Legal Case Retrieval: A Survey of the State of the Art

Yi Feng[1], Chuanyi Li[1]*, Vincent Ng[2]
[1]State Key Laboratory for Novel Software Technology, Nanjing University, China
[2]Human Language Technology Research Institute, University of Texas at Dallas, USA
{fy, lcy}@nju.edu.cn, vince@hlt.utdallas.edu

Abstract

Recent years have seen increasing attention on Legal Case Retrieval (LCR), a key task in the area of Legal AI that concerns the retrieval of cases from a large legal database of historical cases that are similar to a given query. This paper presents a survey of the major milestones made in LCR research, targeting researchers who are finding their way into the field and seek a brief account of the relevant datasets and the recent neural models and their performances.

1 Introduction

Legal Case Retrieval (LCR) is the task of retrieving cases from a large legal database of historical cases that are similar to a given *query* case (Blair and Maron, 1985; Moens, 2001; Breuker et al., 2004; El Jelali et al., 2015; Tran et al., 2020; Li et al., 2021; Hong et al., 2020; Fang et al., 2022). While a query case is composed of the factual description of the event underlying the case, each historical case in the legal database is stored as a legal judgment document that describes not only the facts of the case, but also the court's decision and the reasoning involved in the decision process. Figure 1 shows two examples of a query case and a historical case.

Regardless of the legal system adopted by a country, LCR offers significant practice value. In the Common Law System (such as the U.S.), precedents (prior cases decided in the court) are cited by legal professionals (e.g., lawyers, judges) to support their arguments for their intended outcome in the current case (Shulayeva et al., 2017). Even in the Civil Law System (such as China and Germany), where statutes are cited as arguments, LCR still provides crucial reference information (such as the statute(s) applicable to the historical case and the court's decision) for both legal professionals and laymen in need of legal advice.

Since the first paper on LCR was published in 1994 (Montazeri et al., 1994), LCR has remained an active area of research in the Information Retrieval (IR) community. Note that LCR can naturally be recast as a standard IR task, with the goal of returning a list of historical cases ranked by their relevance to the query case. In fact, this is how early LCR systems were built: a traditional retrieval model (e.g., TF-IDF) is used to measure the relevance of each historical case to a query case and impose a ranking on the historical cases based on the resulting similarity values.

Not surprisingly, like standard IR models, these LCR systems can efficiently rank a large number of documents (Van Opijnen and Santos, 2017; Sansone and Sperlí, 2022). However, their accuracy is fairly poor. This should not be surprising either, as they did not attempt to address the unique challenges associated with LCR that arise primarily from the fact that the definition of relevance in the legal domain is not identical to the typical definition of text relevance (e.g., topic relevance, semantic relevance) (Shao et al., 2020b; Ma et al., 2021b). To exemplify, consider again the two cases in Figure 1(a). They appear to be similar as both are theft events where the item involved is antibiotics and the total values differ only by $90 ($908 vs. $998). However, the two cases are *not* considered similar according to California jurisdiction. Specifically, California law stipulates that any theft in which the amount involved exceeds a certain value is a *grand theft*. In the historical case, the amount exceeded this threshold and is therefore a case of grand theft, whereas the amount involved in the query case is below this threshold and is therefore a case of *petty theft*. In other words, knowledge external to the query case (in this case knowledge of the statutes) is needed to determine case relevance.[1] As we will discuss in Section 4, recently

*Corresponding author.

[1]Additional examples can be found in Appendix A.

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6472–6485

(a) Example 1

Query Case:
[Fact] Respondent Jxx breaked into a Walgreens pharmacy and stole ***$908*** worth of antibiotics.

Historical Case:
[Fact] Respondent Rxx entered into a Walgreens pharmacy without permission and stole ***$998*** worth of antibiotics.
[Reasoning] For his participation in a theft during which he stole ***$998*** worth of antibiotics, Respondent Rxx faced charges of committing a grand theft under the California Penal Code § 487.
[Decision] By K. v. M., 140 U.S. 1021, 40-47, Rxx was sentenced to 16 months of prison

(b) Example 2

Query Case:
[Fact] Respondent Kxx *had conflicts with Uxx*. Kxx knew that Uxx had asthma. To revenge Uxx, Kxx entered Uxx's apartment and *stole Uxx's asthma medicine*. *The theft led to Uxx's death*, the cause of which was asthma attack. After the incident, *Kxx surrendered herself to the law*.

Historical Case:
[Fact] Respondent Jxx *had a grudge against Rxx*. Jxx *stole Rxx's heart medicine*, which caused the victim to *die of a heart attack*. After that, *Jxx fled abroad*.
[Reasoning] The theft event concerned the murder of Rxx under the California Penal Code § 187, and the respondent fled to escape the punishment of the law.
[Decision] By B. v. C., 11 U.S. 99., 10-20, the respondent was sentenced to 15 years of prison.

Figure 1: Examples of query cases and historical cases.

developed LCR models attempt to address this and other challenges associated with LCR.

While LCR has primarily been tackled in the IR community, we believe that this task would also be of interest to NLP researchers since LCR involves information extraction from text data. Our goal in this paper is to increase the awareness of this task among NLP researchers and stimulate their interest in this task, especially since it is far from being solved even after nearly 30 years of research.[2]

2 Modeling Challenges and Issues

For LCR systems to be successfully deployed in practice, not only should they be accurate and efficient, but they should meet the expectation of their target users (i.e., legal professionals and laymen who seek legal advice). With this goal in mind, we discuss seven modeling challenges for LCR researchers. The first four concern accuracy, the fifth one concern efficiency, and the last two concern improving user confidence and satisfaction.

1. Identifying the relevant portions of a case To determine whether two cases are similar, what matters the most are the *case elements*, which are information types and their values that are useful for predicting the court's outcome. For a theft event, what matters would be the type of theft (e.g., stealing someone's wallet, breaking into someone's house), the number of thieves, the items stolen and the total value. However, a case may contain lots of information that is irrelevant to or even misleading for determining case relevance. Developing models that can focus on the important pieces of information in a case is a challenge in LCR research.

2. Exploiting legal knowledge Legal knowledge is often needed to determine whether two cases are similar. For example, certain portions of a case may be described in legal terms, so a LCR system may need to possess legal knowledge in order to understand a case. As another example, external knowledge (i.e., information not present in a case) may be needed to determine case similarity, as discussed previously where the statue(s) applicable to a case would be needed to determine whether the two cases in Figure 1(a) are similar. Identifying the types of legal knowledge that would be useful for LCR as well as possible ways to acquire such knowledge represent another research challenge.

3. Processing complex cases Some cases are complex in that they involve multiple events. Figure 1(b) shows two complex cases, each of which contains four events described in different sentences. Processing complex cases and determining their relevance to other cases are challenging for at least two reasons. First, complex cases tend to be long, and they may be longer than what can be typically handled by state-of-the-art pre-trained language models (PLMs). Second, it may be harder for a model to understand complex cases, which in turn may make it more challenging to determine the relevance of a complex case to other cases. For instance, events follow an underlying temporal order (different from their linear sequence in the text) and may have causal or other types of relationships between them.

4. Modeling time An inherent challenge associated with LCR concerns the streaming nature of evoluation of legal documents with time. For example, statutes may change over time, so cases that were manually identified as similar according to the statutes 10 years ago may no longer be considered similar according to today's statutes. As a specific example, Canada abolished the death penalty in 1976, and criminals that should have been sentenced to death under the circumstances of precedents before 1976 can escape death punishments after 1976. The relevant question, then, is

[2]To our knowledge, there are only two recent surveys that are related to LCR. See Appendix B for a discussion of the differences between these surveys and ours.

Name	Language	Jurisdiction	# of queries	# of candidate cases/query	# of relevant cases/query
FIRE-IRLeD2017 (Mandal et al., 2017)	English	India	200	2000	5
COLIEE 2021 (Rabelo et al., 2022a)	English	Canada	900	4415	4.73
COLIEE 2022 (Kim et al., 2022a)	English	Canada	1198	2963	4.56
COLIEE 2023 (Li et al., 2023c)	English	Canada	1278	3635	4.18
CAIL2019-SCM (Xiao et al., 2019)	Chinese	China	8964	2	1
LeCaRD (Ma et al., 2021b)	Chinese	China	107	100	10.33

Table 1: Comparison of several popularly used corpora for Legal Case Retrieval.

whether LCR models trained on specific time snapshot of documents are temporally robust. If not, the challenge would be to develop temporally robust models that are equipped with mechanisms for mitigating temporal degradation of model performance over new documents. This time problem gets exacerbated when coupled with various phenomena involved such as overruling of precedences, where a historical case is not considered relevant in the current timestamp although it deals with the same foci of dispute as the query case.

5. Ensuring efficiency While traditional retrieval models such as TF-IDF can rank a large number of cases efficiently, the same is not true for complex learners, such as machine-learned rankers. For instance, it is computationally more expensive for these rankers to rank 1000 cases than 10 cases. The worse issue is that legal texts are basically much longer than texts in other domains. How to build complex LCR models that are not only accurate but also efficient is a research challenge.

6. Enabling interpretability Interpretability concerns the ability of a user to understand the reason(s) behind the output of a model. In the context of LCR, an interpretable model would output not only a judgment of whether two cases are similar or not but also an explanation of why they are (dis)similar. This explanation should be written in natural language and in a way that can be understood even by laymen. Constructing interpretable LCR models is important as the explanation provided by a model can increase a user's confidence in it. While there is existing work on building interpretable models, how to build interpretable models that can generate text describing potentially complex logical reasoning steps, which is what a LCR model should ideally be able to do, is a research challenge in light of today's AI technologies.

7. Enabling interactivity When laymen use LCR systems, they often struggle to provide a professional description of a query case, leading to poor retrieval results and user frustration. A key challenge then is to design LCR systems that can interact with users so that the system can provide guidance on how the user's goal can be successfully accomplished, possibly in an iterative fashion.

3 Corpora

In this section, we present six corpora that have been widely used for training and evaluating LCR systems. Table 1 compares these six corpora along five dimensions: (1) the language, (2) the jurisdiction, (3) the total number of queries, (4) the average number of candidate historical cases per query, and (5) the average number of relevant cases per query. Some of these corpora include annotations for not only LCR but also other Legal AI-related tasks. For instance, COLIEE 2021–2023 include annotations that support not only LCR but also Legal Case Entailment (LCE), the task of identifying a specific paragraph from a given supporting case that entails the decision for the query case.

3.1 Dataset Construction Procedure

To understand the limitations in these datasets, we first describe how they are constructed.

Datasets from common law justifications. In these datasets, similarity judgments are typically derived automatically from the citations: two cases are similar if and only if one cites the other.

Cases in COLIEE2021–2023 are drawn from an existing collection of predominantly Federal Court of Canada case law. All query cases and their corresponding precedent cases are provided as a candidate case pool for retrieval.

FIRE-IRLeD2017 contains cases from Indian Supreme Court. Unlike in COLIEE2021–2023, only the precedent cases along with 1000 randomly chosen cases that are not cited by query cases are provided as a candidate pool. Note negative cases in both datasets are manually selected, with cases other than the precedents serving as negative ones.

Datasets from civil law jurisdictions. In these datasets (including CAIL2019-SCM and LeCaRD [3], which are collected from the Supreme People's Court of China), relevance labels have to be provided by legal experts.

LeCaRD first collects the 100 most similar candidate cases from a case pool for each query case based on the similarity scores generated by three IR models (i.e., TF-IDF, BM25 and Language Modeling). Then, human annotators decide which cases are (dis)similar to the query case. All annotators are trained to follow a unified annotation guidance, achieving an agreement of 0.5 in Fleiss's kappa.

Similarly, CAIL2019-SCM first leverages TF-IDF to collect two candidates from a case pool for each query, and the task involves determining which of them is more similar to the query case. Note that even the more similar case may not necessarily be considered similar by a legal expert. It is not clear whether the annotators are trained before annotating and what the annotator agreement is.

3.2 Dataset Limitations

Below we discuss limitations of existing datasets.

Annotation bias. Datasets from common law jurisdictions are typically annotated automatically using citations, as noted before. Ideally, a historical case is cited because of its similarity of facts to the query case. In practice, however, a case is cited for reasons other than factual similarity. It is known that inconsistent practices have been employed in interpretations and that citations get affected by subjective interpretations (Lewis, 2021; Shao et al., 2022). Moreover, there are studies that show that it is not uncommon for judges to be biased when citing cases (Sutton, 1994; Choi and Gulati, 2008). For instance, judges are more likely to cite cases handled by judges who belong to the same political party as themselves as well as judges who cite their own cases frequently. Why a case gets cited and how many cases are cited due to subjective interpretation and personal bais remain open questions, but what we do know is that annotation bias is present in existing LCR datasets. Unfortunately, models trained on these biased annotations could make biased predictions, which is a serious ethical consideration when deploying LCR systems.

Reproducibility issues. While commonly used datasts from civil law jurisdictions (e.g., CAIL2019-SCM, LeCaRD) are annotated by multiple legal experts, the annotation guidelines are not published. For LeCaRD, inter-annotator agreement is not even reported. This is inconsistent with the reproducibility guidelines that are in use today in the NLP community for dataset creation.

[3] A new version of LeCaRD has been released (Li et al., 2023b).

Case 1: Qxx borrowed 5 million from Vxx with an agreed interest rate of 21%, and Qxx failed to repay the loan in time after the maturity date.
Case 2: Gxx borrowed 3 million from Lxx with an agreed interest rate of 7%, and Gxx failed to repay the loan in time after the maturity date.
Existing Explanation: Case 1 involved the failure to perform the contract as promised, and the interest was four times higher than the quoted market loan interest rate; Case 2 involved the failure to perform the contract as promised, and the interest was within the quoted market loan rate.
Desired Explanation: Cases 1 and 2 both involved failure to perform the contract when it expired. However, *according to the Law in the Private Lending § 25*, the interest rate in Case 1 was *four times higher than the loan market interest rate*, while Case 2 was within the legal protection rate. Case 2 had no guidance for Case 1, so they were not similar.

Figure 2: An example of explanations.

Lack of time awareness. Challenge 4 (see Section 2) involves modeling time. Ideally, LCR datasets can support the development of time-aware models. Such datasets would be composed of mutiple specific timeshots of annotated cases rather than a single one, so that models can learn how relevance judgments can change with the evolution of statutes and other legal documents over time. Unfortunately, the construction of existing datasets does not take time into account.

Lack of datasets with laymen-readable explanations. Datasets where similarity judgments are accompanied by human explanations can facilitate the development of interpretable models, but few LCR datasets come with explanations. Even for those that do, the explanations cannot be easily understood by laymen. Consider Figure 2, which shows two cases followed by two different explanations. The first explanation, which is provided by Yu et al. (2022) in their dataset, can be understood by legal experts given their legal knowledge, but it cannot be easily understood by laymen. The reason is that the explanation (as well as all other explanations provided in this corpus) is presented in two sentences, one for each case. In other words, there is no explicit discussion of how the facts in the two cases are related to each other that could allow one to conclude whether the two cases are similar. The second explanation, which is written by us, shows what we think an ideal explanation would look like: every statement is supported by the relevant law whenever applicable and the reasoning steps that lead to the conclusion is written in a way that can

be understood by laymen.

Improper usage. This is not a dataset limitation per se. Rather, LCR researchers have improperly used the information in these datasets. By construction, all the case documents in a dataset, including the query cases, contain all sections of a case, such as the facts, the court's decision, and the reasoning behind this decision. When applying their LCR models to identify the candidate cases that are most similar to a query case, some researchers have included in the query case all of its sections. However, in a realistic scenario, models are typically applied to a query case prior to the final verdict, meaning that researchers should have used only the facts and not the court's decision or the reasoning behind it when training and applying LCR models.

4 Evaluation Metrics

Existing LCR models are either *classification* models, which determine whether a query case and a candidate case are similar, or *ranking* models, which rank a set of candidate cases in terms of their relevance to a query case. Several metrics have been developed to evaluate LCR models.

To evaluate ranking-based models, the metrics used include Accuracy@K, Precision@K, Recall@K, F1@K, NDCG@K (Normalized Discounted Cumulative Gain (Zhu et al., 2022)), and MAP (Mean Average Precision (Tran et al., 2019)), all of which evaluate performance based on the top K cases retrieved by a model (Shao et al., 2020a; Nguyen et al., 2022; Ma et al., 2021b; Li et al., 2021). To evaluate classification-based models, the metrics used include Accuracy as well as micro-averaged Precision, Recall, and F1 (Hong et al., 2020; Xiao et al., 2019; Li et al., 2022; Fang et al., 2022; Peng et al., 2020). To facilitate the comparison of a classification model and a ranking model, researchers have (1) computed Accuracy@K, Precision@K, Recall@K, and F1@K values of the ranking model by setting K to the number of similar historical cases a query case has on average in the evaluation corpus, and then (2) compared the resulting scores directly with the Accuracy, Precision, Recall, and F1 scores achieved by the classification model (Shao et al., 2020b; Ma et al., 2021b).

While Accuracy@K, Precision@K, Recall@K and F1@K can be computed efficiently, they do not consider the order of results and therefore are unable to differentiate models with poor ranking ability from those with better ranking ability. In contrast, NDCG@K and MAP consider the order of results. Compared to MAP, NDCG@k is more sensitive to rank order because it takes into account the position of the relevant items in the ranked list.

5 Approaches to LCR

In this section, we present an overview of existing approaches to LCR.

5.1 Traditional Approaches

In traditional approaches to IR, a case is typically represented using (1) *lexical statistical* features such as n-grams and skipgrams (Kumar et al., 2011; Salton and Buckley, 1988), (2) *hand-crafted* features (Zeng et al., 2007; Li et al., 2023c), and/or (3) *embeddings*, where a case is encoded using a doc2vec model (Sarsa and Hyvönen, 2020; Kulkarni et al., 2017) or a Transformer-based pre-trained language model (Vold and Conrad, 2021; Kim et al., 2022b). Using this representation, the candidate cases that are most similar to the query case can be obtained via one of two approaches. In non-learning-based approaches, text retrieval models such as the Vector Space Model (Salton et al., 1975) and BM25 (Robertson et al., 2009) are used. In contrast, learning-based approaches are either *classification*-based, where the model determines whether a query case and a candidate case is similar (Liu et al., 2009; Hofmann et al., 2013), or *ranking*-based, where the model ranks a set of candidate cases by their relevance to the query case (Wang et al., 2018; Ma et al., 2022; Cao et al., 2007). While non-learning-based approaches are superior in *efficiency*, learning-based approaches are more *accurate* in identifying similar cases. Learning-based approaches outperform BM25 on LeCaRD by nearly 10% in Precision score (Ma et al., 2021b).

5.2 Neural Approaches

Basic models. Early neural LCR models differ from traditional learning-based approaches primarily in terms of how a case is represented. Specifically, a case is represented as a sequence of words, each of which is encoded as a word vector. The resulting sequence is then encoded using encoders such as an LSTM (Liu et al., 2022b; Nanda et al., 2017), BERT (Shao et al., 2021a; Vuong et al., 2022), and RoBERTa (Li et al., 2023a).

Attention. To address Challenge 1 (see Section 2), researchers have employed attention to identify the characters, words, and phrases of a

case that are important for LCR by developing fine-grained attention mechanisms at the character level (Hong et al., 2020) and the word level (Mou et al., 2021; Sivaranjani and Jayabharathy, 2022) for improving Chinese and English LCR respectively.

Paragraph/Sentence-level approaches. To address Challenge 3, researchers examined paragraph/ sentence-level LCR. Rabelo et al. (2022b), for instance, (1) compute the similarity between each paragraph/sentence in a candidate and each paragraph/sentence in the query (using cosine similarity), then (2) use all these pairwise similarity values as features to train a model to determine if two cases are similar at the document level. Paragraph/sentence-level LCR enables us to (partially) address (1) the issue of *long* documents, as the unit of comparison is a paragraph/sentence rather than a document; and (2) the complexity that arises from a case covering multiple events. Consider again Figure 1(b), where both the query case and the candidate case contain four events described in different sentences. Document-level approaches could misclassify these two cases as similar because three of the four events are the same. In contrast, it may be easier for a sentence-level approach to (correctly) classify them as dissimilar when matching the sentence containing the SURRENDER event in the query with the sentence containing the ESCAPE event in the candidate.

Coarse-to-fine approaches. Paragraph/sentence-level LCR approaches are computationally expensive, as the number of relevance computations that needs to be performed for paragraphs/sentences can be much larger than that for documents. To address this efficiency concern (Challenge 5), researchers have adopted a *coarse-to-fine* strategy (Ma et al., 2021a; Li et al., 2023c). where they (1) employ an efficient technique (e.g., an IR-based model such as BM25) to produce a coarse-grained ranking of the candidate paragraphs/sentences for each query paragraph/sentence, filtering the low-ranked candidates; and then (2) produce a fine-grained ranking of the remaining candidates using a neural model.

Knowledge-rich approaches. To address Challenge 2, researchers have developed knowledge-rich approaches where four types of knowledge have been exploited to determine case similarity.

Knowledge of the statutes in the target jurisdiction: Knowledge of the statutes is sometimes required for determining case similarity. To exploit statute information, Fink et al. (2023) augment each case document with the text of each statute applicable to it and compute case similarity using this augmented representation. Bhattacharya et al. (2020) first construct a graph where each node corresponds to either a case or a statute and an edge connects either (1) two case nodes if one cites the other or (2) a case node and a statute node if the statute is applicable to the case, then use node2vec to compute node embeddings, which allow statute information to be integrated into a case.[4]

Domain-specific knowledge of language: SOTA PLMs such as BERT possess *general* knowledge about language. To acquire *legal* knowledge of language, Xiao et al. (2021) have proposed Lawformer, a PLM pre-trained on Chinese civil and criminal legal documents with the goal of acquiring legal-specific knowledge of Chinese civil and criminal cases that could be useful for various legal-related tasks, including LCR. Lawformer and other models that are pre-trained on legal texts (e.g., CLC-RS (Li et al., 2021), LEGAL-BERT (Chalkidis et al., 2020)) have achieved superior performance to their counterparts that are not pre-trained on legal texts (Xiao et al., 2021)).

Knowledge of other legal-related tasks: Knowledge from other legal-related tasks, such as Legal Case Entailment (LCE), could be profitably exploited for LCR. Specifically, Shao et al. (2020b) use LCE-annotated data to fine-tune BERT so that the resulting model, BERT-PLI, can produce better representations of the query and the candidates.

Knowledge provided by lexical knowledge bases: Sometimes the description of a query case can be short, thus making accurate matching difficult. Consequently, researchers have proposed using *query expansion* to improve the retrieval of cases that are similar to the query. Query expansion is a traditional IR technique where the words that are semantically related to those in the initial query are used to augment the query so that the augmented query will improve retrieval results. For instance, Catacora et al. (2022) construct LegalBase, a legal-specific knowledge base, and use it as a source of information to expand a query.[5]

Interpretable approaches. To address Challenge 6, researchers have developed interpretable LCR models, which not only determine case rele-

[4]See Appendix C for details on how knowledge of the statutes are represented and used.

[5]See Appendix D for an overview of LegalBase.

	SOTA	Results	System Description	Strengths	Weaknesses
FIRE-2017	Sampath and Durairaj (2022)	0.632 MAP	(1) trains a sequence-to-sequence model to extract case elements from cases annotated with case elements; (2) uses the extracted case elements to compute semantic and statistical similarity features with a CNN-based module; and (3) feeds these features into a binary classifier to determine if the given case pair is similar or not.	(1) measuring similarity based on case elements rather than documents allows some pairs to be correctly identified due to noise reduction; (2) case element knowledge is exploited; and (3) semantic and statistical features are used to measure similarity.	(1) some (dis)similar pairs cannot be identified simply based on case elements; and (2) errors in case element extraction would propagate to the later modules; and (3) employing a classifier rather than a ranker does not allow candidate cases to be compared against each other, resulting in errors in some cases.
COLIEE21	Ma et al. (2021a)	0.153 P 0.256 R 0.192 F	(1) samples the top-N candidates from the whole candidate pool by a traditional retrieval model (LMIR); (2) fine-tunes a BERT model using the NSP task on a case-entailment dataset to identify whether a paragraph entails another paragraph; (3) divides each case document into paragraphs and use the above fine-tuned BERT model as the encoder to derive paragraph representations from the given case pair; and (4) uses the representations to calculate the similarity of the given pair with a fully connected binary classifier layer.	(1) a coarse-to-fine approach is used to improve efficiency; (2) paragraphs rather than documents are used to obtain fine-grained similarity; and (3) paragraph entailment knowledge and attention are exploited to better construct correlations between paragraphs.	(1) some similar pairs are erroneously filtered by LMIR; and (2) employing a pairwise classification model rather than a ranker does not allow candidate cases to be compared against each other, resulting in errors in certain cases.
COLIEE22	Rabelo et al. (2022b)	0.411 P 0.339 R 0.372 F	(1) removes useless information from cases; (2) divides documents into paragraphs and uses a transformer-based model to generate paragraph embeddings; (3) calculates the similarity between paragraphs from the given case pair; (4) uses these similarities to generate feature vectors (10-bin histograms of all pair-wise comparisons between 2 cases); (5) uses a classifier to determine if those cases should be noticed or not; and (6) uses several post-processing methods to filter resulting cases.	(1) simple but effective simple pre- and pro-processing operations are employed to filter irrelevant pairs; and (2) documents are divided into paragraphs to obtain fine-grained similarity.	(1) some pairs are misclassified due to the fact that the method does not exploit extra entailment annotation knowledge; (2) employing a pairwise classification model rather than a ranker does not allow candidate cases to be compared against each other, resulting in errors in certain cases.
COLIEE23	Li et al. (2023c)	0.238 P 0.406 R 0.300 F	(1) pre-processes cases by removing useless information, extracting summaries; (2) uses different traditional IR (TF-IDF, BM25, etc.) methods to get different types of lexical relevance features; (3) uses a pre-trained language model which pre-trained on a extra case dataset using typical mask language task to get semantic relevance features; (4) uses LightGBM to integrate all features into the final score (5) performs different post-processing strategies (filtering by trial date, filtering query cases) for a more accurate ranking.	(1) simple pre- and pro-processing operations are employed to filter irrelevant pairs; (2) a coarse-to-fine approach is used to achieve high efficiency and effectiveness; and (3) different types of lexical and semantic features are used to measure similarity.	(1) some pairs are misclassified because the method does not exploit extra entailment annotation knowledge; (2) some similar pairs are erroneously filtered by the IR models; and (3) employing a pairwise classifier rather than a ranker does not allow candidate cases to be compared against each other, resulting in errors in some cases.
CAIL19SCM	Bi et al. (2022)	0.739 F	(1) constructs a Legal Hybrid Knowledge Network (i.e., a knowledge graph where each node corresponds to either a legal entity (e.g., Fraud, Theft)) or a legal case, and two nodes are connected by an edge if they are related to each other (e.g., a legal entity is connected to a legal case if it is mentioned in the case); (2) adds the given cases to the graph; (3) generates embedding vectors for these cases where each legal entity mentioned in each case is augmented with its definition extracted from the corresponding nodes in the graph; and (4) uses the cosine similarity of the resulting vectors to determine if the two cases are similar.	(1) simple but effective simple pre- and pro-processing operations are employed to filter irrelevant pairs; (2) a coarse-to-fine approach is used to achieve high efficency and effectiveness; and (3) different types of lexical and semantic features are used to measure case similarity.	(1) some similar pairs are erroneously filtered by the IR models; and (2) employing a pairwise classification model rather than a ranker does not allow candidate cases to be compared against each other, resulting in errors in certain cases.
LeCaRD	Zhu et al. (2022)	0.662 MAP	(1) uses BM25 to extract the top K candidate cases for a query; (2) fine-tunes a BERT model using two query classification tasks (where the query type labels are available in LeCaRD, an auxiliary task-based model) and use the resulting model to encode the query and its top K candidates; and (3) adopts a multi-layer neural network to rank the K candidates via a pair-wise strategy.	(1) extra knowledge is learned from query classification annotations; (2) a ranker rather than classifier is learned, allowing different candidates to be compared against each other.	(1) some similar pairs are erroneously filtered by the BM25; and (2) employing a pairwise classifier rather than a ranker does not allow candidate cases to be compared against each other, resulting in errors in certain cases.

Table 2: State-of-the-art LCR systems on different datasets: their results, strengths and weaknesses.

vance but also provide an explanation for its decision. So far two types of interpretable LCR models have been proposed. If the training data does not contain any hand-annotated explanations, then attention is used for explaining the model's decision, where the words/phrases associated with large attention weights are used as the explanation (Hong et al., 2020). In contrast, if the training cases are hand-annotated with explanations, a model can be trained to *simultaneously* predict similarity and generate explanations (Yu et al., 2022),

Interactive approaches. To address Challenge 7, researchers have proposed an agent-based system where conversational agents ask clarification questions to the user about the user-input query and then use the user feedback to improve the quality of a query (e.g., by rewriting or expanding the query) (Liu et al., 2021, 2022a; Shao et al., 2021b).

Citation-based approaches. Citation-based approaches aim to compute the similarity of cases via citations and, unlike other LCR approaches, they cannot be used to *retrieve* the cases that are most similar to a new query case. Given that they are not central to our discussion of LCR approaches, we defer their discussion to Appendix E.

6 The State of the Art

In Table 2, we present an overview of the systems that have achieved SOTA results on the six corpora described in Section 3 and analyze their strengths and weaknesses. These results suggest that SOTA models have a lot of room for improvement.[6]

7 Ethical Considerations

Several ethical considerations should be taken into account when deploying LCR systems.

Using LCR systems. LCR systems should be designed with the goal of assisting rather than replacing legal professionals and offering consulting suggestions to people who possess little legal knowledge. Like other AI systems, LCR systems are only as good as the data on which they are trained. As mentioned before, bias may exist in the annotations, especially those that are automatically derived, potentially leading to bias in model prediction. Hence, legal decisions should always be made by professionals rather than LCR systems.

Debiasing data. The output of LCR systems may affect the court's decisions. As a result, fairness and justice are important principles underlying the development of LCR systems. To avoid having LCR systems make biased predictions, it is important that these systems are trained on data instances that are not biased. For LCR, we believe that there are at least two sources of data biases. First, the data may contain information that should probably not be used when determining similarity (e.g., the name, gender, age, and/or race of the person involved in a case). To ensure that such information is not exploited by LCR systems, the data should be properly anonymized. Second, the data may not have been properly annotated. For manually annotated datasets, some human annotators may be racially, sexually, or even politically biased, and hence the annotations they produce could be biased. Standard annotation procedures could be used to address this problem. For example, biased annotators could be identified using qualification tests, and biased annotations could be identified if inter-annotator agreement was low. For datasets automatically annotated via citations, citation bias exist. To improve annotation quality, we recommend that a legal expert provide an independent relevance judgment for each data instance.

Mimicking human reasoning. From an ethical perspective, it is critical for a LCR system to follow the human reasoning process when making similarity judgments. In other words, it is not acceptable for a LCR system to provide the right answer (similarity judgment) for the wrong reason. This makes it all the more important to develop interpretable models for providing explanations that can be easily understood even by laymen.

8 Concluding Remarks

In this section, we conclude with several promising directions for LCR research.

Understanding experts' notion of similarity. A key issue surrounding LCR research is that the notion of case similarity according to legal experts remains somewhat elusive to LCR researchers. Current datasets from common law jurisdictions have been constructed automatically from their citations. While we know that a historical case is cited for reasons that go beyond its factual similarity to the

[6]Compared to other legal tasks such as Legal Judgment Prediction (LJP) (Liu et al., 2023), LCR is a task that is far from being solved. LCR only achieves a maximum of 73.9% in F1 score, while LJP obtains over 90% in F1 score.

query case, such as personal bias, we do not know exactly what these reasons are. It would be worth studying the intent beyond such citations and develop a taxonomy of "why a document gets cited".

In contrast, existing datasets from civil law jurisdictions are annotated by legal experts. However, none of them comes with annotation guidelines, so it is not clear what guidelines were provided to the human annotators. Going forward, we encourage dataset creators to publish the notions of similarity to experts. Further down the road, experts should provide an explanation for each similarity judgment they make. Not only will the resulting explanations enable LCR researchers to better understand experts' notion of similarity, but they will facilitate the training of interpretable LCR models.

Developing ethical, interpretable, temporally-robust and interactive LCR systems. Work on building LCR systems that are intepretable and interactive and which can make unbiased and ethical decisions is still in its infancy. In addition, there have been no attempts to develop temporally robust models that are equipped with mechanisms for mitigating temporal degradation of model performance over new documents, as well as models that address inherent challenges such as overruling of precedences. These are promising research directions that deserve attention from LCR researchers.

Limitations

This paper has a limitation. We paid more attention to deep learning-based LCR approaches than traditional IR-based ones, for several reasons. First, given that there is a page limit, it is virtually impossible to provide a detailed overview of both deep learning-based approaches and traditional IR-based approaches, so we decided to focus on one of them. Second, since LCR has always been a hotspot in AI for the judicial field, new LCR approaches are constantly emerging with the continuous development of AI technology. In recent years, deep learning techniques outperform traditional IR-based methods a lot on LCR task. Almost all recent LCR solutions are deep learning-based. So, we decided to focus on deep learning-based LCR more than IR-based ones. Finally, as mentioned in the introduction, traditional IR-based methods have received a fair coverage in a recent survey on Case Law Retrieval (Locke and Zuccon, 2022).

Acknowledgments

We thank all the reviewers for their valuable comments on the earlier draft of this paper. This work was supported by the Overseas Project (KFKT2023A07, KFKT2024A07) and the General Innovation Project (ZZKT2024B02) of the National Key Laboratory for Novel Software Technology in China.

References

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. In *Proceedings of the 43rd SIGIR*, pages 1657–1660.

Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2022. Legal case document similarity: You need both network and text. *Information Processing & Management*, 59(6):103069.

Sheng Bi, Zafar Ali, Meng Wang, Tianxing Wu, and Guilin Qi. 2022. Learning heterogeneous graph embedding for chinese legal document similarity. *Knowledge-Based Systems*, 250:109046.

David C Blair and Melvin E Maron. 1985. An evaluation of retrieval effectiveness for a full-text document-retrieval system. *Communications of the ACM*, 28(3):289–299.

Joost Breuker, André Valente, and Radboud Winkels. 2004. Legal ontologies in knowledge engineering and information management. *Artificial intelligence and law*, 12:241–277.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136.

Joel Arnaldo Gimenez Catacora, Ana Casali, and Claudia Deco. 2022. Legal information retrieval system with entity-based query expansion: Case study in traffic accident litigation. *Journal of Computer Science and Technology*, 22(2):e12–e12.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: "preparing the muppets for court'". In *Findings of EMNLP*, pages 2898–2904.

Stephen J. Choi and G. Mitu Gulati. 2008. Bias in judicial citations: A window into the behavior of judges? *The Journal of Legal Studies*, 37(1):87–129.

Soufiane El Jelali, Elisabetta Fersini, and Enza Messina. 2015. Legal retrieval as support to emediation: matching disputant's case and court decisions. *Artificial Intelligence and Law*, 23:1–22.

Jingxin Fang, Xuwei Li, and Yiguang Liu. 2022. Low-resource similar case matching in legal domain. In *Proceedings of the 31st ICANN*, pages 570–582.

Tobias Fink, Gábor Recski, Wojciech Kusa, and Allan Hanbury. 2023. Statute-enhanced lexical retrieval of court cases for COLIEE 2022. *CoRR*, abs/2304.08188.

Katja Hofmann, Shimon Whiteson, and Maarten de Rijke. 2013. Balancing exploration and exploitation in listwise and pairwise online learning to rank for information retrieval. *Information Retrieval*, 16:63–90.

Zhilong Hong, Qifei Zhou, Rong Zhang, Weiping Li, and Tong Mo. 2020. Legal feature enhanced semantic matching network for similar case matching. In *Proceedings of IJCNN*, pages 1–8.

Mi-Young Kim, Juliano Rabelo, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2022a. Coliee 2022 summary: Methods for legal document retrieval and entailment. In *JSAI International Symposium on Artificial Intelligence*, pages 51–67. Springer.

Mi-Young Kim, Juliano Rabelo, Kingsley Okeke, and Randy Goebel. 2022b. Legal information retrieval and entailment based on bm25, transformer and semantic thesaurus methods. *The Review of Socionetwork Strategies*, 16(1):157–174.

Yogesh H Kulkarni, Rishabh Patil, and Srinivasan Shridharan. 2017. Detection of catchphrases and precedence in legal documents. In *FIRE (Working Notes)*, pages 86–89.

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*, page 17.

Sebastian Lewis. 2021. Precedent and the rule of law. *Oxford journal of legal studies*, 41(4):873–898.

Haitao Li, Qingyao Ai, Jia Chen, Qian Dong, Yueyue Wu, Yiqun Liu, Chong Chen, and Qi Tian. 2023a. Sailer: Structure-aware pre-trained language model for legal case retrieval. *arXiv preprint arXiv:2304.11370*.

Haitao Li, Yunqiu Shao, Yueyue Wu, Qingyao Ai, Yixiao Ma, and Yiqun Liu. 2023b. Lecardv2: A large-scale chinese legal case retrieval dataset. *CoRR*, abs/2310.17609.

Haitao Li, Weihang Su, Changyue Wang, Yueyue Wu, Qingyao Ai, and Yiqun Liu. 2023c. Thuir@ coliee 2023: Incorporating structural knowledge into pre-trained language models for legal case retrieval. *arXiv preprint arXiv:2305.06812*.

Hui Li, Jin Lu, Yuquan Le, and Jiawei He. 2022. Iacn: Interactive attention capsule network for similar case matching. *Intelligent Data Analysis*, 26(2):525–541.

Jieke Li, Min Yang, and Chengming Li. 2021. Clc-rs: A chinese legal case retrieval system with masked language ranking. In *Proceedings of the 30th CIKM*, pages 4734–4738.

Bulou Liu, Yueyue Wu, Yiqun Liu, Fan Zhang, Yunqiu Shao, Chenliang Li, Min Zhang, and Shaoping Ma. 2021. Conversational vs traditional: Comparing search behavior and outcome in legal case retrieval. In *Proceedings of the 44th SIGIR*, pages 1622–1626.

Bulou Liu, Yueyue Wu, Fan Zhang, Yiqun Liu, Zhihong Wang, Chenliang Li, Min Zhang, and Shaoping Ma. 2022a. Query generation and buffer mechanism: Towards a better conversational agent for legal case retrieval. *Information Processing & Management*, 59(5):103051.

Jianping Liu, Xintao Chu, Yingfei Wang, and Meng Wang. 2022b. Deep text retrieval models based on dnn, cnn, rnn and transformer: A review. In *2022 IEEE 8th International Conference on Cloud Computing and Intelligent Systems (CCIS)*, pages 391–400. IEEE.

Tie-Yan Liu et al. 2009. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331.

Yifei Liu, Yiquan Wu, Yating Zhang, Changlong Sun, Weiming Lu, Fei Wu, and Kun Kuang. 2023. Mlljp: Multi-law aware legal judgment prediction. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1023–1034.

Daniel Locke and Guido Zuccon. 2022. Case law retrieval: problems, methods, challenges and evaluations in the last 20 years. *arXiv preprint arXiv:2202.07209*.

Yixiao Ma, Qingyao Ai, Yueyue Wu, Yunqiu Shao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2022. Incorporating retrieval information into the truncation of ranking lists for better legal search. In *Proceedings of SIGIR*, pages 438–448.

Yixiao Ma, Yunqiu Shao, Bulou Liu, Yiqun Liu, Min Zhang, and Shaoping Ma. 2021a. Retrieving legal cases from a large-scale candidate corpus. *Proceedings of the Eighth International Competition on Legal Information Extraction/Entailment, COLIEE2021*.

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

Arpan Mandal, Kripabandhu Ghosh, Arnab Bhattacharya, Arindam Pal, and Saptarshi Ghosh. 2017. Overview of the fire 2017 irled track: Information retrieval from legal documents. In *Proceedings of the 9th FIRE*, pages 63–68.

Akshay Minocha, Navjyoti Singh, and Arjit Srivastava. 2015. Finding relevant indian judgments using dispersion of citation network. In *Proceedings of the 24th WWW*, pages 1085–1088.

Marie-Francine Moens. 2001. Innovative techniques for legal text retrieval. *Artificial Intelligence and Law*, 9:29–57.

Mohammad Ali Montazeri, Mike Brown, and Alison E. Adam. 1994. Cross structural similarity for retrieval of legal cases. In *Proceedings of ICLP Workshop*.

Yanlin Mou, Yang Weng, Songyuan Gu, and Zhu Wang. 2021. Semantic matching of legal cases for large scale judgments of china. In *Proceedings of the 16th ICCSE*, pages 1120–1125.

Rohan Nanda, Kolawole John Adebayo, Luigi Di Caro, Guido Boella, and Livio Robaldo. 2017. Legal information retrieval using topic clustering and neural networks. In *COLIEE@ ICAIL*, pages 68–78.

Ha-Thanh Nguyen, Phi Manh Kien, Ngo Xuan Bach, Vu Tran, Le-Minh Nguyen, and Tu Minh Phuong. 2022. Attentive deep neural networks for legal document retrieval. *CoRR*, abs/2212.13899.

Dunlu Peng, Jiyin Yang, and Jing Lu. 2020. Similar case matching with explicit knowledge-enhanced text representation. *Applied Soft Computing*, 95:106514.

Jay M. Ponte and W. Bruce Croft. 1998. A language modeling approach to information retrieval. SIGIR '98, page 275–281.

Juliano Rabelo, Randy Goebel, Mi-Young Kim, Yoshinobu Kano, Masaharu Yoshioka, and Ken Satoh. 2022a. Overview and discussion of the competition on legal information extraction/entailment (coliee) 2021. *The Review of Socionetwork Strategies*, 16(1):111–133.

Juliano Rabelo, Mi-Young Kim, and Randy Goebel. 2022b. Semantic-based classification of relevant case law. In *JSAI International Symposium on Artificial Intelligence*, pages 84–95. Springer.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

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

Gerard Salton, Anita Wong, and Chung-Shu Yang. 1975. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Kayalvizhi Sampath and Thenmozhi Durairaj. 2022. Prelcap: Precedence retrieval from legal documents using catch phrases. *Neural Processing Letters*, 54(5):3873–3891.

Carlo Sansone and Giancarlo Sperlí. 2022. Legal information retrieval systems: State-of-the-art and open issues. *Information Systems*, 106:101967.

Sami Sarsa and Eero Hyvönen. 2020. Searching case law judgments by using other judgments as a query. In *Artificial Intelligence and Natural Language: 9th Conference, AINL 2020, Helsinki, Finland, October 7–9, 2020, Proceedings 9*, pages 145–157. Springer.

Hsuan-Lei Shao, Yi-Chia Chen, and Sieh-Chuen Huang. 2021a. Bert-based ensemble model for statute law retrieval and legal information entailment. In *New Frontiers in Artificial Intelligence: JSAI-isAI 2020 Workshops, JURISIN, LENLS 2020 Workshops, Virtual Event, November 15–17, 2020, Revised Selected Papers 12*, pages 226–239. Springer.

Yunqiu Shao, Bulou Liu, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2020a. Thuir@coliee-2020: Leveraging semantic understanding and exact matching for legal case retrieval and entailment. *arXiv preprint arXiv:2012.13102*.

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

Yunqiu Shao, Yueyue Wu, Yiqun Liu, Jiaxin Mao, and Shaoping Ma. 2022. Understanding relevance judgments in legal case retrieval. *ACM Transactions on Information Systems*.

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

Olga Shulayeva, Advaith Siddharthan, and Adam Zachary Wyner. 2017. Recognizing cited facts and principles in legal judgements. *Artif. Intell. Law*, 25(1):107–126.

N Sivaranjani and J Jayabharathy. 2022. Neural network towards uncertain legal case retrieval. *Journal of Uncertain Systems*, 15(02):2241001.

Stuart A Sutton. 1994. The role of attorney mental models of law in case relevance determinations: An exploratory analysis. *Journal of the American Society for Information Science*, 45(3):186–200.

Vu Tran, Minh Le Nguyen, Satoshi Tojo, and Ken Satoh. 2020. Encoded summarization: summarizing documents into continuous vector space for legal case retrieval. *Artificial Intelligence and Law*, 28(4):441–467.

Vu Tran, Minh L. Nguyen, and Ken Satoh. 2019. Building legal case retrieval systems with lexical matching and summarization using a pre-trained phrase scoring model. In *Proceedings of the 17th ICAIL*, pages 275–282.

Marc Van Opijnen and Cristiana Santos. 2017. On the concept of relevance in legal information retrieval. *Artificial Intelligence and Law*, 25(1):65–87.

Andrew Vold and Jack G Conrad. 2021. Using transformers to improve answer retrieval for legal questions. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, pages 245–249.

Yen Thi-Hai Vuong, Quan Minh Bui, Ha-Thanh Nguyen, Thi-Thu-Trang Nguyen, Vu Tran, Xuan-Hieu Phan, Ken Satoh, and Le-Minh Nguyen. 2022. Sm-bert-cr: a deep learning approach for case law retrieval with supporting model. *Artificial Intelligence and Law*, pages 1–28.

Pengfei Wang, Ze Yang, Shuzi Niu, Yongfeng Zhang, Lei Zhang, and ShaoZhang Niu. 2018. Modeling dynamic pairwise attention for crime classification over legal articles. In *the 41st international ACM SIGIR conference on research & development in information retrieval*, pages 485–494.

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.

Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Tianyang Zhang, Xianpei Han, Zhen Hu, Heng Wang, et al. 2019. Cail2019-scm: A dataset of similar case matching in legal domain. *arXiv preprint arXiv:1911.08962*.

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

Yiming Zeng, Ruili Wang, John Zeleznikow, and Elizabeth Kemp. 2007. A knowledge representation model for the intelligent retrieval of legal cases. *International Journal of Law and Information Technology*, 15(3):299–319.

Junlin Zhu, Xudong Luo, and Jiaye Wu. 2022. A bert-based two-stage ranking method for legal case retrieval. In *Proceedings of the 15th KSEM*, pages 534–546.

A Additional Examples on (Dis)similar Cases

To enable the reader to better understand the notion of similarity in LCR, we discuss additional examples of (dis)similar cases in this section.

Broadly, relevant cases in the legal domain are those which support practitioners' arguments for their intended outcome during the trail process. Practitioners use relevant cases to formulate and present their arguments persuasively. This typically means that whether two cases are relevant involves three aspects: (1) legally semantic relevance, (2) legally degree relevance and (3) legally outcome relevance.

Query Case:
[Fact] Respondent Fxx breaked into the petitioner Pxx's home and stole a wallet with a total value of $ 1,000.

Historical Case 1:
[Fact] Respondent Dxx as a Walmart pocketed cash from the cash register to buy drinks at a nightclub for herself.
[Reasoning] Dxx fraudulently spent cash, which committed embezzlement charge under the California Penal Code § 503.
[Decision] By J. v. X., 111 U.S. 1321, 24-47, the respondent was sentenced to 6 months year of prison.

Historical Case 2:
[Fact] Respondent Ixx breaked into the petitioner Lxx's house, and took away Lxx's diamond, which was worth of $ 50,000.
[Reasoning] Stealing over $ 950 was prosecuted as grand theft under the California Penal Code § 487.
[Decision] By T. v. N., 110 U.S. 31, 55-60, the respondent was sentenced to 3 years of prison.

Historical Case 3:
[Fact] Respondent Jxx had a grudge against Rxx. Jxx stole Rxx's heart medicine, which caused the victim to die of a heart attack. After that, Jxx fled abroad.
[Reasoning] The theft event concerned the murder of Rxx under the California Penal Code § 187, and the respondent fled to escape the punishment of the law.
[Decision] By B. v. C., 11 U.S. 99., 10-20, the respondent was sentenced to 15 years of prison.

Figure 3: Additional examples of a query case and historical cases.

As an example, consider the query and the first historical case in Figure 3. Intuitively, both texts are similar as they describe theft events. However, the two cases are not similar because they describe different types of theft: one concerns stealing private properties (committing theft), whereas another concerns a staff member stealing public properties (committing embezzlement). In other words, the facts from the two cases are not similar from the perspective of legal semantics. In general, a historical case cannot support the query case unless the two cases are legally semantically relevant.

Next, consider the query and the second historical case. While both texts are legally semantically relevant (both events are about thefts), the two cases are not similar as the respondents commit different degrees of theft legally ($ 1,000 vs. $ 50,000). The respondent in the query case would be sentenced to 16 months in prison, whereas the respondent in the historical case would be sentenced to 3 years in prison. In general, a historical case cannot support the query case unless the two cases are legally degree relevant.

However, even if two cases are legally semantically relevant and legally degree relevant, it does not mean they are similar. As illustrated in the query and the third historical case, they share similar facts and degrees. However, they do not share the same legal outcome as the third case involves

committing a murder charge: while the respondent stole asthma medicine, the respondent was murdering the victim as he knew the victim would die without the medicine. It seems as if the legal outcomes (judgment results) were similar, then the cases would be similar. However, precedents with similar legal outcomes do not necessarily support the current case. Consider the following example. The historical case is sentenced as murder, and the query case would be sentenced as murder without doubt as the defendants all kill the victims. While they share the same legal outcomes, one case involves killing by stealing medicine and the other case involves killing by gun. The gunshot murder case provides no support for the theft murder case.

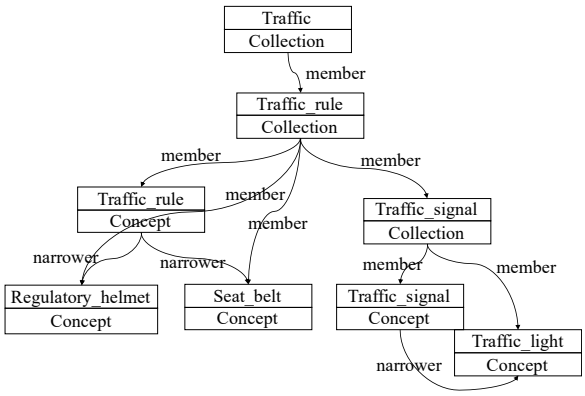

Figure 4: An illustration of the LegalBase knowledge base.

B Related Surveys

To our knowledge, there are only two recent surveys that are related to LCR. The first one is a survey on Legal IR that has the broader goal of providing a general overview of the state-of-the-art (SOTA) on various Legal IR tasks, including information extraction, document classification, ontology construction, and LCR (Sansone and Sperlí, 2022). In particular, the discussion on LCR is relatively terse covering only nine research papers published prior to 2021. The second one is a survey on Case Law Retrieval (CLR) (Locke and Zuccon, 2022). CLR is essentially LCR except that the input can be either a query case or any natural language description provided by the user who is looking for historical cases related to the description. However, this survey covers methods that are published prior to 2021, focusing primarily traditional Information Retrieval (IR) methods and devoting little attention to deep learning methods. In contrast, our survey provides a comprehensive overview of different approaches with a focus on deep learning methods, including those published in the last two years.

C Representation and Use of Statute Knowledge for LCR

To enable the reader to better understand how knowledge of statute(s) is represented and used for LCR, we discuss in this section existing work that exploit such knowledge for LCR. Before we begin, we note that to exploit knowledge of statutes, we first need to know which statute(s) are applicable to the case at hand. However, the task of identifying the statute(s) applicable to a case is a very challenging task in itself. Consequently, all the existing work that exploits statute information assumes that the statute(s) applicable to a case are already given (note that this information can be easily extracted from existing LCR datasets: the applicable statute(s) can be extracted from the Reasoning section of case document where these statute(s) are explicitly cited as part of the court's justification for its decision on the case).

Fink et al. (2023) present a method for representing and using semantic knowledge of statutes for LCR. Specifically, assuming that the statute(s) applicable to a case is given, they (1) augment the text of the query case with the text of the statute(s) applicable to the query case, (2) augment the text of the candidate case with the text of the statute(s) applicable to the candidate case; (3) use TF-IDF to represent the augmented query case and the augmented candidate case; (4) compute their similarity using BM25 using the resulting representation.

Bhattacharya et al. (2020) propose a different method to utilize statute citation knowledge for LCR. They map statutes and cases into a graph, where an edge can connect two case nodes if one case cites the other, and an edge can connect a case node and a statute node if the statute is applicable to the case. Then Node2Vec is utilized to learning node embeddings. Finally, case similarity is computed based on the distance of node embeddings, using cosine similarity as a distance measure.

Note that the assumption underlying the aforementioned methods (i.e., the statute(s) applicable to a case is given) may be impractical. For these methods to be used in realistic settings, however, the statute(s) applicable to a case will need to be extracted automatically.

D Overview of LegalBase

To enable the reader to better understand LegalBase and how it can be used profitably for query expansion, we utilize the traffic-related legal concepts in LegalBase as an example. As illustrated in Figure 4, each node consists of its entity and property. There is an edge between two nodes to represent the relation of the nodes. Here, "Traffic" is a generic collection as a head node. "Traffic_rule" is a member collection belonging to "Traffic". "Traffic_signal" is another member collection belonging to "Traffic_rule". "Regulatory_helmet" and "Seat_belt" are concrete legal concepts of "Traffic_rule". "Traffic_light" is a concrete legal concept of "Traffic_signal".

Next, we illustrate how to use this knowledge graph to expand queries. For a short query "traffic accident, right of way...", a Query Language Model (Ponte and Croft, 1998) is used to retrieve the relevant entities related to the query from the knowledge base. Then these retrieved entities are used to augment the query. For instance, "Traffic_signal" and "Traffic_light" may be retrieved as extra texts as they are relevant to "Traffic_rule" in the knowledge base. For a knowledge base for legal tasks, legal-related rather than semantically related entities should be connected to help models make inferences. For instance, "regulatory_helmet" is not relevant to "Traffic_rule" in semantic, but they are relevant in the legal domain.

E Citation-Based Approaches to LCR

The goal of citation-based approaches is different from other LCR approaches:they aim to compute the similarity of cases via citations. Given a set of historical legal cases, each of which is represented using only the citations that appear in it, a graph is constructed where each node corresponds to a case and two nodes are connected by an edge if the two cases cite each other. For example, in PCNet (Kumar et al., 2011; Minocha et al., 2015), the similarity between two cases is estimated using network measurements such as Bibliographic coupling (Kumar et al., 2011), co-citation (Kumar et al., 2011), or dispersion (Minocha et al., 2015). However, using only citations to precedent cases is insufficient for representing a case. For example, knowledge of statutes is also a key aspect for understanding the similarity between case documents (Bhattacharya et al., 2022). As a result, Bhattacharya et al. (2020) develop Hier-SPCNet, which integrates citation similarity with statute similarity. By construction, these approaches cannot be applied to cases without citations. In particular, they cannot be applied to cases that do not appear in the training data. Hence, it is somewhat misleading to call them "approaches to LCR" because they cannot be used to *retrieve* the cases that are most similar to a new query case. However, researchers can utilize these approaches to extract external knowledge via citations to enhance case representations, i.e., improving the representation of a case by incorporating knowledge from its cited sources.