Logic Rules as Explanations for Legal Case Retrieval

Zhongxiang Sun[1], Kepu Zhang[1], Weijie Yu[2*], Haoyu Wang[1], Jun Xu[1]
[1]Gaoling School of Artificial Intelligence, Renmin University of China
[2]School of Information Technology and Management, University of International Business and Economics
{sunzhongxiang, kepuzhang}@ruc.edu.cn, yu@uibe.edu.cn, {wanghaoyu0924, junxu}@ruc.edu.cn

Abstract

In this paper, we address the issue of using logic rules to explain the results from legal case retrieval. The task is critical to legal case retrieval because the users (e.g., lawyers or judges) are highly specialized and require the system to provide logical, faithful, and interpretable explanations before making legal decisions. Recently, research efforts have been made to learn explainable legal case retrieval models. However, these methods usually select rationales (key sentences) from the legal cases as explanations, failing to provide faithful and logically correct explanations. In this paper, we propose **N**eural-**S**ymbolic enhanced **L**egal **C**ase **R**etrieval (**NS-LCR**), a framework that explicitly conducts reasoning on the matching of legal cases through learning case-level and law-level logic rules. The learned rules are then integrated into the retrieval process in a neuro-symbolic manner. Benefiting from the logic and interpretable nature of the logic rules, NS-LCR is equipped with built-in faithful explainability. We also show that NS-LCR is a model-agnostic framework that can be plugged in for multiple legal retrieval models. To showcase NS-LCR's superiority, we enhance existing benchmarks by adding manually annotated logic rules and introducing a novel explainability metric using Large Language Models (LLMs). Our comprehensive experiments reveal NS-LCR's effectiveness for ranking, alongside its proficiency in delivering reliable explanations for legal case retrieval.

Keywords: Legal Applications, Information Retrieval, Explainability

1. Introduction

Legal case retrieval retrieves relevant cases from a query and is a specialized Information Retrieval task. Due to its vital role in aiding legal practitioners, logical explanations for retrieved cases are essential. Only retrieved cases with accurate logical reasoning can serve as persuasive evidence for legal decisions (Prakken and Sartor, 2015).

Deep learning advances have improved semantic retrieval of legal cases (Shao et al., 2020; Xiao et al., 2021; Qin et al., 2023). Most retrieval models focus on estimating the relevance scores of a target case given the query (shown as the Lawformer tab in Figure 1). Additionally, in response to the need for explainability in legal case retrieval, Yu et al. (2022c) proposed IOT-Match, which generates explanations by extracting rationales (key sentences) from both query and target cases (shown as the IOT-Match tab in Figure 1). However, IOT-Match cannot provide the users with an explicit logic reasoning process on whether the query and candidate cases are relevant or not. Furthermore, these explanations focus only on case facts, overlooking law articles' significance in assessing query and candidate case relevance (Sun et al., 2023a).

Recently, some studies have used logic for explanation. Lee et al. (2022) have demonstrated the effectiveness of learning rules from data for explanation. Furthermore, logic rule-based explanations surpass prior methods in human precision (Alvarez Melis and Jaakkola, 2018). Ciravegna et al. (2023) proposed a unique type of concept-based neural network that provides first-order logic explanations for decision-making. Though these methods have shown promise in providing explanations for tasks in general domains, they cannot be directly adapted to legal retrieval because, in the legal domain, it is required that the judges make decisions based not only on case documents but also on law articles. Moreover, some studies highlight law articles' role in enhancing judgment prediction (Zhong et al., 2018) and legal case matching (Sun et al., 2023a). How to incorporate the corresponding law articles in explicit logic reasoning is important while under-explored problems. It is expected that legal case retrieval models should explain decisions using logic rules from both cases and law articles, as shown in Figure 1.

To tackle these issues, we proposes a model-agnostic framework called NS-LCR which learns logic rules from the query and target cases as the explanations for retrieved legal cases. Unlike studies that solely rely on text semantics for relevance scores, NS-LCR uses two neuro-symbolic modules to learn law-level and case-level logic rules. Specifically, the law-level module forms first-order-logic (FOL) rules for each target case, extracting predicates from the case based on laws and connecting them with logic operations. Then, the legal relevance prediction is formalized as the fine-grained evaluation between the query and the FOL rule, which can be efficiently induced by fuzzy logic such as Łukasiewicz T-norm (Klement et al., 2013). The case-level module forms the relevance rules by

Corresponding author: Weijie Yu.

LREC-COLING 2024, pages 10747–10759

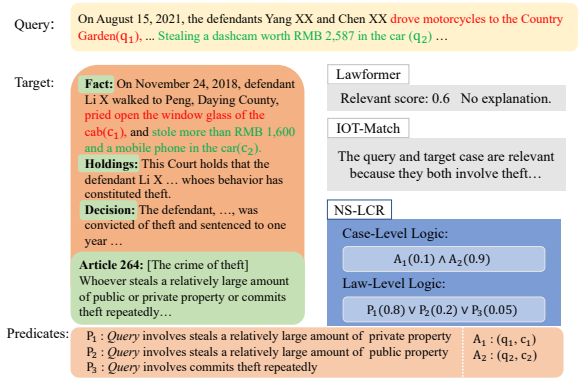

Figure 1: Explanations provided by different legal case retrieval models. Semantic models (e.g., Lawformer) only estimate the matching score. Existing explainable methods (e.g., IOT-Match) provide sentences as explanations. NS-LCR aims to explain matching results with case and law logic rules. Article 264 of PRC Criminal Law, with three key facts P_1, P_2, and P_3, applies to the target case.

identifying the aligned sentences from the query and target cases. The learned relevance rules also work in a fuzzy logic fashion and provide evidence for the relevance prediction. Benefiting from these modules, NS-LCR not only provides the learned logic rules as explanations but also effectively improves the underlying retrieval model's performance in low-resource situations. We take four well-known legal retrieval baselines as the underlying model of NS-LCR and conduct extensive experiments of high-resource and low-resource legal retrieval performance on LeCaRD (Ma et al., 2021) and ELAM (Yu et al., 2022c). To evaluate explanations, we use Large Language Models to check the effectiveness when applied to downstream tasks.

We summarize our contributions as follows: (1) We analyze the importance of explicit logic reasoning in legal case retrieval. We further show that the law-level and case-level logic rules are critical in explaining the retrieved cases. (2) We propose a novel neural symbolic enhanced framework (**NS-LCR**) for explainable legal case retrieval by representing law articles and case documents into logic rules and involving the logic rules in legal case retrieval. (3) We tested on two datasets and introduced a new **LLM-based** evaluation method. Results showed NS-LCR enhanced model performance and validated the importance of case and law-level rules in legal retrieval.

2. Related Work

2.1. Legal Case Retrieval

Traditional techniques emphasized legal issue decomposition and ontology construction (Bench-Capon et al., 2012; Saravanan et al., 2009), while recent advancements segment into network and text-based methods. Network-based strategies, such as the Precedent Citation Network (PCNet) and Hier-SPCNet, focus on citation clustering and domain encapsulation to assess case similarity (Minocha et al., 2015; Bhattacharya et al., 2020). On the other hand, text-based methods leverage semantic analysis of case texts, with innovations like BERT-PLI, OPT-Match and Lawformer modeling paragraph interactions and specializing in case analysis (Shao et al., 2020; Yu et al., 2022b; Xiao et al., 2021). Despite improvements, the explicability of these models remains a challenge, highlighted by the introduction of a tri-stage explainable model by Yu et al. (2022c); Sun et al. (2023b), which still falls short in logical reasoning (Jain and Wallace, 2019).

2.2. Logic as Explanation

Studies indicate logic explains prediction results. Works are classified into concept-based explanations with predicates from concept set inputs, and data-based explanations that learn rules from data. Barbiero et al. (2022) used an entropy criterion to identify relevant concepts and extract First-Order Logic explanations from neural networks. Ciravegna et al. (2023) introduced LENs that predict output concepts and provide First-Order Logic explanations based on input concepts. Jain et al. (2022) enhanced LEN by testing perturbed input words on text classification. Wu et al. (2021) improved Natural Language Inference explainability by aligning detected phrases in sentences. Aligned units form weakly supervised logic reasoning. Lee et al. (2022) developed a framework using human priors to learn logic rules from data. Learned rules explain deep models' output. Feng et al. (2022b) presented a framework combining reinforcement learning and introspective revision for improved reasoning in natural language inference tasks. Our study advances beyond existing approaches by focusing on the legal domain, leveraging the logical structure of the civil law system and the semi-structured nature of legal documents. We introduce the NS-LCR model, which generates more precise and detailed explanations, improving effectiveness in legal case retrieval compared to methods designed for general domains.

3. Background and Preliminaries

3.1. Task Formulation

Suppose that we have a set of collected samples $\mathcal{D} = \{(q, \mathcal{C}, \mathcal{L}, \mathcal{R})\}$. For each data instance, q represents a query case submitted by the legal practitioner; $\mathcal{C} = \{c_1, c_2, \cdots, c_{N_C}\}$ represents a set

of candidate cases (precedents) with size N_C in which $c_i \in \mathcal{C}$ is potentially relevant to q and thus support q's legal judgement; $\mathcal{L} = \{l_1, l_2, \cdots, l_{N_L}\}$ represents the set of applicable laws with size N_L that provides legal basis for the relevance judgement between q and c_i; $\mathcal{R}$ represents the labeled ranking of $\mathcal{C}$ given the query case q. NS-LCR aims at learning a ranking function $f : q \times \mathcal{C} \times \mathcal{L} \rightarrow \mathcal{R} \times \mathcal{E}$, where $\mathcal{E}$ denotes the desired logic explanations corresponding to $\mathcal{R}$.

As mentioned in section 1, we consider two-level explanations for legal retrieval in this study, i.e., $\mathcal{E} = \{e_L, e_C\}$. e_L is a learned logic rule and denotes the law-level explanations that represent the alignment between q and $l \in \mathcal{L}$ applicable for c_i. e_C is another learned logic rule and denotes the case-level explanations that represent the sentence-level alignment between q and c_i. To explicitly model the logic reasoning in legal retrieval, we represent e_L and e_C in the first-order-logic (FOL) format and evaluate them in the fuzzy logic way which we will introduce in the following sections. Benefiting from the logic rules, NS-LCR not only provides explanations for the retrieval but also obtains law-level and case-level relevance scores respectively denoted as r_L and r_C by solving e_L and e_C. NS-LCR further combines r_L, r_C, and semantic relevance score r_N between q and c_i to rank candidates.

3.2. Presenting law articles as FOL

In this study, we present law articles in the FOL format. Specifically, we first manually extract $\mathcal{P}^i = \{P_1^i, P_2^i, \cdots, P_{N_P}^i\}$, a set of predicates from each of $l_i \in \mathcal{L}$, where the predicate represents a key fact or a key circumstance (Ma et al., 2021), N_P denotes the number of predicates. Then, the extracted predicates are connected by logic operators, including conjunction ($\wedge$), disjunction ($\vee$), and negation ($\neg$) to form the clause. As a result, the FOL rules enables the precise expression of the relationships among all of the key facts and circumstances in a legal article, whereby such relationships denote the applicability of this law. For example, as shown in Figure 1, the predicates of "Article 264: [The crime of theft] whoever steals a relatively large amount of public or private property or commits theft repeatedly" include P_1 = "steals a relatively large amount of private property", P_2 = "steals a relatively large amount of public property", and P_3 = "commits theft repeatedly". Based on the relationship among predicates, we represent this law article as a FOL format logic rule $l : (P_1 \vee P_2 \vee P_3 \rightarrow Y)$, where Y = "crime of theft".

3.3. Fuzzy Logic

In this study, as law articles are represented in the FOL format, NS-LCR learns the alignments between the query q and the applicable law $l_i \in \mathcal{L}$ of the candidate case $c \in \mathcal{C}$ by the fuzzy logic. Specifically, at the predicate level, NS-LCR determines whether q satisfies $P \in \mathcal{P}^i$ of l based on $q - P$ similarity, which we called **evaluation**. At the rule level, NS-LCR reasons whether l_i is applicable for q according to all of $q - P$ similarities in l_i, which we called **induction**. In other words, through fuzzy logic-based the evaluation and induction, NS-LCR computes the similarity between q and c with the guidance of l_i since c has already been judged in history and has a clear applicable law, which provides the law-level relevance measurement.

4. OUR APPROACH: NS-LCR

4.1. General framework

As illustrated in Figure 2, our NS-LCR takes a query case q and a candidate case c as the input and predicts multi-level relevance scores along with two logic explanations e_L and e_C. Specifically, NS-LCR achieves this goal through the following four modules:

Neural retrieval module f_{neural} is responsible for predicting the relevance score from the semantic perspective given a pair of (q, c):

$$r_N = f_{neural}(q, c; \theta_{neural}), \tag{1}$$

where $r_N \in \mathbb{R}$ denotes the degree of the relevance between q and c; θ_{neural} denotes the learnable parameters in this module. Considering that NS-LCR is a general framework, f_{neural} can be implemented by existing legal retrieval models, such as cross-encoder methods methods (Xiao et al., 2021; Shao et al., 2020) or dual-encoder methods (Yu et al., 2022c; Sun et al., 2023a).

Law-level module f_{law} incorporates law articles $\mathcal{L}$ in the form of FOL into the input pair (q, c) and outputs the law-level relevance score $r_L \in \mathbb{R}$ and the corresponding explanation e_L in the FOL format:

$$(r_L, e_L) = f_{law}(q, c, \mathcal{L}; \theta_{law}), \tag{2}$$

where θ_{law} is the learnable parameters in this module, f_{law} is implemented in a neuro-symbolic way, which we will introduce the details in subsection 4.2.

Case-level module f_{case} is designed to learn the sentence-level alignment between q and c. The module takes (q, c) as the input and outputs the case-level relevance score $r_C \in \mathbb{R}$ and the corresponding explanation e_C in the FOL format:

$$(r_C, e_C) = f_{case}(q, c; \theta_{case}), \tag{3}$$

where θ_{case} is the learnable parameters in this module, f_{case} is also implemented in a neuro-symbolic way, which we will introduce in subsection 4.3.

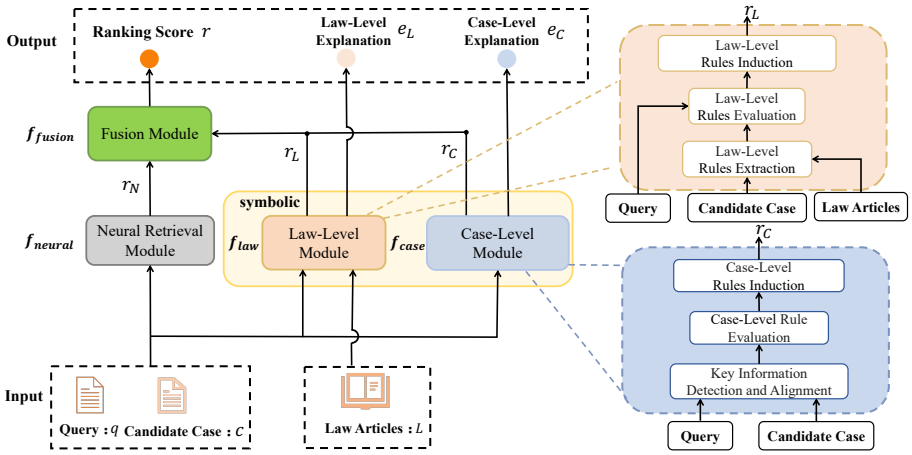

Figure 2: The overall architecture of the proposed model NS-LCR.

Fusion module f_{fusion} combines the outputs of all modules (r_N, r_L, r_C) to computes the final ranking score $r \in \mathbb{R}$ for the candidate case:

$$r = f_{fusion}(r_N, r_L, r_C), \tag{4}$$

where f_{fusion} is the Weighted Reciprocal Rank Fusion (WRRF) to output r by considering all three predicted ranks in a weighted manner:

$$f_{fusion} = \sum_i \frac{w_i^\pi}{\epsilon + \pi(r_i)}, \tag{5}$$

where ϵ is the hyper-parameter for smoothness; $\pi(r_i)$ denotes the predicted rank of module $i \in \{f_{neural}, f_{law}, f_{case}\}$ based on the predicted relevance score r_i; w_i^π is a rank-aware module-specific weight that dynamically adjusts the importance among three types of relevance predictions:

$$w_i^\pi = \begin{cases} 1, & i = f_{neural}, \\ \sin(\frac{\pi(r_i)}{\gamma} \times \frac{\pi}{2}) & i \in \{f_{law}, f_{case}\}, \end{cases} \tag{6}$$

where γ is a hyper-parameter and $\pi(r_i)$ is the predicted rank of the module i. Our Weighted Reciprocal Rank Fusion (WRRF) method enhances the traditional RRF by introducing dynamic weights in the ranking process, diverging from the uniform weighting strategy of RRF (Cormack et al., 2009; Chen et al., 2022). WRRF prioritizes the neural retrieval module for higher-ranked predictions, where it is most confident, and increases the weight of the symbolic module (targeting law and case-specific information) for lower-ranked items to improve accuracy where neural confidence wanes.

4.2. Law-level Module

In this section, we introduce the law-level module f_{law} to model the relevance between a query case and candidates, guided by law articles, as depicted in the top-right of Figure 2.

4.2.1. Law-level Rule Evaluation

Formally, suppose there is a query-candidate pair (q, c) and a set of FOL format law articles $L \subset \mathcal{L}$ corresponding to c [1]. For each of $l_i \in L$ which is represented as a connective of N_P predicates $\mathcal{P}^i = \{P_1^i, P_2^i, \cdots, P_{N_P}^i\}$ as mentioned in subsection 3.2, we first measure the alignment between q and $P_j^i \in \mathcal{P}^i$ by:

$$s_{P_j^i} = f_P(q, P_j^i), \tag{7}$$

where P_j^i denotes the predicate which represents a fact or a circumstance of l_i; $s_{P_j^i} \in [0, 1]$ is the (q, P_i^j) relevance score representing the degree to which q satisfies P_j^i; f_P is implemented by a pre-trained language model (PLM) trained on a large Chinese criminal judgment corpus[2]. Specifically, we construct the input of the PLM in the form of [CLS] + P_j^i + [SEP] + q + [SEP]". The derived embedding of "[CLS]" token in the last layer is then fed to an MLP to compute the score $s_{P_j^i}$.

4.2.2. Law-level Explanation and Induction

Given all of the alignment scores between q and each of $L \subset \mathcal{L}$ corresponding to c, we represent the law-level explanation e_L for a pair (q, c) as:

$$e_L = \bigvee_{i=1}^{N_L} \bigwedge_{j=1}^{N_P} P_j^i \oplus s_{P_j^i}, \tag{8}$$

where $\oplus$ denotes concatenation operation; N_L and N_P respectively denote the number of applicable laws to c and the number of predicates in law l_i;

[1] Each candidate case in civil law system is associated with relevant judged law articles, allowing Law-Level Rules Extraction to be efficiently executed using straightforward text processing techniques.

[2] The corpus is from LeCaRD (Ma et al., 2021). We excluded data from the test set to prevent data leakage

$\wedge$ and $\vee$ respectively denotes the conjunction and disjunction operations. The derived explanation e_L precisely indicates the similarity between query q and all predicates (facts or circumstance) in the law article l_i applicable to candidate case c. Taking Figure 1 as an example, the law-level explanation $e_L = P_1(0.8) \vee P_2(0.2) \vee P_3(0.05)$ means the query respectively has the relevance score 0.8, 0.2, 0.05 to the predicate P_1 = "steals a relatively large amount of private property", predicate P_2 = "steals a relatively large amount of public property", and predicate P_3 = "commits theft repeatedly".

To further induce the law-level relevance score of (q, c), we combine all query-predicates alignment scores across laws using T-norm fuzzy logics (Gottwald and Hájek, 2005):

$$r_L = \frac{\sum_{i=1}^{N_L} \Gamma(\wedge_{j=1}^{N_P} s_{P_j^i})}{N_L}, \tag{9}$$

where N_L and N_P respectively denote the number of applicable laws to c and the number of predicates in law l_i, $\Gamma()$ denotes the Łukasiewicz t-norm (Klement et al., 2013; Li and Srikumar, 2019) that maps the discrete outputs into continuous real values to achieve the induction. As our law-level logic rules are expressed in conjunctive normal form, we can compute r_L by the following steps: (1) use $\Gamma\left(\bigwedge_i P_i\right)$ (or $\Gamma\left(\neg \bigvee_i P_i\right)$) to aggregate predicate-level predictions and get the score for each law l_i; (2) combine all law-level predictions from (1) to get the final score r_L. Please note that in real legal practice, a law may include multiple circumstances connected by $\vee$, indicating that any of the circumstances being satisfied is sufficient to establish the charge. Therefore, step (1) may be computed several times and connected by using $\Gamma\left(\bigvee_i P_i\right)$ (or $\Gamma\left(\neg \bigwedge_i P_i\right)$).

4.3. Case-level Module

The law-level module provides a relevance measure rooted in law articles, yet due to legal cases' semi-structured feature, our legal retrieval model incorporates a case-level module f_{case} for fine-grained sentence alignment between queries and candidates, leveraging logic rules modeling as shown in Figure 2.

4.3.1. Case-level Rule Evaluation

Formally, given a query-candidate pair (q, c), we split q and c into individual sentences respectively denoted by $\{x_i\}_{i=1}^{N_q}$ and $\{y_j\}_{j=1}^{N_c}$ and use use a pre-trained Sentence-BERT (Reimers and Gurevych, 2019) to extract the corresponding embeddings respectively denoted as $\{\mathbf{x}_i\}_{i=1}^{N_q}$, $\{\mathbf{y}_j\}_{j=1}^{N_c}$, where N_q and N_c is the number of sentences in q and c. For each sentence $\mathbf{x}_i$, we seek the K most similar sentences from $\{\mathbf{y}\}_{j=1}^{N_c}$ based on their embeddings, where K is a small value. The case-level logic rule can be constructed as follows:

$$e_C = \bigwedge_{i=1}^{N_q} \bigwedge_{j=1}^{K} (x_i, y_j) \oplus \cos(\mathbf{x}_i, \mathbf{y}_j), \tag{10}$$

where $\cos(\cdot)$ denotes the cosine similarity function [3]. The conjunctive form ensures that learning e_C is a way to help identify relevant facts and circumstances from (q, c). Meanwhile considering e_C is constructed with a small portion of sentences from (q, c), it naturally filters the noise from (q, c).

4.3.2. Case-level Explanation and Induction

Given extracted sentence-pair similarity predictions, we induce the case-level relevance score r_C by:

$$r_C = \left\{ \prod_{i=1}^{N_q} \prod_{j=1}^{K} \cos(\mathbf{x}_i, \mathbf{y}_j) \right\}^{\frac{1}{N_q * K}}. \tag{11}$$

For simplicity, we directly apply the geometric mean to aggregate all sentence pair predictions due to its inclination towards low scores — if there is at least one pair with a low similarity score, then r_C is also low. It expects r_C to illustrate the similarity between all the facts and circumstances in (q, c).

4.4. Model Training

We pre-train θ_{law} using the following steps: (1) For each candidate, we create a pseudo query $\tilde{q}$ from its basic fact description; (2) Using BM25 (Robertson et al., 1995), we select relevant predicates $\tilde{p}^+$ from law articles, marking them as positive (label 1); (3) We also sample negative examples, distinguishing between hard and easy negatives based on chapters [4]. Predicates $\tilde{p}^-$ from these are sampled with balanced ratios and labeled as 0; (4) Using the BCE loss function, we pre-train f_{law}:

$$\ell_{law} = -\frac{1}{n} \sum_{i}^{N} (y_i \log(p_i) + (1 - y_i) \log(1 - p_i)),$$

where y_i is the predicate's label, and $p_i = f_{law}(\tilde{q}_i, \tilde{p}_i, \mathcal{L})$. After this, θ_{law} remains fixed during other training and inference stages.

For θ_{case}, we employ Sentence-BERT (Reimers and Gurevych, 2019) directly without adjustments.

For θ_{neural}, we train the Neural Retrieval Module using the MSE loss function comparing predicted $\hat{r}_N$ and actual relevance scores r for query-candidate pairs: $\ell = \frac{1}{n} \sum_{i=1}^{N} (\hat{r}_N^i - r^i)^2$.

[3] The sentence pairs of cosine similarity ≤ 0 are filtered.

[4] In the civil law system, law articles within the same chapters are similar, while those from different chapters are dissimilar.

5. Experiments

5.1. Experimental settings

5.1.1. Dataset

The experiments[5] were conducted based on two publicly available datasets: ELAM (Yu et al., 2022c) and LeCaRD (Ma et al., 2021).

LeCaRD is a legal case retrieval dataset that contains 107 queries and 43,000 target cases [6]. For each query, 100 target cases are provided, each assigned a 4-level relevance label. All criminal cases were published by the Supreme People's Court of China.

ELAM is an explainable legal case matching dataset, which contains 5000 pairs of annotated cases, and each pair is manually assigned a matching label which is either match (2), partially match (1), or mismatch (0). We transformed this dataset into a legal case retrieval dataset by following the steps: In formulating queries, we capitalize on the paired cases and labels present in the ELAM dataset, enumerating the frequency of each case among these pairs. Instances with a prevalence surpassing six occurrences are designated as queries within match-labeled pairs, culminating in 85 queries. For each specific query, the initially constructed candidate set comprises the cases paired with it in ELAM. Simultaneously, supplementary cases are integrated to expand the candidate pool's dimensions. Considering ELAM's limited data volume, a subset of cases is chosen from LeCaRD's candidate set to establish a corpus. In adherence to LeCaRD's methodology for assembling a candidate pool, we employ a triad of retrieval models BM25 (Robertson et al., 1995) and TF-IDF (Salton and Buckley, 1988) to obtain the top 100 cases from the corpus for each query. Cases that secure a position within the top 100 in a minimum of one model are assimilated as hard negatives into the corresponding candidate set. In contrast, those absent from the top 100 across all three models are annexed as soft negatives. With an approximate 1:4 ratio between hard and soft negatives, the outcome for each query consists of a corresponding set of 50 candidate cases.

We normalize the relevance labels in LeCaRD and ELAM, which have multiple levels, to ensure that the ground-truth relevance scores range from 0 to 1. Table 3 lists the statistics of the datasets.

Since both datasets correspond to Chinese legal case retrieval and NS-LCR targets learning law-level logic rules, we request three annotators manually label the articles of Criminal Law of the People's Republic of China in the FOL format, as subsection 3.2 mentioned. The annotators are postgraduate students in artificial intelligence who have undergone legal training, and the annotation results undergo double-checking by professionals. Some basic statistics of the L are listed in Table 2.

5.1.2. Baselines

To verify the effectiveness of NS-LCR, we apply it to the following underlying models:

Criminal-BERT (Zhong et al., 2019) is a legal domain pre-trained model based on BERT (Devlin et al., 2018), which is fine-tuned on millions of criminal legal document. The cross-encoder architecture has been utilized to predict the relevance between the query and the candidate case.

Lawformer (Xiao et al., 2021) is a Longformer-based pre-trained language model training millions of Chinese legal cases to represent long legal documents better. In the experiment, we concatenate the input cases to Lawformer and use the mean pooling of Lawformer's output to conduct matching.

BERT-PLI (Shao et al., 2020) uses BERT to capture the semantic relationships at the paragraph level. Then it uses RNN and Attention model to infer the relevance between the two cases. Finally, it uses an MLP to calculate the aggregated embeddings similarity score.

BERT-ts-L1 (Shao et al., 2022) optimizes the Criminal-BERT's attention weights with the attention of users majoring in law for relevance prediction in legal case retrieval. The optimized model is then used to predict the relevance of the query and candidate case.

We also compare NS-LCR with three baselines that can also both consider the case-level and law-level in models. The first is an intuitive baseline that appends the contents of cited law articles to the original cases, forming new extended legal cases. Existing matching models of Criminal-BERT, Lawformer, BERT-PLI, and BERT-ts-L1 can be applied to the extended legal cases, denoted as **Cat-Law (Criminal-BERT)**, **Cat-Law (Lawformer)**, **Cat-Law (BERT-PLI)**, and **Cat-Law (BERT-ts-L1)**, respectively. The second baseline is EPM (Feng et al., 2022a), which employs an attention mechanism to incorporate article semantics into the legal judgment prediction models. Existing matching models of Sentence-BERT, Lawformer, BERT-PLI, and BERT-ts-L1 can be applied to EPM, denoted as **EPM (Criminal-BERT)**, **EPM (Lawformer)**, **EPM (BERT-PLI)**, and **EPM (BERT-ts-L1)**, respectively. The third baseline is Law-Match (Sun et al., 2023a), which learns legal case retrieval models by respecting the corresponding law articles as instrumental variables (IVs) and legal cases as treat-

[5]Both the source code and dataset are available at: https://github.com/ke-01/NS-LCR.

[6]Some experiment results are differences from Lecard's paper due to code errors in the original LeCaRD implementation, rectified in commit 89b7bf8.

Table 1: Performance comparisons between NS-LCR and the baselines. The boldface represents the best performance. '†' indicates that the improvements over all of the baselines are statistically significant (t-tests, p-value < 0.05).

Models	LeCaRD						ELAM					
	P@5	P@10	MAP	NDCG@10	NDCG@20	NDCG@30	P@5	P@10	MAP	NDCG@10	NDCG@20	NDCG@30
Criminal-BERT	0.430	0.425	0.498	0.728	0.760	0.799	0.541	0.512	0.598	0.641	0.778	0.821
Cat-Law(Criminal-BERT)	0.410	0.405	0.497	0.745	0.767	0.817	0.447	0.535	0.547	0.603	0.745	0.780
EPM(Criminal-BERT)	0.410	0.415	0.518	0.740	0.782	0.813	0.565	0.541	0.613	0.633	0.778	0.804
Law-Match(Criminal-BERT)	0.420	0.433	0.502	0.746	0.778	0.815	0.626	0.542	0.628	0.657	0.791	0.829
NS-LCR(Criminal-BERT)	**0.440**†	**0.440**†	**0.566**†	**0.758**†	**0.793**†	**0.833**†	**0.647**†	**0.553**†	**0.644**†	**0.688**†	**0.830**†	**0.857**†
Lawformer	0.460	0.425	0.497	0.739	0.759	0.800	0.588	0.582	0.612	0.647	0.775	0.801
Cat-Law(Lawformer)	0.460	0.405	0.464	0.690	0.722	0.764	0.576	0.559	0.618	0.643	0.749	0.810
EPM(Lawformer)	0.450	0.415	0.469	0.695	0.732	0.782	0.553	0.576	0.611	0.640	0.771	0.810
Law-Match(Lawformer)	0.450	0.432	0.496	0.768	0.773	0.819	0.634	0.578	0.644	0.654	0.790	0.818
NS-LCR(Lawformer)	**0.480**†	**0.450**†	**0.531**†	**0.786**†	**0.810**†	**0.841**†	**0.635**	**0.594**†	**0.656**†	**0.701**†	**0.829**†	**0.859**†
Bert-PLI	0.420	0.455	0.490	0.781	0.821	0.868	0.506	0.524	0.580	0.598	0.772	0.794
Cat-Law(BERT-PLI)	0.420	0.420	0.490	0.773	0.817	0.861	0.518	0.535	0.588	0.636	0.777	0.811
EPM(BERT-PLI)	0.430	0.425	0.486	**0.793**	0.823	0.874	0.541	0.524	0.612	0.635	0.796	0.827
Law-Match(BERT-PLI)	**0.450**	0.425	0.519	0.789	0.829	0.868	0.600	0.556	0.629	0.687	0.823	0.843
NS-LCR(BERT-PLI)	**0.450**	**0.460**†	**0.544**†	0.792	**0.831**	**0.876**	**0.635**†	**0.576**†	**0.643**†	**0.688**	**0.836**†	**0.858**†
BERT-ts-L1	0.430	0.425	0.516	0.736	0.776	0.819	0.518	0.529	0.577	0.613	0.759	0.795
Cat-Law(BERT-ts-L1)	0.430	0.405	0.508	0.733	0.754	0.805	0.459	0.535	0.544	0.601	0.743	0.778
EPM(BERT-ts-L1)	0.420	0.420	0.520	0.744	0.790	0.815	0.553	**0.582**	0.595	0.631	0.759	0.797
Law-Match(BERT-ts-L1)	0.430	0.427	0.528	0.759	0.793	0.829	0.529	0.494	0.591	0.609	0.749	0.807
NS-LCR(BERT-ts-L1)	**0.440**†	**0.440**†	**0.550**†	**0.771**†	**0.798**†	**0.847**†	**0.565**†	0.576	**0.640**†	**0.687**†	**0.821**†	**0.852**†

Table 2: Statistics of the annotated FOL rules of law articles.

total articles	FOL			
	¬	∧	∨	→
441	44	2259	1625	1232

Table 3: Statistics of LeCaRD and ELAM

	LeCaRD	ELAM
# total queries	107	85
# candidate cases per query	100	50
avg. # relevant cases per query	10.33	9.06
avg. # sentences per query	6.91	16.24
avg. # sentences per candidate case	86.63	43.43
avg. # cited law articles per candidate case	6.5	5.62

ments. Then, IV decomposition and recombination are used to learn the legal case embedding. Law-Match is model-agnostic and can apply to Sentence-BERT, Lawformer, BERT-PLI, and BERT-ts-L1, denoted as **Law-Match (Criminal-BERT)**, **Law-Match (Lawformer)**, **Law-Match (BERT-PLI)**, and **Law-Match (BERT-ts-L1)**, respectively.

The proposed NS-LCR is also model-agnostic. In the experiments, we applied NS-LCR to the underlying models of Criminal-BERT, Lawformer, BERT-PLI, and BERT-ts-L1, achieving four versions and referred to as **NS-LCR (Criminal-BERT)**, **NS-LCR (Lawformer)**, **NS-LCR (BERT-PLI)**, and **NS-LCR (BERT-ts-L1)** respectively.

5.2. Implementation Details

We optimize the hyperparameters of NS-LCR's base models through grid search on the validation set, employing Adam (Kingma and Adam, 2015). The batch size is selected from {2, 8, 16}, while the learning rate is chosen from {2e-5, 3e-6}. The other parameters of the base models remain in line with their original paper. In the law-level module, we fine-tune Criminal-BERT with a batch size of 24 and a learning rate 2e-5 to acquire the Predicate Evaluation Module. As for the case-level module, we tune K from $\{1, 3, 5\}$. In terms of the fusion module, we set the hyper-parameter $\epsilon = 60$ for smoothness and tune $\gamma \in \{0, 1, 2, 50\}$ [7] to balance the effects of the three modules.

5.3. Evaluate Explainable Legal Case Retrieval

In assessing retrieval accuracy, we adopt precision metrics like P@5, P@10, MAP, and ranking metrics such as NDCG@10, NDCG@20, and NDCG@30, as per (Ma et al., 2021).

We introduce a novel explanation evaluation method with help of large language models (LLMs). Leveraging retrieval argument LLMs allows updated knowledge acquisition, improving generation (Mialon et al., 2023). Believing explainable retrieval assists in comprehension, we posit that explanations can aid LLMs in downstream tasks. Thus, we transformed ELAM and LeCaRD dataset queries into a legal judgment task for LLMs, using four prompt types, including zero-shot and few-shot prompts with/without explanations (Brown et al., 2020; Sun, 2023) [8]:

Zero-shot with/without explanation:
Please answer the criminal name for the query fact description based on the relevant cases.
The query is [fact description].

[7] $\gamma = 0$ represents equal weights assigned to $f_{neural}, f_{law}, f_{case}$ outputs.

[8] Explanations, based on logic rules, exclude predicates with scores below 0.5. Logic operators are translated to natural language: 'and' for ∧, 'or' for ∨, 'not' for ¬. The prompts were translated to Chinese.

Table 4: Performance of LLMs across different prompts on legal judgement prediction on LeCaRD.

LLMs	Prompt	Accuracy
text-davinci-003	Zero-shot w/o explanation	0.621
	Zero-shot w/ explanation	**0.656**
	Few-shot w/o explanation	0.652
	Few-shot w/ explanation	**0.707**
gpt-3.5-turbo	Zero-shot w/o explanation	0.675
	Zero-shot w/ explanation	**0.769**
	Few-shot w/o explanation	0.707
	Few-shot w/ explanation	**0.832**

Table 5: Performance of LLMs across different prompts on legal judgement prediction on ELAM.

LLMs	Prompt	Accuracy
text-davinci-003	Zero-shot w/o explanation	0.672
	Zero-shot w/ explanation (IOT-Match)	0.841
	Zero-shot w/ explanation (NS-LCR)	**0.927**
	Few-shot w/o explanation	0.782
	Few-shot w/ explanation (IOT-Match)	0.853
	Few-shot w/ explanation (NS-LCR)	**0.951**
gpt-3.5-turbo	Zero-shot w/o explanation	0.780
	Zero-shot w/ explanation (IOT-Match)	0.876
	Zero-shot w/ explanation (NS-LCR)	**0.916**
	Few-shot w/o explanation	0.794
	Few-shot w/ explanation (IOT-Match)	0.888
	Few-shot w/ explanation (NS-LCR)	**0.969**

Evidence: [relevant case]+ [explanation].
The answer is:

Few-shot with/without explanation:
Please answer the criminal name for the query fact description based on the relevant cases.
The query is [fact description].
Evidence: [relevant case]+ [explanation].
The answer is: [criminal name].
The query is [fact description].
Evidence: [relevant case]+ [explanation].
The answer is:

Explanation quality is determined by LLM prediction accuracy for various prompts.

We employ **text-davinci-003** (Ouyang et al., 2022) and **gpt-3.5-turbo**. With the temperature set to 0, we chose one relevant case per prompt due to model length constraints.

5.4. Experimental results and analysis

5.4.1. Comparison against baselines.

Based on results in Table 1, NS-LCR variants (including Criminal-Bert, Lawformer, Bert-PLI, BERT-ts-L1) surpassed their neural retrieval counterparts in six metrics on LeCaRD and ELAM with significance (t-tests, p-value <0.05). This underscores the symbolic module's potency in melding law article knowledge via law-level logic rules and enhancing retrieval by extracting key details from cases using case-level logic rules. Additionally, NS-LCR outperforms frameworks like Cat-Law, EPM, and Law-Match, highlighting the superiority of integrating detailed legal behaviors through logic rules over direct embedding encoding.

5.4.2. Quality of Logic Rules as Explanations.

Tables 4 and 5 show outcomes from logic explanations on LeCaRD and ELAM. We infer: (1) Across both LLMs (text-davinci-003 and gpt-3.5-turbo) and datasets (LeCaRD and ELAM), evidence with cases and explanations outperformed evidence with only cases for both zero-shot and few-shot prompts. This highlights the significance of explainable legal case retrieval models in boosting human understanding and LLMs' comprehension of relevant cases; (2) Table 5 contrasts IOT-Match (Yu et al., 2022c) and NS-LCR explanations on ELAM[9]. NS-LCR explanations were found superior, indicating their high-quality nature.

5.4.3. Ablation study.

To evaluate the effectiveness of NS-LCR's symbolic components, we conducted an ablation study on LeCaRD[10]. Our integrations included: (1) + **Law-level Module**, (2) + **Case-level Module**, and (3) + **NS-LCR** (combining both modules) into the base model. As shown in Table 6, both modules significantly boosted retrieval metrics, with the Law-level Module incorporating FOL-form law articles, and the Case-level Module applying logic rules from queries and cases. The combined NS-LCR framework markedly surpassed the base model, highlighting the importance of merging law and case information via logic rules for neural retrieval model efficacy. Notably, the Law-level module's impact waned at higher K values, especially alongside the Case-level module, suggesting its optimality for complex differentiation tasks may diminish with broader top-ranking item arrays, potentially overshadowing its utility.

5.4.4. Low-resource scenarios.

In this section, we assess if NS-LCR can address the challenge of limited labeled data in LCR, given the high cost of expert annotation and diverse data distributions [10]. Through experiments with varied training data amounts, Figure 3 shows NS-LCR consistently improved performance across all base models at 10%, 50%, and 100% LeCaRD training data proportions. The most notable gains occurred with less data, underscoring the advantage

[9]We report only on ELAM due to IOT-Match's need for explanation labels present only in ELAM.

[10]We only focused on LeCaRD due to ELAM's adaptation to the legal case retrieval dataset.

Table 6: Ablation study of NS-LCR on LeCaRD.

Models	P@5	P@10	MAP	NDCG@10	NDCG@20	NDCG@30
Criminal-BERT	0.430	0.425	0.498	0.728	0.760	0.799
+ Law-level Module	0.430	0.430	0.530	0.734	0.767	0.805
+ Case-level Module	0.430	0.425	0.506	0.741	**0.793**	**0.844**
+ NS-LCR	**0.440**	**0.440**	**0.566**	**0.758**	**0.793**	0.833
Lawformer	0.460	0.425	0.497	0.739	0.759	0.800
+ Law-level Module	**0.480**	0.435	0.517	0.772	0.773	0.812
+ Case-level Module	0.470	0.445	0.508	0.766	0.802	**0.849**
+ NS-LCR	**0.480**	**0.450**	**0.531**	**0.786**	**0.810**	0.841
BERT-PLI	0.420	0.455	0.490	0.781	0.821	0.868
+ Law-level Module	0.440	0.455	0.510	0.782	0.823	0.868
+ Case-level Module	0.430	0.455	0.492	0.784	0.827	0.872
+ NS-LCR	**0.450**	**0.460**	**0.544**	**0.792**	**0.831**	**0.876**
BERT-ts-L1	0.430	0.425	0.516	0.736	0.776	0.819
+ Law-level Module	0.430	0.435	0.545	0.755	0.780	0.823
+ Case-level Module	0.430	0.430	0.526	0.743	0.787	**0.858**
+ NS-LCR	**0.440**	**0.440**	**0.550**	**0.771**	**0.798**	0.847

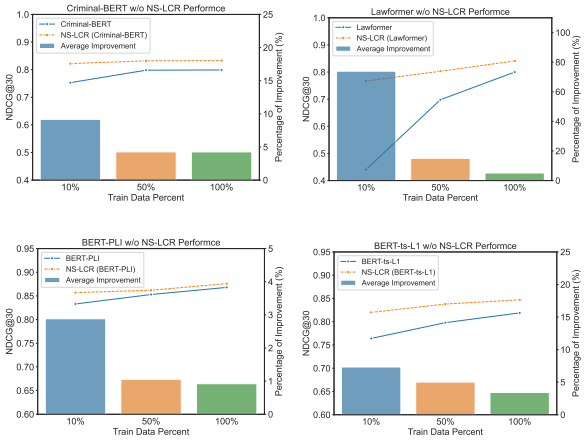

Figure 3: Various base models' performance w/ and wo/ NS-LCR on LeCaRD is shown. The lines show the NDCG@30 scores and the bars show the improvement percentage due to NS-LCR.

of our logic module in low-resource contexts. We also employed a hybrid approach that combines human evaluation and GPT-4 (OpenAI, 2023) to evaluate the explanations (Bills et al., 2023) directly. Specifically, we randomly sampled 50 queries from ELAM, retrieved the relevant candidate cases, and generated the explanations from IOT-Match, gpt-3.5-turbo [11], and NS-LCR. Then, we asked three annotators to judge the quality of explanations from IOT-Match, gpt-3.5-turbo, and NS-LCR based on the degree of alignment between the explanations, the original texts (query and candidate case), and the law articles. At the same time, we used GPT-4 to score the quality of explanations from IOT-Match, gpt-3.5-turbo, and NS-LCR [12]. As shown in Table 7, both the manual scores from the three annotators and the scores from GPT-4, NS-LCR, and gpt-3.5-turbo consistently achieved higher scores than IOT-Match. Besides, gpt-3.5-turbo has not undergone legal task pre-training, and has a relatively limited understanding of legal knowledge (Chalkidis, 2023), resulting in slightly lower performance compared to NS-LCR. In summary, the results verify the effectiveness of NS-LCR in generating high-quality explanations.

[11] The prompt template for generating the explanations is: *"Please confirm whether the query and the candidate case below are relevant, and provide an explanation: [query];[candidate case]"*

[12] The prompt is: *"Rate the explanations for IOT-Match, ChatGPT, and NS-LCR separately, with scores ranging from 0 to 10:"*.

Table 7: Human and LLM evaluations of the explanation quality over 50 randomly sampled queries from ELAM (score range is from 0 (low) to 10 (high)).

	IOT-Match	gpt-3.5-turbo	NS-LCR
Human	7.74	8.75	**8.95**
GPT-4	7.86	8.71	**8.86**

6. Conclusion

This paper introduces the NS-LCR that neuro-symbolically combines case and law logic rules, providing faithful explainability. It aligns with different legal retrieval models. By updating LeCaRD and ELAM benchmarks with logic rules and introducing an explainability metric using LLMs, results show NS-LCR's excellent performance, reliable explanations, and improvement of base models in limited-resource scenarios.

7. Acknowledgements

This work was funded by the National Key R&D Program of China (2023YFA1008704), the National Natural Science Foundation of China (No.62376275, No.62377044), Beijing Key Laboratory of Big Data Management and Analysis Methods, Major Innovation & Planning Inter-disciplinary Platform for the "Double-First Class" Initiative, funds for building world-class universities (disciplines) of Renmin University of China. Supported by the Outstanding Innovative Talents Cultivation Funded Programs 2024 of Renmin University of China.

8. Bibliographical References

David Alvarez Melis and Tommi Jaakkola. 2018. Towards robust interpretability with self-explaining neural networks. *Advances in neural information processing systems*, 31.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity.

Pietro Barbiero, Gabriele Ciravegna, Francesco Giannini, Pietro Lió, Marco Gori, and Stefano Melacci. 2022. Entropy-based logic explanations of neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6046–6054.

Trevor Bench-Capon, Michał Araszkiewicz, Kevin Ashley, Katie Atkinson, Floris Bex, Filipe Borges, Daniele Bourcier, Paul Bourgine, Jack G Conrad, Enrico Francesconi, et al. 2012. A history of ai and law in 50 papers: 25 years of the international conference on ai and law. *Artificial Intelligence and Law*, 20:215–319.

Paheli Bhattacharya and Ghosh et al. 2020. Methods for computing legal document similarity: A comparative study. *arXiv preprint*.

Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2020. *Hier-SPCNet: A Legal Statute Hierarchy-Based Heterogeneous Network for Computing Legal Case Document Similarity*, page 1657–1660. Association for Computing Machinery, New York, NY, USA.

Adrien Bibal, Michael Lognoul, Alexandre De Streel, and Benoît Frénay. 2021. Legal requirements on explainability in machine learning. *Artificial Intelligence and Law*, 29(2):149–169.

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. 2023. Language models can explain neurons in language models.

Max Black. 1956. Why cannot an effect precede its cause? *Analysis*, 16(3):49–58.

Michael J Bommarito II, Daniel Martin Katz, and Eric M Detterman. 2021. Lexnlp: Natural language processing and information extraction for legal and regulatory texts. In *Research Handbook on Big Data Law*, pages 216–227. Edward Elgar Publishing.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Ilias Chalkidis. 2023. Chatgpt may pass the bar exam soon, but has a long way to go for the lexglue benchmark. *arXiv preprint arXiv:2304.12202*.

Tao Chen, Mingyang Zhang, Jing Lu, Michael Bendersky, and Marc Najork. 2022. Out-of-domain semantics to the rescue! zero-shot hybrid retrieval models. In *Advances in Information Retrieval: 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10–14, 2022, Proceedings, Part I*, pages 95–110. Springer.

Gabriele Ciravegna, Pietro Barbiero, Francesco Giannini, Marco Gori, Pietro Lió, Marco Maggini, and Stefano Melacci. 2023. Logic explained networks. *Artificial Intelligence*, 314:103822.

Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 758–759.

Luc De Raedt and Thomas Demeester. 2019. Neuro-symbolic= neural+ logical+ probabilistic. In *NeSy'19@ IJCAI, the 14th International Workshop on Neural-Symbolic Learning and Reasoning*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Didier Dubois, Henri Prade, and Claudette Testemale. 1988. Weighted fuzzy pattern matching. *Fuzzy sets and systems*, 28(3):313–331.

Yi Feng, Chuanyi Li, and Vincent Ng. 2022a. Legal judgment prediction via event extraction with constraints. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 648–664,

Dublin, Ireland. Association for Computational Linguistics.

Yufei Feng, Xiaoyu Yang, Xiaodan Zhu, and Michael Greenspan. 2022b. Neuro-symbolic natural logic with introspective revision for natural language inference. *Transactions of the Association for Computational Linguistics*, 10:240–256.

Leilei Gan, Kun Kuang, Yi Yang, and Fei Wu. 2021. Judgment prediction via injecting legal knowledge into neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12866–12874.

Siegfried Gottwald and Petr Hájek. 2005. Triangular norm-based mathematical fuzzy logics. In *Logical, algebraic, analytic and probabilistic aspects of triangular norms*, pages 275–299. Elsevier.

Eleni Ilkou and Maria Koutraki. 2020. Symbolic vs sub-symbolic ai methods: Friends or enemies? In *CIKM (Workshops)*.

Rishabh Jain, Gabriele Ciravegna, Pietro Barbiero, Francesco Giannini, Davide Buffelli, and Pietro Lio. 2022. Extending logic explained networks to text classification. In *Empirical Methods in Natural Language Processing*.

Sarthak Jain and Byron C Wallace. 2019. Attention is not explanation. In *Proceedings of NAACL-HLT*, pages 3543–3556.

J Betty Jane and EN Ganesh. 2019. A review on big data with machine learning and fuzzy logic for better decision making. *International Journal of Scientific & Technology Research*, 8(10):1121–1125.

Hang Jiang, Sairam Gurajada, Qiuhao Lu, Sumit Neelam, Lucian Popa, Prithviraj Sen, Yunyao Li, and Alexander Gray. 2021. Lnn-el: A neuro-symbolic approach to short-text entity linking. *arXiv preprint arXiv:2106.09795*.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Diederik P Kingma and Jimmy Ba Adam. 2015. A method for stochastic. *Optimization. In, ICLR*, 5.

Erich Peter Klement, Radko Mesiar, and Endre Pap. 2013. *Triangular norms*, volume 8. Springer Science & Business Media.

Sushanta Kumar, P. Krishna Reddy, V. Balakista Reddy, and Aditya Singh. 2011. Similarity analysis of legal judgments. In *Proceedings of the 4th Bangalore Annual Compute Conference, Compute 2011, Bangalore, India, March 25-26, 2011*, page 17. ACM.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering. *arXiv preprint arXiv:2203.05115*.

Seungeon Lee, Xiting Wang, Sungwon Han, Xiaoyuan Yi, Xing Xie, and Meeyoung Cha. 2022. Self-explaining deep models with logic rule reasoning. In *36th Annual Conference on Neural Information Processing Systems (NeurIPS 2022)*. Neural Information Processing Systems (NeurIPS).

Tao Li and Vivek Srikumar. 2019. Augmenting neural networks with first-order logic. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 292–302.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Xiao Liu, Da Yin, Yansong Feng, Yuting Wu, and Dongyan Zhao. 2021. Everything has a cause: Leveraging causal inference in legal text analysis. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1928–1941.

Yixiao Ma, Yunqiu Shao, Yueyue Wu, Yiqun Liu, Ruizhe Zhang, Min Zhang, and Shaoping Ma. 2021. Lecard: a legal case retrieval dataset for chinese law system. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 2342–2348.

Robin Manhaeve, Sebastijan Dumancic, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. 2018. Deepproblog: Neural probabilistic logic programming. *advances in neural information processing systems*, 31.

Grégoire Mialon, Roberto Dessì, Maria Lomeli, Christoforos Nalmpantis, Ram Pasunuru, Roberta Raileanu, Baptiste Rozière, Timo Schick, Jane Dwivedi-Yu, Asli Celikyilmaz, et al. 2023. Augmented language models: a survey. *arXiv preprint arXiv:2302.07842*.

Akshay Minocha, Navjyoti Singh, and Arjit Srivastava. 2015. Finding relevant indian judgments using dispersion of citation network. In *Proceedings of the 24th International Conference on World Wide Web*, page 1085–1088.

Alfredo López Monroy, Hiram Calvo, Alexander F. Gelbukh, and Georgina García Pacheco. 2013. Link analysis for representing and retrieving legal information. In *Computational Linguistics and Intelligent Text Processing - 14th International Conference, CICLing 2013, Samos, Greece, March 24-30, 2013, Proceedings, Part II*, volume 7817, pages 380–393. Springer.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Jay M Ponte and W Bruce Croft. 2017. A language modeling approach to information retrieval. In *ACM SIGIR Forum*, volume 51, pages 202–208. ACM New York, NY, USA.

Henry Prakken. 1997. *The Role of Logic in Legal Reasoning*, pages 15–31. Springer Netherlands, Dordrecht.

Henry Prakken and Giovanni Sartor. 2015. Law and logic: A review from an argumentation perspective. *Artificial Intelligence*, 227:214–245.

Weicong Qin, Zelin Cao, Weijie Yu, Zihua Si, Sirui Chen, and Jun Xu. 2023. Incorporating judgment prediction into legal case retrieval via law-aware generative retrieval. *arXiv preprint arXiv:2312.09591*.

Abhiramon Rajasekharan, Yankai Zeng, Parth Padalkar, and Gopal Gupta. 2023. Reliable natural language understanding with large language models and answer set programming.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Ryan Riegel, Alexander Gray, Francois Luus, Naweed Khan, Ndivhuwo Makondo, Ismail Yunus Akhalwaya, Haifeng Qian, Ronald Fagin, Francisco Barahona, Udit Sharma, et al. 2020. Logical neural networks. *arXiv preprint arXiv:2006.13155*.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at trec-3. *Nist Special Publication Sp*, 109:109.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523.

Manavalan Saravanan, Balaraman Ravindran, and Shivani Raman. 2009. Improving legal information retrieval using an ontological framework. *Artificial Intelligence and Law*, 17:101–124.

Yunqiu Shao, Jiaxin Mao, Yiqun Liu, Weizhi Ma, Ken Satoh, Min Zhang, and Shaoping Ma. 2020. Bert-pli: Modeling paragraph-level interactions for legal case retrieval. In *IJCAI*, pages 3501–3507.

Yunqiu Shao, Yueyue Wu, Yiqun Liu, Jiaxin Mao, and Shaoping Ma. 2022. Understandingunderstanding relevance judgments in legal case retrieval. *ACM Trans. Inf. Syst.*

Yunqiu Shao, Yueyue Wu, Yiqun Liu, Jiaxin Mao, Min Zhang, and Shaoping Ma. 2021. Investigating user behavior in legal case retrieval. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 962–972.

Shaoyun Shi, Hanxiong Chen, Weizhi Ma, Jiaxin Mao, Min Zhang, and Yongfeng Zhang. 2020. Neural logic reasoning. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1365–1374.

Zhongxiang Sun. 2023. A short survey of viewing large language models in legal aspect. *arXiv preprint arXiv:2303.09136*.

Zhongxiang Sun, Jun Xu, Xiao Zhang, Zhenhua Dong, and Ji-Rong Wen. 2023a. Law article-enhanced legal case matching: A causal learning approach. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1549–1558.

Zhongxiang Sun, Weijie Yu, Zihua Si, Jun Xu, Zhenhua Dong, Xu Chen, Hongteng Xu, and Ji-Rong Wen. 2023b. Explainable legal case matching via graph optimal transport. *IEEE Transactions on Knowledge and Data Engineering*.

Wenya Wang and Sinno Jialin Pan. 2020. Integrating deep learning with logic fusion for information extraction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9225–9232.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny

Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Zijun Wu, Atharva Naik, Zi Xuan Zhang, and Lili Mou. 2021. Weakly supervised explainable phrasal reasoning with neural fuzzy logic. *arXiv preprint arXiv:2109.08927*.

Chaojun Xiao, Xueyu Hu, Zhiyuan Liu, Cunchao Tu, and Maosong Sun. 2021. Lawformer: A pre-trained language model for chinese legal long documents. *AI Open*, 2:79–84.

Fangyi Yu, Lee Quartey, and Frank Schilder. 2022a. Legal prompting: Teaching a language model to think like a lawyer. *arXiv preprint arXiv:2212.01326*.

Weijie Yu, Liang Pang, Jun Xu, Bing Su, Zhenhua Dong, and Ji-Rong Wen. 2022b. Optimal partial transport based sentence selection for long-form document matching. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2363–2373, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Weijie Yu, Zhongxiang Sun, Jun Xu, Zhenhua Dong, Xu Chen, Hongteng Xu, and Ji-Rong Wen. 2022c. Explainable legal case matching via inverse optimal transport-based rationale extraction. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 657–668.

Lotfi A Zadeh. 1988. Fuzzy logic. *Computer*, 21(4):83–93.

Yiming Zeng, Ruili Wang, John Zeleznikow, and Elizabeth Kemp. 2005. Knowledge representation for the intelligent legal case retrieval. In *Knowledge-Based Intelligent Information and Engineering Systems: 9th International Conference, KES 2005, Melbourne, Australia, September 14-16, 2005, Proceedings, Part I 9*, pages 339–345. Springer.

Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal judgment prediction via topological learning. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 3540–3549.

Haoxi Zhong, Zhengyan Zhang, Zhiyuan Liu, and Maosong Sun. 2019. Open chinese language pre-trained model zoo. Technical report.