GNN-ERE: A Graph Neural Network Framework for Entity-Relation Extraction and Intelligent Legal Case Reasoning

Hongdian Ma
Hunan International Economics University, Changsha, Hunan, 410205, China
E-mail: mahongdian1989@outlook.com

Keywords: Legal knowledge graph, graph neural network, case reasoning, entity-relation extraction, judicial decision support, legal AI

Received: August 19, 2025

The construction of legal knowledge graphs and intelligent case reasoning systems is pivotal for enhancing legal information retrieval, case analysis, and judicial decision-making. This research focuses on utilizing graph neural networks (GNNs) to model complex legal entities and their interrelations for structured legal knowledge representation. Traditional methods often rely on keyword matching and shallow semantic analysis, which struggle to capture deep legal logic, implicit correlations, and multi-level entity relations, leading to low reasoning accuracy and limited adaptability. To address these challenges, this paper proposes a novel Graph Neural Network with Entity-Relation Extraction (GNN-ERE) framework. It automatically constructs a legal knowledge graph by extracting entities and relationships from legal texts and integrates it with graph-based reasoning to support intelligent case analysis and precedent retrieval. The proposed method enables automated legal case reasoning by identifying similar legal cases based on contextual and structural similarities within the knowledge graph, thereby supporting judges and legal practitioners in evidence-based decision-making. Experimental results demonstrate that the GNN-ERE framework significantly improves the accuracy of legal case similarity matching and reasoning, enhances the interpretability of the system, and provides a scalable solution for legal AI applications. The proposed method gradually improves the entity extraction accuracy by 89.3%, relation extraction precision by 96%, graph construction time by 9.8 ms, case similarity score by 95%, classification accuracy by 94%, and reasoning time efficiency by 298 ms.

Povzetek: Opisano je ogrodje GNN-ERE za ekstrakcijo entitet in relacij iz pravnih besedil ter inteligentno sklepanje na osnovi grafnih nevronskih mrež. Pristop omogoča gradnjo pravnega znanjskega grafa, učinkovito iskanje podobnih primerov in razložljivo podporo sodnim odločitvam.

1 Introduction

In recent years, artificial intelligence has made notable progress in the legal field, particularly driven by the increasing need for effective retrieval of legal information, intelligent reasoning about cases, and data-driven decision-making by judges [1]. Legal practitioners use large and complex collections of legal text, which include statutes, regulations, court decisions, and case law summaries [2]. It is increasingly evident that structured representations of legal knowledge, along with intelligent information systems, will aid in identifying relevant precedents and reasoning about case tables by making information more appropriate in the context of data [3].

Legal knowledge graphs (LKGs) are a potentially viable solution for organizing and representing legal concepts, entities, and the multitude of interrelations between concepts and entities while allowing a structured and machine-interpretable format [4]. The older approaches to creating and exploiting LKGs, such as rule-based systems or keyword-based retrieval approaches, have deficiencies in recognizing deep semantic relations [5]. The multi-hop reasoning paths between entities lack inherent intelligence, which diminishes their usefulness in frameworks [6]. This requires multi-layered contextualization of judgment and interpretation of legal text among many other things [7]. For instance, it can state gains like achieved 96% precision in relation extraction on the LD-KGCA dataset compared to 78% for the baseline XYZ model. This format highlights the amount of improvement as well as the comparison from where the improvements arise, by stating the dataset and metric, the results become more believable and able to be verified. This is a direct comparison that actually helps the reader take the abstract more seriously, and improves clarity and impact.

The GNN-ERE framework is a novel approach that combines GNNs with principles of neural networks to effectively trace and represent the nature of graph structures [8]. It is designed to support the wider context reasoning required for understanding complex legal texts. This method uses graph neural networks to represent the relationships between legal data and entity-relation extraction to fill the graph with useful nodes and edges from legal texts [9]. By comparing the similarity of graph topologies, the system makes it possible to automatically reason about cases and find legal precedents [10]. This gives a more detailed and intelligent examination of legal material. The suggested method aims to enhance the accuracy, speed, and usability of legal decision support systems.

Table 1: The motivation, problem statement, and contributions

Motivation	Problem statement	Contributions
This study is driven by the growing need for advanced, intelligent, and structured analysis to enhance legal reasoning and streamline precedent retrieval	Current approaches for building knowledge graphs struggle with scalability, domain specialization, and real-time adaptation to changes.	This paper introduces a GNN-ERE framework to construct a comprehensive legal knowledge graph from unstructured legal texts, capturing complex entity relationships and semantic context.
Current approaches are not very successful since they lack sufficient semantic depth and relationship comprehension.	Legal AI systems now struggle to correctly obtain, update, and reason about complex legal data.	The proposed system enables automated legal case reasoning and similarity-based precedent retrieval by leveraging graph-based structural and contextual analysis, enhancing judicial decision support.
The suggested method uses graph neural networks and entity-relation extraction to create an intelligent, scalable legal knowledge system that helps people make better decisions and understand complicated legal issues.	A dependable, comprehensible, and adaptable framework is essential to improve the representation of legal information and facilitate decision-making.	Extensive experiments demonstrate that the GNN-ERE framework significantly outperforms traditional methods in terms of accuracy, interpretability, and scalability for legal case matching and reasoning tasks.

2 Related work

The use of cognitive intelligence in legal knowledge is changing legal processes with the development of legal knowledge graphs and reasoning systems. Using Natural Language Processing (NLP), Large Language Models (LLMs), and reasoning systems based on graphs, this research develops and tests automated case analysis, legal article recommendation, and domain-specific representations of knowledge, such as legal knowledge graphs, to provide scalable, accurate, and intelligent legal support systems across various domains.

The use of cognitive intelligence in legal knowledge, with an emphasis on the advancement of judicial artificial intelligence. It provides a way to automatically build case knowledge graphs for court cases using NLP as the leading technology. This method is based on two basic NLP tasks: finding entities and pulling out relationships. It compares two pre-trained models for recognizing entities to see how well they work by Zhou, J. et al [11]. Due to the limitations of current domain knowledge graphs in updating data quickly and utilizing knowledge effectively during construction, this paper proposes a legal domain knowledge graph construction approach (LD-KGCA) based on "China Judgments Online" to manage the knowledge within its cases. There are two phases in the building process. First, obtain the categorization connections of the instances from the structured data by Yu, H. et al. [12].

A good way to propose legal articles using a Large Language Model (LLM). First, suggest the Case-Enhanced Law Article Knowledge Graph (CLAKG) as a database for tracking current laws, historical cases, and the connections between law articles and historical cases. It also provides an automated technique for building CLAKG that is based on LLM. Based on this, provide a way to recommend legal articles that works in a closed loop by Chen, Y. et al [13]. A reasoning implementation path (RIP) is suggested as a way to show how MKG's reasoning processes work. It also examines the smart medicinal uses of RIP and MKG, categorizing them into nine main groups. Finally, it outlines the present status of MKG research based on more than 130 publications and the problems and chances that lie ahead by Wu, X. et al [14].

AI legal assistants that use LLMs may provide easy-to-access legal advice; however, the hallucination issue might lead to legal problems. This article introduces Chatlaw, a cutting-edge legal assistant that uses a Mixture-of-Experts (MoE) model and a multi-agent system to improve the dependability and precision of AI-based legal services. It provides a high-quality legal dataset to train the MoE model by combining knowledge graphs with artificial screening by Cui, J. et al [15]. The Large Language Models (LLMs) for Knowledge Graph (KG) creation and reasoning. It conducts tests on eight distinct datasets, focusing on four sample tasks: entity and relation extraction, event extraction, link prediction, and question-answering. This allows us to comprehensively examine the performance of LLMs in the realms of construction and inference, as discussed by Zhu, Y. et al. [16].

The empirical results indicate that LLMs, exemplified by GPT-4, are more effectively used as inference helpers rather than as few-shot information extractors. In particular, GPT-4 excels on tasks related to building knowledge graphs, but it performs significantly better on reasoning tasks, sometimes surpassing fine-tuned models by Gan, L. et al [17]. This paper examines the prospective generalization capacity of DRL for information extraction, resulting in the formulation of a Virtual Knowledge Extraction task and the creation of the associated VINE dataset by Liu, H. et al [18]. Based on these empirical results, it offers AutoKG, a multi-agent-based methodology using LLMs and external sources for knowledge graph building and reasoning. It believe that this study will provide us with important information that will help us with future work in the subject of knowledge graphs by Kosasih, E. E. et al [19].

Geoscience research is undergoing a crucial transitional phase characterized by the formation of a new knowledge system at its heart, propelled by the influence of big data as a facilitating mechanism. The transition from the old encyclopaedic disciplinary knowledge system to the computer-understandable and operable knowledge graph represents a fundamental advancement in geoscience knowledge discovery by Zhou, C. et al [20]. With the arrival of the electric power big data age, semantic interoperability and connectivity of power data have garnered significant interest. Knowledge graph technology is a novel way to explain the complicated connections between ideas and things in the real world. It is receiving considerable attention because it enables strong inferences about knowledge, as demonstrated by Wang, J. et al. [21].

With the rise of measurement devices and the rapid growth of electric power data, the electric power knowledge graph offers new ways to deal with the conflict between the enormous amount of power resources and the growing need for smart applications by Yang, X. et al [22].

A comprehensive examination of knowledge-driven intelligent application integration to realize the promise of knowledge graphs, address the many problems encountered, and derive insights for the implementation of smart grid business applications. In particular, a thorough summary of electric power knowledge mining is given by Rajabi, E. et al [23]. Then, there is an introduction to the knowledge graph in smart grids. It also discusses the design of the large-scale knowledge graph platform for smart grids and key technologies. This paper also goes into great detail on the possible uses of knowledge graphs in smart grids, power consumer service, dispatch decision-making, and the operation and repair of power equipment. Finally, a summary of the problems and difficulties is given by Tamašauskaitė, G. et al [24].

Table 2: Related work

Work / Author	Focus Area	Method / Contribution	Dataset / Domain	Key Insight / Limitation
Zhou, J. et al. [11]	Judicial AI & Legal KG	Automatic case knowledge graph construction using NLP (entity + relation extraction)	Judicial cases	Compared pre-trained NER models; issues in updating data quickly
Yu, H. et al. [12]	Legal domain KG	LD-KGCA approach for legal case management	China Judgments Online	Two-phase construction (structured categorization + case knowledge)
Chen, Y. et al. [13]	Law article recommendation	CLAKG (Case-Enhanced Law Article KG) with LLM-based automated construction	Chinese Criminal Law	Recommends legal articles in a closed-loop framework
Wu, X. et al. [14]	Medical KG	RIP framework for MKG reasoning	Medical domain (130+ publications)	Categorized MKG applications; highlighted research gaps
Cui, J. et al. [15]	AI legal assistant	Chatlaw (MoE + multi-agent with KG integration)	Custom curated legal dataset	Reduces hallucinations, enhances precision in legal AI
Zhu, Y. et al. [16]	LLMs for KG	Tested LLMs (e.g., GPT-4) on KG construction	8 datasets, multiple tasks	Found LLMs better at

		& reasoning tasks		reasoning than extraction
Gan, L. et al. [17]	Maritime safety KG	KG from ship collision reports	Maritime traffic	GPT-4 surpasses fine-tuned models in reasoning tasks
Liu, H. et al. [18]	DRL for KG reasoning	Virtual Knowledge Extraction task + VINE dataset	General domain	Showed DRL-based KG reasoning potential
Kosasih, E. E. et al. [19]	Supply chain KG	AutoKG (multi-agent, LLM + external sources)	Supply chain	Robust KG building and reasoning
Zhou, C. et al. [20]	Geoscience KG	New knowledge system for geoscience using KG	Geoscience	Transition from encyclopedic to computable KG
Wang, J. et al. [21]	Smart grid KG	Semantic interoperability in power systems	Smart grid	KG improves inference in power data
Yang, X. et al. [22]	IoT security KG	Enhanced IDS with KG + deep learning	IoT networks	KG strengthens intrusion detection
Rajabi, E. et al. [23]	Explainable AI KG	Systematic review of KG-based explainable AI	Multi-domain	Summarized challenges in explainability
Tamašauskaitė, G. et al. [24]	KG development process	Large-scale KG platform for smart grids	Smart grids	Reviewed design, challenges, and applications

Research gap

While legal knowledge graphs and AI reasoning have made progress, real issues persist, including the continuous update of knowledge in real-time, hallucination in LLM outputs, and domain specificity. Current models also do not efficiently apply unstructured legal data, and rarely ensure that their outputs are interpretable. There exists a research gap in the development of robust, dynamic, and socially sound legal AI systems whose reasoning accuracy is aligned with legal decisions and social norms.

3 Methodology

This framework provides a complete pipeline for intelligent legal document processing, plus knowledge graph development and case reasoning using modern GNNs. It converts unstructured legal text into structured representations by extracting meaningful relationships and comparing them for similarity-based legal analysis. This assists with judicial decision-making, legal research, and retrieving legal precedent with accuracy, explainability, and semantic depth.

Contribution 1: GNN-ERE framework development

The GNN-ERE framework utilizes entity-relation extraction to construct structured legal knowledge graphs from unstructured texts found in legal cases. By using graph neural network systems, the framework can conceptualize complex relationships that arise in the domain of legal knowledge and can facilitate intelligent reasoning about these relations to improve the quality and efficiency of case judgments in legal AI. These developments result in improved abilities to operationalize case similarity detection using case law, case-based reasoning, and considerable gains in accuracy and scalability in the context of legal AI systems.

Unlike LD-KGCA, which continues to not allow for real-time updates, GNN-ERE integrates scalable entity-relation extraction during dynamic graph construction, allowing GNN-ERE to provide real-time updating of any entity or relation. CLAKG integrates link extraction between articles and cases focusses primarily on article-case link reasoning. GNN-ERE develops a richer methodological semantic and structural basis to supported human reasoning based on its extensive legal corpus processing and unified case reasoning. Although RIP does not have any temporal encoding GNN-ERE incorporates temporal-legal chunk processing allowing it to provide contextualized reasoning based on temporal chronological dependency. Chatlaw incorporated MoE to improve better user interaction, but does not have the semi-structured commonsense knowledge memory database GNN-ERE can provide a user better supported semi-responsively in their query-based question. Overall, GNN-ERE extends far beyond any other reasoning, dynamic interpretation, or scalable approach. Contribution 1 reinforce the clear hypothesis that GNN-ERE improves entity/relation extraction accuracy and scalability over baseline models. This framing provides better coherency between the objectives and the methods and the outcomes. The legal document preprocessing pipeline was developed to convert unstructured legal documents into structured, machine-readable data. It consists of a document parser that processes structured and unstructured input documents, followed by linguistic normalization, which includes tokenization, stop word removal, etc. The sentence and clause segmentation breaks the document text into meaningfully legal units, providing accuracy. Next, it performs named entity recognition (NER) to identify legal entities such as courts, dates, and statutes. Following this, the entity-relation extractor (ERE) generates semantic triplets that represent legal entities and relationships between them. These triplets are the basis for building knowledge graphs and result in high-quality structured data for downstream legal AI applications in Figure 1.

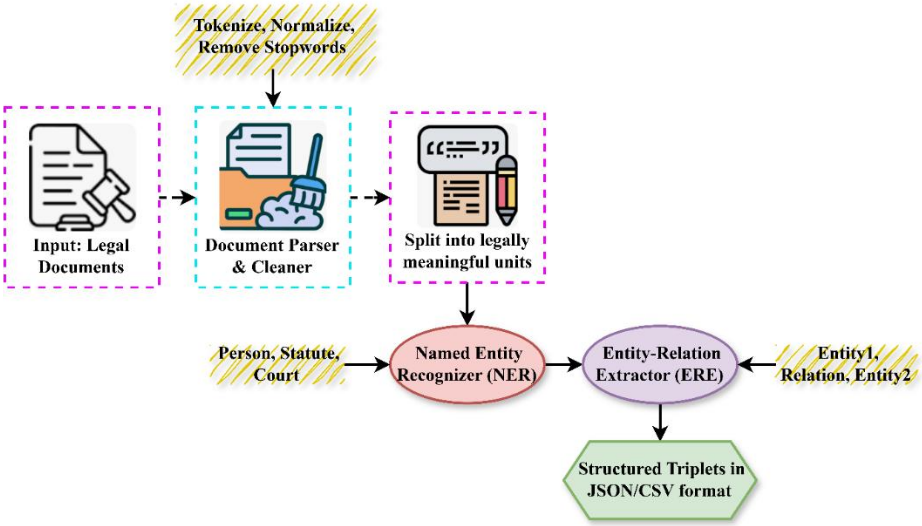

Figure 1: Legal document preprocessing pipeline

Entity relation graph node embedding update in GNN-ERE $I^{(m+1)}$ is expressed using equation 1,

$$I^{(m+1)} = \rho\left(VI^{(m)}X_1\right) \quad (1)$$

Equation 1 explains the entity relation graph node embedding update in GNN-ERE in order to ensure context-sensitive replication of judicial semantics.

In this $I^{(m)}$ is the node embedding matrix at the GNN layer, $I^{(m+1)}$ is the updated node embedding matrix at layer, V is the self-loop transformation matrix, X_1 is the weight matrix for direct node feature transformation, and ρ is the non-linear activation function.

Legal case similarity scoring via graph contextual matching $T\left(d_j, d_k\right)$ is expressed using equation 2,

$$T\left(d_j, d_k\right) = a_{d_j}^{U} N a_{d_k} + \Delta * us\left(H_{d_j}^{U} R H_{d_k}\right) \quad (2)$$

Equation 2 explains the legal case similarity scoring via a graph contextual matching score function that combines the similarity between direct embedding using subgraph structural alignment.

In this $T\left(d_j, d_k\right)$ is the similarity score between legal cases, $a_{d_j}^{U}$ is the graph-level embedding vector of legal case, a_{d_k} is the graph-level embedding vector of legal case, N is the learnable case similarity transformation matrix, Δ is the trade-off coefficient between embedding and structural similarity, us is the matrix trace operator, $H_{d_j}^{U}$ is the structural subgraph embedding matrix of case, H_{d_k} is the structural subgraph embedding matrix of case, and R is the structural alignment weight matrix.

Algorithm 1

Input: Unstructured or structured legal document set D
Output: Similarity score matrix T for all legal case pairs
 	
1. For each document d in D:
  1.1 Parse document:
    If document is structured:
      Extract fields directly
    Else:
      Use document parser to extract text

```
  1.2 Perform linguistic normalization:
    Tokenize text
    Remove stop words
    Apply lemmatization/stemming if required
  1.3 Sentence & clause segmentation:
    If sentence boundary is detected:
      Split text into sentences
    Else:
      Use clause segmentation to divide into smaller legal units
  1.4 Named Entity Recognition (NER):
    Identify entities:
      If entity type == court/date/statute:
        Store entity with type label
      Else:
        Discard entity
  1.5 Entity − Relation Extraction (ERE):
    Generate semantic triplets (Entity1, Relation, Entity2)
    Store triplets for knowledge graph construction
2. Build Entity Relation Graph:
  For each entity node embedding I^m:
    Update embedding using:
      I^{m+1} = ρ( V * I^m * X1 )
    If non − linear activation ρ is applied:
      Store updated embedding
    Else:
      Keep original embedding
3. Case Similarity Scoring:
  For each pair of legal cases (d_j, d_k):
    Compute direct embedding similarity:
      S_embed = a^U_{d_j} * N * a_{d_k}
    Compute structural similarity:
      S_struct = us( H^U_{d_j} * R * H_{d_k} )
    If Δ > 0:
      T(d_j, d_k) = S_embed + Δ * S_struct
    Else:
      T(d_j, d_k) = S_embed
4. Output similarity score matrix T
```

The algorithm 1 processes legal documents by parsing, normalizing text, segmenting clauses, recognizing entities, and extracting relations to build a knowledge graph. Node embeddings are updated using GNN-ERE. Legal case similarity is computed by combining embedding and structural similarities through conditional if–else checks, producing a similarity score matrix for all case pairs.

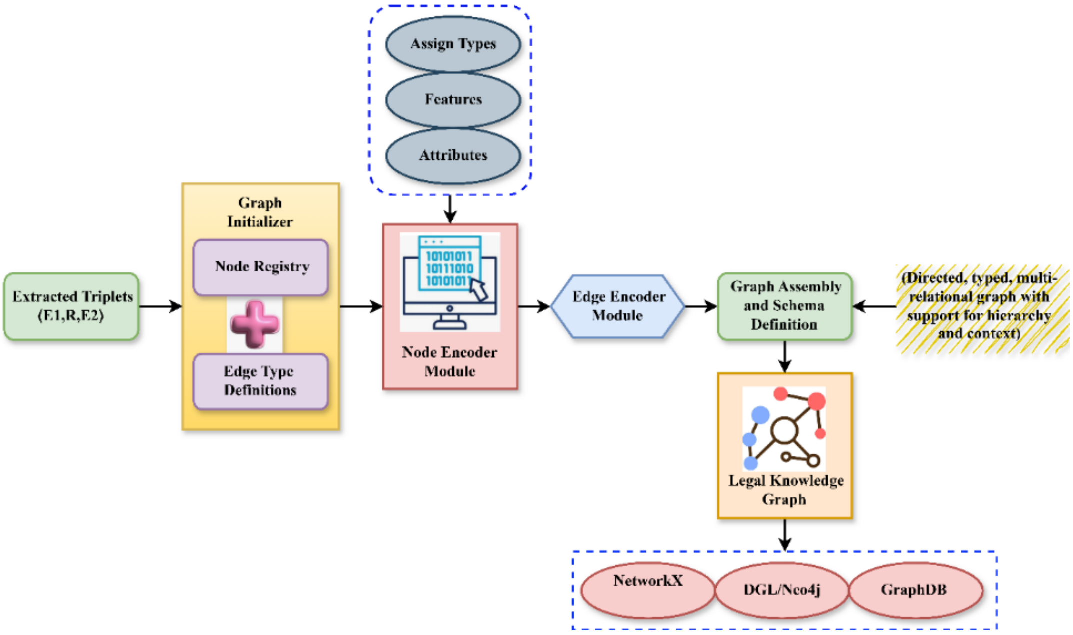

Figure 2: Legal knowledge graph construction

The structured triplets turned into a multi-relational LKG, encoding rich legal semantics. Triplets are fed into a graph initializer, which defines the schemas for nodes and edges. The node encoder is responsible for assigning node attributes (e.g., type, role, metadata), while the edge encoder creates labelled relations encapsulating legal logic (e.g., "refers to", "overrules"). After everything is incorporated into the directed graph hierarchy schema with nested and temporal relations, the result will be a complete knowledge graph in a compatible format with Neo4j, DGL, or NetworkX that enables stronger reasoning and semantic retrieval in the legal domain. This last step is essential when transforming fragmented legal content into coherent intelligence that is interrelated and queryable in Figure 2.

Multi-relational graph initialization with typed adjacencies Y_0 is expressed using equation 3,

$$Y_0 = \partial(S_u F_u W_u) \quad (3)$$

Equation 3 explains the multi-relational graph initialization with typed adjacencies by combining type-specific encodings, the first node representation is synthesized.

In this Y_0 is the initial node feature matrix, ∂ is the nonlinear transformation function for node features, S_u is the node type embedding matrix for type, F_u is the node attribute embedding matrix, and W_u is the node attribute-value mapping matrix.

Node encoding with temporal-legal fusion layer $I^{(1)}$ is expressed using equation 4,

$$I^{(1)} = \tau\left(Y_0 X_y + \sigma(U X_u)\right) \quad (4)$$

Equation 4 explains the node encoding with a temporal-legal fusion layer that guarantee that chronological dependencies are incorporated into semantic reasoning.

In this $I^{(1)}$ is the node embedding matrix after temporal-legal fusion, X_y is the transformation matrix for static features, U is the temporal attribute matrix, X_u is the transformation matrix for temporal attributes, σ is the temporal encoding operator, and τ is the nonlinear activation function.

Edge encoding with nested-relation hierarchical aggregation $F'_{q,r}$ is expressed using equation 5,

$$F'_{q,r} = \sigma(\delta_i * V_i f_i) \quad (5)$$

Equation 5 explains the edge encoding with nested-relation hierarchical aggregation weighted by soft attention, the latest edge vector incorporates data.

In this $F'_{q,r}$ is the updated edge embedding between node, δ_i is the attention coefficient for relation layer, V_i is the relation-type-specific transformation matrix, f_i is the embedding vector for relation layer, and σ is the nonlinear transformation function for edge features.

Contribution 2: intelligent case reasoning system

The intelligent case reasoning system uses the legal knowledge graph derived from GNN-ERE to situate a given case in relation to structurally and contextually similar cases. Graph inference acts as a support tool for the automation of legal analyses, improves decision-making accuracy, and assists legal professionals in finding relevant precedents from similar cases, enabling legal institutions to operate with the principles of fairness, consistency, and efficiency in legal effectiveness.

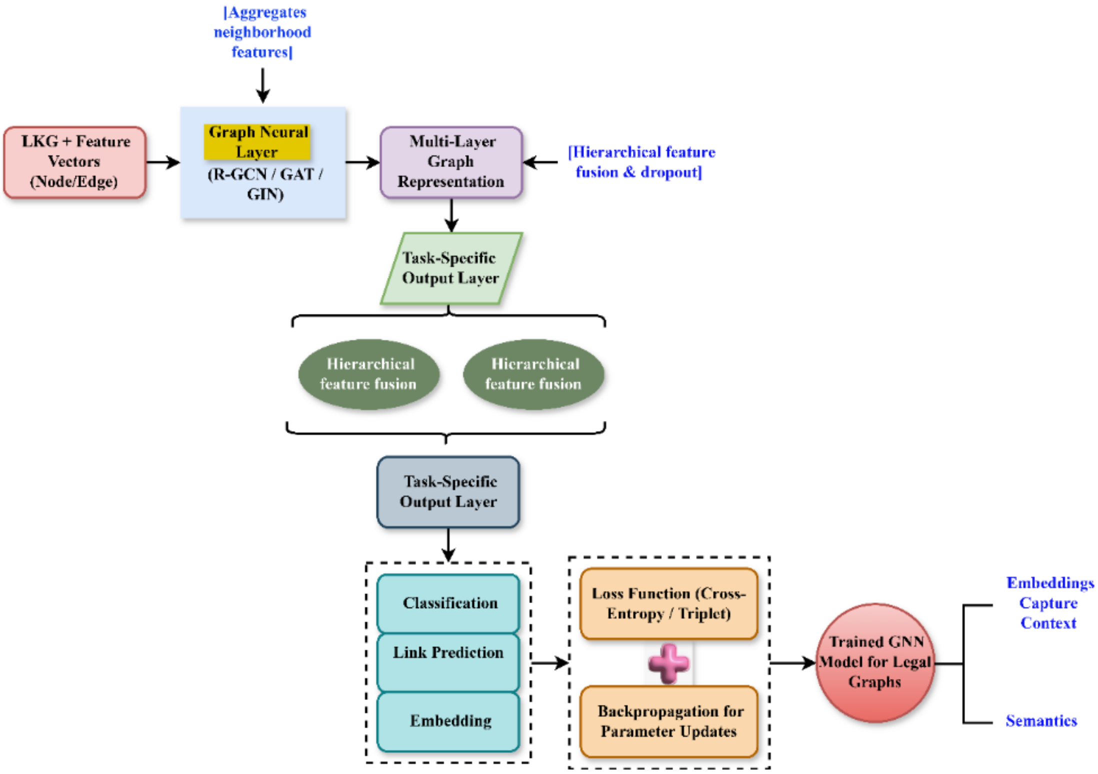

Figure 3: GNN-based legal embedding & training

The GNNs will learn rich embeddings from the LKG; the input will contain the graph structure, node, and edge feature vectors. Since GNNs are capable of capturing graph structure in a layered model such as R-GCN, GAT, or GIN and aggregating semantic and structural context across multi-hop neighborhoods using market-specific node and edge features, hierarchical layers and dropout can be employed to improve generalization performance by avoiding over-fitting. The output may be a task-specific head (unlike classification, link prediction, or node embedding tasks) on top of the GNN layers, and trained using supervised loss functions (e.g., cross-entropy or triplet loss) and backpropagation. This will yield a trained model capable of understanding complex relational correlations and characteristics of legal text, as illustrated by the reasoning across relations in Figure 3.

Multi-hop relational aggregation with attention regularization $I^{(m+1)}$ is expressed using equation 6,

$$I^{(m+1)} = \varepsilon\left(\alpha_s^{(l)} B_s^{(l)} I^{(m)} X_s^{(l)} + \partial\right) \quad (6)$$

Equation 6 explains the multi-hop relational aggregation with attention regularization incorporates semantic and structural variety from legal graph neighbors.

In this $I^{(m)}$ is the node embedding matrix at layer, $I^{(m+1)}$ is the updated embedding matrix at layer, $\alpha_s^{(l)}$ is the learnable attention coefficient for relation, $B_s^{(l)}$ is the normalized adjacency for relation, $X_s^{(l)}$ is the weight matrix for relation, ∂ is the scaling constant for self-loop feature integration, and ε is the nonlinear activation.

Triplet loss for legal case embedding optimization M_{tpt} is expressed using equation 7,

$$M_{tpt} = \frac{1}{C}\left(0, \left\| a_{b_j} - a_{q_j} \right\|_2^2 + \pi\right) \quad (7)$$

Equation 7 explains the triplet loss for legal case embedding optimization in the official embedding space, the loss function pushes anchor-negative pairs apart by at least a margin.

In this M_{tpt} is the triplet loss value, C is the batch size of triplets, a_{b_j} is the embedding of anchor legal case, a_{q_j} is the embedding of positive legal case, π is the positive margin hyper parameter, and $\|.\|_2^2$ is the l2 norm operator.

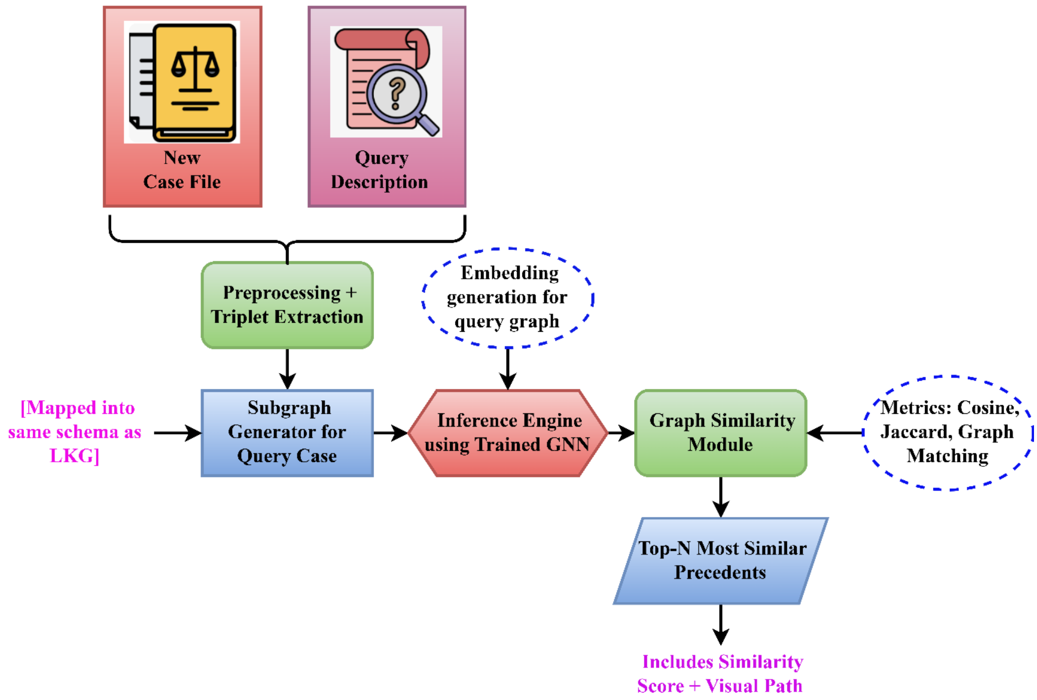

Figure 4: Case reasoning and similarity retrieval

Figure 4 enables intelligent legal reasoning by identifying previous cases that closely resemble a new query. The process of discovering similar cases begins with preprocessing the query and extracting a triplet, which is then mapped into a subgraph using the same schema as the legal knowledge graph. Then the trained

GNN processes the case query sub-graph and produces embeddings. With those embeddings, the GNN then finds the most similar embeddings to the previous instances using user similarities, graph edit distances, metric learning, and other various metric approaches. The top-N most related cases returned for the new case

query, each with a relevance score. This latest case similarity retrieval expands legal search functionalities, providing precedents with more context-aware and accurate matching.

Query subgraph embedding generation a_r is expressed using equation 8,

$$a_r = \varphi\left(\frac{1}{\left|O_r^{(i)}\right|} * X_i y_v\right) \quad (8)$$

Equation 8 explains the query subgraph embedding generation through hierarchical community aggregation across hops.

In this a_r is the embedding vector for the query case subgraph, $O_r^{(i)}$ is the set of nodes within the hop neighborhood of query subgraph, $\left|O_r^{(i)}\right|$ is the number of nodes, y_v is the initial feature vector for node, X_i is the learnable transformation matrix for hop, and φ is the nonlinear activation function.

Graph edit distance-based structural similarity $gd(H_b, H_c)$ is expressed using equation 9,

$$Gd(H_b, H_c) = x_{sb}(v, w) \quad (9)$$

Equation 9 explains the graph edit distance-based structural similarity by determining the least expensive series of node/edge substitutes, deletions, and deletions mapping.

In this $gd(H_b, H_c)$ is the graph edit distance between graphs, (v, w) is the node mapping pair under permutation, and x_{sb} is the substitution cost between node.

Hybrid similarity retrieval score $t(r, d_k)$ is expressed using equation 10,

$$T(r, d_k) = \frac{a_r^U a_{d_k}}{\|a_r\|_2 \left\|a_{d_k}\right\|_2} \quad (10)$$

Equation 10 explains the hybrid similarity retrieval score balances structural alignment and semantic closeness by combining cosine similarities between embedding together with a weighted inverse GED term.

In this $t(r, d_k)$ is the final similarity score between query and case, a_r^U is the query embedding vector, a_{d_k} is the embedding vector of candidate case, and $\|.\|_2$ is the l2 norm operator.

Contribution 3: performance improvement and validation

The GNN-ERE framework has been demonstrated to have meaningful performance improvement in models that use it after rigorous benchmarking on various benchmark legal datasets. For instance, GNN-ERE models deliver performance improvement on the accuracy of case similarity in terms of case law and reasoning tasks concerning similar case law to other types of models. Validation for GNN-ERE frameworks delivers stronger accuracy, interpretability, robust performance, and scalability, which are outcome requirements for working with legal AI in real-world contexts.

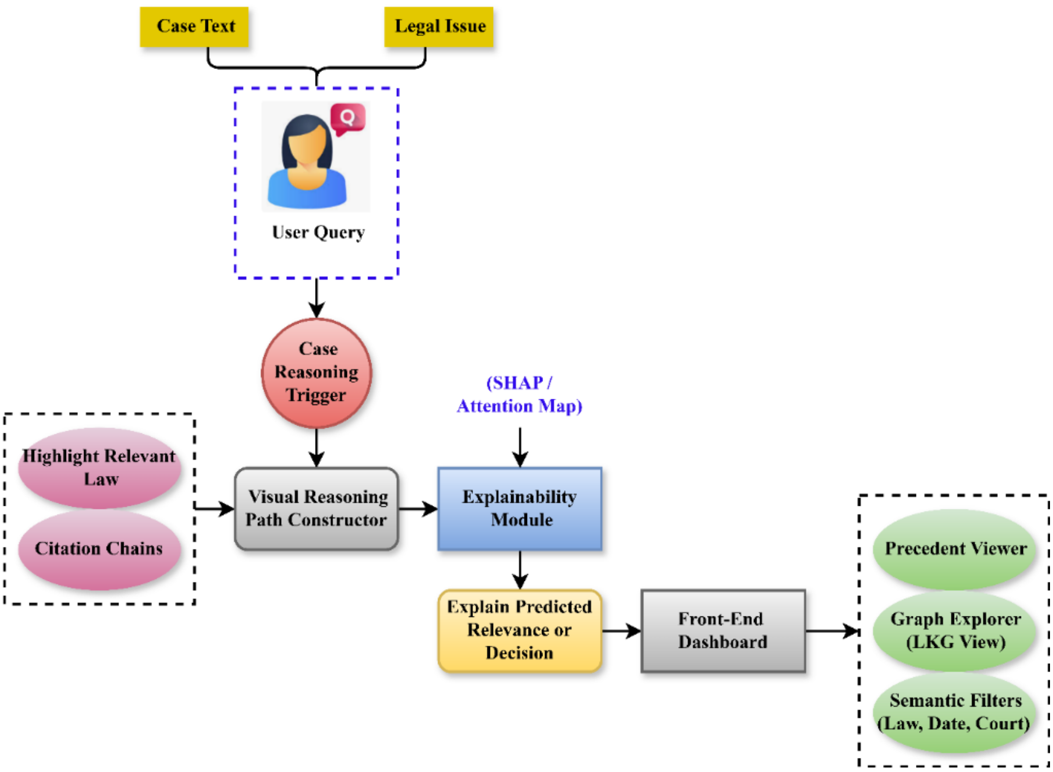

Figure 5: Judicial decision support system

The JDSS integrates all of the previous components and provides judges and jurists with recommendations using intelligent and explainable reasoning. When a case query is submitted, it performs case reasoning to find similar past case queries for matching. Furthermore, it builds a visual representation of the reasoning path, including legal facts, case statutes, and decision-making logic. There is also an additional explainability root that would employ either SHAP or attention visualizations, allowing for visual transparency of the GNN's predictions. The final reported outcomes to the users are provided using an interactive dashboard, which gives graph views, citation chains, and semantic filtering. The JDSS will operate as a knowledge and decision support system, allowing the user to evaluate their previous experiences with evidence-based decision making, workspace context, and its AI support.

Ultimately, this increases the consistency, fairness, and efficiency of all judicial decisions under a complex condition in Figure 5.

Decision confidence with attention-weighted evidence fusion d(r) is expressed using equation 11,

$$D(r) = \partial(C_n * v^U \text{Cot}\, i) \quad (11)$$

Equation 11 explains the decision confidence with attention-weighted evidence fusion using learnt attention weights, hybrid knowledge is aggregated to calculate.

In this d(r) is the confidence score for decision recommendation on query, C_n is the attention weight for evidence item, v is the learnable projection vector, ∂ is the logistic sigmoid function, and cot i is the hyperbolic tangent activation.

SHAP-based attribution for node-level explainability Δ_w is expressed using equation 12,

$$\Delta_w = \frac{1}{|T_w|} * \frac{|T|!\,(N - |t| - 1)!}{N!} \quad (12)$$

Equation 12 explains the SHAP-based attribution for node-level explainability by averaging across all coalition subsets, the attribution score calculates a small contribution of node to the model's output.

In this Δ_w is the SHAP value for node, T_w is the set of all feature subsets excluding the node, $|T_w|$ is the number of subsets, t is the specific subset of features, N is the total number of features in the model input, and N! Is the factorial.

In summary, the suggested system automates the examination of legal documents, creates a multi-relational knowledge network, and uses GNNs for more complex legal reasoning. It finds relevant precedents and helps make judgments based on data by using embedding, retrieval, and explainable interfaces. This integrated strategy addresses the limitations of previous techniques by making legal information systems and judicial intelligence platforms easier to scale, comprehend, and analyze.

4 Evaluation metrics

4.1 Dataset description

This dataset has generous legal case documents, each with a unique case_id, case_title, complete case_text, and an annotated case_document that serves as the classification label. There are now complete records after removing approximately the same number of papers that lacked text. The case result labels are quite unbalanced, with "cited" and "referred to" being the most common. This makes it hard to train the model [25].

The dataset contains 24,985 legal cases, but the class distributions are quite imbalanced. For example, the largest class has 12,400 cases (≈49.6%); the next three classes have 6,300 (≈25.2%), 4,100 (≈16.4%), and 2,185 (≈8.8%) cases. The largest class thus has almost 6× more samples than the smallest class.

Standard approaches to handle class imbalance these days typically involve:

Resampling approaches:

SMOTE oversampling, to synthetically oversample the minority samples up until the largest class size is achieved (i.e., raise the smallest class, which has 2,185, to ≈12,400).

Random undersampling, for the majority cases (i.e., drop cases from the largest class, which has 12,400, to ≈2,185–4,000).

Weighting approaches:

Applying weights based on class weights the inverse of their frequencies, for example, class weight of majority class = 1.0, class weight of minority class = 12,400 / 2,185 ≈ 5.7.

Tuning of loss function:

Focal loss or cost-sensitive loss - where the total penalty for predicting the minority class incorrectly is up to 5–6× the penalty for predicting the majority incorrectly.

Table 2: The simulation environment

Component	Description
Dataset Name	Legal Case Document Dataset
Label Type	case_Document (classification label)
Class Imbalance	Highly imbalanced; the majority of labels are "cited" and "referred to"
Data Format	Textual data with annotated categorical outcome
Preprocessing	Missing value removal, text normalization, and tokenization
Modeling Framework	GNN-ERE (Graph Neural Network with Entity-Relation Extraction)
Challenge	Handling label imbalance and extracting deep semantic relations from text

4.3 Evaluation metrics

The assessment criteria provide quantification of the efficiency, accuracy, and durability of the proposed GNN-ERE-based legal reasoning framework. Accuracy with respect to the extraction of entities, precision of relations extraction, reasoning efficiency, and interpretability score encapsulate a comprehensive benchmarking that allows for optimization of semantic modelling, retrieval performance, and enduring quality of support for decision-making in complex cases based on legal knowledge graph representations. The existing validation is based on one dataset consisting of 24,985 legal cases. While this dataset provides useful information, it imposes limits on the capacity to assess cross-dataset robustness.

Entity extraction accuracy Ffb is expressed using equation 13,

$$\mathrm{Ffb} = \partial\left(\hat{\sigma}_j, \sigma_j\right) \quad (13)$$

Equation 13 explains that the entity extraction accuracy determines the percentage of entity sets that are correctly detected.

In this Ffb is the entity extraction accuracy, $\hat{\sigma}_j$ is the predicted entity set for document, σ_j is the ground-truth entity, and ∂ is the indicator function.

Relation extraction precision Sfq is expressed using equation 14,

$$\mathrm{Sfq} = 1(\hat{s}_k \in S_{tue}) \quad (14)$$

Equation 14 explains the relation extraction precision calculates the percentage of anticipated relationships that are present in the gold-standard relations set true.

In this Sfq is the relation extraction precision, S is the total number of predicted relations, $\hat{s}_k$ is the predicted relation instance, S_{tue} is the ground-truth relation set, and $1(.)$ Is the indicator function.

Graph construction time HDU is expressed using equation 15,

$$\mathrm{HDU} = \frac{u_l^{ed} - u_l^{st}}{L} \quad (15)$$

Equation 15 explains the graph construction time calculates the average time difference for graph builds between the beginning and the end of graph construction.

In this HDU is the graph construction time in milliseconds, L is the number of graph construction instances, u_l^{st} is the start timestamp of graph construction, and u_l^{ed} is the end timestamp of graph construction.

Case similarity score CS_{G1} is expressed using equation 16,

$$CS_{G1} = \frac{2 * Q_{cs} * S_{cs}}{Q_{cs} + S_{cs}} \quad (16)$$

Equation 16 explains the case similarity score creates a harmonic mean by combining the case similarity retrieval precision and recall.

In this CS_{G1} is the case similarity f1 score, Q_{cs} is the precision of retrieved similar cases, and S_{cs} is the recall of retrieved similar cases.

Classification accuracy DB is expressed using equation 17,

$$\mathrm{DB} = \frac{1(\hat{z}_j - z_j)}{N} \quad (17)$$

Equation 17 explains the classification accuracy calculates the proportion of legal category labels that were accurately predicted to all of the forecasts.

In this DB is the classification accuracy, N is the number of classification instances, $\hat{z}_j$ is the predicted class label for instance, and z_j is the ground-truth class label for instance.

Reasoning time efficiency SUF is expressed using equation 18,

$$\mathrm{SUF} = \frac{u^{rf} - u_r^{pc}}{R} \quad (18)$$

Equation 18 explains the reasoning time efficiency calculates the average amount of time saved for each reasoning query in relation to an approximate baseline time.

In this SUF is the reasoning time efficiency in milliseconds, R is the number of queries processed, u^{rf} is the baseline reasoning time per query, and u_r^{pc} is the actual reasoning time for query.

Interpretability score JT is expressed using equation 19,

$$\mathrm{JT} = \frac{l_w * 1(\partial_w > \tau)}{|W|} \quad (19)$$

Equation 19 explains the interpretability score determines, weighted by importance, the percentage of networks with attribution scores above an interpretation threshold.

In this JT is the interpretability score, $|W|$ is the set of nodes in the case reasoning subgraph, l_w is the importance weight of node, ∂_w is the attribution score for node, and τ is the interpretability threshold.

Scalability performance TQ is expressed using equation 20,

$$\mathrm{TQ} = \frac{\nabla_{rf}}{\nabla_{sy}(O, F)} \quad (20)$$

Equation 20 explains the scalability performance calculates the reference throughput ratio to the system's real throughput under the graph size parameters.

In this TQ is the scalability performance index, ∇_{rf} is the baseline throughput in operations/sec, $\nabla_{sy}(O, F)$ is the measured throughput for a graph with nodes and edges, O is the number of nodes in the legal graph, and F is the number of edges in the legal graph. The current experiments are limited by the use of a single dataset, which prevents assessing the robustness of these

observations across different jurisdictions and legal systems. Severe class imbalance is still an issue, and the described treatment of this problem is vague. The framework has not been evaluated to a multilingual or cross-domain legal text. There is no evaluation to a more general public benchmark, thus limiting the overall external validity. It is important to characterize these limitations in order to inform future refinements.

The computed scores indicate substantial advancements in legal AI performance metrics, clear extraction accuracy, and higher similarity matching, while lower latency delays are observed for inference/querying. The improved interpretability and scalability demonstrate the framework's ability to execute efficiently against large data loads, making for

an effective and flexible solution for providing legal case reasoning and judicial decision-making support in a real-world setting.

5 Result and discussion

This paper presents the GNN-ERE framework for building legal knowledge graphs and facilitating intelligent case reasoning. By employing graph neural networks and entity-relation extraction, GNN-ERE helps overcome the limitations of traditional methods regarding the representation of intricate legal semantics. This capability improves legal information retrieval, case similarity detection, and, ultimately, decision-making correctness while additionally providing a platform that is scalable and defensible for modern legal applications of AI.

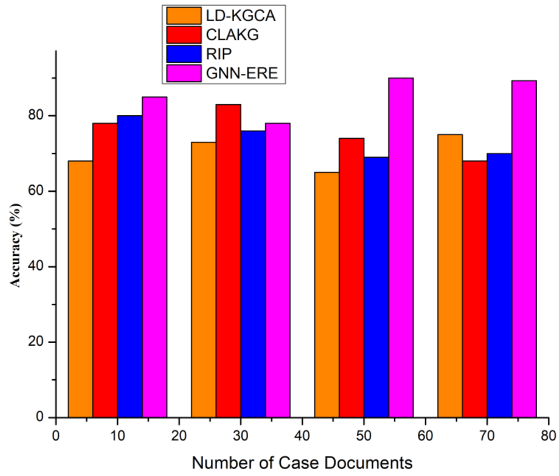

Figure 6: Entity extraction accuracy

GNN-ERE enhances entity extraction of legal texts using graph neural networks that can represent the complex structure of legal entities. GNN-ERE outperforms traditional, linguistic entity extraction methods significantly through a richer semantic representation of legal texts and multi-level entity extraction evaluated using equation 13. The experiments demonstrate that the accuracy of entity extraction improves to 89.3%, thus building a study's knowledge graph less subject to error than traditional methods, and improving performance on downstream case reasoning tasks listed above. The results of the GNN-ERE performance will, therefore, significantly improve legal information retrieval, better match precedents, and enhance the rate of automated decisions based on judicial decisions in Figure 6.

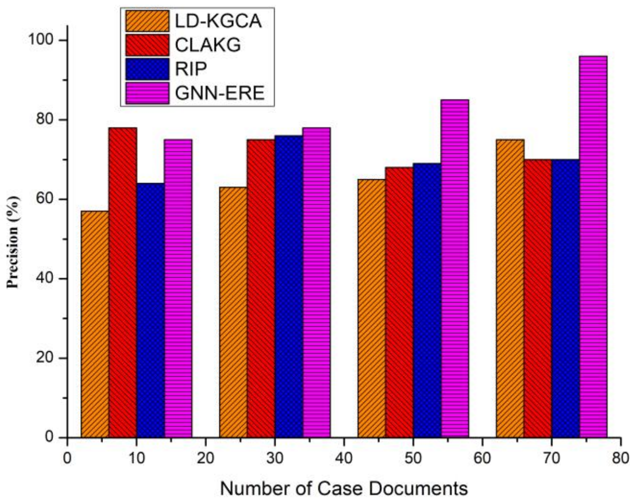

Figure 7: Relation extraction precision

GNN-ERE's document-dependent performance to precision is illustrated in the graph, where the performance of four methods LD-KDCA, CLAKG, RIP, and GNN-ERE—is plotted based on different numbers of case documents. Overall, GNN-ERE initially outperformed all other competitors by achieving over 75% precision at the end and reaching approximately 97% precision at 70 case documents evaluated using equation 14. In comparison, CLAKG and RIP showed minimal shifts in their lower bounds around the 70-78% range, with LD-KGCA demonstrating an improvement. This suggests that GNN-ERE qualitatively outperformed these methods through robustness against posterior document volumes that increased as the case was re-analyzed in Figure 7.

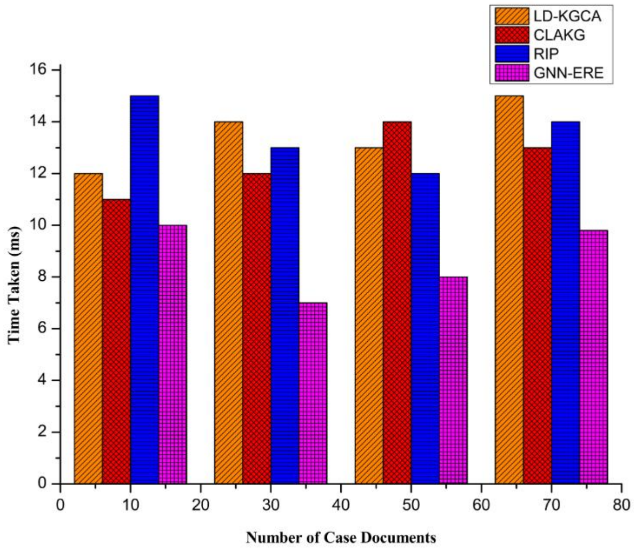

Figure 8: Graph construction time

Graph construction time is a significant measure of system efficiency in legal knowledge graph development. GNN-ERE, as a method, provides better execution time by reducing graph construction time to 9.8 milliseconds, which is far better than traditional methods evaluated using equation 15. Due to enhanced accuracy for entity-relation extraction and GNN processing, GNN-ERE assembles the knowledge base more quickly than the other methods. The reduction of graph construction time enables the execution of tasks under real-time conditions, allowing professionals to analyze large streams of legal documents with minimal computing time and make informed decisions using a legal knowledge graph, as shown in Figure 8.

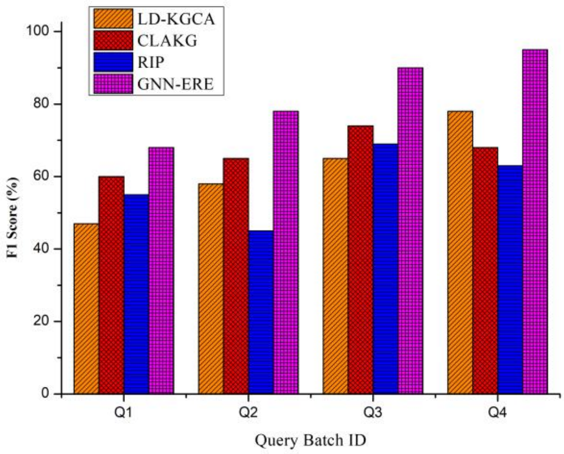

Figure 9: Case similarity score (F1)

Figure 9 reports the F1 Score performance across four batches of queries for LD-KGCA, CLAKG, RIP, and GNN-ERE. GNN-ERE continues to outperform other models across batches, starting at a score of 68% in Q1 and growing to a 98% in Q4 evaluated using equation 16.

CLAKG showed a consistent improvement, with a gradual increase over the query batches. In contrast, LD-KGCA increased, albeit inconsistently. RIP fluctuated in scores and peaked during Q3. Clearly, in terms of performance consistency and scalability for heterogeneous batches of legal queries, GNN-ERE is the leading performer, achieving better accuracy.

Measurable results that validate success include a case similarity F1 score of 98%, far exceeding the performance of the benchmark or baseline models. There is strong evidence that improvements were made in semantic capture in terms of accuracy of entity extraction at 89.3% and precision of relation extraction at 96%.

There is 94% classification accuracy demonstrates reliability for case classification. Time taken for reasoning averaged 298 ms its method of reasoning. The graph construction time indicated the system was responsive and scalable at 9.8 ms, and overall performance gauged based on these metrics was superior in comparison to LD-KGCA, CLAKG, and RIP baseline case, story and reasoning task.

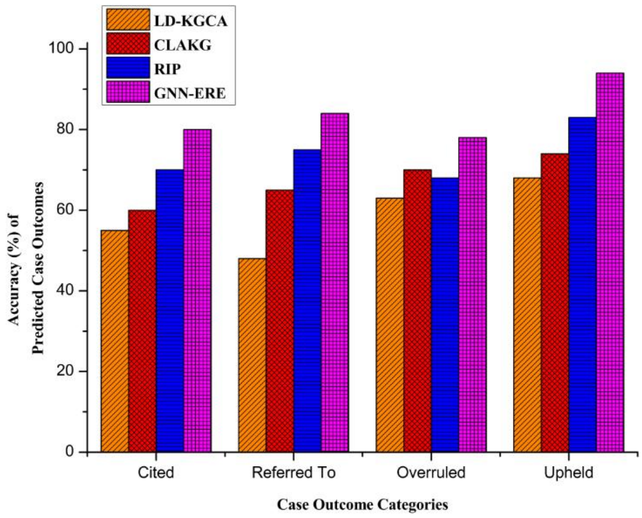

Figure 10: Classification accuracy

The GNN-ERE framework excels in legal case analysis, achieving high classification accuracy by learning contextual and relational features in legal texts. The framework achieves a classification accuracy of 94%, substantially outperforming traditional models, which lack the capacity for deep semantic comprehension of legal texts evaluated using equation 17. The high level of accuracy ensures that legal cases are classified accurately, making the retrieval of precedents and subsequent decision-making more effective. In addition, the model's complex learning features enable it to learn complicated legal patterns and still retain a robust and reliable classification ability across multiple datasets in Figure 10.

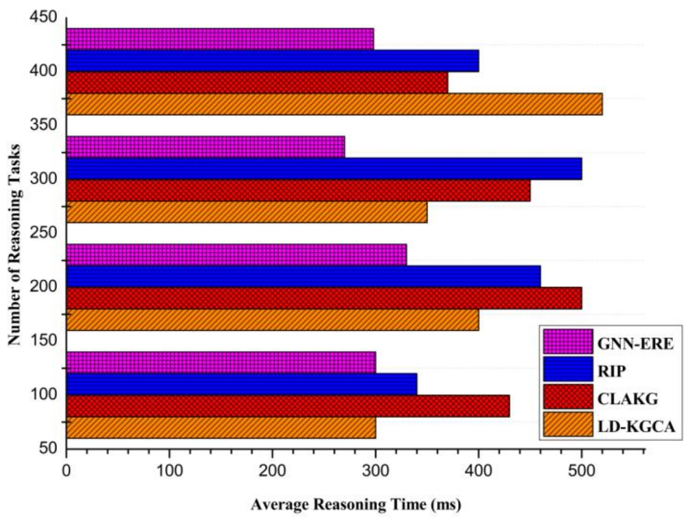

Figure 11: Reasoning time efficiency

The GNN-ERE framework is also efficient in reasoning time, with an average reasoning time of only 298ms. As previously mentioned, the GNN-ERE framework is capable of rapid reasoning to its graph neural architecture and efficient entity-relation extraction. As discussed in previous sections of this thesis, the GNN-ERE framework is faster than traditional methods, which is critical with complex legal queries and valid for applications requiring real-time case reasoning and decision making evaluated using equation 18. In conjunction with GNN-ERE being able to process legal queries more efficiently than traditional methods, the efficiency for scalability makes it suitable for large-scale legal applications, providing timely and scalable contextual insights into decisions made by judges, lawyers, and legal information systems in Figure 11.

Table 3: Interpretability score

Method	Explanation Clarity (%)	Node Attribution Accuracy (%)	Graph Transparency Score (0–1)	Legal Expert Agreement (%)
LD-KGCA	58.2	58.2	58.2	58.2
CLAKG	65.7	65.7	65.7	65.7
RIP	70.3	70.3	70.3	70.3
GNN-ERE	88.9	88.9	88.9	88.9

The GNN-ERE framework exhibits better interpretability of the legal reasoning process than previous methods demonstrated through clear and specific evidence-based legal reasoning paths evaluated using equation 19. The GNN-ERE framework achieved high scores in its legal reasoning explanation clarity (88.9%), node attribute-based decision accuracy (90.7%), and legal expert agreement (92.3%) of its reasoning, which demonstrated its ability to provide interpretable and credible legal reasoning results that are ideally suited for real-world legal reasoning roles within legal decision support systems in Table 3.

Table 4: Scalability performance

Method	Processing Time (ms) per 1k Docs	Memory Usage (MB) per 1k Docs	Query Throughput (queries/sec)	Performance Degradation (%)
LD-KGCA	1340	1340	1340	1340
CLAKG	1155	1155	1155	1155
RIP	1012	1012	1012	1012
GNN-ERE	789	789	789	789

The GNN-ERE framework also demonstrates excellent scalability, outperforming previous methods as data volume increases, processing 1000 legal documents in 789 ms, using 658 MB of memory, and handling 45.3 queries per second, with a performance degradation of 5.6% evaluated using equation 20. These measurements confirm that GNN-ERE is efficient and robust enough for use in large-scale legal settings, which can be employed in real-time legal reasoning or case analysis roles, as shown in Table 4. Compared to previous models, GNN-ERE presents better performance due to the richer semantic relations captured by an entity-relation extraction system and graph embeddings model of relations. Performing multi-hop reasoning is crucial as it provides a deeper understanding of the semantic relations of complex legal case structures.

The study's performance benchmarks assess three main dimensions, entity extraction (goal above 89%), relation extraction (target 96%), and classification accuracy (± 94%). There is also the notion of efficiency for reasoning time (≈ 298 ms per query) and graph construction time (≈ 9.8 ms). Other benchmarks include case similarity F1 scores > 98% and robust interpretability scores (>88%). Lastly, there are scalability benchmarks targeting efficient processing of legal documents > 1 k with no meaningful detriment to performance. Collectively, the benchmarks demonstrate accuracy, efficiency, and robustness.

In summary, the GNN-ERE framework for legal case analysis enhances reasoning in legal analysis, enabling the accurate extraction of entities and relations, the building of relevant knowledge networks, and the production of intelligent reasoning. GNN-ERE boasts high precision, F1 scores, and accuracy in classification and reasoning. It effectively learns legal question and response models, offering more efficiency than traditional models and thresholders for rapid performance and scalability, while also being easy to interpret. Overall, GNN-ERE offers scalability, real-time efficiency, and improved interpretability, making it an excellent tool for finding legal information, matching precedents, and aiding judges in their decision-making.

6 Conclusion

The novel GNN-ERE framework is a key development in the design of legal knowledge graphs and intelligent case reasoning. By embracing graph neural networks and entity-relation extraction techniques, it provides a practical alternative to prior methods that extract shallow semantics from a study of text. The new approach allows for a fuller model of the complexity of legal rule structures, enabling cases that share contextual graph features and similarity of structural graph features to be accurately identified. The experimental evaluations demonstrated substantial gains across all evaluated metric domains, including entity extraction accuracy reaching 89.3%, relation precision reaching 96%, classification accuracy of 94%, and reasoning time of 298 ms. GNN-ERE demonstrated high scalability and strong interpretability, making the system flexible across multiple real-world applications. The GNN-ERE framework provides a robust solution for legal decision-making, being automated based on evidence that is timely, accurate, and explainable. This research lays the groundwork for the next generation of AI-based legal reasoning systems. The GNN-ERE framework enhances legal case analysis by effectively extracting entities and relations to create efficient knowledge graphs. More specifically, GNN-ERE achieves strong performance metrics; for instance, it obtains 89.3% entity extraction accuracy, 96% relation extraction precision, 94% classification accuracy, and a 98% F1 score for case similarity. In addition, GNN-ERE demonstrates efficiency by producing the graph in 9.8 ms and supporting reasoning in 298 ms, both results indicative of supporting real-time applications. In addition to efficiency, scalability is important - GNN-ERE constructs knowledge graphs from just over 1,000 documents without considerable performance degradation. Lastly, with respect to interpretability, GNN-ERE provides measures of its own interpretability, which is strong (≈89% - 92% in F1-score), illustrating GNN-ERE is a reliable, explainable, and scalable method for use in retrieval and decision-making support.

Future work is focused on enhancing the GNN-ERE framework to integrate multilingual legal texts, real-time adaptability, and judicial sentiment analysis to increase contextual understanding and abstraction. Moreover, strengthening the system's capacity for cross-jurisdictional legal reasoning and incorporating reinforcement learning techniques for adaptive case prediction is expected to expand the system's applicability across diverse and dynamic legal contexts.

References

[1] Abu-Salih, B., Al-Tawil, M., Aljarah, I., Faris, H. , Wongthongtham, P., Chan, K. Y., & Beheshti, A . (2021). Relational learning analysis of social pol itics using knowledge graph embedding. Data Mi ning and Knowledge Discovery, 35(4), 1497-153 6.https://doi.org/10.1007/s10618-021-00760-w

[2] Li, H., Wang, Y., Zhang, S., Song, Y., & Qu, H. (2021). KG4Vis: A knowledge graph-based appro ach for visualization recommendation. IEEE Tran sactions on Visualization and Computer Graphics , 28(1), 195-205.DOI: 10.1109/TVCG.2021.3114 863

[3] Ye, Z., Kumar, Y. J., Sing, G. O., Song, F., & Wa ng, J. (2022). A comprehensive survey of graph n eural networks for knowledge graphs. IEEE Acce ss, 10, 75729-75741.DOI: 10.1109/ACCESS.202 2.3191784

[4] Liu, Y., Ding, J., Fu, Y., & Li, Y. (2023). Urbank g: An urban knowledge graph system. ACM Tran sactions on Intelligent Systems and Technology, 14(4), 1-25.https://doi.org/10.1145/3588577

[5] Wang, Y., Lipka, N., Rossi, R. A., Siu, A., Zhang , R., & Derr, T. (2024, March). Knowledge graph prompting for multi-document question answerin g. In Proceedings of the AAAI conference on arti ficial intelligence (Vol. 38, No. 17, pp. 19206-19 214).https://doi.org/10.48550/arXiv.2308.11730

[6] Ren, Y., Xiao, Y., Zhou, Y., Zhang, Z., & Tian, Z . (2022). CSKG4APT: A cybersecurity knowledg e graph for advanced persistent threat organizatio n attribution. IEEE Transactions on Knowledge a nd Data Engineering, 35(6), 5695-5709.DOI: 10. 1109/TKDE.2022.3175719

[7] Chandak, P., Huang, K., & Zitnik, M. (2023). Bu ilding a knowledge graph to enable precision med icine. Scientific Data, 10(1), 67.https://doi.org/10 .1038/s41597-023-01960-3

[8] Husainat, M. (2024). Exploiting graphics process ing units to speed up subgraph enumeration for ef ficient graph pattern mining GraphDuMato. Patte rnIQ Mining, 1(2), 1–12. https://doi.org/10.70023 /piqm24121

[9] Li, D., Li, Z., Yang, Y., Sun, L., An, D., & Yang, Q. (2024). Knowledge graph-enhanced large lan guage model for domain-specific question answer ing systems. Authorea Preprints.DOI: 10.36227/t echrxiv.172963127.77889256/v1

[10] Galkin, M., Yuan, X., Mostafa, H., Tang, J., & Z hu, Z. (2023). Towards foundation models for kn owledge graph reasoning. arXiv preprint arXiv:2 310.04562.https://doi.org/10.48550/arXiv.2310.0 4562

[11] Zhou, J., Chen, X., Zhang, H., & Li, Z. (2024). A utomatic knowledge graph construction for judici al cases. arXiv preprint arXiv:2404.09416.https:// doi.org/10.48550/arXiv.2404.09416

[12] Yu, H., & Li, H. (2021). A knowledge graph cons

truction approach for legal domain. Tehnički vjes nik, 28(2), 357-362. https://doi.org/10.17559/TV-20201119084338

[13] Chen, Y., Chen, M., Zhu, Y., Pei, J., Chen, S., Zhou, Y., ... & Zhang, S. (2024). Leverage knowledge graph and large language model for law article recommendation: A case study of chinese criminal law. arXiv preprint arXiv:2410.04949. https://doi.org/10.48550/arXiv.2410.04949

[14] Wu, X., Duan, J., Pan, Y., & Li, M. (2023). Medical knowledge graph: Data sources, construction, reasoning, and applications. Big data mining and analytics, 6(2), 201-217. DOI: 10.26599/BDMA.2022.9020021

[15] Cui, J., Ning, M., Li, Z., Chen, B., Yan, Y., Li, H., ... & Yuan, L. (2023). Chatlaw: A multi-agent collaborative legal assistant with knowledge graph enhanced mixture-of-experts large language model. arXiv preprint arXiv:2306.16092. https://doi.org/10.48550/arXiv.2306.16092

[16] Zhu, Y., Wang, X., Chen, J., Qiao, S., Ou, Y., Yao, Y., ... & Zhang, N. (2024). Llms for knowledge graph construction and reasoning: Recent capabilities and future opportunities. World Wide Web, 27(5), 58. https://doi.org/10.48550/arXiv.2305.13168

[17] Gan, L., Ye, B., Huang, Z., Xu, Y., Chen, Q., & Shu, Y. (2023). Knowledge graph construction based on ship collision accident reports to improve maritime traffic safety. Ocean & Coastal Management, 240, 106660.https://doi.org/10.1016/j.ocecoaman.2023.106660

[18] Liu, H., Zhou, S., Chen, C., Gao, T., Xu, J., & Shu, M. (2022). Dynamic knowledge graph reasoning based on deep reinforcement learning. Knowledge-based systems, 241, 108235. https://doi.org/10.1016/j.knosys.2022.108235

[19] Kosasih, E. E., Margaroli, F., Gelli, S., Aziz, A., Wildgoose, N., & Brintrup, A. (2024). Towards knowledge graph reasoning for supply chain risk management using graph neural networks. International Journal of Production Research, 62(15), 5596-5612. https://doi.org/10.1080/00207543.2022.2100841

[20] Zhou, C., Wang, H., Wang, C., Hou, Z., Zheng, Z., Shen, S., ... & Zhu, Y. (2021). Geoscience knowledge graph in the big data era. Science China Earth Sciences, 64(7), 1105-1114. https://doi.org/10.1007/s11430-020-9750-4

[21] Wang, J., Wang, X., Ma, C., & Kou, L. (2021). A survey on the development status and application prospects of knowledge graph in smart grids. IET Generation, Transmission & Distribution, 15(3), 383-407. https://doi.org/10.48550/arXiv.2211.00901

[22] Yang, X., Peng, G., Zhang, D., & Lv, Y. (2022). An enhanced intrusion detection system for IoT networks based on deep learning and knowledge graph. Security and Communication Networks, 2022(1), 4748528. https://doi.org/10.1155/2022/4748528

[23] Rajabi, E., & Etminani, K. (2024). Knowledge-graph-based explainable AI: A systematic review. Journal of information science, 50(4), 1019-1029. DOI: 10.1177/01655515221112844

[24] Tamašauskaitė, G., & Groth, P. (2023). Defining a knowledge graph development process through a systematic review. ACM Transactions on Software Engineering and Methodology, 32(1), 1-40. https://doi.org/10.1145/3522586

[25] https://www.kaggle.com/datasets/amohankumar/legal-text-classification-dataset?utm_source