This highlights its inability to handle ranking tasks that involve complex semantic nuances.

Furthermore, DSI's lack of capability to leverage explicit document-query interactions is a key limitation that constrains its effectiveness. In contrast, BM25+ (Round 2) utilizes iterative training with hard negatives, refining ranking capabilities to achieve a MAP@10 of 29.0, far exceeding DSI's 13.6. These metrics underscore the robustness and adaptability of the proposed approach in addressing nuanced relationships required for high-quality document retrieval. This makes it a clearly superior alternative to generative paradigms.

The two-phase training strategy employed in the proposed methods underpins their superior performance. In Phase 1, pretraining with the MLM task establishes a strong foundational understanding of the dataset. Phase 2's multi-stage approach, which involves the initial BM25+, refinement with positive and negative samples using LMs, such as XLM-RoBERTa, and iterative training with hard negatives, ensures a comprehensive and nuanced model training process. This allows the proposed methods to consistently outperform traditional sparse retrieval techniques and state-of-the-art Generative Retrieval models.

In general, these findings emphasize the robustness and adaptability of the proposed retrieval framework, demonstrating its ability to excel across a diverse range of retrieval metrics. The significant performance gains achieved by BM25+ (Round 2) highlight the value of incorporating advanced training strategies and iterative refinement in modern retrieval tasks.

C. Hyperparameter Settings

To ensure the reproducibility of these experiments and allow fair comparison between different retrieval pipelines, consistent training settings were adopted throughout the proposed approach, unless otherwise stated.

For the LMS (Round 1) and LMS (Finetuned MLM) Round 1 pipelines, the training process begins with Phase 1, where the encoder model is initialized with the pretrained weights of a multilingual Transformer-based language model and further optimized using an MLM objective. The model is trained for 50 epochs using a batch size of 8 per device, with input sequences truncated or padded to a maximum length of 512 tokens. The masking probability is set to 15%, following standard practice: 80% of masked tokens are replaced with a [MASK] token, 10% with a random token, and 10% remain unchanged. The Adam optimizer is employed with parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 1e^{-8}$, along with a linear decay learning rate scheduler starting at $5e^{-5}$. No weight decay or warm-up steps are used. Dropout is applied at a rate of 0.1, as defined in the original model configuration. Checkpoints are saved every 25,000 steps, and training is conducted in full-precision mode.

In Phase 2, query-document pairs are constructed by pairing each query with a small set of relevant documents (positive examples) and a larger set of non-relevant documents (negative examples). For each query, the system retrieves the top 50 non-relevant documents based on embedding similarity to serve as negative candidates. These positive/negative document pairs form the training set for contrastive fine-tuning.

The encoder is then fine-tuned using a contrastive loss function, encouraging the model to bring relevant documents closer to the query while pushing non-relevant documents away in the embedding space. The architecture remains consistent: the base Transformer encoder with mean pooling over token embeddings. Training is performed for 10 epochs with a batch size of 32 in Round 1, using the AdamW optimizer with a learning rate of $1e^{-5}$ and 1,000 warm-up steps. Mixed-precision training is enabled for efficiency. Evaluation is carried out every 1,000 steps on a held-out validation set containing 1,000 document pairs, and the best-performing checkpoint is selected.

For both LMS (Round 2) and LMS (Finetuned MLM) Round 2, a refinement stage is introduced, in which the encoder is retrained on hard negative examples. These hard negatives are generated by re-encoding the corpus using the encoder trained in Round 1 and selecting the most similar non-relevant documents for each query. The fine-tuning setup in Round 2 mirrors that of Round 1, with the exception that the batch size is reduced to 16 to accommodate increased computational demands.

Overall, these hyperparameter choices are grounded in best practices for Transformer-based retrieval systems and are consistently applied across all pipeline variants to ensure valid and fair performance comparisons.

D. Ablation Study on Similarity Metrics in Contrastive Loss

To evaluate the impact of different similarity (or dissimilarity) metrics D_W in the contrastive loss function, an ablation study compared three widely used metrics: cosine similarity, Manhattan distance, and Euclidean distance. This experiment was carried out using the proposed pipeline LMS (Finetuned MLM) Round 2 under consistent training conditions. For all configurations, the margin hyperparameter was fixed at $margin = 0.5$ to ensure fair comparison. Table IV presents the results across several retrieval metrics. Cosine similarity consistently outperformed Manhattan and Euclidean metrics across all key measures—Recalls@3/5/10/20/200, MAP@10, MRR@10, and nDCG@10. Manhattan distance remains competitive at higher recall thresholds (R@50/100), whereas Euclidean distance lags overall.

These findings reinforce the choice of cosine similarity as the preferred similarity metric in this implementation. Its superior performance can be attributed to its ability to measure angular similarity, which aligns well with semantic closeness in dense text embeddings. In contrast, Euclidean and Manhattan distances are more sensitive to vector magnitudes and may not capture directional similarity as effectively in high-dimensional embedding spaces.

TABLE IV. COMPARISON OF SIMILARITY METRICS IN CONTRASTIVE LOSS ON THE JAPANESE LEGAL RETRIEVAL DATASET

D_W/Metric	R@3	R@5	R@10	R@20	R@50	R@100	R@200	MAP@10	MRR@10	nDCG@10
Manhattan	0.6064	0.6571	0.7252	0.7797	**0.8720**	**0.9339**	0.9567	0.5755	0.8167	0.6449
Euclidean	0.5962	0.6377	0.7260	0.7960	0.8597	0.8897	0.9333	0.5572	0.7658	0.6336
Cosine	**0.6872**	**0.7250**	**0.7953**	**0.8257**	0.8567	0.9158	**0.9746**	**0.6542**	**0.8238**	**0.7265**

Each row corresponds to a similarity measure, each column to a retrieval metric.

Bold indicates the highest score per column.

VI. CONCLUSION

This study presents a novel approach to Japanese legal text retrieval, introducing a tailored two-phase retrieval pipeline and an ensemble model to enhance retrieval performance. Using advanced LMs, this approach establishes a new state-of-the-art over baselines on the Japanese legal dataset while delivering competitive performance on widely recognized benchmarks. The proposed pipeline addresses the unique challenges of the Japanese legal context, providing a robust and adaptable solution for complex retrieval tasks. Empirical evaluations demonstrate the effectiveness of the proposed approach, validating its applicability across diverse scenarios and datasets. In addition, the integration of an ensemble model ensures consistent performance improvements, further establishing the versatility of the proposed framework.

Future work will focus on exploring semantic chunking, splitting each document into semantically coherent segments, and embedding both chunks and the user's query with the shared encoder to better handle very long documents. The top-k most relevant chunks will be retrieved, and then, a prompt-based LLM filter will be applied to select the passages most pertinent to the query. For complex, multiple-term queries, an LLM can be used to decompose a user's input into finer subqueries, retrieve results for each subquery independently, and then aggregate these results into a unified set of highly relevant documents. These extensions aim to further improve the system's robustness and precision when faced with long texts and nuanced queries typical of legal tasks.

ACKNOWLEDGMENT

This work was supported and conducted at AJ Technologies.

REFERENCES

[1] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, Jan. 1988, https://doi.org/10.1016/0306-4573(88)90021-0.

[2] S. Robertson, H. Zaragoza, and M. Taylor, "Simple BM25 extension to multiple weighted fields," in *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, Washington, DC, USA, Nov. 2004, pp. 42–49, https://doi.org/10.1145/1031171.1031181.

[3] S. E. Robertson and S. Walker, "Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval," in *SIGIR*

'94, B. W. Croft and C. J. Van Rijsbergen, Eds. Springer London, 1994, pp. 232–241.

[4] S. Khalid, S. Khusro, I. Ullah, and G. Dawson-Amoah, "On The Current State of Scholarly Retrieval Systems," *Engineering, Technology & Applied Science Research*, vol. 9, no. 1, pp. 3863–3870, Feb. 2019, https://doi.org/10.48084/etasr.2448.

[5] S. Khalid and S. Wu, "Supporting Scholarly Search by Query Expansion and Citation Analysis," *Engineering, Technology & Applied Science Research*, vol. 10, no. 4, pp. 6102–6108, Aug. 2020, https://doi.org/10.48084/etasr.3655.

[6] Y. Lv and C. Zhai, "Lower-bounding term frequency normalization," in *Proceedings of the 20th ACM International Conference on Information and Knowledge management*, Oct. 2011, pp. 7–16, https://doi.org/10.1145/2063576.2063584.

[7] L. Gao and J. Callan, "Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, Ireland, 2022, pp. 2843–2853, https://doi.org/10.18653/v1/2022.acl-long.203.

[8] A. Sil *et al.*, "PrimeQA: The Prime Repository for State-of-the-Art Multilingual Question Answering Research and Development," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Toronto, Canada, 2023, pp. 51–62, https://doi.org/10.18653/v1/2023.acl-demo.5.

[9] Q. Jin, A. Shin, and Z. Lu, "LADER: Log-Augmented DEnse Retrieval for Biomedical Literature Search," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Taipei, Taiwan, Jul. 2023, pp. 2092–2097, https://doi.org/10.1145/3539618.3592005.

[10] S. Zhuang *et al.*, "Bridging the Gap Between Indexing and Retrieval for Differentiable Search Index with Query Generation." arXiv, 2022, https://doi.org/10.48550/ARXIV.2206.10128.

[11] W. Sun *et al.*, "Learning to Tokenize for Generative Retrieval," *Advances in Neural Information Processing Systems*, vol. 36, pp. 46345–46361, Dec. 2023.

[12] T. Gao, X. Yao, and D. Chen, "SimCSE: Simple Contrastive Learning of Sentence Embeddings," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic, 2021, pp. 6894–6910, https://doi.org/10.18653/v1/2021.emnlp-main.552.

[13] L. Xiong *et al.*, "Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval." arXiv, Oct. 20, 2020, https://doi.org/10.48550/arXiv.2007.00808.

[14] J. Niklaus, V. Matoshi, P. Rani, A. Galassi, M. Stürmer, and I. Chalkidis, "LEXTREME: A Multi-Lingual and Multi-Task Benchmark for the Legal Domain," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, Singapore, 2023, pp. 3016–3054, https://doi.org/10.18653/v1/2023.findings-emnlp.200.

[15] H. Li, Y. Shao, Y. Wu, Q. Ai, Y. Ma, and Y. Liu, "LeCaRDv2: A Large-Scale Chinese Legal Case Retrieval Dataset," in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Washington, DC, USA, Jul. 2024, pp. 2251–2260, https://doi.org/10.1145/3626772.3657887.

[16] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality Reduction by Learning an Invariant Mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06)*, New York, NY, USA, 2006, vol. 2, pp. 1735–1742, https://doi.org/10.1109/CVPR.2006.100.

[17] Q. H. Trung, N. V. H. Phuc, L. T. Hoang, Q. H. Hieu, and V. N. L. Duy, "Adaptive Two-Phase Finetuning LLMs for Japanese Legal Text Retrieval." arXiv, Dec. 03, 2024, https://doi.org/10.48550/arXiv.2412.13205.

[18] T. Nguyen *et al.*, "MS MARCO: A Human Generated MAchine Reading COmprehension Dataset," presented at the 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona Spain, 2016, vol. 1773.

[19] S. Khalid, S. Wu, and F. Zhang, "A multi-objective approach to determining the usefulness of papers in academic search," *Data Technologies and Applications*, vol. 55, no. 5, pp. 734–748, Oct. 2021, https://doi.org/10.1108/DTA-05-2020-0104.

[20] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 2019, pp. 3980–3990, https://doi.org/10.18653/v1/D19-1410.

[21] Y. Yang *et al.*, "Multilingual Universal Sentence Encoder for Semantic Retrieval," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020, pp. 87–94, https://doi.org/10.18653/v1/2020.acl-demos.12.

[22] V. Karpukhin *et al.*, "Dense Passage Retrieval for Open-Domain Question Answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 6769–6781, https://doi.org/10.18653/v1/2020.emnlp-main.550.

[23] Y. Sasazawa, K. Yokote, O. Imaichi, and Y. Sogawa, "Text Retrieval with Multi-Stage Re-Ranking Models." arXiv, 2023, https://doi.org/10.48550/ARXIV.2311.07994.

[24] Y. Tay *et al.*, "Transformer Memory as a Differentiable Search Index," *Advances in Neural Information Processing Systems*, vol. 35, pp. 21831–21843, Dec. 2022.