*Journal of Network Intelligence* ©2026 ISSN 2414-8105
**Taiwan Ubiquitous Information** Volume **11**, Number **1**, February **2026**

# Intelligent Retrieval and Storage Performance Improvement of Legal Documents Based on GraphRAG and SVM

Yi-Fei Chen[1]

[1]Law School,
Anhui University of Finance & Economics, Bengbu 233030, P. R. China
yifeiaufe@126.com

Zheng-Kun Yan[2]

[2]School of Languages and Media,
Anhui University of Finance & Economics, Bengbu 233030, P. R. China
yanaufe23@126.com

Minghsun Yang[3,*]

[3]Tan Siu Lin Business School,
Quanzhou Normal University, Quanzhou 362000, P. R. China
solicitoryang@gmail.com

Jun-Hua Zhou[4]

[4]Solutionswon group, Melbourne VIC3000, Australia
Timz900719@gmail.com

[*]Corresponding author: Minghsun Yang

Received March 05, 2025, revised June 20, 2025, accepted October 03, 2025.

Abstract. *Legal document retrieval, one of the main areas of legal research in the current information society, confronts ever difficult problems. Mostly depending on keyword matching, traditional document retrieval techniques cannot adequately handle the semantic relationships and intricate structures in legal documents. Researchers have progressively tried to increase the accuracy and efficiency of legal document retrieval using more sophisticated algorithms. Still, many models are challenging to adequately depict the intricate semantic relationships between documents and the logical connections between documents. This work presents an intelligent retrieval model for legal documents based on Graph Retrieval-Augmented Generation (GraphRAG) and Support Vector Machine (SVM) LegalSVM-RAG, which makes full use of Graph Neural Networks (GNN) to retrieve documents from each other by combining the advantages of graph structural information and generative models, so solving this problem. The model makes full use of graph neural network (GNN) to model the relationship between documents and SVM to effectively classify documents, so combining the benefits of graph structural information and generative model. GraphRAG not only manages the similarity and citation relationship between documents but also optimizes the document representation through the generative process. In this work, notably in cross-category retrieval and complicated document feature representation, we experimentally validate the efficiency of the proposed model in legal document retrieval. Strong practical value and a fresh concept and approach for the process of legal document retrieval are offered by the model.*

**Keywords:** GraphRAG; SVM; GNN; cross-category document retrieval

## 1. Introduction.

The application of intelligent information retrieval technology in many spheres has attracted great attention given the fast development of information technology, notably the continual progress of natural language processing (NLP) and machine learning technology [1, 2]. As an information-intensive sector, the legal sector has a lot of legal documents and case data including background information, case law analyses and court logic in addition to complicated legal phrases and conventions. Effective management, storage, and retrieval of these legal papers have grown to be pressing issues [3]. Manual classification and keyword matching define traditional legal retrieval techniques; nevertheless, with the growing complexity of case types and legal texts, manual retrieval is ineffective and error-prone and fails to satisfy the demand of the current legal sector for quick and accurate retrieval.

### 1.1. Related work.

With the fast expansion of information technology in the past decades, particularly the ongoing development of Natural Language Processing (NLP) and machine learning, intelligent document retrieval techniques have been extensively applied in many disciplines [4]. Retrieving legal documents is a difficulty for one of the information-intensive disciplines since the intricacy of many legal texts, cases, and legal terminologies is involved. Particularly in legal terminology, implicit semantics, and complex syntactic structures, traditional legal document retrieval techniques mostly rely on keyword matching and Boolean retrieval, which match the keywords appearing in the user query with the corresponding keywords in the document by finding the keywords, which are simple to operate but lack of understanding of the semantics of the document resulting in retrieval results that are often inaccurate and unsatisfactory.

More and more studies have begun to present cutting-edge methods to enhance the retrieval of legal documents as machine learning and deep learning technologies continue to evolve [5]. Based on Support Vector Machine (SVM), one of the traditional strategies maximizes the inter-category spacing and identifies the ideal hyperplane for effective document categorization and retrieval. SVM cannot adequately manage the complicated links between documents, nevertheless, and its effectiveness depends more on hand feature engineering [6]. In this regard, deep learning-based neural network models have been progressively applied. Legal documents are automatically extracted high-level feature representations using deep learning techniques including convolutional neural networks (CNNs) and recurrent neural networks (RNNs). CNNs have been extensively applied for text classification chores since they are good in spotting local elements in short texts [7]. Conversely, RNNs-especially their variation Long Short-Term Memory Network (LSTM)-are able to effectively handle sequential dependencies in long texts, therefore enabling contextual semantics and sophisticated structure capture in legal documents. These deep learning techniques not only automatically extract features but also understand the intricate relationships and implicit semantics in the text, so considerably enhancing the accuracy of legal document retrieval with end-to-end training.

Graph-based retrieval techniques have also progressively taken front stage in legal document retrieval at the same time [8, 9]. Legal documents contain not only independent language but also often related information via references, case studies, or like rulings. This relationship commonly exhibits the features of graph structure, in which the document as a node of the graph, the relationship between the nodes through the edge connection. The graph retrieval approach is especially appropriate for the legal sector since the graph structure may adequately depict the similarity of documents, citation links, etc., which reflects their relevance [10]. Scholars have started to model the relationships between legal texts using Graph Neural Network (GNN), therefore enhancing the semantic comprehension of the retrieval system. Graph Retrieval-Augmented Generation (GraphRAG) is a unique method combining the generative model with the graph retrieval.

Deep learning in the field of natural language processing undergoes a revolution when bidirectional encoder representations from Transformers (BERT) and other pre-trained language models appear [11, 12]. In many NLP applications, bidirectional encoding has produced notable success and helps to capture the bidirectional interdependence of contexts in sentences. While BERT greatly enhances the effectiveness of text retrieval by deep contextual comprehension, legal documents typically contain multi-level structural information and complicated contextual linkages that cannot be sufficiently reflected by conventional unidirectional models [13]. More and more studies have begun to employ BERT and its derivatives (e.g., RoBERTa, DistilBERT) to pre-training and fine-tune legal texts to cope with complicated language structures and particular legal terminology in the legal realm. These techniques have grown to be crucial tools for contemporary legal document retrieval since they offer great support for jobs including document categorization, case prediction, and legal question answering.

Though current methods have made great advancement possible, intelligent retrieval of legal documents remains a difficult task. Thus, by combining the benefits of GraphRAG and SVM, this work presents a new model for intelligent retrieval of legal documents LegalSVM-RAG with a goal to offer an efficient solution to cross-category document retrieval and legal document feature representation.

### 1.2. Contribution.

In this work, we offer an intelligent retrieval model LegalSVM-RAG for legal documents based on GraphRAG and SVM, which creatively solves the constraints found in conventional legal document retrieval approaches.

First, most legal document retrieval systems rely on SVM-based categorization or keyword matching, which could make it challenging to completely investigate document interactions. Therefore, the GraphRAG model presented in this paper can include rich relational information into document representation by combining graph retrieval and generative modeling.

Second, this work maximizes the cooperative effect of SVM's high classification capability by ingeniously combining graph structure enhancement learning with GraphRAG uses graph structure modeling to compile implicit semantic information from papers; SVM uses the best hyperplane to maximize retrieval of papers.

Both in feature space optimization and data preparation, this work is novel. These feature extraction and data cleaning techniques for legal document attributes guarantee that the model can perform well in a high-noise, low-labelled data environment, therefore enabling it to better fit the complicated legal environment.

This work presents a novel technique that greatly enhances the performance of legal document retrieval and offers fresh concepts and approaches for cross-category document retrieval and legal document feature representation by aggregating GraphRAG with SVM.

## 2. Theoretical analysis.

### 2.1. Graph Retrieval-Augmented Generation.

GraphRAG advances information retrieval accuracy and generating quality by means of graph structures and generative models. While they fail to capture deep semantic linkages in complex documents, especially legal documents, which have many references, clauses, and contextual information, traditional information retrieval models generally use text-based similarity metrics like TF-IDF or BM25 to retrieve simple text. GraphRAG models document content and relationships using graph structure, therefore enhancing document retrieval and generation by means of GNN and generative models. GraphRAG presents the document collecting as a graph structure; hence, let $D$ be the document collection and then $D$ may be shown as:

$$D = \{d_1, d_2, \ldots, d_n\} \tag{1}$$

Every document $d_i$ has numerous components. GraphRAG considers the elements of a document (e.g., clauses, paragraphs, keywords, etc.) as nodes in a graph, therefore generating a graph structure $G$ that may be stated as capturing the semantic relationships inside a document:

$$G = (V, E) \tag{2}$$

where $V$ also finds expression as:

$$V = \{v_1, v_2, \ldots, v_n\} \tag{3}$$

where $V$ is the collection of graph nodes; $E$ is the set of edges between these nodes, therefore indicating the logical or semantic connections between the elements of the document. By means of this graph structure representation, the possible links between several sections of a document can be efficiently represented, so facilitating the semantic content of every section to be more precisely fused [14].

GraphRAG learns the node representation within the GNN architecture by iterative message passing mechanism [15]. At the $t^{th}$ layer of GNN, the update of the node can be expressed assuming that the initial feature of node $v_i$ is $h_i^0$ as follows:

$$h_i^{(t+1)} = \sigma\left(\sum_{v_j \in N(v_i)} \frac{1}{N(v_i)} W^{(t)} h_j + b^{(t)}\right) \tag{4}$$

where $W^{(t)}$ and $b^{(t)}$ are the weight matrix and bias term of the $t^{th}$ layer of the GNN correspondingly; $\sigma(\cdot)$ is the activation function and $N(v_i)$ is the collection of nearby nodes of node $v_i$. Messaging at each level combines data from the surrounding nodes of every node, therefore allowing the representation of the node to contain data from several areas [16]. This layer-by-layer message transfer mechanism lets the node representation efficiently catch the more general semantic connections in the document.

GraphRAG retrieves by computing the similarity between query $q$ and document node $h_i$ in the retrieval phase. Usually in order to get a query feature vector, the query $q$ is encoded using some sort of encoding, such TF-IDF or embedded representation of deep learning models. Conversely, the document node $h_i$ is a GNN processing node embedding representation. Usually, cosine similarity with the formula helps one to find the similarity between a query and a document node.

$$\text{sim}(q, h_i) = \frac{q \cdot h_i}{\|q\| \|h_i\|} \tag{5}$$

The Euclidean paradigm is represented by $\| \cdot \|$; the dot product operation by $\cdot$. By computing the angular difference between a query and a document node, cosine similarity gauges the query's relevance to the node feature vectors. The search is more pertinent to the document node the more similar the two are. GraphRAG therefore allows the most pertinent portions from documents depending on the query to be accessed [17].

GraphRAG unlike conventional retrieval techniques not only depends on the relevance measure of the retrieval but also presents a generative model to raise the quality of the outputs. After obtaining the pertinent papers and aggregating the query data, the generative model's job is to provide the last response or summary. In the generative model, the query $q$ and the set of obtained pertinent documents $\{d_{i_1}, d_{i_2}, \ldots, d_{i_k}\}$ feed into each other to generate the last result $y$. One can convey the procedure by means of the following equation:

$$y = \mathrm{Gen}(q, \{d_{i_1}, d_{i_2}, \ldots, d_{i_k}\}) \tag{6}$$

GraphRAG also presents a retrieval enhancing strategy to help even more. GraphRAG's objective in the retrieval phase is to maximise the correlation between the query and the documents therefore filtering out the most pertinent sections of the documents from the query. GraphRAG uses the retrieval loss function $L_{\mathrm{ret}}$ with an optimisation aim to reach this target:

$$L_{\mathrm{ret}} = -\sum_{i=1}^{k} \log P_{\mathrm{ret}}(d_i|q) \tag{7}$$

where $P_{\mathrm{ret}}(d_i|q)$ denotes the likelihood of obtaining document $d_i$ given query $q$. GraphRAG can choose the most pertinent documents by maximising this likelihood, therefore optimising the retrieval process.

GraphRAG also has a generating objective function $L_{\mathrm{gen}}$ in the generation phase, whose optimisation aim is to optimize the likelihood of producing an answer:

$$L_{\mathrm{gen}} = -\sum_{i=1}^{k} \log P_{\mathrm{gen}}(y|q, \{d_{i_1}, d_{i_2}, \ldots, d_{i_k}\}) \tag{8}$$

Building a loss function aims to produce an answer $y$ more precisely reflecting the information in the query $q$ and the related documents. GraphRAG maximizes the produced answers to be more relevant and correct by maximizing the generating probability.

GraphRAG's advantage resides in its efficient use of graph structure, which can adequately depict the intricate semantic links among the components of a document, particularly for the purpose of handling challenging textual material. GraphRAG not only increases the quality of the responses by the generative model, which finally accomplishes effective and accurate information retrieval and generation, but also enhances the document representation through the multi-layer message transmission of GNN.

2.2. **Support Vector Machine.** Widely applied in classification and regression issues, SVM is a supervised learning method [18]. SVM is fundamentally based on building one or more hyperplanes to classify data in feature space. The aim is to identify a perfect hyperplane maximizing the distance between samples of every class and efficiently separating data points of various classes [19]. With high-dimensional data, SVM is very appropriate for classification issues; it has shown great performance in many practical applications, as seen by Figure 1.

The SVM aims to identify a hyperplane such that the data point $x_i$ is appropriately classified and the point furthest from the hyperplane (i.e., the support vector) has the biggest distance to the hyperplane given a training set $\{(x_i, y_i)\}_{i=1}^{n}$ whereby $x_i \in \mathbb{R}^d$ is a data point and $y_i \in \{-1, +1\}$ is the related label. The SVM maximizes the interval of this hyperplane, which serves as the decision boundary, therefore enhancing the classification effect [20].

First take into account the situation when the data is linearly separable—that is, where a hyperplane can entirely separate it. This will help to solve the problem. In this instance, the SVM's aim of optimisation is to maximise the interval. The aim of optimisation is Eq:

$$L_1 = \min_{w,b} \frac{1}{2}\|w\|^2 \tag{9}$$

The restriction is that every data point fulfills the classification criterion, meaning:

$$y_i(w \cdot x_i + b) \geq 1, \quad \forall i = 1, 2, \ldots, n \tag{10}$$

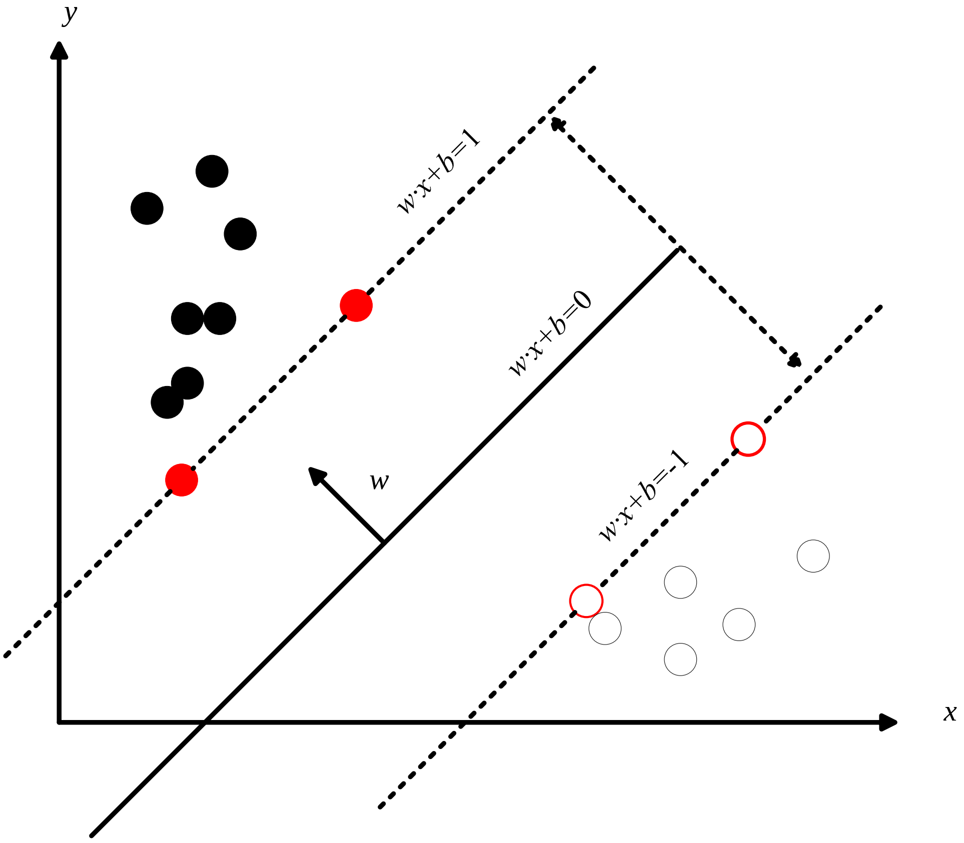

Figure 1. Structure of SVM

This goal implies that the SVM seeks a hyperplane such that the distance from the support vector to the hyperplane is maximised, hence optimising the spacing between categories [21].

Usually, the Lagrange multiplier method is applied to address optimization issues. Introducing the Lagrange multipliers $\alpha_i$, the dyadic SVM issue can be converted into:

$$L_q = \max_{\alpha} \left( \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \right) \tag{11}$$

The Limitations are:

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \quad \alpha_i \geq 0 \tag{12}$$

where $\langle x_i, x_j \rangle$ represents the inner product between the sample points $x_i$ and $x_j$. Solving this dyadic issue yields the optimal value of $\alpha_i$, which determines the hyperplane's parameters $w$ and $b$ in turn.

The decision function allows one to classify after the ideal $w$ and $b$ are found. The following decision function produces the classification result for a fresh sample point $x$:

$$f(x) = w \cdot x + b \tag{13}$$

By maximizing the interval, SVM guarantees robustness of classification, so the classification results are quite resistant to noise and minor data fluctuations [22, 23]. Often utilized in fields including text classification and picture recognition, SVM performs especially well with high-dimensional data.

Many real datasets, however, are not linearly separable, hence SVM uses the "Kernel Trick" in such cases. The "Kernel Trick's" central concept is to map data from the original space to a higher dimensional space therefore rendering the data linearly separable. SVMs build hyperplanes in high-dimensional spaces by substituting the kernel function $K(x, x')$ for the original inner product $(x, x')$, therefore permitting nonlinear classification. Commonly used kernel functions comprise radial basis and linear kernel functions.

For data maybe noisy and indivisible, SVM additionally presents the "Soft Margin" approach [24]. "Soft Margin" introduces a slack variable $\xi_i$ to let some data points lie on the incorrect side of the hyperplane, therefore improving the generalisation of the model. The goal of optimisation turns to be:

$$L = \min_{w,b,\xi_i} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i \tag{14}$$

The Limitations are:

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \tag{15}$$

where the degree of penalty for misclassification is managed using a regularisation value called $C$. One can find a compromise between classification accuracy and model complexity by changing $C$, therefore preventing either underfitting or overfitting.

Though it finds great use in many disciplines, SVM particularly shines in managing high-dimensional data [25]. SVM is extensively applied in text classification, image recognition, speech processing, and other domains due to its strong generalisation performance and powerful classification capacity. Through proper kernel function selection and optimisation techniques, SVM can achieve good performance on large-scale datasets even if its computing complexity during training is significant.

## 3. Model for intelligent retrieval and storage optimisation of legal documents.

**3.1. Model architecture.** See Figure 2 to understand the LegalSVM-RAG model, which seeks to efficiently mix GraphRAG and SVM techniques to maximize legal document retrieval and storage performance.

The complete design is split into various modules, each with a particular function and cooperating to drive the performance of the whole model. More specifically, the model comprises mostly in GraphRAG and SVM classification modules, data preprocessing, and co-optimization tools.

The procedure of the model begins with data preparation overall. The data preparation module's job is to clean, normalize, extract features from the source legal documents and translate them into a standardised feature set for use by next modules [26]. First the legal document's text is cleaned to eliminate noise including stop words and punctuation marks. To build the feature matrix of the document, text features then are obtained using TF-IDF or bag-of-words modeling [27]. Producing a feature representation $X$ of a document with dimensions $d \times n$ where $d$ is the number of features and $n$ is the number of documents.

One may define data preparation as follows:

$$X = \text{Preprocess}(D) \tag{16}$$

where $D$ is the original legal document collecting; $X$ is the cleaned and normalized feature collecting. Further processing of the feature set $X$ helps to guarantee that the model can effectively represent the semantics of the text in next processing, therefore meeting the input criteria of GraphRAG and SVM.

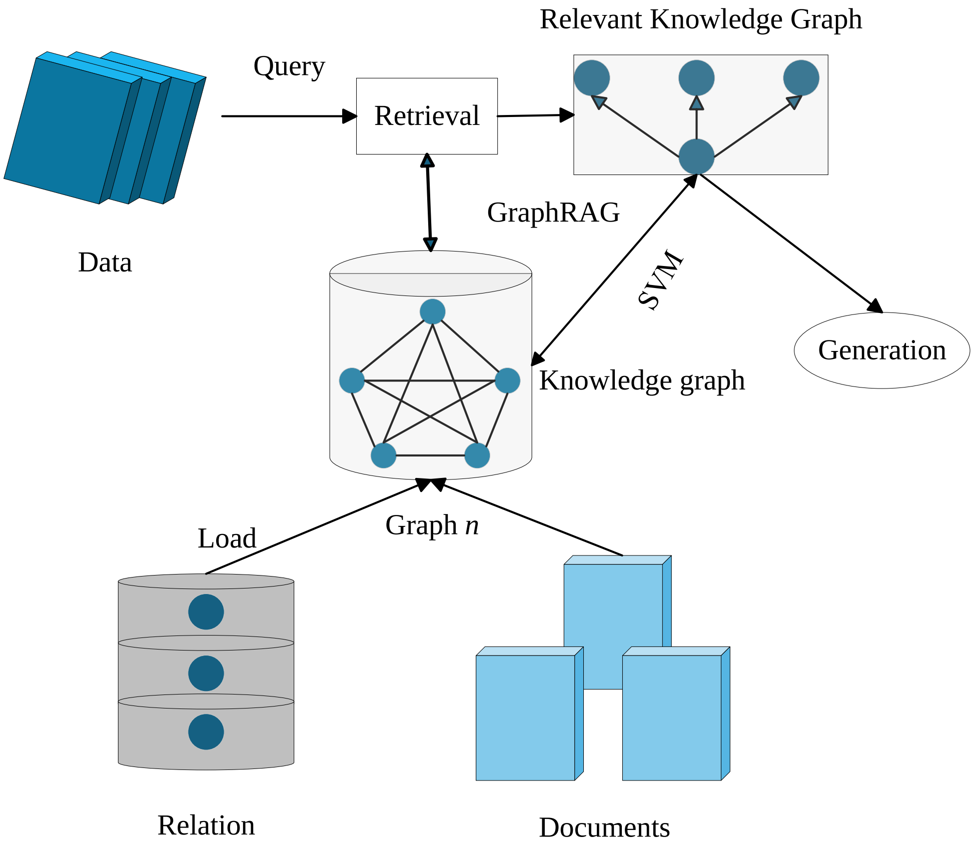

Figure 2. LegalSVM-RAG model

First built in the GraphRAG module depending on the entities and their relationships taken from the document, is the graph structure. Every entity is shown as a node; relationships between entities are shown as edges. The node representation is iteratively updated and information is spread among graph nodes using GNN [28]. At each layer, the non-linearly activated node representations are weighted and averaged; so, at last the representation of every node includes the contextual information of their neighbourhood. One may characterize the process of creating the graph structure by means of the following equation:

$$G = \text{Graph}(X, R) \tag{17}$$

where $G$ is the built graph structure; $X$ is the collection of entities taken from the document; $R$ is the set of links among entities. The graph neural network modulates the node representation by means of multi-layer information transmission as follows:

$$h_i^{(k+1)} = \sigma\left(W_k \sum_{j \in N(i)} h_j^{(k)} + B_k\right) \tag{18}$$

where $W_k$ and $B_k$ are the weight matrix and bias term at the $k^{th}$ layer; $h_i^{(k)}$ represents the $i^{th}$ node; $N(i)$ is the set of surrounding nodes of node $i$; $\sigma$ is the activation function. The node representation gradually fuses the neighbourhood information by means of multi-layer information propagation, so improving the semantic comprehension of the nodes. Following the graph structure augmentation, the GraphRAG module offers the graph structure representation for the SVM classification module as its input features. SVM's decision-making is as follows:

$$f(x) = \text{sign}\left(\sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b\right) \tag{19}$$

where $\text{sign}(\cdot)$ is the sign function; $\alpha_i$ is the Lagrange multiplier; $y_i$ is the label; $K(x_i, x)$ is the kernel function; $b$ is the bias term and $f(x)$ is the classification result of the input sample $x$. This decision capability of the SVM helps to sort the input samples into several groups.

SVM uses the kernel trick to translate data to a higher dimensional space, hence addressing nonlinearly differentiable problems [29]. The kernel function forms:

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \tag{20}$$

where $K(x_i, x_j)$ is the inner product of the samples $x_i$ and $x_j$ in the high-dimensional space and $\phi(x)$ is the mapping function.

The LegalSVM-RAG model is based on the cooperative effort of GraphRAG and SVM, whereby the two modules work together to jointly maximize the performance of the model. GraphRAG creates graph structure representations that SVM uses as input features; they are then classified via hyperplane optimization. The two modules are collaboratively optimised to raise the general performance. GraphRAG and SVM's parameters are tweaked to one another throughout the combined optimisation process so that the two modules cooperate to generate best results in the classification problem. The joint process loss function in:

$$L = \min_{\theta_1, \theta_2} L_{\text{GraphRAG}}(\theta_1) + L_{\text{SVM}}(\theta_2) \tag{21}$$

where $L_{\text{GraphRAG}}$ and $L_{\text{SVM}}$ are loss functions of GraphRAG and SVM respectively; $\theta_1$ and $\theta_2$ are parameters of GraphRAG and SVM respectively.

Moreover, the graph created by GraphRAG interacts with SVM's decision function by means of the following equation, therefore improving the collaboration process:

$$f(x) = \text{sign}\left(\sum_{i=1}^{n} \alpha_i y_i \langle \text{GraphRAG}(x_i), \text{GraphRAG}(x) \rangle + b\right) \tag{22}$$

where $\text{GraphRAG}(x)$ represents the increased characteristics of the GraphRAG module-generated graph structure. This formula reveals that the decision function of SVM depends not only on the original input features but also on the created graph features by GraphRAG, hence attaining a synergy between the two.

At last, by means of joint optimisation, the entire model constantly modifies GraphRAG and SVM's parameters to minimise the total loss function, so enhancing the retrieval efficiency and classification accuracy of legal documents.

3.2. **Model evaluation.** This work evaluates the Legal SVM-RAG model using weighted F1 value, genuine similarity, and document rearrangement accuracy. Legal applications would find these measures perfect since they can faithfully represent the classification impact, semantic comprehension capacity, and retrieval accuracy of the model when processing legal texts.

We evaluate category imbalance with weighted F1 value. Legal documents seem to show some less frequent categories, hence the F1 number could not fairly represent their performance. By weighting the F1 value for every category, the weighted F1 value considers variances in category sample size [30, 31]. Weighted F1's formula is:

$$\text{Weighted F1-Score} = \sum_{i=1}^{C} \frac{n_i}{N} \cdot F1_i \tag{23}$$

where $C$ is the number of categories; $n_i$ is the sample size of category $i$; $N$ is the total number of samples; $F1_i$ is the F1 value of category $i$. This weighting method guarantees that the predictive ability of the model on rare categories is not neglected, particularly considering the unequal distribution of categories such as legal documents, therefore enabling better analysis of the model's impact.

The legal document semantics of the model is gauged by actual similarity. Semantic similarity between model-predicted document pairs and actual labels reveals the model's reading ability. Its calculating method:

$$\text{True Similarity} = \frac{\sum_{i=1}^{N} \text{sim}(y_i, \hat{y}_i)}{N} \tag{24}$$

where $\text{sim}(y_i, \hat{y}_i)$ represents the cosine similarity or other text similarity metric between the expected category $\hat{y}_i$ and the actual category $y_i$.

A major retrieval model performance statistic is document reclassification accuracy. The legal document retrieval and storage model has to classify and efficiently arrange acquired documents to arrange the most pertinent first. Mathematical formula for computation:

$$\text{Re-ranking Precision} = \frac{|\{\hat{y}_i \mid y_i \in \text{Relevant Documents}\}|}{\text{Total Retrieved Documents}} \tag{25}$$

After model reordering, $\hat{y}_i$ stands for documents; $y_i \in$ Relevant Documents stands for query-related documents; and Total Retrieved Documents shows the overall count of obtained papers. This indicator can evaluate the relevance ranking of the model in legal document retrieval, extract the most relevant information from several documents, and thus improve retrieval efficiency.

These criteria let one evaluate the categorization, semantic understanding, and document retrieval efficiency of the Legal SVM-RAG model. The weighted F1 value addresses category imbalance, true similarity measures model semantics, and document reordering accuracy measures retrieval efficiency. These evaluation metrics taken together will enable practical implementation and model optimization.

## 4. **Performance testing and analysis.**

### 4.1. **Experimental data.**

The "Legal Case Report Dataset" is used in this work to provide the model rich data in the legal domain by means of a large number of legal case reports spanning several forms of cases (e.g., criminal, civil, administrative, etc.).

The dataset's content spans a broad spectrum of legal documents, including judgements, court decisions, litigation merits, case rulings, and many more elements that can totally support the experimental needs of the document categorization and retrieval model. By means of these records, the model learns to classify them based on case content and thereby enhances the accuracy of document retrieval via relevance sorting. Table 1 provides the fundamental details of The Legal Case Report Dataset:

Using this dataset, the model can accomplish a range of experimental tasks such legal reasoning analysis based on legal texts and judgements, document retrieval based on similarity of cases, or categorization of documents depending on case categories. This dataset offers enough experimental data for this research to guarantee the validity and representativeness of the experimental outcomes.

Table 1. The Legal Case Report Dataset information

| **Item** | **Content** |
|---|---|
| Data Source | Obtained from open legal data platforms, covering various types of legal documents |
| Dataset Size | Approximately 100,000 case records |
| Data Format | JSON or XML format, including case descriptions, judgment text, court information, judgment dates, and legal references |
| Applicable Tasks | Legal document classification, case retrieval, document relevance ranking, etc. |
| Sample Data | Detailed reports of various legal cases, including criminal, civil, and administrative cases |

4.2. **Cross-category document retrieval experiment.** This work evaluates the LegalSVM-RAG model for cross-category document retrieval. If the model can effectively return documents related to a query from several legal domains or categories, cross-category document retrieval tests, the model search civil or administrative cases connected to a criminal case query. Legal document retrieval is a challenging issue since different instances may have same legal relevance but different arrangement and content [32, 33].

Selected for this study were criminal, civil, and administrative cases from "The Legal Case Report Dataset". For this project, questions were developed for every sort of scenario to gather materials in all spheres. Using document relevance to the query, the LegalSVM-RAG model arranges retrieval results. The retrieval results are assessed and the cross-category retrieval efficacy of the model is computed depending on manually labeled relevance. Figure 3 contain the experimental outcomes.

LegalSVM-RAG performs effectively in criminal and civil case retrieval, as the image shows. The high weighted F1 value of the model and document relevance accuracy between criminal and civil cases imply it can better retrieve documents. The model shows its retrieval capacity by correctly determining the most pertinent papers to the search.

The model can stabilize cross-category retrieval between criminal and administrative scenarios even with a performance decline. The model can still capture legal relevance between cases and show a certain degree of document relevance accuracy, therefore proving that it can still be used in many kinds of scenarios.

Cross-category civil and administrative case retrieval of the model gets better. The model shows great weighted F1 values and document relevance accuracy despite significant document differences across civil and administrative cases and criminal cases, therefore proving its flexibility and robustness in cross-domain retrieval tasks.

Particularly for criminal and civil cases, the model shines in cross-category retrieval. Although administrative case retrieval is challenging, the LegalSVM-RAG model shows strong general performance, implying appropriate scalability in multi-category document retrieval applications.

4.3. **Experiments on document distribution in feature space.** This experiment assesses the performance in the feature space of the LegalSVM-RAG-based model, namely with regard to document retrieval efficacy and distribution. This work confirms the synergy between graph structure and SVM by means of feature space analysis of document categories, observation of whether documents can be clustered and differentiated, and

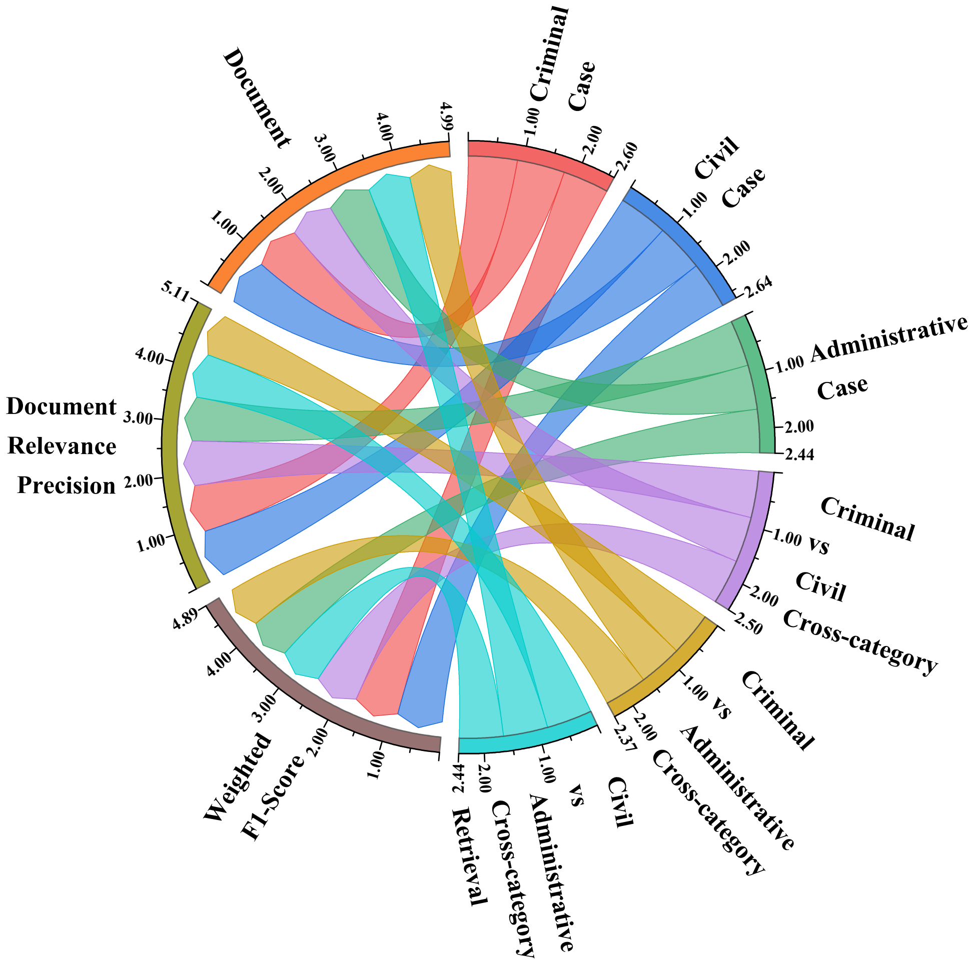

Figure 3. Experimental results of cross-category document retrieval

evaluation of the cross-category retrieval performance of the model. The "Legal Case Report Dataset"—which comprises opinions, case studies, publications, and judgments—was employed in the experiment.

The LegalSVM-RAG model first generated a high-dimensional feature vector by extracting elements from every dataset document. Following dimensionality reduction, we display the 2D feature space distribution of every document. Retrieval in this feature space is evaluated using weighted F1 value, relevance accuracy, and rearrangement accuracy of every document. Figure 4 contain the experimental outcomes.

Different document categories affect the performance of the LegalSVM-RAG-based model, according to findings. Especially case study documents, which have 0.75, 0.82, and 0.78, respectively, showing that the model can better capture and differentiate their properties, judgment and case study documents have higher weighted F1 values, actual similarity, and document rearrangement accuracy. Legal writings fall short in the foregoing standards, most especially with regard to document rearrangement accuracy (0.73). Legal documents could be more difficult to obtain and filter for the model depending on their standardizing and simplicity.

Further investigation reveals that, particularly in reordering, the combined LegalSVM-RAG model can efficiently manage documents with complicated semantic connections by precisely ranking pertinent articles in the front. The model shines at cross-category document retrieval even if legal texts suffer in performance. By means of feature space

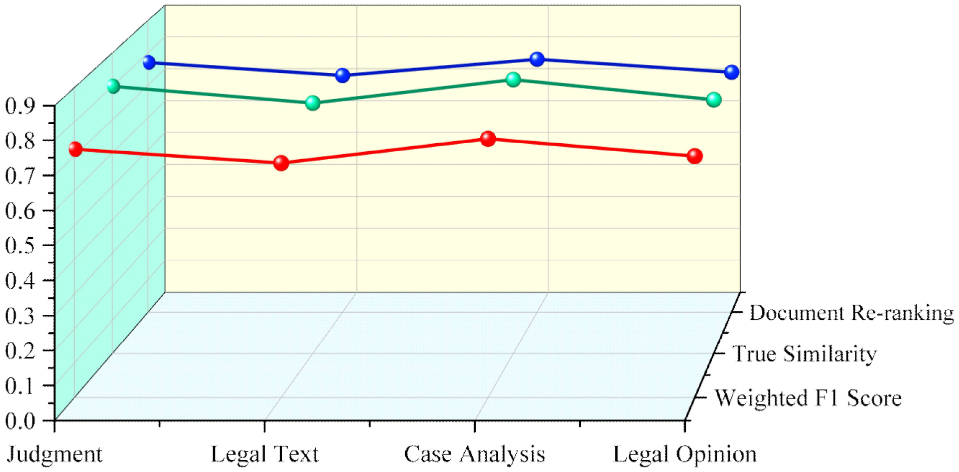

Figure 4. Experimental results of document distribution in feature space

optimization, the model increases retrieval and ranking accuracy, therefore offering a suitable solution for intelligent legal document retrieval.

5. **Conclusion.** The LegalSVM-RAG model is proposed in this work to enhance legal document retrieval and storage by combining GraphRAG with SVM. This work uses the GraphRAG model to improve legal document feature extraction and representation as well as SVM classification model document retrieval and accuracy. By means of testing, this paper demonstrates the legal document retrieval efficiency of the model. This paper suggests a fresh paradigm based on intelligent legal document retrieval and storage that improves accuracy and efficiency.

This study also has shortcomings. Though less varied and complicated, "The Legal Case Report Dataset" was used for experimental validation. Legal documents may in fact comprise international law, tax law, and others; consequently, the cross-domain performance of the model has to be investigated. Second, although GraphRAG and SVM together can enhance document retrieval, the model performs badly with some complicated legal documents—especially administrative situations. Future studies thus have to focus on how to maximize the generalization of the SVM classification model and the GraphRAG model for multi-domain legal materials.

Future study can go in many directions. More contextual information and domain knowledge can be used together with an expert system for model training to improve the retrieval of administrative cases and thus the processing of particular document types. Second, legal document type diversity makes cross-domain legal document retrieval and storage still absolutely vital. To improve the model's digestion of complex papers, we can then include more strong deep learning models such as Transformer architecture. Another study field, particularly in light of globalization, is multilingual and cross-lingual retrieval of legal documents. Important subjects for next studies will be how to manage legal papers in several languages and enhance cross-lingual retrieval accuracy.

Finally, although this work has great potential with its LegalSVM-RAG model for intelligent legal document retrieval and storage, it has restrictions that offer insightful analysis for next studies. Improved accuracy, robustness, cross-domain and cross-language generalization of the model will help to enhance application results in more legal document processing environments.

**Acknowledgment.** This work is supported by the Anhui Provincial Department of Education named: Research on personal information infringement damages litigation (No. 2024AH052116).

## REFERENCES

[1] T. Wu, S. He, J. Liu, S. Sun, K. Liu, Q.-L. Han, and Y. Tang, "A brief overview of ChatGPT: The history, status quo and potential future development," *IEEE/CAA Journal of Automatica Sinica*, vol. 10, no. 5, pp. 1122-1136, 2023.

[2] T.-Y. Wu, H. Li, S. Kumari, and C.-M. Chen, "A Spectral Convolutional Neural Network Model Based on Adaptive Fick's Law for Hyperspectral Image Classification," *Computers, Materials & Continua*, vol. 79, no. 1, pp. 19-46, 2024.

[3] B. Alouffi, M. Hasnain, A. Alharbi, W. Alosaimi, H. Alyami, and M. Ayaz, "A systematic literature review on cloud computing security: threats and mitigation strategies," *IEEE Access*, vol. 9, pp. 57792-58707, 2021.

[4] D. Khurana, A. Koli, K. Khatter, and S. Singh, "Natural language processing: state of the art, current trends and challenges," *Multimedia Tools and Applications*, vol. 82, no. 3, pp. 3713-3744, 2023.

[5] M. Paramesha, N. L. Rane, and J. Rane, "Big data analytics, artificial intelligence, machine learning, internet of things, and blockchain for enhanced business intelligence," *Partners Universal Multidisciplinary Research Journal*, vol. 1, no. 2, pp. 110-133, 2024.

[6] D. Theng, and K. K. Bhoyar, "Feature selection techniques for machine learning: a survey of more than two decades of research," *Knowledge and Information Systems*, vol. 66, no. 3, pp. 1575-1637, 2024.

[7] T. K. Mishra, M. Kolhar, S. R. Mishra, H. Mohapatra, F. Al-Turjman, and A. K. Rath, "Local features-based evidence glossary for generic recognition of handwritten characters," *Neural Computing and Applications*, vol. 36, no. 2, pp. 685-695, 2024.

[8] N. Lettieri, A. Guarino, D. Malandrino, and R. Zaccagnino, "Knowledge mining and social dangerousness assessment in criminal justice: metaheuristic integration of machine learning and graph-based inference," *Artificial Intelligence and Law*, vol. 31, no. 4, pp. 653-702, 2023.

[9] D. Xiao, M. Dianati, W. G. Geiger, and R. Woodman, "Review of graph-based hazardous event detection methods for autonomous driving systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 5, pp. 4697-4715, 2023.

[10] F. d. Oliveira, and J. M. P. d. Oliveira, "A RDF-based graph to representing and searching parts of legal documents," *Artificial Intelligence and Law*, vol. 32, no. 3, pp. 667-695, 2024.

[11] M. A. K. Raiaan, M. S. H. Mukta, K. Fatema, N. M. Fahad, S. Sakib, M. M. J. Mim, J. Ahmad, M. E. Ali, and S. Azam, "A review on large language models: Architectures, applications, taxonomies, open issues and challenges," *IEEE Access*, vol. 12, pp. 26839-26874, 2024.

[12] M. A. Ferrag, M. Ndhlovu, N. Tihanyi, L. C. Cordeiro, M. Debbah, T. Lestable, and N. S. Thandi, "Revolutionizing cyber threat detection with large language models: A privacy-preserving bert-based lightweight model for iot/iiot devices," *IEEE Access*, vol. 12, pp. 23733-23750, 2024.

[13] Z. Hu, W. Hou, and X. Liu, "Deep learning for named entity recognition: a survey," *Neural Computing and Applications*, vol. 36, no. 16, pp. 8995-9022, 2024.

[14] Š. Čebirić, F. Goasdoué, H. Kondylakis, D. Kotzinos, I. Manolescu, G. Troullinou, and M. Zneika, "Summarizing semantic graphs: a survey," *The VLDB Journal*, vol. 28, pp. 295-327, 2019.

[15] V. Vasudevan, M. Bassenne, M. T. Islam, and L. Xing, "Image classification using graph neural network and multiscale wavelet superpixels," *Pattern Recognition Letters*, vol. 166, pp. 89-96, 2023.

[16] Z. Zhong, C.-T. Li, and J. Pang, "Hierarchical message-passing graph neural networks," *Data Mining and Knowledge Discovery*, vol. 37, no. 1, pp. 381-408, 2023.

[17] X. He, Y. Tian, Y. Sun, N. Chawla, T. Laurent, Y. LeCun, X. Bresson, and B. Hooi, "G-retriever: Retrieval-augmented generation for textual graph understanding and question answering," *Advances in Neural Information Processing Systems*, vol. 37, pp. 132876-132907, 2025.

[18] D. M. Abdullah, and A. M. Abdulazeez, "Machine learning applications based on SVM classification a review," *Qubahan Academic Journal*, vol. 1, no. 2, pp. 81-90, 2021.

[19] M. Tanveer, T. Rajani, R. Rastogi, Y.-H. Shao, and M. Ganaie, "Comprehensive review on twin support vector machines," *Annals of Operations Research*, vol. 339, no. 3, pp. 1223-1268, 2024.

[20] S. Li, G. Chai, Y. Wang, G. Zhou, Z. Li, D. Yu, and R. Gao, "Crsf: An intrusion detection framework for industrial internet of things based on pretrained cnn2d-rnn and svm," *IEEE Access*, vol. 11, pp. 92041-92054, 2023.

[21] D. K. Jain, S. B. Dubey, R. K. Choubey, A. Sinhal, S. K. Arjaria, A. Jain, and H. Wang, "An approach for hyperspectral image classification by optimizing SVM using self organizing map," *Journal of Computational Science*, vol. 25, pp. 252-259, 2018.

[22] M. Singla, D. Ghosh, and K. Shukla, "A survey of robust optimization based machine learning with special reference to support vector machines," *International Journal of Machine Learning and Cybernetics*, vol. 11, no. 7, pp. 1359-1385, 2020.

[23] E. Cook, S. Luo, and Y. Weng, "Solar panel identification via deep semi-supervised learning and deep one-class classification," *IEEE Transactions on Power Systems*, vol. 37, no. 4, pp. 2516-2526, 2021.

[24] W. Lv, T. Li, H. Ren, S. Zeng, and J. Zhou, "Inequality distance hyperplane multiclass support vector machines," *International Journal of Intelligent Systems*, vol. 37, no. 3, pp. 2046-2060, 2022.

[25] S. Das, S. P. Nayak, B. Sahoo, and S. C. Nayak, "Machine learning in healthcare analytics: a state-of-the-art review," *Archives of Computational Methods in Engineering*, vol. 31, no. 7, pp. 3923-3962, 2024.

[26] J. Luengo, D. García-Gil, S. Ramírez-Gallego, S. Garcí, and F. Herrera, *Big data preprocessing*. Cham: Springer, vol. 1, pp. 1-186, 2020.

[27] A. P. Tuan, B. Tran, T. H. Nguyen, L. N. Van, and K. Than, "Bag of biterms modeling for short texts," *Knowledge and Information Systems*, vol. 62, no. 10, pp. 4055-4090, 2020.

[28] R. Bhattacharya, N. K. Nagwani, and S. Tripathi, "Detecting influential nodes with topological structure via graph neural network approach in social networks," *International Journal of Information Technology*, vol. 15, no. 4, pp. 2233-2246, 2023.

[29] S. Weerasinghe, T. Alpcan, S. M. Erfani, and C. Leckie, "Defending support vector machines against data poisoning attacks," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2566-2578, 2021.

[30] K. Takahashi, K. Yamamoto, A. Kuchiba, and T. Koyama, "Confidence interval for micro-averaged F 1 and macro-averaged F 1 scores," *Applied Intelligence*, vol. 52, no. 5, pp. 4961-4972, 2022.

[31] L. Jiang, Y. Xie, X. Wen, and T. Ren, "Modeling highly imbalanced crash severity data by ensemble methods and global sensitivity analysis," *Journal of Transportation Safety & Security*, vol. 14, no. 4, pp. 562-584, 2022.

[32] A. Savelyev, "Copyright in the blockchain era: Promises and challenges," *Computer Law & Security Review*, vol. 34, no. 3, pp. 550-561, 2018.

[33] G. Governatori, F. Idelberger, Z. Milosevic, R. Riveret, G. Sartor, and X. Xu, "On legal contracts, imperative and declarative smart contracts, and blockchain systems," *Artificial Intelligence and Law*, vol. 26, pp. 377-409, 2018.