U-CREAT: Unsupervised Case Retrieval using Events extrAcTion

Abhinav Joshi* **Akshat Sharma*** **Sai Kiran Tanikella*** **Ashutosh Modi**
Indian Institute of Technology Kanpur (IIT Kanpur)
{ajoshi, akshatsh, tskiran, ashutoshm}@cse.iitk.ac.in

Abstract

The task of Prior Case Retrieval (PCR) in the legal domain is about automatically citing relevant (based on facts and precedence) prior legal cases in a given query case. To further promote research in PCR, in this paper, we propose a new large benchmark (in English) for the PCR task: IL-PCR (Indian Legal Prior Case Retrieval) corpus. Given the complex nature of case relevance and the long size of legal documents, BM25 remains a strong baseline for ranking the cited prior documents. In this work, we explore the role of events in legal case retrieval and propose an unsupervised retrieval method-based pipeline U-CREAT (Unsupervised Case Retrieval using Events Extraction). We find that the proposed unsupervised retrieval method significantly increases performance compared to BM25 and makes retrieval faster by a considerable margin, making it applicable to real-time case retrieval systems. Our proposed system is generic, we show that it generalizes across two different legal systems (Indian and Canadian), and it shows state-of-the-art performance on the benchmarks for both the legal systems (IL-PCR and COLIEE corpora).

1 Introduction

Traditionally, in the legal domain, for a given legal case (query document) at hand, lawyers and judges have relied on their expertise and experience to cite relevant past precedents (cited documents). Moreover, even when legal professionals have made limited use of technology, it has been mainly restricted to Boolean queries and keywords. However, as cases increase, it becomes difficult for even experienced legal professionals to cite older precedents. NLP-based technologies can aid legal professionals in this regard. The task of *Prior Case Retrieval* (PCR) has been formulated to address this problem (Rabelo et al., 2022). More concretely, the task of *Prior Case Retrieval* involves retrieving all the previous legal documents that should be cited in the current legal document based on factual and precedent relevance. PCR can be particularly important in populous countries like India, where the number of cases has been growing exponentially, for example, there are 41 million pending cases in India (National Judicial Data Grid, 2021). Technology-based solutions such as PCR can make the process streamlined and efficient, expediting case disposal. PCR is different from standard document retrieval tasks. It is primarily due to the nature of legal documents themselves. Legal documents, in general, are quite long (tens to hundreds of pages), which makes each document in both the query and candidate pool long. Legal documents are unstructured and sometimes noisy (for example, in many common law countries like India, legal documents

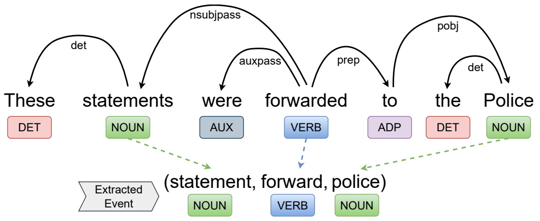

Figure 1: Dependency parse of the sentence (along with extracted event) from the **IL-PCR** corpus: “These statements were forwarded to the Police".

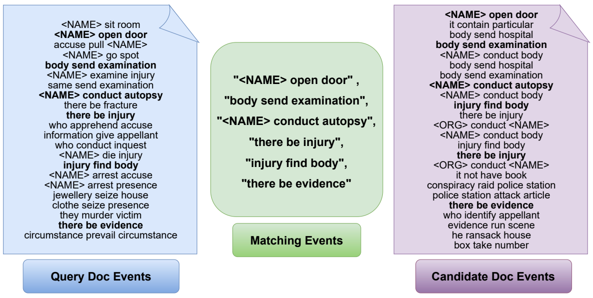

Figure 2: The Figure shows common events (highlighted in **bold**) for a positive query-candidate pair (example taken from the **IL-PCR** corpus)

*Equal Contributions

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 13899–13915
July 9-14, 2023 ©2023 Association for Computational Linguistics

are manually typed and prone to grammatical and spelling mistakes). Moreover, in a common-law system, where the judges can overrule an existing precedence, there is some degree of subjectivity involved, making the task of document processing and retrieval challenging.

In this paper, we propose a new large PCR corpus for the Indian legal setting referred to as Indian Legal Prior Case Retrieval (`IL-PCR`) corpus. Further, we propose an unsupervised approach for the task of prior case retrieval based on events structure in the document. Events are defined in terms of predicate and its corresponding arguments (see Figure 1) obtained via a syntactic dependency parser. The proposed event-based representation technique performs better than the existing state-of-the-art approaches both in terms of retrieval efficiency as well as inference time. We conjecture that events obtained via a dependency parser play an essential role in providing a short summary of long judgment documents, hence reducing the noise (task-dependent non-relevant information) by a considerable margin (also shown in Fig. 2).

The focus of this paper is an unsupervised and fast approach for retrieving relevant legal documents, in contrast to resource and compute-intensive supervised approaches. In the legal domain, supervised algorithms often require hand-crafted engineering/tuning with considerable experimentation to enable deployment in a real-time scenario, making them harder to adapt to an industrial setting. Although not a fair comparison, our proposed method shows an improvement of 5.27 F1 score over a recent state-of-the-art supervised method (Abolghasemi et al., 2022) for the existing PCR benchmark dataset of COLIEE'21 (§5.2). In a nutshell, we make the following contributions:

- Considering the lack of available benchmarks for the Indian legal setting, we create a new benchmark for Prior Case Retrieval focused on the Indian legal system (`IL-PCR`) and provide a detailed analysis of the created benchmark. Due to the large size of the corpus, the created benchmark could serve as a helpful resource for building information retrieval systems for legal documents (§3). We release the corpus and model code for the purpose of research usage via GitHub: `https://github.com/Exploration-Lab/IL-PCR`.
- We propose a new framework for legal document retrieval: U-CREAT (Unsupervised Case Retrieval using Events Extraction), based on the events extracted from documents. We propose different event-based models for the PCR task. We show that these perform better than existing state-of-the-art methods both in terms of retrieval efficiency as well as inference time (§5).
- Further, we show that the proposed event-based framework and models generalize well across different legal systems (Indian and Canadian systems) without any law/demography-specific tuning of models.

2 Related Work

Automating processes and tasks in the legal domain has been an active area of research in the NLP and IR community in the past few years. For example, several tasks/research problems and solutions have been proposed, e.g., Catchphrase Extraction (Galgani et al., 2012), Crime Classification (Wang et al., 2018, 2019), Summarization (Tran et al., 2019), Rhetorical Role prediction (Malik et al., 2022; Kalamkar et al., 2022) and Judgment Prediction (Zhong et al., 2020; Malik et al., 2021; Chalkidis et al., 2019; Aletras et al., 2016; Chen et al., 2019; Long et al., 2019; Xu et al., 2020; Yang et al., 2019; Kapoor et al., 2022).

Some earlier works (Al-Kofahi et al., 2001; Jackson et al., 2003) in Prior Case Retrieval have used feature-based machine learning models such as SVM. Since the past few years, the Competition on Legal Information Extraction and Entailment (COLIEE) has been organized annually (Rabelo et al., 2022). COLIEE has spurred research in PCR. Researchers participating at COLIEE have shown that BM-25 based method is a strong baseline. Most of the participating systems in COLIEE have used models based on BM-25 combined with other techniques like TF-IDF, language models, transformers, and XG-Boost (e.g., (Rosa et al., 2021; Rabelo et al., 2022; Askari et al., 2021; Ma et al., 2021; Nguyen et al., 2021; Shao et al., 2020; Bithel and Malagi, 2021)). Citation network-based approaches (Kumar et al., 2011; Minocha et al., 2015; Bhattacharya et al., 2020; Mandal et al., 2017; Kumar et al., 2013) are not meaningfully applicable to PCR as the legal citation networks are quite sparse. Abolghasemi et al. (2022) proposed BERT-based Query-by-Document Retrieval method with Multi-Task Optimization. We also experimented with transformer-based methods for retrieving prior cases as described in §5.1.

In the NLP community, researchers have used event-based information for many different Natural Language Understanding (NLU), and commonsense reasoning tasks (Chen et al.; Chambers and Jurafsky, 2008, 2009; Modi and Titov, 2014; Modi, 2016; Modi et al., 2017). For example, Glavaš and Šnajder (2014) extracted events from a document and used the event-centric graph representation for information retrieval and multi-document summarization tasks, where they define an event as a tuple of predicate (action) and corresponding arguments (participants/actors). In the legal-NLP domain, event-based representations have not been explored much, as also pointed out in the survey by Feng et al. (2022). In this work, we employ event-based representation for PCR.

3 IL-PCR Corpus and PCR Task

To spur research in the area of PCR, we propose the creation of a new corpus for the task of PCR: Indian Legal Prior Case Retrieval (**IL-PCR**) corpus. **IL-PCR** corpus is a corpus of Indian legal documents in English containing 7070 legal documents.

3.1 IL-PCR Corpus Creation

The corpus is created by scraping legal judgment documents (in the public domain) from the website IndianKanoon (https://indiankanoon.org/). We started by scraping documents corresponding to the top 100 most cited Supreme Court of India (SCI) cases (these are termed the zero-hop set). To gather more cases, we scraped documents cited within the zero-hop cases to obtain the one-hop cases. Scraping in this manner ensured a sufficient number of cited cases for each document. In practice, gathering cases till the second hop was sufficient for a corpus of desirable size. The desirable size is decided by comparing it relatively to the size of the existing PCR benchmarks like COLIEE. Any empty/non-existent cases were removed. Zero and one-hop cases were merged into a large query pool, which was further split into the train (70%), validation (10%), and test (20%) queries. To facilitate generalization among developed models, we did not put any temporal constraints on the scraped documents (as also justified in (Malik et al., 2021)); the cases range from 1950 to 2020. We followed a similar corpus creation methodology as done by the COLIEE benchmark.

Pre-Processing: All documents are normalized for names and organization names using a NER model (Honnibal Matthew and Van Landeghem Sofie, 2020) and a manually compiled gazetteer. This step helps to create more generic event representations. As done in the case of other PCR corpora such as COLIEE (Rabelo et al., 2022), the text segment associated with each citation (these are in the form of hyperlinks in scraped documents) is replaced with a citation marker <CITATION>. The text segments corresponding to statutes (acts and laws) are not replaced since our focus is prior case retrieval and not statute retrieval (Kim et al., 2019). We also experimented with another version of the corpus where the entire sentence containing the citation is removed (details in §5.3).

Dataset	COLIEE'21	IL-PCR
# Documents	4415	7070
Avg. Document Size	5813.66	8093.19
# query Documents	900	1182
Vocab Size	80577	113340
Total Citation Links	4211	8008
Avg. Citation Links per query	4.678	6.775
Language	English	English
Legal System	Canadian	Indian

Table 1: The table compares the created **IL-PCR** corpus with COLIEE'21 corpus.

Comparison with Existing Corpora: We compare existing PCR corpus from COLEE'21 and **IL-PCR** in Table 1. **IL-PCR** is almost 1.6 times COLIEE 2021 and average length of document in **IL-PCR** is almost 1.4 times. **IL-PCR** has a much larger vocabulary and more citations per document. Both COLIEE 2021 and **IL-PCR** are primarily in English but address different legal systems, namely, Canadian and Indian legal systems respectively.

3.2 PCR Task Definition

Given a legal document as a query $\mathcal{Q}_i$ and a pool of N legal documents as candidates $\{\mathcal{C}_1, \mathcal{C}_2., \ldots, \mathcal{C}_N\}$, the Prior Case Retrieval task is to retrieve the legal documents from the candidate pool which are relevant (and hence cited) in the given query document. As also pointed out by the legal expert, relevance in the legal domain is mainly about similar factual situations and previous legal precedents.

4 Event Based Representations

A story or an incident is best described in terms of a sequence of events (Chambers and Jurafsky, 2008, 2009; Chen et al.). If we consider a case judgment document to be a narrative about how things (e.g., situations in the form of facts) developed, then it is best to represent a legal document in terms of events. We define an event as a tuple containing predicate (describing the main action, typically it is verb/verb-compound) and its main arguments (describing main actors/participants, typically these correspond to subject, object, indirect object, and prepositional object) as shown in Fig. 1.

4.1 Event Extraction

To extract events, legal documents are first preprocessed to remove noise (unwanted characters and symbols) using regex-based patterns. For example, initials (not picked by NER) in the names (e.g., initials A.R. in the name A. R. Lakshman) are removed. Similarly, characters other than letters and citation markers are removed. Honorifics like Dr., Mr., Mrs., etc. are removed as these were wrongly picked up as the end of the sentence during sentence splitting and during event extraction. Other short forms like no., nos., addl., etc., are replaced by corresponding full words. Subsequently, a dependency parser is used to extract events from texts.

A dependency parser represents a sentence in the form of a directed graph $G : (V, E)$, where V are vertices representing words and E are the directed edges that capture the grammatical (syntactic) relationship between words (Kübler et al., 2009). Sentences in the document are parsed with the dependency parser (we use spaCy: (Honnibal Matthew and Van Landeghem Sofie, 2020)) to extract the list of verbs. These verbs form the root of the dependency graph. As observed, mostly the sentences in legal documents are in active voice. The left children of each verb are examined to find the subjects with syntactic dependency relationships like nsubj, nsubjpass, and csubj. The right children of a verb are considered for relationships like dobj, pobj, and dative to indicate the object's presence. Further, the lefts and rights are examined for conjunctions and compounds to get all the possible subjects and (indirect) objects. Each of the words in the extracted event is lemmatized to make the event more generic. Further, incomplete events and empty events (generated due to incorrect sentence splitting) are discarded. Both query and candidate documents are processed with the dependency parser to get the events. After removing noisy events, we did not observe any significant mistakes in the extracted events. Manual examination of the verb-argument tuples showed plausible events.

Events play an important role in establishing the relationship between a case and a cited (precedent) case. If a case has a precedent, then most likely, both are related based on the nature of the facts, evidence, and judgment. The events in a prior case form a basis for the arguments and judgments in such similar cases. Based on the experimental results, we conjecture that events further help to summarize documents in terms of main actions (e.g., related to facts) and hence help to filter out noise.

5 Experiments, Results and Analysis

Datasets: We experimented with the COLIEE-21 and **IL-PCR** corpora. Since the two corpora are different, it enables checking the generalization capabilities of models.

Evaluation Metric: We use a micro-averaged F1 score as the evaluation metric (as done in COLIEE-21[1]). In practice, models predict a relevance score for each candidate for a given query. Top-K-ranked candidates are considered for prediction (i.e., whether a candidate is cited or not). As done in previous work (Rabelo et al., 2022), we select K based on the best performance on the validation set and report the F1 on the test set using the same K value (metric definition is provided in App. A).

5.1 Baselines, Proposed Models and Results

For the baseline models, we selected the prominent approaches used for the PCR task. Considering the findings reported in COLIEE-21, BM25 marks a strong baseline (Rosa et al., 2021; Rabelo et al., 2022) for document retrieval tasks in the legal domain. Moreover, most of the re-ranking-based supervised methods (Askari et al., 2021; Nguyen et al., 2021; Bithel and Malagi, 2021; Shao et al., 2020; Abolghasemi et al., 2022) also use BM25 as a pre-filtering step for document retrieval. Broadly, we consider three types of unsupervised retrieval models as baselines, 1) Word-based (Count-based), which are lexical models using words directly; 2) Transformer-based models, which capture the semantics using distributed representations of words; and 3) Sentence Transformer based models, which capture semantics at the sentence level. We provide experimental results for all the baseline models on COLIEE'21 and **IL-PCR** datasets in Table 2. We describe baseline models next.

[1] Section 3.1 in https://sites.ualberta.ca/~rabelo/COLIEE2021/

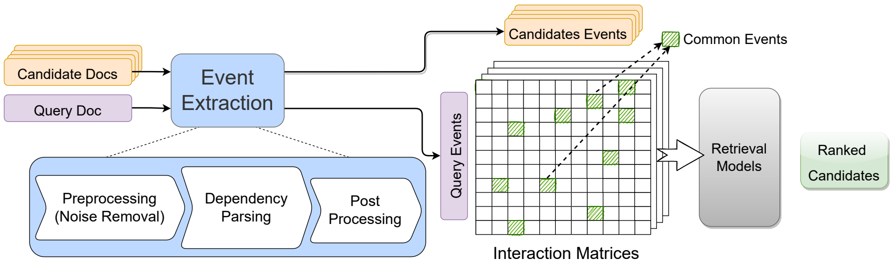

Figure 3: U-CREAT pipeline based on events extraction, for the PCR task.

Word-Based (Count-Based): We use a standard implementation of BM25 (Sklearn's (Pedregosa et al., 2011) TfidfVectorizer module) to compute scores for each query-candidate pair. We experiment with two widely used versions of BM25, unigram, and bigram. The bigram variant of BM25 improves the retrieval performance (Table 2) by a considerable margin, from 14.72% to 22.14% in COLIEE'21 and 13.85% to 28.59% in `IL-PCR`. However, the large runtime overhead of the bigram setting makes it ineffective for a real-time retrieval system and hence is usually not the preferred choice.

Transformer-Based: We use two widely used transformer models for generating word embedding: pre-trained BERT (Devlin et al., 2019) and DistilBERT (Sanh et al., 2019). We also experiment with a fine-tuned version. We fine-tune the model on the train split of the respective datasets (`IL-PCR` and COLIEE'21) using standard masked language modeling (MLM) objective (details in App. B). In addition, we also experiment with Indian legal domain-specific language models: InCaseLawBERT and InLegalBERT (Paul et al., 2022). We use transformer models in two settings, one using the entire document and the other using the top 512 tokens. Due to limitations on the input size of transformer models, to learn the representation of the entire document, we divide the document into multiple segments (each of 10 sentences) with a stride of 5 sentences (to ensure overlap). Subsequently, an interaction matrix (having relevance score) between query and candidate segments is created using cosine similarity between respective representations and this is followed by an aggregation step (avg. or max) to come up with a score. In the other setting, we consider only the top 512 tokens as input to the transformer and discard the remaining information. Our experiments highlight that fine-tuning these models slightly improves the performance in the case of transformers with top 512 tokens and slightly worsens the performance in the case of full document transformers. (Table 2). We observe that InCaseLawBERT and InLegalBERT perform quite poorly, possibly due to noise in legal documents.

Sentence Transformer-Based (SBERT): We also experiment with sentence embeddings-based methods that capture the similarity at the sentence level. We experiment with two popularly used sentence embedding methods[2]: SBERT-BERT and SBERT-DistilRoBERTA (Reimers and Gurevych, 2019). To finetune the transformers in an unsupervised fashion, we follow SimCSE's (Gao et al., 2021) strategy (details in App. B and App. C). For all the methods, we use cosine similarity between all query-candidate sentence pairs to generate an interaction matrix and consider the max of the matrix to be the relevance score for the pair. In general, compared to full document and vanilla transformers SBERT based approaches have better performance (Table 2).

Event Based Models: The general pipeline for event-based models is shown in Fig. 3. We refer to this pipeline as U-CREAT (Unsupervised Case Retrieval using Event extrAcTion). We first extract event representations from the query and candidate documents, and these are used to calculate an interaction matrix between each query-candidate pair. The interaction matrix captures similarities between events (relevance scores); subsequently, a retrieval model is used to rank the candidates. The methods proposed below differ in the document representation, interaction matrix, and retrieval model.

Atomic Events: In this variant, an event (predicate and arguments tuple) is considered as an atomic unit (like a word), and a document is represented only by these atomic events. An approach to generating the relevance scores can be using Jaccard similarity (IOU: Intersection Over Union) over the

[2] For model implementation, we used the SBERT library (`https://www.sbert.net/examples/unsupervised_learning/SimCSE/README.html`). We used the hyperparameters corresponding to the best-performing model on the leaderboard for the sentence similarity task.

obtained set of events. For a given query candidate pair $(\mathcal{Q}_i, \mathcal{C}_j)$, we extract the events corresponding to each document, $\mathcal{E}^{(\mathcal{Q}_i)} = \{e_1^{(\mathcal{Q}_i)}, \dots, e_n^{(\mathcal{Q}_i)}\}$, and $\mathcal{E}^{(\mathcal{C}_j)} = \{e_1^{(\mathcal{C}_j)}, \dots, e_m^{(\mathcal{C}_j)}\}$ which is used to compute the Jaccard similarity, i.e., Relevance Score = $\frac{|\mathcal{E}^{(\mathcal{Q}_i)} \cap \mathcal{E}^{(\mathcal{C}_j)}|}{|\mathcal{E}^{(\mathcal{Q}_i)} \cup \mathcal{E}^{(\mathcal{C}_j)}|}$. As shown in Table 2, this trivial strategy of computing Jaccard similarity over the set of events improves performance on both datasets compared to BM25. Though the gain is less in COLIEE'21 (increase by ~ 8 F1 score), in **IL-PCR**, the improvement is significant (increase by ~ 20 F1 score). We speculate that given the legal document's diverse and lengthy nature, events help filter out the noise and improve performance significantly. Another way of getting the relevance score would be to take all the extracted events $\mathcal{E}^{(\mathcal{Q}_i)}$ and $\mathcal{E}^{(\mathcal{C}_j)}$ and perform a BM25 over atomic events instead of words; this setting helps to capture the relation between various events present in both the docs. We experiment with multiple settings of BM25. The results highlight that the BM25's unigram setting performs similarly to the Jaccard similarity with a drop in performance when increased to bigram, trigram, the reason being the lower frequency of bigram/trigram events present in the document pairs.

Non-atomic Events: For this setting, we consider the words (predicates and arguments) that are present in the extracted events $\mathcal{E}^{(\mathcal{Q}_i)}$ and $\mathcal{E}^{(\mathcal{C}_j)}$ separately. This setting removes the event as an atomic unit, and it considers words of each event as an independent unit, i.e., a document is represented only by individual words in the extracted events. We run various variants for BM25 to generate relevance scores. We found that the trigram version of BM25 (the best model for non-atomic events) has a similar performance to the best model for atomic events (BM25).

Events filtered Docs: As the primary role of events is to capture the relevance between the query and the candidate doc, for this variant, we select the complete sentences corresponding to the overlapping events $|\mathcal{E}^{(\mathcal{Q}_i)} \cap \mathcal{E}^{(\mathcal{C}_j)}|$. For example, if a common event $e_p^{\mathcal{Q}_i}$ emanates from sentences $\mathcal{S}_t$ and $\mathcal{S}_v$ in the query and candidate document, respectively, we consider the sentence $\mathcal{S}_t$ from the query and $\mathcal{S}_v$ from the candidate. Selecting sentences for each overlapping event results in sentences selected for every doc. We refer to this updated version of the doc as the events filtered doc and use this new version for classical retrieval methods like BM25. We observe that this setting further improves the retrieval scores by 2.62 in **IL-PCR** and 3.19 in COLIEE'21, compared to the best non-atomic event-based methods. Overall, this setting outperforms all the other methods for both datasets and shows a performance boost of 25.3 F1 score in **IL-PCR** and 12.6 F1 score in COLIEE'21 compared to the standard BM-25 baseline.

Event Embeddings: We also tried models based on event embeddings obtained by composing embeddings of predicates and arguments, e.g., via transformer models or deep NNs (Modi, 2016; Modi and Titov, 2014); however, these approaches gave a worse performance than vanilla transformer based approaches. Moreover, these approaches have an extra overhead of training (and learning) event embeddings.

Rhetorical Roles Filtered Docs: In the legal domain, Rhetorical Roles (RR) (Malik et al., 2022; Kalamkar et al., 2022) have been introduced to segment a document into semantically coherent textual units corresponding to 7 main rhetorical roles: Facts, Arguments, Statues, Ruling, Precedents, Ratio, and Judgment. For more details, please refer to Malik et al. (2022). The main idea is to label each sentence in the legal document with one of the rhetorical roles. For RR, we used the pre-trained transformer-based model utilizing multi-task learning provided by Malik et al. (2022) to predict sentence-level labels for legal documents in COLIEE21 and **IL-PCR**. We used some specific RR labels (that capture relevance as per legal experts) to filter out sentences from a query (RRs used: facts, argument, ratio) and candidates (RRs used: facts, argument, ratio, and judgment). Using all RRs labels gave a worse performance, possibly due to the introduced noise. The filtered query and candidate documents are then used for BM-25-based baselines. Table 2 shows that a pre-filtering step done using a pre-trained RR model is a strong retrieval method and provides a significant performance boost (increase of 24.97 in COLIEE'21 and 37.72 in the case of **IL-PCR**). However, the events-based filtering methods remain the outperforming model (27.32 increase in COLIEE'21 and 39.15 boost in F1 score in **IL-PCR**). However, in the case of RR, inference time in the case of quad-gram and penta-gram increases drastically, making them impractical (§5.3). RR-based models have lesser improvement on COLIEE'21 as the pre-trained

Model		COLIEE'21	IL-PCR
Word Level (Count Based)	BM25	14.72 **(Baseline)**	13.85 **(Baseline)**
	BM25 (Bigram)	22.14 (↑ 7.42)	28.59 (↑ 14.74)
Segmented-Doc Transformer (full document)	BERT	5.10 (↓ 9.62)	9.24 (↓ 4.61)
	BERT (finetuned)	4.58 (↓ 10.14)	7.91 (↓ 5.94)
	DistilBERT	10.04 (↓ 4.68)	16.61 (↑ 2.76)
	DistilBERT (finetuned)	4.73 (↓ 9.99)	7.86 (↓ 5.99)
	InCaseLawBERT	1.71 (↓ 13.01)	3.62 (↓ 10.23)
	InLegalBERT	2.79 (↓ 11.93)	7.57 (↓ 6.28)
Transformer (top 512 tokens)	BERT	0.53 (↓ 14.19)	0.56 (↓ 13.29)
	BERT (finetuned)	0.46 (↓ 14.26)	0.88 (↓ 12.97)
	DistilBERT	0.54 (↓ 14.18)	0.50 (↓ 13.35)
	DistilBERT (finetuned)	0.34 (↓ 14.38)	0.75 (↓ 13.1)
	InCaseLawBERT	0.78 (↓ 13.94)	0.75 (↓ 13.1)
	InLegalBERT	0.50 (↓ 14.22)	0.71 (↓ 13.14)
Sentence Transformer (SBERT)	BERT	6.79 (↓ 7.93)	5.94 (↓ 7.91)
	DistilRoBERTa	3.63 (↓ 11.09)	3.91 (↓ 9.94)
	BERT (finetuned)	7.68 (↓ 7.04)	6.01 (↓ 7.84)
	DistilRoBERTa (finetuned)	1.26 (↓ 13.46)	2.14 (↓ 11.17)
Atomic Events	Jaccard similarity	23.08 (↑ 8.36)	34.17 (↑ 20.32)
	BM25	23.45 (↑ 8.73)	36.77 (↑ 22.92)
	BM25 (Bigram)	22.42 (↑ 7.70)	31.81 (↑ 17.96)
	BM25 (Trigram)	21.12 (↑ 6.40)	27.61 (↑ 13.76)
Non-atomic Events	BM25	14.19 (↑ 0.53)	11.99 (↓ 1.86)
	BM25 (Bigram)	23.59 (↑ 8.87)	32.27 (↑ 18.42)
	BM25 (Trigram)	24.13 (↑ 9.41)	36.53 (↑ 22.68)
	BM25 (Quad-gram)	22.69 (↑ 7.97)	34.76 (↑ 20.91)
	BM25 (Penta-gram)	21.81 (↑ 7.09)	33.54 (↑ 19.69)
Events Filtered Docs	BM25	18.97 (↑ 4.25)	19.64 (↑ 5.79)
	BM25 (Bigram)	23.3 (↑ 8.58)	30.28 (↑ 16.43)
	BM25 (Trigram)	**27.32** (↑ **12.60**)	37.17 (↑ 23.32)
	BM25 (Quad-gram)	26.94 (↑ 12.22)	**39.15** (↑ **25.3**)
	BM25 (Penta-gram)	25.81 (↑ 11.09)	38.61 (↑ 24.76)
RR Filtered Docs	BM25	12.97 (↓ 1.75)	13.05 (↓ 0.80)
	BM25 (Bigram)	21.06 (↑ 6.34)	24.67 (↑ 10.82)
	BM25 (Trigram)	24.97 (↑ 10.25)	34.22 (↑ 20.37)
	BM25 (Quad-gram)	24.90 (↑ 10.18)	36.77 (↑ 22.92)
	BM25 (Penta-gram)	23.72 (↑ 9.00)	37.72 (↑ 23.87)

Table 2: The table shows the performance comparison (F1 scores in %, with top K retrieved documents selected using validation set) of the proposed method with the baseline unsupervised methods on the COLIEE-21 (Rabelo et al., 2022) and proposed IL-PCR benchmark. The numbers in the bracket highlight the performance difference compared to the BM25 (Baseline, Table's first row). ↑ shows the increase, and ↓ shows the drop in performance.

Method	Brief Description	Unsupervised	F1
JNLP (Nguyen et al., 2021)	Top-100,Paragraph,BM25,BERT,Union Score	✓	0.19
TR (Rabelo et al., 2022)	Top-1000 TF-IDF, Xgboost	✓	0.46
DSSIR (Althammer et al., 2021)	vanilla BERT	✗	2.79
DSSIR (Althammer et al., 2021)	paragraph level BM25, lawDPR	✗	2.72
SIAT (Rabelo et al., 2022)	Top-50 BM25, BERT-Legal	✓	3.00
DSSIR (Althammer et al., 2021)	BM25	✓	4.11
TLIR (Ma et al., 2021)	LMIR, BERT-PLI on paragraphs	✗	4.56
NM (Rosa et al., 2021)	Vanilla BM25-Segments	✓	9.37
TLIR (Ma et al., 2021)	Language Model for IR and paragraph filtering	✓	19.17
MTFT-BERT (Abolghasemi et al., 2022)	Multi-task optimization over $BM25_{optimized}$	✗	22.05
U-CREAT	BM25 (Tri-gram) over Events Filtered Docs	✓	**27.32**

Table 3: The table shows the performance comparison of the proposed method with the existing methods on the COLIEE-21 (Rabelo et al., 2022) dataset. The F1 scores (in %) represent the numbers reported in respective methods. The table highlights a significant performance boost with respect to the current state-of-the-art MTFT-BERT (supervised method trained on COLIEE-21 corpus).

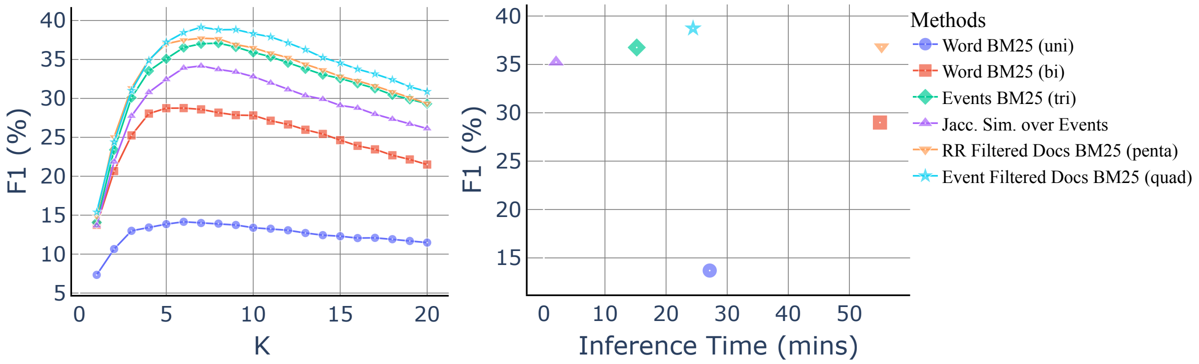

Figure 4: The figure on the left shows the performance curves and the right shows inference time vs. performance of various methods. Also see Appendix Table 4.

RR model (Malik et al., 2022) used for labeling is trained on Indian legal documents.

5.2 Comparison with Existing Methods

For a fair comparison with the existing methods, we compare the proposed event-based approaches with the state-of-the-art methods for the COLIEE'21 benchmark. A recent supervised retrieval approach by Abolghasemi et al. (2022) uses a multitasking framework to improve upon the optimized BM25 retrieval scores. To the best of our knowledge, this approach is the current state-of-the-art method for the COLIEE-21 document retrieval task. Table 3 shows the F1 scores obtained by multiple methods, as given in (Rabelo et al., 2022). The proposed event-based methods outperform the existing approaches by a significant margin highlighting the effectiveness of events in legal document retrieval. A noteworthy point here is that the event-based techniques are completely unsupervised, making them more applicable to current systems without corpus-specific training. Moreover, these approaches generalize well over legal documents in different legal systems, as shown using two different legal system datasets.

5.3 Analysis

Variation with K: To provide a detailed insight into the performance of various methods, we also show the F1 score at different K values (top retrieved documents) on **IL-PCR**. Figure 4 (left side) highlights the improvements in the F1 curves obtained by event methods compared to the popularly used BM25 baselines. The performance peaks for K = 3 to 7, this is similar to what has been observed on the COLIEE dataset (Rabelo et al., 2022). We show the variation of Precision and Recall scores with the value of K in Figure 5. As can be observed (and is expected based on the evaluation metric definition) for each of the models, precision falls and recall improves with increasing K values, resulting in the hump shape in Fig. 4. The Precision, Recall, and F1 scores corresponding to best K are tabulated in Appendix Table 5.

Inference Time: An important property of a retrieval algorithm often not stressed by existing methods is inference time. For a retrieval system to be adaptable to industrial solutions, it is not only the retrieval efficiency but also the inference time required by the system. We compare inference times of various methods to provide a more transparent insight. We use the queries in the entire test split (237 query documents) of the **IL-PCR** corpus to calculate inference time. We benchmark the relevance score generation time for all the queries on a single core of an Intel(R) Xeon(R) Silver 4210R CPU @ 2.40GHz processor. We compute the event extraction time along with the relevance score generation time for the proposed event-based methods. Figure 4 (right side) shows the inference vs. performance comparison for the prominent methods (also see exact numbers in App. Table 4).

The inference time for the different models varies greatly, the Jaccard Similarity over Events (IOU) stands out with the fastest time of 2 minutes, while the Word BM25 (bigram) model has the longest inference time of 55 minutes. The Events BM25 (trigram) model has a much faster inference time of 15.2 minutes, which is approximately 50% faster than the Word BM25 (unigram) model. The Event Filtered Docs BM25 (quadgram) model also has a relatively fast inference time of 24.42 minutes, which is about 10% faster than the Word BM25 (unigram) model. Overall, the proposed Event Filtered Docs BM25 (quadgram) has a relatively fast inference time of 24.42 minutes com-

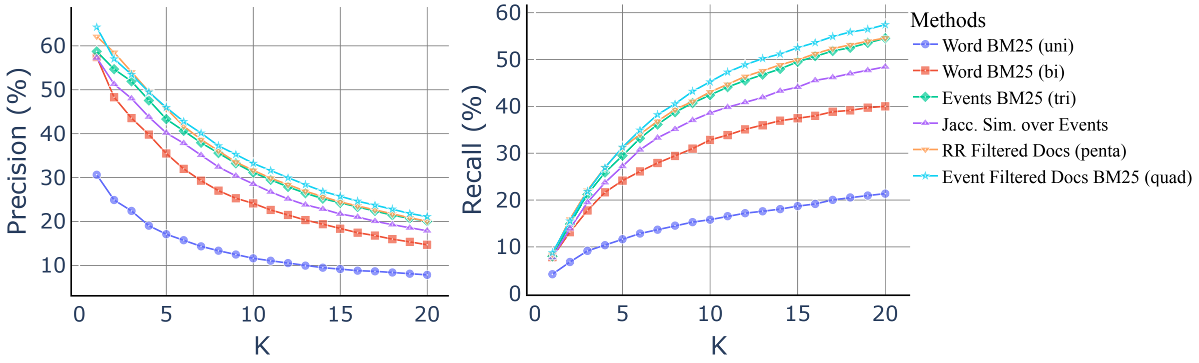

Figure 5: The precision and recall curves at different K (top retrieved documents) values used for retrieval.

pared to the other models and represents a significant improvement in performance. This time is about 10% faster than the Word BM25 (unigram) model and significantly faster than the Word BM25 (bigram) model, which has the longest inference time of 55 minutes. In the F1 score, the Event Filtered Docs BM25 (quadgram) model also outperforms the Jaccard Similarity over Events (IOU) model, which has the quickest time at 2 minutes, and the Events BM25 (trigram) model, which has a time of 15.2 minutes. The proposed model stands out as a strong performer in terms of inference time and F1, providing a significant improvement in performance compared to the other models.

In terms of inference time, the retrieval method based on Jaccard similarity shows a significant performance boost along with a significant improvement in the F1 score. Overall, the increase in document size results in a longer inference time in BM25-based methods. Moreover, going from unigram to bigram also results in a considerable increase in inference time, making the word-based BM25 bigram ineffective for real-time retrieval systems. The inference time results for event-based methods highlight the effectiveness both in terms of inference time and retrieval efficiency.

A noteworthy trend in the current deep learning-based supervised methods in legal document retrieval is the use of BM25 as a pre-filtering step (Askari et al., 2021; Nguyen et al., 2021; Bithel and Malagi, 2021; Shao et al., 2020; Abolghasemi et al., 2022). The scores obtained from a word-based BM25 provide a strong pre-filtering, enabling the re-ranking-based algorithm to improve the scores over the top-K% retrieved documents. This re-ranking setting for inference on a deployable system would require BM25 inference time and deep model inference time to generate the retrieval scores. In contrast, the proposed event-based approaches lead to a much faster inference time and improved retrieval performance. It would facilitate the current research on supervised retrieval methods as well.

Other Observations: We also experimented with another version of the corpus where we removed the sentences containing the citation to prevent the model from exploiting any neighboring information. The results are shown in the Appendix Table 6, there is a slight drop in performance; however the overall trends (as in Table 2) remain the same.

5.4 Discussion

An important point to note is that the PCR task has inherent limitations; the relevant cases are considered based on official citations as ground truth. However, there might be cases that were not mentioned by the judge (document writer) due to subjectivity involved in the common-law system; finding correct annotation for relevance is always a challenge for a domain like legal, where the number of documents is enormous.

6 Conclusion

In this paper, we proposed a new large dataset (**IL-PCR**) for Prior Case Retrieval and the U-CREAT pipeline for performing event-based retrieval for legal documents. We ran a battery of experiments with different types of models to show that event-based methods have better performance and much better inference times (and hence amenable to production settings) compared to existing unsupervised approaches and some of the supervised approaches (e.g., ~ 5.27 F1 score improvement on COLIEE) on two completely different datasets. In the future, we plan to combine event-based methods with supervised techniques such as contrastive learning to develop more efficient models.

Limitations

In this paper, we propose a simple model for prior case retrieval. As shown in experiments and results, the models could improve and score better. There is a big room for improvement. All the previously proposed approaches for PCR have calculated relevance as some form of lexical/semantic similarity between a case and its citations. However, cited case relevance may sometimes differ from lexical/semantic similarity. Modeling the document in terms of events only partially addresses this. Consequently, what is required is the inclusion of more legal information. We made an attempt towards that via experiments using Rhetorical Roles. Similarly, one could use the information coming via statutes and laws since similar cases are likely to invoke similar statutes. Another approach is learning representations using contrastive models that score relevant cases higher than non-relevant ones. In the future, we plan to investigate these approaches to improve the task of PCR.

This paper considers a simple structure for an event as a tuple of predicates and arguments. However, more sophisticated formulations are possible, as outlined in the survey/tutorial by Chen et al.. Moreover, we are taking events in isolation and ignoring the sequential nature of events that help to form narratives. In the future, we would like to develop a model that captures a more sophisticated structure and sequential nature of events in the case. Though we covered an extensive set of experiments for the proposed event-based matching technique, many more combinations can be experimented with to understand the role of events in legal documents. This unique finding of events missing from the legal literature would facilitate exploring new directions in the legal domain.

In this paper, we evaluated only two datasets as we could not find any publicly available PCR datasets. However, in the future, if we can find more PCR datasets, we would like to evaluate them to see if the trends generalize over other legal corpora.

Ethical Concerns

This paper proposes a system for retrieving (recommending) relevant documents. The system is not involved in any decision-making process. The motivation for proposing the system is to augment legal experts rather than replace them. Moreover, for training the system, we used publicly available legal documents. We took steps to normalize documents concerning named entities to prevent a model from developing any known biases. To the best of our knowledge, we addressed any biases that the model might learn from the data.

References

Amin Abolghasemi, Suzan Verberne, and Leif Azzopardi. 2022. Improving BERT-Based Query-by-Document Retrieval with Multi-Task Optimization. In *Advances in Information Retrieval: 44th European Conference on IR Research, (ECIR)*.

Khalid Al-Kofahi, Alex Tyrrell, Arun Vachher, and Peter Jackson. 2001. A Machine Learning Approach to Prior Case Retrieval. In *Proceedings of the Eighth International Conference on Artificial Intelligence and Law (ICAIL)*.

Nikolaos Aletras, Dimitrios Tsarapatsanis, Daniel Preotiuc-Pietro, and Vasileios Lampos. 2016. Predicting Judicial Decisions of the European Court of Human Rights: a Natural Language Processing Perspective. *PeerJ Computer Science*.

Sophia Althammer, Arian Askari, Suzan Verberne, and Allan Hanbury. 2021. DoSSIER@COLIEE 2021: Leveraging Dense Retrieval and Summarization-based Re-ranking for Case Law Retrieval. *arXiv preprint arXiv:2108.03937*.

AA Askari, SV Verberne, O Alonso, S Marchesin, M Najork, and G Silvello. 2021. Combining Lexical and Neural Retrieval with Longformer-Based Summarization for Effective Case Law retrieva. In *Proceedings of the Second International Conference on Design of Experimental Search & Information REtrieval Systems (DESIRES)*.

Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2020. Methods for Computing Legal Document Similarity: A Comparative Study. *arXiv preprint arXiv:2004.12307*.

Shivangi Bithel and Sumitra S Malagi. 2021. Unsupervised Identification of Relevant Prior Cases. *arXiv preprint arXiv:2107.08973*.

Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. 2019. Neural Legal Judgment Prediction in English. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics(ACL)*.

Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised Learning of Narrative Event Chains. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08:HLT)*.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised Learning of Narrative Schemas and Their Participants. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Huajie Chen, Deng Cai, Wei Dai, Zehui Dai, and Yadong Ding. 2019. Charge-Based Prison Term Prediction with Deep Gating Network. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, (EMNLP-IJCNLP)*.

Muhao Chen, Hongming Zhang, Qiang Ning, Manling Li, Heng Ji, Kathleen McKeown, and Dan Roth. Event-Centric Natural Language Processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Tutorial Abstracts (ACL-IJCNLP)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Yi Feng, Chuanyi Li, and Vincent Ng. 2022. Legal Judgment Prediction: A Survey of the State of the Art. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI)*.

Filippo Galgani, Paul Compton, and Achim G. Hoffmann. 2012. Towards Automatic Generation of Catchphrases for Legal Case Reports. In *Computational Linguistics and Intelligent Text Processing - 13th International Conference, (CICLing)*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Goran Glavaš and Jan Šnajder. 2014. Event Graphs for Information Retrieval and Multi-Document Summarization. *Expert Systems with Applications, Elsevier*.

Montani Ines Honnibal Matthew and Boyd Adriane Van Landeghem Sofie. 2020. spaCy: Industrial-Strength Natural Language Processing in Python.

Peter Jackson, Khalid Al-Kofahi, Alex Tyrrell, and Arun Vachher. 2003. Information Extraction from Case Law and Retrieval of Prior Cases. *Artificial Intelligence, Elsevier*.

Prathamesh Kalamkar, Aman Tiwari, Astha Agarwal, Saurabh Karn, Smita Gupta, Vivek Raghavan, and Ashutosh Modi. 2022. Corpus for Automatic Structuring of Legal Documents. In *Proceedings of the 13th Language Resources and Evaluation Conference -Association for Computational Linguistics (ACL-LREC)*.

Arnav Kapoor, Mudit Dhawan, Anmol Goel, Arjun T H, Akshala Bhatnagar, Vibhu Agrawal, Amul Agrawal, Arnab Bhattacharya, Ponnurangam Kumaraguru, and Ashutosh Modi. 2022. HLDC: Hindi Legal Documents Corpus. In *Findings of the Association for Computational Linguistics (ACL)*.

Mi-Young Kim, Juliano Rabelo, and Randy Goebel. 2019. Statute Law Information Retrieval and Entailment. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law (ICAIL)*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*.

Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. Dependency parsing. *Synthesis Lectures on Human Language Technologies (SLHLT), Springer*.

Sushanta Kumar, P. Krishna Reddy, V. Balakista Reddy, and Aditya Singh. 2011. Similarity Analysis of Legal Judgments. In *COMPUTE '11: Proceedings of the 4th Annual Association for Computing Machinery*.

Sushanta Kumar, P. Krishna Reddy, V. Balakista Reddy, and Malti Suri. 2013. Finding Similar Legal Judgements under Common Law System. In *Databases in Networked Information Systems (DNIS),Springer*.

Shangbang Long, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2019. Automatic Judgment Prediction via Legal Reading Comprehension. In *Chinese Computational Linguistics - 18th China National Conference, (CCL) Springer*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations (ICLR)*.

Yixiao Ma, Yunqiu Shao, Bulou Liu, Yiqun Liu, Min Zhang, and Shaoping Ma. 2021. Retrieving Legal Cases from a Large-scale Candidate Corpus. In *Proceedings of the Eighth International Competition on Legal Information Extraction/Entailment (COLIEE)*.

Vijit Malik, Rishabh Sanjay, Shouvik Kumar Guha, Angshuman Hazarika, Shubham Nigam, Arnab Bhattacharya, and Ashutosh Modi. 2022. Semantic Segmentation of Legal Documents via Rhetorical Roles. In *Proceedings of the Natural Legal Language Processing Workshop (NLLP) EMNLP*.

Vijit Malik, Rishabh Sanjay, Shubham Kumar Nigam, Kripabandhu Ghosh, Shouvik Kumar Guha, Arnab Bhattacharya, and Ashutosh Modi. 2021. ILDC for CJPE: Indian Legal Documents Corpus for Court Judgment Prediction and Explanation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*.

Arpan Mandal, Raktim Chaki, Sarbajit Saha, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2017. Measuring Similarity among Legal Court Case Documents. In *Compute '17: Proceedings of the 10th Annual ACM India Compute Conference*.

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*.

Ashutosh Modi. 2016. Event Embeddings for Semantic Script Modeling. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL)*.

Ashutosh Modi and Ivan Titov. 2014. Inducing Neural Models of Script Knowledge. In *Proceedings of the 18th SIGNLL Conference on Computational Natural Language Learning (CoNLL)*.

Ashutosh Modi, Ivan Titov, Vera Demberg, Asad Sayeed, and Manfred Pinkal. 2017. Modeling semantic expectation: Using script knowledge for referent prediction. *Transactions of the Association for Computational Linguistics (TACL)*.

National Judicial Data Grid. 2021. National judicial data grid statistics. `https://www.njdg.ecourts.gov.in/njdgnew/index.php`.

Ha-Thanh Nguyen, Phuong Minh Nguyen, Thi-Hai-Yen Vuong, Quan Minh Bui, Chau Minh Nguyen, Binh Tran Dang, Vu Tran, Minh Le Nguyen, and Ken Satoh. 2021. JNLP Team: Deep Learning Approaches for Legal Processing Tasks in COLIEE 2021. *arXiv preprint arXiv:2106.13405*.

Shounak Paul, Arpan Mandal, Pawan Goyal, and Saptarshi Ghosh. 2022. Pre-training Transformers on Indian Legal Text. *arXiv preprint arXiv:2209.06049*.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research (JMLR)*.

Juliano Rabelo, Randy Goebel, Mi-Young Kim, Yoshinobu Kano, Masaharu Yoshioka, and Ken Satoh. 2022. Overview and Discussion of the Competition on Legal Information Extraction/Entailment (COLIEE) 2021. *The Review of Socionetwork Strategies*.

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)*.

Guilherme Moraes Rosa, Ruan Chaves Rodrigues, Roberto Lotufo, and Rodrigo Nogueira. 2021. Yes, BM25 is a strong baseline for legal case retrieval. *arXiv preprint arXiv:2105.05686*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

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 *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI)*.

Vu Tran, Minh Le 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 Seventeenth International Conference on Artificial Intelligence and Law (ICAIL)*.

Pengfei Wang, Yu Fan, Shuzi Niu, Ze Yang, Yongfeng Zhang, and Jiafeng Guo. 2019. Hierarchical Matching Network for Crime Classification. In *Proceedings of the 42nd International ACM Conference on Research and Development in Information Retrieval, (SIGIR)*.

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 Conference on Research & Development in Information Retrieval (SIGIR)* .

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP)*.

Nuo Xu, Pinghui Wang, Long Chen, Li Pan, Xiaoyan Wang, and Junzhou Zhao. 2020. Distinguish Confusing Law Articles for Legal Judgment Prediction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)* .

Wenmian Yang, Weijia Jia, Xiaojie Zhou, and Yutao Luo. 2019. Legal Judgment Prediction via Multi-Perspective Bi-Feedback Network. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI)*.

Haoxi Zhong, Yuzhong Wang, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. Iteratively Questioning and Answering for Interpretable Legal Judgment Prediction. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference (IAAI), The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI)*.

Appendix

A Evaluation Metric Definition

$$\text{Precision} = \frac{(\text{\# correctly retrieved cases } \forall \text{ queries})}{(\text{\# retrieved cases } \forall \text{ queries})},$$

$$\text{Recall} = \frac{(\text{\# correctly retrieved cases } \forall \text{ queries})}{(\text{\# relevant cases } \forall \text{ queries})},$$

$$\text{F1} = \frac{(2 \text{ x Precision x Recall})}{(\text{Precision + Recall})}$$

B Hyper-Parameters

Transformer-Based Models: We train the standard BERT and DistilBERT models using PyTorch and HuggingFace library-based (Wolf et al., 2020) implementations for 6 epochs with a batch size of 32 and AdamW (Loshchilov and Hutter, 2019) optimizer with a learning rate of 1×10^{-5}.
Sentence Transformer-Based Models: We use a batch size of 512 and fine-tune the models for 20 epochs with Adam (Kingma and Ba, 2015) of learning rate 5×10^{-5}

C SBERT Fine Tuning Strategy

SBERT is finetuned using SimCSE (Gao et al., 2021) based checkpoints present in SBERT package (Reimers and Gurevych, 2019), due to the unavailability of annotated similar sentence pairs present for the datasets, SimCSE is trained in unsupervised manner by predicting the input sentence itself using dropout for noisy representation of the sentence.

D Precision and Recall Scores

Table 5 shows the Precision, Recall and F1 scores for various models in given in the main paper. Table 6 shows the Precision, Recall and F1 scores for various models on the version of **IL-PCR** without citation sentences.

E Inference Time of Models

Table 4 shows the inference time for algorithms shown in Fig 4.

Algorithm	Inference Time (mins)
Word BM25 (unigram)	27.14
Word BM25 (bigram)	55.00
Events BM25 (trigram)	15.20
Jaccard sim. over events	2.00
RR filtered BM25 (penta)	55.27
Events filtered BM25 (quad)	24.42

Table 4: Inference Times for various models..

Model		K	Precision	Recall	F1
Word Level	BM25	5	17.11	11.64	13.85
	BM25 (Bigram)	7	29.30	27.91	28.59
Segmented-Doc Transformer (full document)	BERT	6	10.28	8.40	9.24
	BERT (finetuned)	6	8.79	7.18	7.90
	DistilBERT	7	17.02	16.21	16.61
	DistilBERT (finetuned)	5	9.70	6.60	7.86
	InCaseLawBERT	11	3.02	4.52	3.62
	InLegalBERT	12	6.10	9.96	7.56
Transformer (top 512 tokens)	BERT	20	0.38	1.04	0.56
	BERT (finetuned)	15	0.65	1.33	0.87
	DistilBERT	20	0.34	0.93	0.50
	DistilBERT (finetuned)	20	0.51	1.39	0.75
	InCaseLawBERT	20	0.51	1.39	0.75
	InLegalBERT	19	0.49	1.27	0.71
Sentence Transformer (SBERT)	BERT	5	7.35	4.98	5.94
	DistilRoBERTa	4	5.56	3.01	3.91
	BERT (finetuned)	5	7.44	5.04	6.01
	DistilRoBERTa (finetuned)	7	2.20	2.08	2.14
Atomic Events	Jaccard Similarity	7	35.12	33.28	34.17
	BM25	7	37.69	35.90	36.77
	BM25 (Bigram)	6	35.39	28.89	31.81
	BM25 (Trigram)	6	30.71	25.07	27.61
Non-atomic Events	BM25	6	13.33	10.89	11.99
	BM25 (Bigram)	7	33.07	31.50	32.27
	BM25 (Trigram)	6	40.64	33.18	36.53
	BM25 (Quad-gram)	7	35.62	33.93	34.76
	BM25 (Penta-gram)	6	37.30	30.46	33.54
Events Filtered Docs	BM25	5	24.26	16.50	19.64
	BM25 (Bigram)	6	33.69	27.50	30.28
	BM25 (Trigram)	6	41.35	33.76	37.17
	BM25 (Quad-gram)	7	40.12	38.22	39.15
	BM25 (Penta-gram)	7	39.57	37.70	38.61
RR Filtered Docs	BM25	7	13.37	12.74	13.05
	BM25 (Bigram)	7	25.29	24.09	24.67
	BM25 (Trigram)	7	35.08	33.41	34.22
	BM25 (Quad-gram)	7	37.69	35.90	36.77
	BM25 (Penta-gram)	7	38.66	36.83	37.72

Table 5: The table shows the K values, Precision, Recall and F1 scores for each model.

Model		IL-PCR	IL-PCR$_{\neg\text{sent}}$ (without citation sents.)
Word Level	BM25	13.85	13.23
	BM25 (Bigram)	28.59	27.52
Segmented-Doc Transformer (full document)	BERT	9.24	9.58
	BERT (finetuned)	7.90	8.41
	DistilBERT	16.61	17.58
	DistilBERT (finetuned)	7.86	8.21
	InCaseLawBERT	3.62	3.25
	InLegalBERT	7.56	7.96
Transformer (top 512 tokens)	BERT	0.56	0.36
	BERT (finetuned)	0.87	0.67
	DistilBERT	0.50	0.52
	DistilBERT (finetuned)	0.75	0.68
	InCaseLawBERT	0.75	0.68
	InLegalBERT	0.71	0.68
Sentence Transformer (SBERT)	BERT	5.94	4.73
	DistilRoBERTa	3.91	2.94
	BERT (finetuned)	6.01	5.01
	DistilRoBERTa (finetuned)	2.14	1.01
Atomic Events	Jaccard Similarity	34.17	32.38
	BM25	36.77	35.26
	BM25 (Bigram)	31.81	30.96
	BM25 (Trigram)	27.61	26.59
Non-atomic Events	BM25	11.99	11.99
	BM25 (Bigram)	32.27	31.91
	BM25 (Trigram)	36.53	36.02
	BM25 (Quad-gram)	34.76	33.75
	BM25 (Penta-gram)	33.54	32.38
Events Filtered Docs	BM25	19.64	19.78
	BM25 (Bigram)	30.28	30.35
	BM25 (Trigram)	37.17	36.40
	BM25 (Quad-gram)	39.15	38.32
	BM25 (Penta-gram)	38.61	37.66
RR Filtered Docs	BM25	13.05	13.65
	BM25 (Bigram)	24.67	24.80
	BM25 (Trigram)	34.22	33.15
	BM25 (Quad-gram)	36.77	36.77
	BM25 (Penta-gram)	37.72	36.93

Table 6: The table shows the performance comparison (F1 scores in %) of the proposed method with the baseline unsupervised methods on the COLIEE-21 (Rabelo et al., 2022), the **IL-PCR** benchmark and the dataset with sentences having the citation removed: **IL-PCR**$_{\neg\text{sent}}$)

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Yes, After the conclusion section: Limitations

☑ A2. Did you discuss any potential risks of your work?
Yes, in the Ethics Section

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Yes, Abstract and Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Yes section 3, 4, and 5

□ B1. Did you cite the creators of artifacts you used?
No response.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

Yes, Section 4 and 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Yes, Section 4, 5 and Appendix

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Yes, Section 4, 5 and Appendix

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Yes, Section 4 and 5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Yes, Section 4, 5 and Appendix

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.