The Knowledge Engineering Review

RESEARCH ARTICLE

Open Access

https://doi.org/10.48130/ker-0026-0018

The Knowledge Engineering Review **2026**, 41: e017

Structure-aware legal case retrieval via graph-based structural modeling and dense semantic retrieval

Chunling Dong*, Yang Yang and Mingsen Li

School of Computer and Cyber Sciences, Communication University of China, Beijing 100024, China

* Correspondence: dongcl15@tsinghua.org.cn (Dong C)

Abstract

Legal case retrieval is a fundamental task in intelligent judicial systems, where both accuracy and interpretability are critical for providing reliable decision support. However, the domain-specific terminology, complex structures, and diverse expressions inherent in legal texts hinder the effectiveness of methods that rely solely on dense retrieval or graph-based modeling. To address these challenges, we propose Graph-enhanced Dense Passage Retrieval (G-DPR), a structure-aware legal case retrieval framework that integrates graph-based structural modeling with Dense Passage Retrieval (DPR) to jointly capture legal structural dependencies and semantic relevance. In the graph-based channel, an automated legal knowledge graph is constructed using a fine-tuned legal large language model and encoded via Relational Graph Convolutional Networks (RGCN) to capture structural dependencies and calculate graph-based similarities. In the dense retrieval channel, LegalBERT enhances the DPR dual-encoder, enabling fine-grained semantic representation of queries and candidate cases. Furthermore, a joint re-ranking strategy is introduced to fuse graph-based and semantic similarities. Experiments conducted on the LeCaRD dataset demonstrate that G-DPR consistently outperforms representative baseline methods across multiple evaluation metrics, while further improving retrieval interpretability and exhibiting strong potential for real-world legal AI applications.

Keywords: Legal case retrieval, Graph-based structural modeling, Legal knowledge graph, Dense passage retrieval, Structure-aware retrieval

Citation: Dong C, Yang Y, Li M. 2026. Structure-aware legal case retrieval via graph-based structural modeling and dense semantic retrieval. *The Knowledge Engineering Review* 41: e017 https://doi.org/10.48130/ker-0026-0018

1 Introduction

Legal Case Retrieval (LCR) is a pivotal component of intelligent legal systems. Its primary objective is to identify highly relevant historical cases from legal databases based on the factual descriptions or statutory provisions of a target case. In judicial practice, accurate LCR not only enhances the consistency and fairness of court verdicts but also improves the interpretability of legal reasoning and decision-making processes. Existing keyword-based retrieval methods often have significant limitations in capturing the deep semantic associations between legal provisions and case facts. As a result, the relevance and accuracy of retrieval results are frequently insufficient to support complex legal tasks.

In recent years, Dense Retrieval (DR) techniques have achieved remarkable success in the field of information retrieval. A representative model, Dense Passage Retrieval (DPR), has performed effectively in tasks such as open-domain question answering and knowledge retrieval. DPR adopts a dual-encoder architecture that leverages deep neural networks to learn low-dimensional dense representations of textual data, enabling both queries and documents to be mapped into a shared semantic space. Compared to traditional sparse retrieval methods such as BM25, DPR is more effective in modeling sentence-level semantic similarity and benefits from efficient retrieval and indexing mechanisms that significantly reduce computational overhead.

Applying DPR to legal case retrieval provides a promising direction for improving retrieval effectiveness. However, several domain-specific challenges remain. Legal documents are typically lengthy, structurally complex, and characterized by specialized terminology and implicit legal logic. Directly transferring standard DPR models to the legal domain often struggles to capture these characteristics, thereby limiting retrieval performance. More importantly, legal case retrieval depends not only on semantic similarity but also on structural dependencies among facts, legal provisions, and reasoning processes. Existing dense retrieval methods mainly focus on semantic matching and lack explicit mechanisms to model such structural relationships. Although DR improves semantic alignment, it remains insufficient for capturing the underlying legal structures that influence judicial decisions.

To address these limitations, this paper proposes Graph-enhanced Dense Passage Retrieval (G-DPR), a structure-aware legal case retrieval framework that integrates graph-based structural modeling with dense semantic retrieval. Specifically, a legal knowledge graph is constructed to represent structural dependencies among cases, enabling explicit modeling of factual, statutory, and relational structures that are often implicit in raw text. Compared with purely semantic representations, the graph module is capable of capturing long-range dependencies and inter-case relationships, thereby improving the model's ability to distinguish cases with similar surface semantics but different legal implications. Graph representation learning is then employed to capture structural similarity in this graph space, while DPR is adopted to model semantic relevance. A joint reranking strategy is further introduced to fuse structural and semantic signals, improving retrieval effectiveness and interpretability for legal case retrieval tasks.

2 Related work

Considerable progress has been made in advancing the understanding and retrieval of legal documents, with extensive research focusing on adapting pre-trained language models, integrating legal knowledge, and leveraging structured representations.

© The Author(s)

https://doi.org/10.48130/ker-0026-0018

2.1 From general pre-trained models to domain-specific legal models

General pre-trained models such as BERT[1] and RoBERTa[2] have demonstrated strong performance in NLP tasks but face challenges in the legal domain due to domain-specific terminology, lengthy texts, and complex logical structures. To address these issues, LEGAL-BERT[3] was introduced by further pre-training BERT on large-scale legal corpora, significantly enhancing legal semantic understanding. To mitigate the scarcity of labeled data in similar case retrieval, Gao et al.[4] proposed the LEAD dataset, which automatically generates large-scale training samples through template matching and content filtering. In retrieval tasks, DPR[5] employs a dual-encoder architecture to independently encode queries and documents, improving the efficiency and accuracy of dense representation learning and inspiring its application in legal case matching. With the development of legal large language models (LLMs), domain adaptation has gained momentum. In China, efforts such as the Faxin LLM and Peking University's ChatLaw[6,7] target statute matching, legal reasoning, and structured legal understanding. Internationally, benchmarks like LegalBench[8] and models such as LawyerGPT[9] promote systematic evaluation and enhance legal LLM capabilities in tasks like legal reasoning and statute citation. These developments reflect a growing trend toward tailoring general LLMs to meet the unique requirements of legal applications.

2.2 Structure-aware and knowledge-enhanced retrieval models

To better model ultra-long legal documents, various techniques have been proposed to enhance contextual understanding. Vuong et al.[10] leveraged paragraph-level relational modeling; Nguyen et al.[11] introduced sparse attention mechanisms, and Beltagy et al.[12] combined local and task-driven global attention to improve long-document comprehension. In similar case retrieval, where sensitivity to key legal elements and document structure is critical, domain-specific models have been developed. CaseEncoder[13] incorporates statute-guided sample selection and Biased Circle Loss to improve discrimination. Bhattacharya et al.[14] enhanced retrieval by exploiting citation network information. Tang et al. introduced CaseGNN[15] and CaseLink[16], using graph neural networks (GNNs) to model legal document structure and inter-case references. The SLR framework[17] further integrates intra- and inter-document structural signals to improve case matching. To capture key legal semantics, SAILER[18] employs an asymmetric encoder-decoder architecture with multiple pre-training objectives, achieving strong performance even in low-resource settings. Bi et al. proposed L-HetGRL[19], which constructs a heterogeneous graph combining legal entities and document content and applies a case-aware semantic alignment strategy. Later, the EMV framework[20] incorporated event knowledge into heterogeneous graph structures, demonstrating the effectiveness of structured and knowledge-enriched representations in boosting retrieval performance.

2.3 Causality-aware and structure-enhanced legal retrieval

Beyond semantic matching, recent studies have explored incorporating causal and structural information to better model legal reasoning processes. Sun et al.[21] introduced causal structures between statutes and case facts to reduce confounding effects and improve consistency in legal case matching. Building upon causal modeling, Zhang et al.[22] proposed CFGL-LCR, which combines counterfactual reasoning with graph neural networks to uncover latent relational paths among cases and capture complex legal dependencies. To further enhance semantic consistency across legal representations, Wang et al.[23] developed a multi-view contrastive learning framework that jointly models legal elements, charges, and entities under causal alignment objectives. These studies demonstrate that explicitly modeling structural and relational dependencies can significantly improve legal retrieval beyond pure semantic similarity.

2.4 Hybrid Retrieval and Retrieval-Augmented Frameworks

Existing hybrid retrieval and RAG frameworks have explored complementary evidence sources for knowledge-intensive tasks. Sarmah et al.[24] proposed HybridRAG, which integrates knowledge-graph retrieval with dense vector retrieval to support information extraction and answer generation. Kalra et al.[25] introduced HyPA-RAG for legal and policy applications, combining dense, sparse, and graph-based retrieval under adaptive parameter control. Wiratunga et al.[26] proposed CBR-RAG, which incorporates case-based reasoning into the RAG pipeline to improve the consistency of legal question answering.

However, the methodological distinction between these frameworks and G-DPR should not be framed primarily in terms of the presence or absence of a generation stage. Instead, the core difference lies in how structured knowledge is constructed and operationalized for retrieval. HybridRAG builds open-ended document-level triplet graphs and concatenates retrieved graph contexts with vector-retrieved text as input to the generation module. HyPA-RAG extracts relational triplets from a legal-policy knowledge graph and dynamically appends them to text chunks according to query complexity, primarily to enhance the faithfulness of generated responses. CBR-RAG organizes legal cases through case-based similarity containers, rather than constructing an explicitly schema-constrained legal knowledge graph for structural matching.

In contrast, G-DPR adopts a schema-constrained, case-oriented graph construction strategy. It separately models factual structures from case narratives and normative knowledge from criminal law provisions, with entities and relations strictly bounded by a predefined legal schema and further filtered by entity-consistency and relation-type compatibility checks. The resulting heterogeneous graphs are encoded via Relational Graph Convolutional Networks (RGCN) to derive structural similarity scores, which are directly fused into candidate case reranking. In this framework, the graph does not merely serve as auxiliary context for generation; it defines a structure-aware similarity space that distinguishes cases with similar surface semantics but divergent legal elements or causal structures. The absence of a generation stage in G-DPR is a natural consequence of its retrieval-ranking objective, rather than the principal source of its methodological novelty.

3 Methodology

This chapter proposes a hybrid case retrieval method that integrates graph-based retrieval with dense vector retrieval, termed Graph-enhanced Dense Passage Retrieval (G-DPR). The proposed approach combines structured graph information with semantic vector representations to perform multi-level retrieval of legal cases, thereby improving both the accuracy and relevance of similar case retrieval.

The overall framework consists of three core modules: a graph-based structural retrieval module, dense passage retrieval (DPR), and a re-ranking module. Specifically, the graph-based structural retrieval module leverages entities and relational information within the legal knowledge graph to retrieve and enrich structured semantic representations of cases. The DPR module, built upon deep semantic embedding

techniques, performs dense vector matching over case texts to identify semantically similar candidate cases.

Based on the initial retrieval results, a re-ranking strategy is further applied to refine the candidate set by assigning relevance scores, ultimately selecting the most relevant similar cases with respect to the target case. The architecture of the proposed G-DPR framework is illustrated in Fig. 1.

3.1 Graph similarity retrieval based on graph-enhanced structural retrieval

To effectively capture the complex logical relationships among factual elements in legal cases and enhance the matching of structured semantic information between cases, we propose a graph similarity retrieval method based on graph-enhanced structural retrieval. This method integrates the semantic modeling capabilities of large language models (LLMs) with the structure-aware representation power of graph neural networks, as shown in Fig. 2. By leveraging entity relations and knowledge structures inherent in legal contexts, the approach enhances the accuracy and interpretability of similar case retrieval at the knowledge graph level.

3.1.1 Fine-tuning and construction of a domain-specific legal LLM

To adapt to the linguistic characteristics and stylistic nuances of legal texts, this study fine-tunes the DeepSeek R1-7B model using both supervised and unsupervised judicial corpora, thereby developing a domain-specific LLM named DeepSeek-Law. The training data encompasses a wide range of legal documents, including prototypical court decisions, interpretive texts of criminal statutes, judicial rationales, and case summaries. The fine-tuning process is specifically designed to improve performance on key legal tasks such as legal entity recognition, causal chain construction, and statute citation understanding. This targeted adaptation enhances the model's capacity for legal text comprehension and generation. An overview of the fine-tuning objectives and data composition is illustrated below:

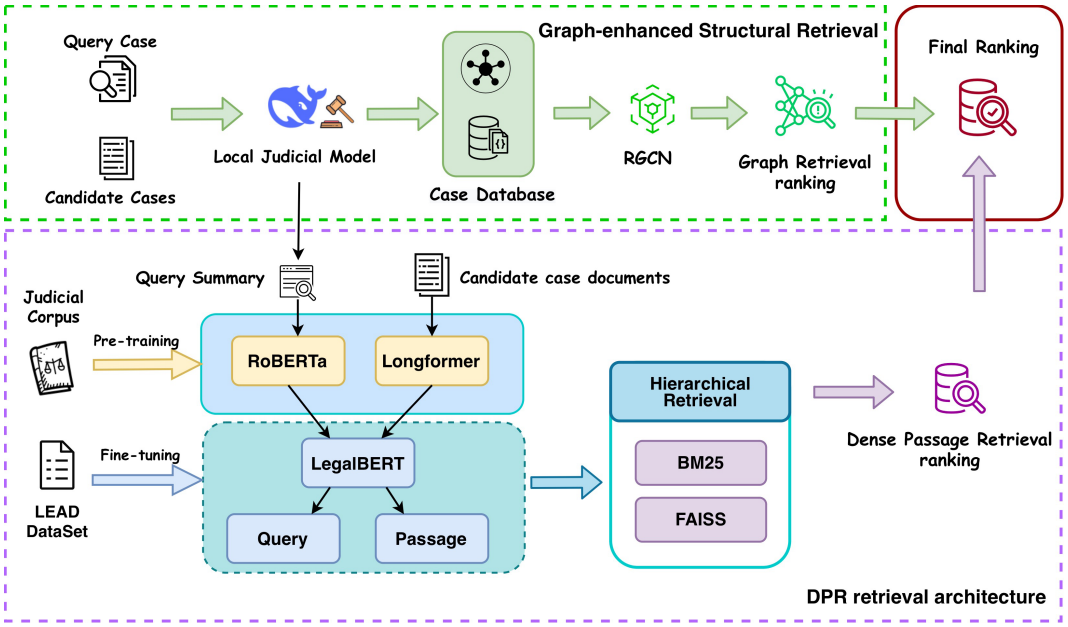


Fig. 1 The proposed architecture of the G-DPR framework.

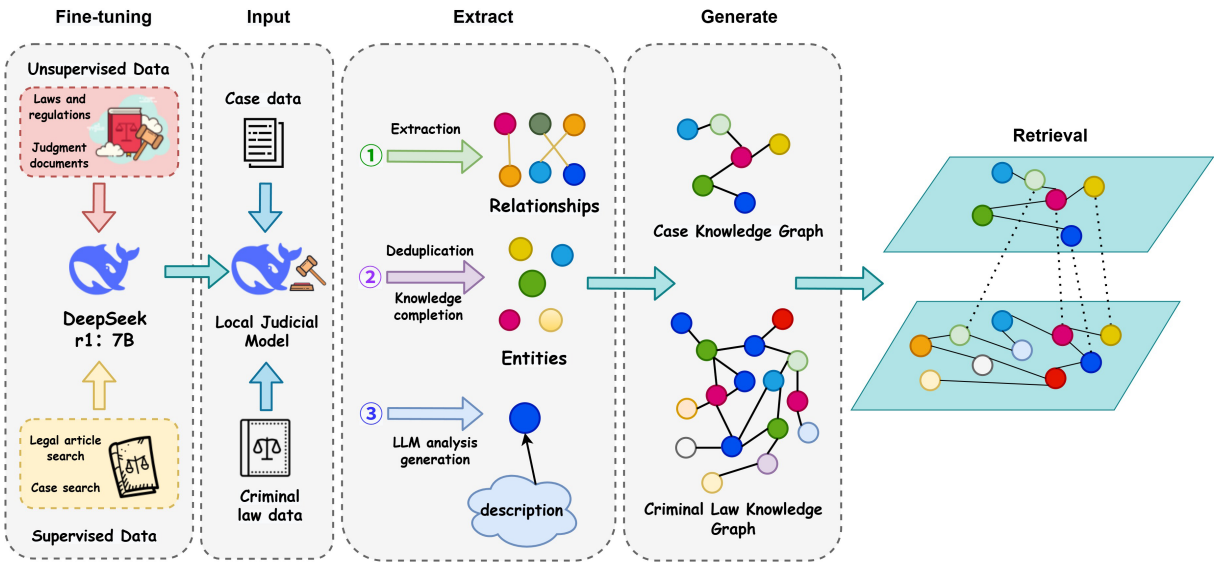


Fig. 2 The procedure of graph similarity retrieval.

https://doi.org/10.48130/ker-0026-0018

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{NER} + \lambda_2 \mathcal{L}_{Casual} + \lambda_3 \mathcal{L}_{LawRef} \quad (1)$$

where, $\mathcal{L}_{NER}$ denotes the cross-entropy loss for legal entity recognition, $\mathcal{L}_{Causal}$ is the sequence labeling loss for causal chain extraction, $\mathcal{L}_{LawRef}$ represents the contrastive loss for statute citation matching. λ_1, λ_2, λ_3 are the corresponding task-specific weights controlling the contribution of each component.

3.1.2 Construction of case and criminal law knowledge graphs

To effectively capture the structured semantic information embedded in legal texts and enhance the downstream retrieval model's understanding of legal logic, we propose an automated knowledge graph construction framework tailored to the judicial domain. This framework is built upon a domain-adapted LLM and integrates semantic extraction, structural normalization, and knowledge completion. It enables unified modeling of both criminal statutes and representative case documents, resulting in the construction of two distinct but structurally compatible graphs: Criminal Law Knowledge Graphs and Case Knowledge Graphs. These graphs serve as the foundation for graph-based similarity modeling and cross-graph alignment in subsequent retrieval tasks.

3.1.2.1 Semantic unit recognition and initial graph construction

The construction process begins with DeepSeek-Law, a domain-specialized LLM fine-tuned on legal corpora. The input includes both statutory provisions and factual descriptions of legal cases. The model performs semantic parsing to automatically identify key legal entities—such as perpetrator, victim, time points, objects of action, and legal consequences—as well as the semantic relations between them, such as commits, causes, violates, and based on. These entities and relations are then used to form the initial entity-relation graph, which captures the core facts and logical structure underlying each legal article or case scenario.

The full schema used for construction is predefined according to criminal law logic. Entity types include Person (perpetrator/victim), Act, Object, Time, Legal Provision, Charge, and Legal Consequence. The core relation types and their valid head-tail entity pairs are commits (Person → Act), causes (Act → Legal Consequence), violates (Act → Legal Provision), corresponds_to (Case Fact → Charge), based_on (Judgment → Legal Provision), results_in (Act → Charge). All extracted candidate triples are first checked against this schema to filter structurally invalid combinations.

3.1.2.2 Structural normalization and consistency validation

After initial extraction, a rule-based validation and normalization step is applied before graph encoding. First, incomplete triples with missing head or tail entities and unresolved ambiguous entity mentions are filtered out. Second, type-incompatible edges that do not conform to the predefined schema constraints are removed. Third, extraction results with confidence scores below the threshold are discarded to reduce low-quality noise. The remaining triples are then normalized in entity naming and deduplicated to eliminate redundant nodes and repeated edges. This procedure reduces semantically invalid structures and improves the structural consistency of the heterogeneous graphs fed into the subsequent RGCN encoder.

3.1.2.3 Knowledge completion and semantic enhancement

To mitigate the problem of semantic discontinuities within the initial graphs, an automated knowledge completion mechanism is employed. This component integrates contextual semantics, domain-specific knowledge bases, and graph topological features to enrich the graph structure. It automatically adds missing relational edges, infers latent semantic paths, and applies lexical expansion techniques to incorporate hypernyms and hyponyms. These enhancements introduce additional semantic bridges across graph components, thereby improving both the connectivity and coverage of the knowledge graphs.

To enhance the completeness of the constructed legal knowledge graph, we further perform lightweight knowledge completion during the graph construction stage. Specifically, missing relations between legal entities (e.g., cases, charges, and legal provisions) are inferred based on co-occurrence patterns and structured rule constraints derived from annotated case texts. This process is guided by the legal domain schema to ensure that only semantically and legally plausible relations are introduced. To mitigate potential hallucinations from the LLM during relation extraction, we adopt a constrained extraction strategy, where the LLM is required to generate relations strictly within a predefined relation set.

For quality assurance of the constructed knowledge graph, we conduct a manual quality inspection on a randomly selected subset of 50 cases from LeCaRD, covering approximately 800 extracted triples. Legal researchers evaluate each triple according to three criteria: entity correctness, relation correctness, and schema compliance. The inspection achieves 89.16% accuracy at the entity level, 87.28% accuracy at the relation level, and an 84.01% schema-compliance rate. These results show that the automatically constructed graphs reach acceptable extraction quality, while also indicating that structural noise still exists in a portion of triples. The aforementioned schema-based validation, normalization, and deduplication procedures are designed to mitigate such noise and stabilize the graph quality for downstream representation learning.

3.1.2.4 Semantic description and heterogeneous graph modeling

To enhance the interpretability of the constructed knowledge graphs, the system generates natural language descriptions for key nodes along critical reasoning paths. These textual annotations support subsequent legal case explanation and improve the transparency of the retrieval process. All graphs are uniformly represented as multi-relational heterogeneous structures, where node types (e.g., person, action, object) and relation types (e.g., causality, citation, affiliation) are explicitly distinguished. This heterogeneous modeling framework lays a solid foundation for downstream graph neural network modeling and structural similarity matching. The automatic construction process for knowledge graphs is formally described in Algorithm 1.

3.1.3 Graph similarity calculation based on RGCN

Accurately capturing the structural semantic similarity between a query case and candidate cases is a critical step toward achieving high-quality retrieval. In this study, we adopt the relational graph convolutional networks (RGCN)[27] to perform graph representation learning on the constructed legal knowledge graphs. By incorporating relational edge types and heterogeneous node features, the RGCN enables

Algorithm 1. Build_knowledge_graph.

Input: *text*: Case or statute document
model: Domain-adapted DeepSeek-Law model
Output: *G*: Multi-relational heterogeneous knowledge graph
Step1: *entities*, *relations* = model.extract_entities_and_relations(*text*)
Step2: *G* = Graph()
for *entity* in *entities*: *G*.add_node(entity)
for (*head*, *relation*, *tail*) in relations: *G*.add_edge(*head*, *tail*, *relation*)
Step3: *G* = remove_redundant_edges_and_nodes(*G*)
G = resolve_semantic_ambiguities(*G*)
Step4: *G* = complete_missing_links(*G*)
G = enrich_with_hypernyms_and_reasoned_paths(*G*)
Step5: for *path* in key_paths(*G*):
description = generate_textual_summary(*path*)
attach_description_to_path(*G*, *path*, *description*)
return *G*

the model to effectively encode complex legal structures and latent logical dependencies. The learned graph embeddings are subsequently used to compute pairwise similarities between the query and candidate graphs, supporting efficient Top-K retrieval based on structural alignment. Specifically, RGCN encodes both the query graph and candidate case graphs to learn structural semantic embeddings representing the entire graph. These embeddings capture not only node attribute information but also the contextual position of each node within the graph and the relational paths connecting it to other nodes. Formally, for each node at the l^{th} layer, its representation $h_v^{(l)}$ is updated as follows:

$$h_v^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{u\in\mathcal{N}_r(v)}\frac{1}{C_{v,r}}W_r^{(l)}h_u^{(l-1)} + W_0^{(l)}h_u^{(l-1)}\right) \quad (2)$$

where, $\mathcal{R}$ denotes the set of relation types, $\mathcal{N}_r(v)$ is the set of neighboring nodes connected to node v via relation r, $C_{v,r}$ is a normalization constant, $W_r^{(l)}$ and $W_0^{(l)}$ are learnable weight matrices specific to relation r and self-loop respectively, and σ is a non-linear activation function.

To quantitatively assess the similarity between graphs, this study employs cosine similarity as the fundamental metric to compare the embedding representations of the query graph and each candidate graph. Given a query case graph $\mathcal{G}_q$ and a candidate case graph $\mathcal{G}_c$, their similarity is computed using Eq. (3). The resulting similarity score effectively reflects the degree of alignment between cases in terms of factual structure and legal applicability. Once similarity scores are obtained for all candidate graphs, the system ranks them and selects the Top-K most similar cases to form the final set of recommended precedents:

$$Sim\left(\mathcal{G}_q,\mathcal{G}_q\right) = \frac{f\left(\mathcal{G}_q\right)\cdot f\left(\mathcal{G}_c\right)}{\left\|f\left(\mathcal{G}_q\right)\right\|\cdot\left\|f\left(\mathcal{G}_c\right)\right\|} \quad (3)$$

where, $\mathcal{G}$ denotes the structural embeddings of the entire graph, obtained through the RGCN encoder. The numerator represents the dot product of the two embedding vectors, while the denominator corresponds to the product of their Euclidean norms.

3.2 Vector-based retrieval with DPR

DPR is a method that has made significant progress in text retrieval tasks in recent years. To improve the performance of DPR in legal case retrieval, this study adopts RoBERTa and Longformer as backbone encoders, both of which are further pretrained on large-scale legal corpora to enhance the model's understanding of legal texts. Subsequently, LegalBERT is fine-tuned on a case-level retrieval task, aligning legal case summaries with entries in the case base to optimize retrieval performance for downstream legal applications. The DPR-based retrieval framework employed in this study is illustrated in Fig. 3.

3.2.1 Pre-training optimization of RoBERTa and Longformer

3.2.1.1 RoBERTa pre-training optimization strategy based on contrastive learning

This study conducts preliminary pretraining of RoBERTa on a large-scale legal corpus to enhance its comprehension of legal texts and improve its applicability to dense vector retrieval tasks. For corpus construction, a domain-specific legal dataset is built based on the LEAD corpus, integrating multi-source data including statutes, judicial decisions, and legal interpretations to form a cross-domain legal textual resource. Given the significant differences between legal texts and general-domain corpora in terms of syntactic structures, specialized terminology, and discourse logic, a dynamic masking strategy is adopted during pretraining to prevent the model from memorizing fixed masked positions, thereby improving its contextual generalization capabilities.

Considering the complex semantic similarity requirements between case summaries and documents in the case repository, this study incorporates a contrastive learning strategy during the pretraining phase of RoBERTa to enhance the model's ability to distinguish between semantically similar legal texts. Specifically, positive and negative sample pairs are constructed, where positive samples are drawn from related cases, similar judgments, or documents applying the same legal provisions, while negative samples are randomly selected from unrelated legal domains. The training objective is formulated using a contrastive loss function as follows:

$$\mathcal{L}_{CL} = -\log\frac{\exp\left(sim\left(h_i,h_i^+\right)/t\right)}{\sum_{j=1}^{N}\exp\left(sim\left(h_i,h_j\right)/t\right)} \quad (4)$$

Here, h_i and h_i^+ represent the output vectors of the RoBERTa encoder for the query and the corresponding positive example, respectively. The function *sim* denotes cosine similarity, and t is a temperature hyperparameter that adjusts the model's sensitivity to similarity differences.

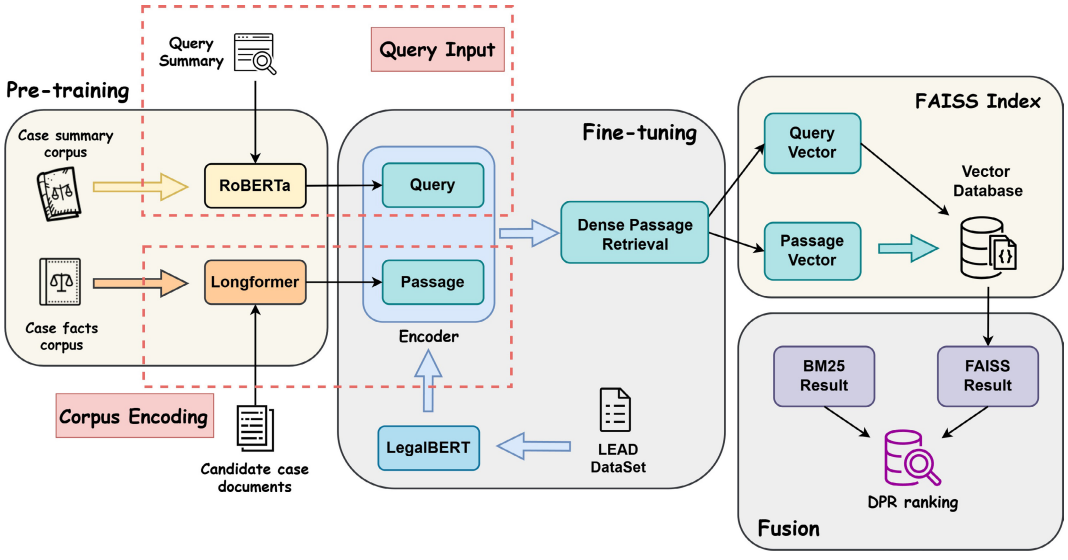


Fig. 3 Architecture of the DPR-based retrieval framework.

https://doi.org/10.48130/ker-0026-0018

This loss function encourages the model to pull representations of positive pairs closer while pushing apart those of negative pairs, thereby significantly enhancing RoBERTa's semantic discrimination capability in legal contexts. Additionally, to prevent gradient explosion during large-scale corpus training and to ensure training stability over extended periods, gradient accumulation and mixed-precision training techniques are employed to accommodate limited computational resources.

3.2.1.2 Longformer-based long text layered encoding strategy

In legal case retrieval tasks, legal documents such as judgments, statutes, and case analyses are typically much longer than the 512-token input limit of traditional models like BERT and RoBERTa. Truncating these texts risks losing critical information, while segmenting them disrupts semantic coherence. To address this issue, this study introduces Longformer as the encoder to enhance the modeling capacity for lengthy legal texts, thereby improving the accuracy and robustness of dense vector retrieval. The core innovation of Longformer lies in its sparse attention mechanism. Unlike the fully connected attention structure of standard Transformers, Longformer's attention matrix retains only a subset of token interactions. It combines local window-based attention with global token attention, significantly reducing computational complexity. The attention weight computation is expressed as follows:

$$A_{i,j} = \begin{cases} \text{Softmax}\left(\dfrac{Q_i \cdot K_j^T}{\sqrt{d_k}}\right), & j \in \mathcal{W}(i) \cup \mathcal{G} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where, $\mathcal{W}(i)$ denotes the local sliding window surrounding token i, and $\mathcal{G}$ represents the set of globally attended token positions. Q and K are the query and key vectors, respectively, and d_k is the scaling factor corresponding to the dimensionality of the vectors. This structure ensures that the model can effectively maintain attention to critical global information while significantly reducing computational overhead, making it well-suited for processing lengthy legal documents.

To ensure semantic consistency between legal facts and judicial reasoning, this study proposes a hierarchical encoding strategy tailored for long legal texts, aiming to capture both global and local information for improved modeling of case similarity and judgment semantics. Specifically, the entire judgment document is first encoded using Longformer to obtain a global semantic representation h_{long}. Concurrently, RoBERTa is applied to encode key information segments—such as case highlights and applicable legal provisions—yielding a fine-grained local representation h_{local}. The final representation is constructed by concatenating these two vectors:

$$h_{final} = Conact\left(h_{long}, h_{local}\right) \tag{6}$$

3.2.2 Retrieval strategy integrating DPR fine-tuning and hierarchical semantic matching

3.2.2.1 DPR fine-tuning method based on LegalBERT and hard negative sample mining

Since traditional DPR models are primarily trained on open-domain data, their encoders struggle to capture the domain-specific terminology, complex syntax, and reasoning logic inherent in legal texts. As a result, directly applying such models may lead to reduced accuracy in legal case matching. To address this challenge, the present study employs LegalBERT as both the query encoder and the case encoder within the DPR framework, as illustrated in Algorithm 2. This setup encodes the case summary and candidate case documents separately, optimizing the representation of legal texts in vector space so that semantically relevant cases are positioned closer together, thereby enhancing retrieval precision.

During training, a hard negative sampling strategy is adopted. Specifically, BM25 is used to obtain preliminary retrieval results, from which cases that are lexically similar to the query but differ significantly in legal reasoning or judgment outcomes are selected as negative samples. These hard negatives—characterized by high surface-level similarity yet divergent legal implications—help improve the model's semantic discrimination capability in legal contexts. This mitigates overgeneralization in the vector space and promotes more targeted and accurate retrieval performance.

Specifically, the representations of the query q and the document d are obtained using a shared-parameter LegalBERT encoder, yielding the vectors q_θ^T and d_ϕ, respectively. Their similarity is computed via dot product: $sim(q,d) = q_\theta^T d_\phi$. The training objective adopts the InfoNCE loss from contrastive learning to maximize the similarity between the positive pair (q, d^+), while minimizing the similarity between the query and hard negative samples d^-. The loss function is defined as:

$$\mathcal{L}_{DPR} = -\log \frac{\exp(sim(q,d^+)/t)}{\exp(sim(q,d^+)/t) + \sum_{d^- \in D^-} \exp(sim(q,d^-)/t)} \tag{7}$$

where, t is the temperature hyperparameter that controls the sensitivity to similarity differences, and D^- denotes the set of hard negative samples.

3.2.2.2 Retrieval method based on hierarchical filtering and semantic matching

To improve retrieval efficiency in large-scale legal case databases, this study employs FAISS (Facebook AI Similarity Search) as a high-performance vector retrieval engine. Considering the long-text nature and semantic complexity of legal documents, a hierarchical retrieval mechanism is adopted. Specifically, BM25 is first used to perform sparse vector-based preliminary filtering, selecting a candidate set of cases with high term-level relevance from the database. Then, the DPR model conducts dense semantic matching within this candidate set. This two-stage retrieval framework balances recall efficiency and matching accuracy, significantly alleviating the computational burden of dense retrieval in large-scale scenarios.

To further integrate the strengths of sparse and dense representations, this study proposes a Hybrid Scoring Strategy, which combines the BM25 term-matching score with the semantic similarity score from DPR through weighted fusion. Let the BM25 score between the query and candidate document be denoted as S_{BM25}, and the DPR-based semantic similarity score as S_{DPR}. The final fused score S_{Final} is calculated as follows:

Algorithm 2. LegalBERT-based DPR fine-tuning.

Input:	Query set Q, Case document set D, Pre-trained model *LegalBERT*
Output:	Optimized DPR model
Step1:	Initialize *query_encoder*, *document_encoder* = *LegalBERT* (shared parameters)
Step2:	For each query q in Q: *positive_doc* = SelectPositiveCase(q, D) *candidate_docs* = BM25InitialRetrieval(q, D, t*op_k*) *negative_docs* = SelectHardNegatives(*candidate_docs*, *positive_doc*)
Step3:	*q_vector* = query_encoder(q) *d_positive_vector* = document_encoder(*positive_doc*) *d_negative_vectors* = document_encoder(*negative_docs*)
Step4:	*sim_positive* = DotProduct(*q_vector*, *d_positive_vector*) *sim_negatives* = [DotProduct(*q_vector*, *d_neg*) for *d_neg* in *d_negative_vectors*]
Step5:	*loss* = InfoNCELoss(*sim_positive*, *sim_negatives*) UpdateModelParameters(*loss*, *query_encoder*, *document_encoder*) Return fine-tuned *query_encoder*, *document_encoder*

$$S_{Final} = \lambda \cdot S_{DPR} + (1-\lambda) \cdot S_{BM25}, \lambda \in [0,1] \quad (8)$$

where, λ is a weighting hyperparameter that adjusts the relative importance between dense semantic matching and sparse keyword matching.

3.3 Re-ranking by integrating graph and semantic similarities

Relying on a single similarity metric often fails to simultaneously capture both semantic content and structural features of legal cases. DPR, based on language model-derived embeddings, excels at capturing the overall semantic context of case documents. In contrast, the graph-based similarity retrieval method is adept at uncovering deeper structural information such as causal chains of events, references to legal provisions, and behavioral logic patterns. Leveraging either method in isolation falls short of fully modeling the intricate relationships between legal cases. To address this issue and enhance both the precision and interpretability of case matching, we propose a re-ranking mechanism that integrates semantic vector similarity with graph structural similarity. This hybrid approach refines the initially retrieved candidate set by jointly considering surface-level semantic alignment and deeper structural correspondence.

The implementation proceeds as follows. Given an initial set of retrieved candidate cases $\{d_1, d_2, \ldots, d_k\}$, we compute two types of similarity scores between each candidate d_i and the query case Q: a semantic similarity score S_{vec} using the DPR model, and a structural similarity score S_{graph} using the RGCN-based graph encoder. To effectively integrate both similarity dimensions, we adopt a linear weighted fusion strategy to compute the final ranking score for each candidate case:

$$S_{Final}(Q,d_i) = \alpha \cdot S_{vec}(Q,d_i) + (1-\alpha) \cdot S_{graph}(Q,d_i), \alpha \in [0,1] \quad (9)$$

where, α is a tunable parameter that balances the contribution of semantic and structural similarities. The candidate cases are then re-ranked according to S_{Final}, and the Top-K cases are selected as the final retrieval results.

This fusion mechanism not only preserves the DPR model's capability for capturing the global semantic understanding of legal texts, but also incorporates the RGCN model's strength in modeling causal relationships and adjudicative structures. By jointly leveraging both semantic and structural cues, the proposed approach significantly enhances the accuracy and interpretability of case matching under complex judicial contexts.

4 Experimental analysis

In this section, we conduct a comprehensive evaluation of the proposed G-DPR case retrieval algorithm on the LeCaRD dataset. The performance of G-DPR is evaluated against a set of representative retrieval baselines to demonstrate its capabilities in the legal case retrieval task. In addition, ablation studies are performed to investigate the contribution of graph-structured information to the overall retrieval performance.

4.1 Datasets and metrics

LeCaRD[28] is a benchmark dataset specifically designed to evaluate the performance of legal case retrieval systems. It contains standardized case descriptions, annotated relevant case pairs, and detailed judgment documents. LeCaRD provides expert-labeled similarity annotations, which enhance the legal validity and authority of retrieval evaluations. In this study, we use the LEAD dataset—an open-source corpus released by Tsinghua University—as the training data. LeCaRD is employed as the test set to evaluate the effectiveness of the proposed method under real-world legal retrieval scenarios.

To assess the accuracy, ranking quality, and overall performance of the retrieval system, three core evaluation metrics are employed: Precision (P), Normalized Discounted Cumulative Gain (NDCG), and Mean Average Precision (MAP).

4.1.1 Implementation Details

For the dense retrieval module, the training batch size is set to 128, the learning rate is set to 1×10^{-5}, and the model is trained for up to 100 epochs. Adam is adopted as the optimizer, combined with linear learning rate scheduling and warm-up. The dropout rate is set to 0.1, and the maximum input sequence length is set to 2,048.

The checkpoint achieving the best performance on the development set is selected for final evaluation. Regarding the retrieval candidate pool, each query in the standard LeCaRD evaluation protocol is associated with 100 candidate cases. Accordingly, all retrieval and ranking procedures are performed within this 100-case candidate set.

We also specify the fusion parameters in Eqs. (8) and (9). The BM25–DPR fusion weight λ in Eq. (8) is set to 0.4, while the graph-semantic reranking weight α in Eq. (9) is set to 0.6. Both parameters are tuned according to the performance on the development set. The slightly larger value of α assigns a greater contribution to graph structural similarity during the final reranking stage, while retaining the complementary semantic information provided by DPR.

4.2 Results and analysis

To ensure a fair comparison, all baseline models are re-evaluated under a unified experimental setting following recent legal retrieval studies (e.g., LEAD), where models are trained and evaluated under the same retrieval formulation and data preprocessing pipeline. This setting reduces discrepancies caused by differences in dataset construction, query formulation, and evaluation protocols across prior work. It is worth noting that Lawformer reports results under a different configuration, including full-length case queries and five-fold cross-validation. In contrast, our setting uses compressed queries and a dense retrieval-based ranking framework. Therefore, the reported Lawformer results are reproduced under the unified protocol and are not directly comparable to the original paper.

Table 1 and Fig. 4 present a performance comparison between the proposed G-DPR model and a wide range of baseline methods on the LeCaRD dataset. Overall, G-DPR consistently achieves the best results across the majority of evaluation metrics, demonstrating its effectiveness in legal case retrieval. In particular, it outperforms all baselines in terms of MAP and P@k (k = 5, 10, 20), indicating a clear advantage in both overall retrieval accuracy and precision at different cutoff levels. Moreover, G-DPR attains the highest scores on NDCG@5 and NDCG@10, further confirming its superior capability in ranking highly relevant cases at the top of the retrieved list.

A closer examination reveals that the performance gains are especially pronounced in top-ranked metrics, which are critical for practical legal retrieval scenarios. Compared to strong baselines such as SAILER and CaseEncoder, the proposed method demonstrates more accurate early-stage ranking, suggesting that the integration of graph-based structural reasoning with dense semantic retrieval enables more effective identification of key legal relationships. Specifically, while existing methods primarily rely on semantic similarity, they often struggle to distinguish between cases with overlapping lexical patterns but differing legal implications. In contrast, G-DPR incorporates structured relational signals, allowing it to capture deeper dependencies

https://doi.org/10.48130/ker-0026-0018

Table 1. Performance comparison of G-DPR and baseline models on evaluation metrics.

Model	MAP	P@*k*			NDCG@*k*			
		k = 5	*k* = 10	*k* = 20	*k* = 5	*k* = 10	*k* = 20	*k* = 30
Lawformer	46.7	40.2	37.7	34.9	70.2	73.6	79.7	88.3
SAILER	57.7	49.5	44.3	38.8	80.9	84.7	**88.9**	**93.7**
CaseEncoder	57.2	49.2	45.8	39.1	79.8	83.5	87.5	92.9
EventBERT	45.45	40.9	38.6	35.7	75.7	77.7	82.5	86.1
FTGS-SCR	48.12	41.8	40.0	37.3	78.4	80.5	83.7	86.7
GTE-Qwen1.5-7B-instruct	51.3	45.6	40.7	37.5	74.3	77.9	83.0	90.4
LEAD (0%)	51.7	45.0	42.0	38.6	74.1	77.8	82.8	90.1
G-DPR (ours)	**58.31**	**50.09**	**46.54**	**40.51**	**83.34**	**85.04**	88.52	93.35

Bold values represent the best experimental results under each evaluation metric.

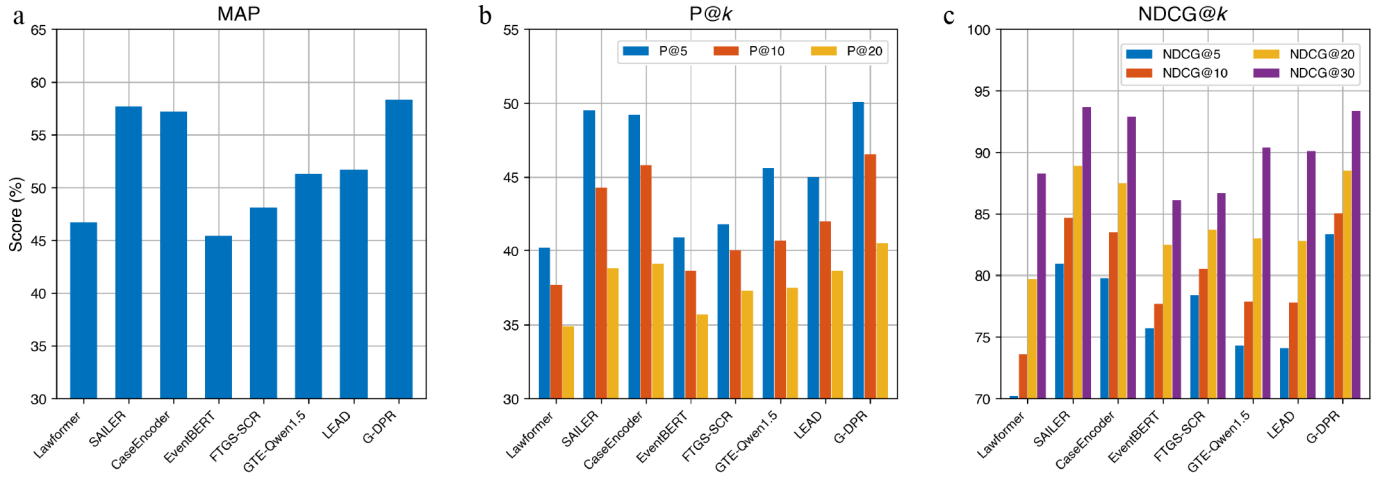


Fig. 4 Performance comparison of G-DPR and baseline models on evaluation metrics: (a) MAP, (b) P@*k*, and (c) NDCG@*k*.

among legal elements and thereby improving fine-grained ranking precision.

From a methodological perspective, the observed improvements can be attributed to the complementary strengths of the two core components. The DPR module provides robust semantic representations that ensure high recall of relevant candidates, while the graph module introduces explicit relational reasoning that enhances discriminative capacity during ranking. This joint modeling strategy effectively mitigates the limitations of single-view retrieval approaches, leading to more balanced performance across different evaluation dimensions.

On the other hand, G-DPR achieves slightly lower performance on NDCG@20 and NDCG@30, where it ranks second to SAILER. This phenomenon suggests that as the retrieval depth increases, the advantage of fine-grained structural reasoning becomes less dominant, while the influence of global ranking stability becomes more pronounced. In larger candidate sets, semantically similar but less relevant cases are more likely to be included, which may dilute the effectiveness of structural signals in lower-ranked positions. In contrast, SAILER exhibits relatively stronger robustness in maintaining ranking consistency across deeper retrieval levels, likely due to its optimization strategy focusing on global semantic alignment.

Overall, the results indicate that G-DPR is particularly effective in identifying and prioritizing highly relevant cases, achieving superior performance in top-ranked retrieval. This characteristic is of great practical significance in legal applications, where users typically focus on a limited number of the most relevant precedents. The findings further validate the importance of jointly leveraging semantic representations and structured relational knowledge for advancing legal case retrieval systems.

4.3 Ablation study

To rigorously quantify the contribution of each component in the proposed framework, we perform a comprehensive ablation study by systematically removing the graph structure module and the DPR-based dense retrieval module separately. Accordingly, two ablated variants are constructed:

4.3.1 w/o Graph:

Disables the graph reasoning channel and relies solely on semantic representations learned by DPR.

4.3.2 w/o DPR:

Removes the dense retrieval mechanism while retaining the graph-based structural reasoning component.

The experimental results on the LeCaRD dataset are summarized in Table 2, where multiple evaluation metrics, including MAP, P@*k*, and NDCG@*k*, are reported for a holistic assessment of retrieval effectiveness. As observed, the full G-DPR model consistently achieves the best performance across all metrics, demonstrating the superiority of jointly modeling semantic relevance and structural dependencies.

A detailed comparison reveals that removing the graph structure module (w/o Graph) leads to a systematic degradation in performance, with noticeable declines in MAP (−2.29), P@5 (−2.99), and NDCG@5 (−4.05). These reductions are particularly pronounced in top-ranked metrics, highlighting the critical role of graph-based structural reasoning in refining early-stage ranking quality. This finding suggests that purely semantic representations, while effective for coarse-grained matching, remain insufficient for capturing nuanced legal relationships, especially when cases exhibit high lexical similarity but differ in underlying factual or legal structures. By explicitly

Table 2. Ablation study of G-DPR on the LeCaRD dataset.

Model	MAP	P@5	P@10	P@20	NDCG @5	NDCG @10	NDCG @20	NDCG @30
G-DPR	**58.31**	**50.09**	**46.54**	**40.51**	**83.34**	**85.04**	**88.52**	**93.35**
w/o Graph	56.02	47.10	45.42	40.42	79.29	82.37	86.58	91.88
w/o DPR	55.35	49.35	44.30	38.04	78.87	80.17	81.91	85.86

Bold values represent the best experimental results under each evaluation metric.

modeling inter-case relational dependencies, the graph module introduces complementary structural signals that enhance the model's discriminative capacity, thereby improving fine-grained ranking precision.

Conversely, the removal of the DPR module (w/o DPR) results in a more substantial degradation in global ranking quality, as evidenced by the decreases in MAP (−2.96), P@10 (−2.24), P@20 (−2.47), and NDCG@20 (−6.61). This trend highlights the indispensable role of dense semantic retrieval in ensuring robust candidate recall and maintaining overall ranking consistency. Without DPR, the model relies solely on structural signals, which, although informative, lack sufficient coverage to effectively retrieve semantically relevant candidates in large-scale corpora.

The two modules improve retrieval performance from different dimensions, and their combination delivers more balanced and comprehensive results. This demonstrates that graph structural signals and dense semantic embeddings play complementary roles rather than functioning as independent additive factors: the DPR module mainly supports high-recall semantic matching, while the graph structure module strengthens fine-grained discrimination via structured reasoning. Their synergistic integration allows the G-DPR model to achieve superior retrieval performance, particularly in complex legal scenarios with subtle semantic variations and intricate relational structures. This complementarity emphasizes the need to jointly utilize unstructured textual semantics and structured relational knowledge for advanced legal case retrieval.

5 Conclusions

The proposed approach leverages the language understanding and abstract reasoning capabilities of fine-tuned legal large language models (LLMs) to construct a multi-relational heterogeneous graph that integrates case facts and statutory semantics. A relational graph convolutional network (RGCN) is employed to conduct deep structural representation learning and inter-graph similarity modeling, enabling fine-grained structural alignment between legal cases. On the semantic representation front, RoBERTa and Longformer undergo domain-specific pretraining to capture the unique linguistic and contextual characteristics of legal texts. Furthermore, a dual-encoder framework based on LegalBERT, fine-tuned on the LEAD dataset, is integrated to significantly enhance the precision and robustness of semantic similarity estimation between legal cases. A fusion-based reranking mechanism is subsequently introduced to jointly model semantic similarity and structural similarity, thereby enabling more robust and interpretable candidate ranking. Experimental results on the LeCaRD legal case retrieval dataset demonstrate that the G-DPR model significantly outperforms existing baseline methods across multiple evaluation metrics. These findings validate the effectiveness of integrating structural modeling with semantic encoding. Additionally, the explicit comparison paths provided by the graph-based channel enhance the interpretability of the retrieval process, thereby improving model transparency and fostering greater user trust.

Future work will focus on three key directions: first, we will systematically verify the cross-dataset generalization performance of G-DPR on more legal case retrieval benchmarks and further optimize the adaptability of the framework to different data schemas. The core retrieval and ranking modules of G-DPR follow a general modular design with inherent reusability across datasets; cross-dataset deployment mainly requires engineering adaptation at the knowledge graph schema layer, such as entity-relation alignment and data field mapping. This direction will focus on reducing adaptation costs and improving the framework's portability in practical scenarios. Second, deeper integration of event-level causal chains will be explored to establish a more advanced legal semantic reasoning mechanism, thereby enhancing the model's ability to comprehend and assess complex case narratives. Third, efforts will be devoted to further extending the capabilities of the graph-based retrieval module in G-DPR. This involves leveraging domain-adapted LLMs to more effectively incorporate and utilize external legal knowledge. By integrating open legal case repositories with structured legal knowledge resources, the objective is to develop an intelligent case retrieval assistant that can provide practical support in real-world judicial scenarios.

Author contributions

The authors confirm their contributions to the paper as follows: construction of theoretical methods: Dong C; paper writing: Dong C, Yang Y; experimental coding, statistical analysis: Yang Y; model training, experimental analysis, and paper drafting: Li M. All authors reviewed the results and approved the final version of the manuscript.

Data availability

The data used in this study are from the publicly available LEAD dataset and the LeCaRD benchmark dataset. All datasets are publicly accessible via their official channels, and additional relevant data or code can be obtained from the corresponding author upon reasonable request.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant Nos 62572446, 62176240), the Beijing Education Science 14th Five-Year Plan Project (Grant No. CDDB25231), the Fundamental Research Funds for the Central Universities (Grant No. CUC26XZ02), and the China Scholarship Council (Grant No. 202507050074).

Conflict of interest

The authors declare that they have no conflict of interest.

Dates

Received 3 April 2026; Revised 2 September 2026; Accepted 16 September 2026; Published online 14 October 2026

References

[1] Devlin J, Chang MW, Lee K, Toutanova K. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, eds. Burstein J, Doran C, Solorio T. Minneapolis, Minnesota: Association for Computational Linguistics. pp. 4171−4186 doi: 10.18653/v1/N19-1423

[2] Liu Y, Ott M, Goyal N, Du J, Joshi M, et al. 2019. RoBERTa: a robustly optimized BERT pretraining approach. *arXiv* Preprint:1907.11692

[3] Chalkidis I, Fergadiotis M, Malakasiotis P, Aletras N, Androutsopoulos I. 2020. LEGAL-BERT: the muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, eds. Cohn T, He Y, Liu Y. Minneapolis, Minnesota: Association for Computational Linguistics. pp. 2898−2904 doi: 10.18653/v1/2020.findings-emnlp.261

[4] Gao C, Xiao C, Liu Z, Chen H, Liu Z, et al. 2024. Enhancing legal case retrieval *via* scaling high-quality synthetic query-candidate pairs. *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, eds. Al-Onaizan Y, Bansal M, Chen YN. Miami, Florida, USA: Association for Computational Linguistics. pp. 7086−7100 doi: 10.18653/v1/2024.emnlp-main.402

[5] Karpukhin V, Oguz B, Min S, Lewis P, Wu L, et al. 2020. Dense passage retrieval for open-domain question answering. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, eds. Webber B, Cohn T, He Y, Liu Y. Online: Association for Computational Linguistics. pp. 6769−6781 doi: 10.18653/v1/2020.emnlp-main.550

[6] Cui J, Ning M, Li Z, Chen B, Yan Y, et al. 2023. Chatlaw: a multi-agent legal assistant based on a role-aligned mixture-of-experts architecture. *arXiv* Preprint:2306.16092

[7] Cui J, Ning M, Li Z, Li H, Ya Y, et al. 2026. Chatlaw: a multi-agent legal assistant based on a role-aligned mixture-of-experts architecture. *Fundamental Research* 2026.03.026

[8] Guha N, Nyarko J, Ho D, Ré C, Chilton A, et al. 2023. LegalBench: a collaboratively built benchmark for measuring legal reasoning in large language models. In *Advances in Neural Information Processing Systems 36*. USA: Neural Information Processing Systems Foundation, Inc. (NeurIPS). pp. 44123−44279 doi: 10.52202/075280-1915

[9] Yao S, Ke Q, Wang Q, Li K, Hu J. 2024. Lawyer GPT: a legal large language model with enhanced domain knowledge and reasoning capabilities. *Proceedings of the 2024 3rd International Symposium on Robotics, Artificial Intelligence and Information Engineering, Singapore, 5−7 July, 2024*. New York, NY, USA: Association for Computing Machinery. pp. 108−112 doi: 10.1145/3689299.3689319

[10] Vuong YTH, Bui QM, Nguyen HT, Nguyen TTT, Tran V, et al. 2023. SM-BERT-CR: a deep learning approach for case law retrieval with supporting model. *Artificial Intelligence and Law* 31:601−628

[11] Nguyen HT, Phi MK, Ngo XB, Tran V, Nguyen LM, et al. 2022. Attentive deep neural networks for legal document retrieval. *Artificial Intelligence and Law* 32:57−86

[12] Beltagy I, Peters ME, Cohan A. 2020. Longformer: the long-document transformer. *arXiv* Preprint:2004.05150

[13] Ma Y, Wu Y, Su W, Ai Q, Liu Y. 2023. CaseEncoder: a knowledge-enhanced pre-trained model for legal case encoding. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, eds. Bouamor H, Pino J, Bali K. Singapore: Association for Computational Linguistics. pp. 7134−7143 doi: 10.18653/v1/2023.emnlp-main.441

[14] Bhattacharya P, Ghosh K, Pal A, Ghosh S. 2022. Legal case document similarity: you need both network and text. *Information Processing & Management* 59:103069

[15] Tang Y, Qiu R, Liu Y, Li X, Huang Z. 2024. CaseGNN: graph neural networks for legal case retrieval with text-attributed graphs. *Advances in Information Retrieval*, eds. Goharian N, Tonellotto N, He Y, Lipani A, McDonald G, et al. Cham: Springer. pp. 80−95 doi: 10.1007/978-3-031-56060-6_6

[16] Tang Y, Qiu R, Yin H, Li X, Huang Z. 2024. CaseLink: inductive graph learning for legal case retrieval. *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington DC, USA 14−18 July, 2024*. New York, NY, USA: Association for Computing Machinery. pp. 2199−2209 doi: 10.1145/3626772.3657693

[17] Ma Y, Wu Y, Ai Q, Liu Y, Shao Y, et al. 2023. Incorporating structural information into legal case retrieval. *ACM Transactions on Information Systems* 42:1−28

[18] Li H, Ai Q, Chen J, Dong Q, Wu Y, et al. 2023. SAILER: structure-aware pre-trained language model for legal case retrieval. *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, Taipei, Taiwan, China, 23−27 July, 2023*. New York, NY, USA: Association for Computing Machinery. pp. 1035−1044 doi: 10.1145/3539618.3591761

[19] Bi S, Ali Z, Wang M, Wu T, Qi G. 2022. Learning heterogeneous graph embedding for Chinese legal document similarity. *Knowledge-Based Systems* 250:109046

[20] Zhang Y, Zhai S, Meng Y, Bi S, Chen Y, et al. 2024. Event is more valuable than you think: improving the Similar Legal Case Retrieval *via* event knowledge. *Information Processing & Management* 61:103729

[21] Sun Z, Xu J, Zhang X, Dong Z, Wen JR. 2023. Law article-enhanced legal case matching: a causal learning approach. *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, Taipei, Taiwan, China, 23−27 July, 2023*. New York, NY, USA: Association for Computing Machinery. pp. 1549−1558 doi: 10.1145/3539618.3591709

[22] Zhang K, Chen C, Wang Y, Tian Q, Bai L. 2023. CFGL-LCR: a counterfactual graph learning framework for legal case retrieval. *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Long Beach CA, USA, 6−10 August, 2023*. New York, NY, USA: Association for Computing Machinery. pp. 3332−3341 doi: 10.1145/3580305.3599273

[23] Wang Z. 2022. Legal element-oriented modeling with multi-view contrastive learning for legal case retrieval. *2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, 18−23 July, 2022*. USA: IEEE. pp. 1−10 doi: 10.1109/IJCNN55064.2022.9892487

[24] Sarmah B, Mehta D, Hall B, Rao R, Patel S, et al. 2024. HybridRAG: integrating knowledge graphs and vector retrieval augmented generation for efficient information extraction. *Proceedings of the 5th ACM International Conference on AI in Finance, Brooklyn NY, USA, 14−17 November, 2024*. New York, NY, USA: Association for Computing Machinery. pp. 608−616 doi: 10.1145/3677052.3698671

[25] Kalra R, Wu Z, Gulley A, Hilliard A, Guan X, et al. 2024. HyPA-RAG: a hybrid parameter adaptive retrieval-augmented generation system for AI legal and policy applications. *Proceedings of the 1st Workshop on Customizable NLP: Progress and Challenges in Customizing NLP for a Domain, Application, Group, or Individual (CustomNLP4U)*, eds. Kumar S, Balachandran V, Park CY, Shi W, Hayati SA, et al. Miami, Florida, USA: Association for Computational Linguistics. pp. 237−256 doi: 10.18653/v1/2024.customnlp4u-1.18

[26] Wiratunga N, Abeyratne R, Jayawardena L, Martin K, Massie S, et al. 2024. CBR-RAG: Case-Based Reasoning for Retrieval Augmented Generation in LLMs for Legal Question Answering. In *Case-Based Reasoning Research and Development*, eds. Recio-Garcia JA, Orozco-del-Castillo MG, Bridge D. Cham: Springer Nature Switzerland. pp. 445−460 doi: 10.1007/978-3-031-63646-2_29

[27] Schlichtkrull M, Kipf TN, Bloem P, van den Berg R, Titov I, et al. 2018. Modeling relational data with graph convolutional networks. In *The Semantic Web*, eds. Gangemi A, Navigli R, Vidal ME, Hitzler P, Troncy R, et al. Cham: Springer. pp. 593−607 doi: 10.1007/978-3-319-93417-4_38

[28] Ma Y, Shao Y, Wu Y, Liu Y, Zhang R, et al. 2021. LeCaRD: a legal case retrieval dataset for Chinese law system. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, 11−15 July, 2021*. New York, NY, USA: Association for Computing Machinery. pp. 2342−2348 doi: 10.1145/3404835.3463250

Copyright: © 2026 by the author(s). Published by Maximum Academic Press, Fayetteville, GA. This article is an open access article distributed under Creative Commons Attribution License (CC BY 4.0), visit https://creativecommons.org/licenses/by/4.0/.