RESEARCH ARTICLE

RSP Science Hub

International Research Journal on Advanced Science Hub
2582-4376
www.rspsciencehub.com
Vol. 07, Issue 12 December

Check for updates

http://dx.doi.org/10.47392/IRJASH.2025.124

Legal Entity Tracking Over Time

K K Sruthi[1], C Vishnu Mohan[2]
[1]UG, Department of Computer Science, Sacred Heart College, Thevara, Kerala-682013, India.
[2]Assistant Professor, Department of Computer Science, Sacred Heart College, Thevara, Kerala, India.
Emails: *23umcp12110@shcollege.ac.in[1], vishnumohan@shcollege.ac.in[2]*

Article history
Received: 25 September 2025
Accepted: 26 October 2025
Published:26 December 2025

Keywords:
Case Law Analytics; Legal Data Mining; Named Entity Recognition(NER); Text Classification.

Abstract
Legal documents, such as court judgments and case files, are often lengthy and contain complex information. One way to understand this information is by using computer programs that can automatically find and highlight names, such as people, laws, courts, or organisations. This process is called Named Entity Recognition (NER). Many tools can locate these names in a single document. In this paper, we take it a step further for Legal Entity Tracking Over Time, which tracks how often and where these names appear across multiple documents over time. This data would enable insights into legal behaviour, such as identifying repeat petitioners, monitoring the activity of judges, or analysing the influence of legal statutes over time. It demonstrates how temporal entity tracking can enhance legal research, support predictive analytics, and contribute to more transparent legal data systems. By organising this information, the project makes legal data easier to understand and more useful for researchers, students, and legal professionals.

1. Introduction

Courts generate a significant volume of legal documents every day, including judgments, case summaries, and orders. These documents contain important information about the parties involved, relevant legal provisions, judicial interpretations, and established precedents. All this together forms the foundation of the legal system and influences future decisions, policies, and the public's understanding of the law. These documents are often lengthy, written in obtuse language, and difficult to navigate. This leads to difficulty in extracting specific information from these documents. For this, we have been utilizing Named Entity Recognition (NER) to scan text and identify and classify entities, including the names of people (petitioners, respondents, judges), organizations, courts, statutes, and legal provisions (cited laws and referenced cases). This automated extraction converts unstructured legal text into structured data, enabling faster retrieval, analysis, and integration with other information systems. Identifying the entities' addresses is only a part of the problem. Legal information often spans multiple cases and periods, and the absence of time-based entity tracking creates several issues. For example, the same judge may preside over several important rulings, or a company may appear in various disputes. Cases can come across disconnected, and opportunities for deeper analysis can be lost. These patterns remain hidden as there is no system to track them. Researchers and professionals need to do manual searches across multiple sources, which is inefficient, prone to oversight, and limits the scale of possible analysis. If patterns of repeated litigation or judicial involvement go unnoticed, it will be difficult to hold recurring offenders accountable. Policymakers and analysts may struggle to understand how specific laws are applied or challenged over extended periods. Furthermore, the broader public has limited access to clear, consolidated insights into legal activity, reducing transparency. A Legal Entity Tracking Over Time system will not only extract entities from legal documents but also link them to their appearances across multiple cases and time-frames. For example, a record of the cases handled by a particular judge, the frequency of litigation involving a specific company, or the timeline of how a statute has been applied. To be able to connect the past and present legal events will help in making informed decisions, enhance transparency, and lead to comprehensive research. The potential users of this system include legal researchers and academics seeking to analyze long-term trends, journalists investigating patterns in judicial behavior or case outcomes, lawyers preparing for litigation by reviewing prior cases involving the same parties, policymakers evaluating the application of laws, and public interest organizations monitoring legal actions on key social or regulatory issues. For each of these groups, the ability to view entities within a temporal framework transforms fragmented data into actionable knowledge. This project applies the concept to Indian court judgments in English, using a dataset that has cases from multiple decades. The system uses an NER to identify key entities, including petitioners, respondents, judges, statutes, provisions, and courts. These entities are then organized into a structured, time-based record, allowing for chronological analysis and visualization. Overall, this system will convert legal records from static, isolated texts into connected historical datasets. It would enable users to identify patterns, developments, and conduct targeted research efficiently. In a legal environment where precision, speed, and transparency are essential, such a system is not merely a technological convenience but an important tool for understanding and improving the administration of justice.

2. Related Works and Background

Named Entity Recognition in NLP focuses on identifying and categorizing predefined classes of entities in unstructured text. These entities can be names of people, organizations, locations, dates, etc. In a specific domain, as in here we are considering the legal field, the scope and nature of entities expand considerably to include case identifiers, citations to statutes, specific provisions, names of petitioners and respondents, judges, and court names. Legal NER therefore becomes an indispensable tool for enabling structured access to the massive volumes of legal text produced daily by courts, legislatures, and related bodies [20]. Legal documents are known to be complex due to their length, density, and the high concentration of domain-specific language. A single judgment may include multiple cross references to prior cases, statutes, and procedural events spanning years or decades. And manual extraction of these details is both time-consuming and prone to human error. It has also been noticed that the format and style of legal writing vary significantly across jurisdictions, with differences in citation style, abbreviation use, and even the ordering of document components [24]. These characteristics necessitate domain specific NER systems that go beyond general-purpose models. When it comes to legal documents, unlike journalistic or scientific text, these often contain overlapping entity types where a single phrase might be both a case title and the name of an involved organization, which further complicates the classification process.

2.1. Traditional and Transformer-Based NER

Transformer-driven approaches have significantly advanced Named Entity Recognition (NER), especially in domains requiring deep contextual understanding such as legal text. Earlier NER systems employed BiLSTM-CRF models, which captured sequential information and ensured coherent label prediction. Research on Indian-language legal NER demonstrated that combining word-level, character-level, and affix-level embeddings enhanced the model's ability to capture rich linguistic structure in resource-constrained settings [1]. Although these hybrid BiLSTM-CRF models achieved strong accuracy, subsequent work showed that transformer encoders such as RoBERTa provided superior performance on long, complex judicial documents due to better handling of long-range dependencies [1]. A broad comparative evaluation of NER models across domains further reinforced this shift, with BERT, RoBERTa, and similar architectures outperforming

BiLSTM-CNN-CRF baselines in terms of recall and overall F1 scores [2]. However, precision did not always increase, revealing that transformers may introduce more false positives unless supported by domain adaptation or specialized post-processing [2]. This observation underscores the importance of domain-specific models in legal NLP. A major advancement for legal text processing came from Legal-BERT, a domain-pretrained transformer tuned specifically on legal corpora. Chalkidis et al. demonstrated that legal-specific pretraining significantly improves entity detection, contextual understanding, and robustness to domain-specific vocabulary compared to generic BERT models. This confirms that specialized encoders are essential for high-quality NER in legal domains [4]. Models such as TENER further refine transformer performance by incorporating direction- and distance-aware attention to improve entity boundary detection in structurally complex text [3].

2.2. Advances in Sequence Labeling

Sequence labeling methods have driven progress in Named Entity Recognition (NER) and related tasks, evolving from feature-rich statistical models to end-to-end neural architectures that jointly learn representations and label dependencies. A recent survey [11] breaks down the advances in this field into three main parts: embedding modules, context encoders, and inference layers. The summary shows how innovations such as character-level embeddings, bidirectional recurrent encoders, and structured decoders like CRFs have consistently improved generalization, robustness to out-of-vocabulary to kens, and boundary consistency in tasks such as NER, part of-speech tagging, and chunking. The BiLSTM-CNNs-CRF model is the most influential architecture. It brings together character-level CNNs, BiLSTM word encoders, and a CRF decoding layer [7]. This model demonstrated strong performance on benchmark NER datasets by using learned subword features, bidirectional con text modeling, and sequence-consistent decoding, without re lying on extensive handcrafted features. Bidirectional LSTM CRF models more broadly have established themselves as the canonical neural sequence taggers [9]. These architectures improved boundary detection by directly modeling both forward and backward dependencies in text and adding a CRF layer. They reduced inconsistent label assignments across multi-token entities, making them widely adopted baselines for domain-specific NER tasks. Expanding on the CRF framework, Neural CRF Transducers introduced a more expressive strategy by merging an RNN-based observation encoder with an RNN that models label sequences [5]. This architecture goes beyond linear chain dependencies, capturing long-range label interactions that occur frequently in complex or nested constructs, which are common in technical and legal documents. The results showed consistent improvements over standard BiLSTM-CRF models in maintaining global label coherence. Reimers et al. introduced Sentence-BERT (SBERT), an adaptation of BERT designed to produce efficient sentence-level embeddings using a Siamese network setup. Unlike standard BERT, which requires costly pairwise comparisons, SBERT enables fast semantic similarity scoring across large document collections [6] The model effectively captures fine-grained meaning between sentences, making it well suited for tasks such as cross-document entity linking and identifying semantically similar legal passages. Its ability to represent legal statements at the sentence level provides a strong complement to token-level NER models in complex judicial corpora. In Turkish product name extraction, BiLSTM-CRF showed that combining subword embeddings with sequence tagging improves recall in morphologically rich, low-resource languages [10]. These findings match results from keyphrase extraction in scholarly documents. In that setting, BiLSTM CRF effectively modeled domain-specific terminology and outperformed unsupervised and rule-based baselines [8]. This body of work highlights a stable architectural recipe for effective sequence labeling. It uses subword-aware embed dings to handle morphology and unseen tokens, bidirectional recurrent networks to capture contextual dependencies, and structured decoders such as CRFs or transducers to enforce global label consistency. While transformers have more recently dominated, BiLSTM-CRF and its variants remain competitive, interpretable, and well-suited for specialized domains, particularly where annotated data is limited or morphological complexity is high [9] [7] [5] [8] [6] [11] [10].

2.3. Argument Mining in Legal Text Summarization

Argument mining is an essential NLP method for detecting argumentative structures in text. It is especially valuable in legal domains, where reasoning is central. Xu, ˇSavelka, and Ashley (2020) had presented the notion of legal argument triples as a structured framework to capture the flow of judicial reasoning. They were Issue, Reason, and Conclusion (IRC). These triples serve as the core of case summaries as they allow for concise representations of complex judgments. The authors used a dataset from the Canadian Legal Information Institute (CanLII) with over 28,000 case-summary pairs for their study. Law students manually annotated a subset of 574 summaries and 109 full cases for IRC components. Inter-annotator agreement showed strong reliability. Annotators found Issues and Conclusions easier to identify than Reasons, since Reasons often overlapped with factual details. They conducted four main experiments: multi-class classification on summaries and full texts, and binary classification distinguishing IRC from non-IRC sentences. Comparisons of Random Forests, LSTM, CNN, and FastText showed that CNNs performed best for summaries. Random Forests with sampling strategies worked better for full texts because of class imbalance. The findings show that summaries are easier to classify automatically, but full judgments are harder without balancing strategies. The findings showed that it is easier to detect Conclusions and Issues than Reasons, because Conclusions and Issues have clearer linguistic cues. Despite this, detecting conclusions across cases consistently shows that they can anchor auto mated summarization. The study provides a valuable annotated dataset for legal argument mining and fills a resource gap. Ultimately, the research demonstrated that argument mining can support legal text summarization. Further progress requires more effective extraction of complex reasoning components [12].

2.4. Information Extraction in Legal Documents

Information extraction (IE) in legal texts aims to convert unstructured provisions, clauses, and decisions into structured representations that support retrieval, compliance checking, and reasoning. The paper provides a full rundown of legal IE, initially introducing the three complementary axes: NLP pipelines, deep learning, and knowledge base population, which are essential for understanding the field. It argues that mature systems typically combine these strands to address legal language variability and document complexity [17]. The survey catalogues core tasks (named entity recognition, relation extraction, semantic role labeling, dependency parsing). It emphasizes ontology grounding and canonicalization for interoperability across sources and time [17]. Following the discussion on core tasks, ontology-centered IE emerges as one of the earliest and most persistent approaches, playing a crucial role in structuring legal information. Automatic legal document analysis demonstrates how layered ontologies can model legal concepts and their linguistic realizations, enabling semantic annotation and faceted retrieval over heterogeneous collections, thereby improving the understanding of legal information [15]. Complementary research on automatic semantics extraction enhances the ontology-centered IE by mapping textual segments to deontic and functional roles such as obligation, permission, prohibition, actor, and action, thereby bridging raw text with machine-readable knowledge structures [14]. These studies demonstrate how ontologies serve as target schemas and validation mechanisms, improving the precision and reuse of extracted information. Building on ontology-centered IE, domain-specific applications illustrate the practicality of hybrid, ontology-aware IE. Automated construction specification review integrates NER and rule-based checks with a domain ontology, allowing the system to detect nonconformities, missing requirements, and inconsistent references in technical specifications [13]. This application shows that effective IE must combine linguistic extraction with domain constraints and identifiers to produce actionable outputs for practitioners. After exploring domain-specific applications, the discussion naturally progresses to hybrid pipelines, which explicitly combine syntactic, semantic, and logical processing to further reduce ambiguity in norm extraction. A representative frame work parses sentences with a constituency/dependency parser, normalizes vocabulary via lexical resources, and composes logical forms (e.g., with a semantic parser) before mapping to an ontology of obligations, permissions, and prohibitions [16]. Evaluated on a real regulatory code, this approach outperforms single-method baselines because it captures long-distance dependencies and clarifies scope and exceptions, recurring pain points in legal drafting [16]. The contributions discussed, including surveys, ontology centric methods, domain applications, and hybrid NLP–logic pipelines, collectively reveal a consistent pattern: robust legal IE requires (i) high-recall linguistic extraction (entities, roles, relations), (ii) ontology-driven normalization and linking to stabilize meaning across documents, and (iii) task- or domain specific constraints to elevate precision. Integrating ontologies with statistical or neural extractors is key to scaling legal IE while preserving legal fidelity.

2.5. Annotation for Legal Text

Current research, such as Savelka's, also focuses on the automated semantic annotation of legal texts using large language models (LLMs). Semantic annotation, a key focus of current research, is crucial for comprehending and processing legal documents as it enables the extraction of semantic information from legal questions and facilitates efficient legal analysis. LLMs propose a novel solution to this issue, and in zero-shot settings in particular, they enable researchers to employ models without re-training them. Zero-shot setups aim to reduce the cost and complexity of creating large annotated corpora. One specific study, 'The Unreasonable Effectiveness of Large Language Models in Zero-Shot Annotation of Legal Texts,' tests whether LLMs in the state of the art can provide semantic role labeling to legal sentences conditioned only on the text-based descriptions of annotation categories [18]. Our study discovered that LLMs can perform on par with traditional statistical models and fine-tuned transformer models for legal and statutory clauses. They are also comparable in adapting to different types of legal documents, such as contracts, regulations, and decisions, while being more efficient. Nevertheless, while LLMs are effective at capturing semantic context, they can be brittle, producing unreliable predictions in complex lexical scenarios, such as those involving intricate legal terms. This vagueness makes them unreliable for some applications. For example, Savelka's work builds on this and performs benchmarking of GPT-3.5 and various classical statistical models, such as random forest, and pre-trained transformer models, including RoBERTa. It was done to evaluate its effectiveness on legal text annotation [19]. Three categories were evaluated in the study: adjudicatory determinations, contract terms, and statutory/regulatory provisions. GPT-3.5 substantially outperformed simple similarity measures employed as benchmarks, and it was competitive in some tasks with supervised models learned on a small amount of data. For instance, in contract annotation, GPT obtained an F1 score as high as 0.86, demonstrating its ability to expedite annotation for documents in online legal-services applications like contract review or case analysis. However, performance differed between the domains, being particularly low in the adjudicatory "Reasoning" sub structures and in the legislation domain. The combined results from the work of Savelka and others indicate that LLMs, which do not require extensive training, offer a strong direction for cost saving in legal annotation by eliminating the requirement for large labeled datasets. Their competitive performance and adaptability across legal domains provide strong evidence of their potential despite limitations in handling terminology. Their ability to generalize from definitions without labeled training data offers a scalable solution for enriching the semantics of legal texts. These ideas are valuable for legal researchers, practitioners, and policymakers seeking to integrate AI-driven annotation into practical applications like document review, compliance, and legal analytics pipelines [18] [19]

2.6. Datasets and Data-Efficient Annotation for Legal NLP

High-quality, domain-specific datasets and annotation strategies support the scaling of legal NLP and tasks such as Legal Entity Tracking Over Time (LETOT). Current works have advanced both the creation of large legal corpora and methods for reducing annotation cost while retaining model performance. Kalamkar et al. introduced a specialized dataset for Indian court judgments designed for legal NER. The authors release a corpus comprising 14 legal entity types, which reports 46,545 annotated entity mentions drawn from 14,444 judgment sentences and 2,126 preambles. They also provide a transformer based NER baseline and document-level post-processing rules to improve extraction and coreference resolution [20]. This resource is really useful for LETOT because it combines fine-grained entity typing with representative sampling across 29 Indian courts and eight case types, enabling longitudinal studies of entities within an Indian judicial context [20]. For contractual provisions and large-scale classification, Tuggener et al. gave a complementary perspective with LEDGAR. It is a large, semi-automatically labelled corpus of contract provisions (constructed from SEC EDGAR filings) containing on the order of 105 provisions and a label set originally exceeding 12,000 provision types. The authors use data cleaning, label-hierarchy extraction, and subsampling techniques to handle extreme multi-label classification [22] LEDGAR is useful for LETOT-style analyses that require tracking provision types, clause recurrence, and corporate contract behaviour over time across thousands of contracts [22]. Mamooler et al. propose an active-learning pipeline for legal text classification, noting that labeled data in this domain is costly. Their approach (1) task-adapts a pretrained LM with unlabeled in-domain text, (2) uses knowledge distillation to produce semantically meaningful embeddings, and (3) initializes annotation via clustering medoids to drastically reduce the number of annotator actions. On Contract-NLI and LEDGAR benchmarks, the pipeline yields substantial efficiency and quality gains (e.g., large F1 improvements and up to 63% fewer initial annotation actions on skewed data). The paper demonstrates active learning and task adaptation that can approach fully supervised performance at a significantly lower annotation cost [21]. With LexSumm and LexT5, Santosh et al. provide both a generative benchmark and a seq2seq model tailored for legal summarization using eight English datasets covering the US, UK, EU, and India. The LexSumm effort highlights the difficulty of long-document legal generation and provides LexT5 as a pre-trained seq2seq backbone for long-context legal summarization and probing; these resources help evaluate how well models preserve legal facts and entity references when producing summaries—vital for downstream LETOT tasks that rely on faithful, entity-preserving generation or extraction [23]. Together, these works show a practical path for LETOT, which is a combined domain-specific corpora (inJudgments, LEDGAR, LexFiles/LexSumm) with task-adaptation and active-learning strategies to scale annotation, improve ex traction quality, and enable robust temporal linking of entities across large, heterogeneous legal collections.

2.7. Evaluation Methodologies and NER Challenges

Evaluation is the backbone of reliable NER, as it determines what actually works and identifies the system's weaknesses that need to be addressed before it is used in real-world applications. Peddavenkatagari et al. (2024) evaluated the models based on their token-level and entity-level precision, recall, and F1. They emphasized the need to consistently report the matching rules (i.e., strict vs. partial-span), as legal and scientific texts often contain numerous multi-token entities and intricate citations that partial matches might resolve ambiguously. They also propose robustness testing, cross-validation, held-out domain splits, out-of-sample testing, and, when applicable, error analysis and human evaluation, as supplements to scores for high-stakes applications [25]. These strategies help to ensure reported gains in results are generalizable to the real world as opposed to tuning to the datasets [25] Pakhale et al. identify evaluation challenges, such as nested entities, ambiguous legal terms, frequent acronyms, OCR errors in scanned texts, and drifting entities, including company rebranding or judge reassignment. They demonstrate how these problems can bias evaluations when not accounted for in the dataset design and test-set split. The methods in some of these are presented in the paper, such as span-based scoring, nested-NER benchmarks, uncertainty-aware models, and distant supervision for robustness improvement [24]. Together, Peddavenkatagari et al. and Pakhale et al. suggest a pragmatic evaluation roadmap for legal NER and LETOT: (a) report strict and partial-span metrics broken down by recall and precision per entity type and span length; (b) evaluate in cross-domain and temporal holdout scenarios; (c) streamline error analysis (boundary errors, type confusions, OCR-induced errors) and (d) combine automated metrics with targeted human validation focused on legal utility [24] [25].

2.8. Privacy-Preserving Approaches for NER

Privacy-preserving methods have become central when applying NER to sensitive domains (healthcare, legal, user data). Four complementary strategies have occurred in recent work: mimic (student–teacher) learning, federated learning, automatic/noisy annotation with distributed training, and entity masking or type-replacement preprocessing. Mimic learning (teacher-student) trains a high-performing teacher model on sensitive data from inside the company. It uses it to make”silver” labels on public or unlabeled corpora. You can then share student models that were trained on those silver labels without giving away the original private data. Bannour et al. illustrate this methodology for French Clinical Named Entity Recognition. A private teacher who has been trained on limited clinical notes makes silver annotations for several public corpora. Student models that have been trained on those labels reach a useful level of teacher performance while also allowing safe model sharing. The study defines distinct privacy-utility trade-offs and offers released silver annotations to facilitate reproducibility. [26] Federated learning protects privacy by keeping data on the user's device and sending model updates instead of raw text. FedNER breaks down client models into a shared module (centralized aggregation) and private modules (local adaptation). This lets different sites use different annotation schemes while still getting the benefits of multi-site supervision. Experiments demonstrate that this shared/private decomposition enhances cross-site generalization relative to naive federated averaging. This is beneficial when organizations can't share data but want to use distributed labels. FedNER also talks about real-world engineering problems, like label-schema mismatch and how well communication works. [29] Another useful way is to use programmatic/noisy annotation with distributed training. Hathurusinghe et al. developed WikiPII, an automatically labeled PII corpus derived from Wikipedia infoboxes, demonstrating that substantial amounts of noisy labels can facilitate remotely trained transformer NER models. They further examine federated training workflows for PII extraction and assess the influence of label noise and dataset scale on performance. Their findings suggest that programmatic annotation combined with distributed training can diminish manual labeling efforts while maintaining satisfactory accuracy under specific trust and data volume conditions. [28] A complementary, lightweight approach uses NER as a privacy-preserving preprocessing step: Kutbi shows replacing detected named entities with their type labels (e.g., PERSON, ORG) improves downstream text-classification robustness and reduces feature sparsity while masking sensitive tokens. This technique is simple to deploy, lowers data exposure in pipelines, and can be combined with other privacy-preserving strategies, though it may remove some predictive signal for tasks that rely on specific names. [27]

Common trade-offs across these studies are consistent:

- Privacy gains typically come with some performance cost (e.g., silver-student gap, communication overhead, noisy labels).
- Label-schema alignment issues in federated setups.
- The need for human-in-the-loop validation for high-stakes uses.

Best practices emerging from the literature recommend hybrid solutions:

- Mimic learning or programmatic annotation to generate shareable training data.
- Federated training with shared/private modules for cross institutional gains.
- Lightweight masking for low-risk scenarios, combined with careful evaluation of privacy leakage risks and targeted expert validation.

3. Method

The methodology of this project is designed to extract, normalize, and track legal entities across judicial documents over time. Unlike large-scale industrial systems that combine multiple deep learning architectures, this work adopts a

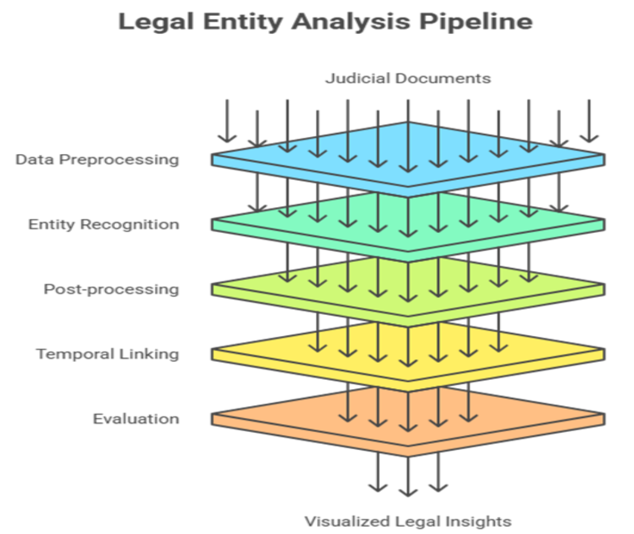

Figure 1 **Overview of The Methodology Pipeline**

streamlined pipeline built on two core pretrained models, Legal-BERT and Sentence-BERT, augmented with rule-based normalization and lightweight database management. This combination offers both conceptual depth and practical feasibility for academic implementation. The approach is organized into six major stages.

3.1. Data Collection and Preprocessing

- **Dataset Selection:** The dataset is drawn from the Indian Court Judgments dataset [20], which contains thousands of annotated legal documents sourced from the Supreme Court and multiple High Courts of India. This dataset spans decades and case categories (civil, criminal, constitutional, and regulatory), ensuring that the system captures diverse entities such as statutes, provisions, judges, and litigants. This corpus is particularly well-suited for Legal Entity Tracking Over Time (LETOT) due to its extensive temporal coverage and representative sampling of judgments.
- **Text Cleaning:** Legal documents, particularly older judgments, often include OCR artifacts, page headers, and for matting irregularities. Preprocessing begins with rule-based cleaning using Python regular expressions. OCR noise, such as broken words and repeated headers, is removed. Boilerplate legal phrases (e.g., "Heard learned counsel for both parties") are preserved, since they often mark the presence of named entities like counsels or petitioners.
- **Sentence Segmentation and Tokenization:** The cleaned text is segmented into sentences using spaCy's dependency aware tokenizer, adapted for long legal sentences. Tokenization is carried out using WordPiece tokenization from the Legal BERT model. This ensures that rare or complex legal expressions (e.g., habeas corpus, res judicata, Section 138 NI Act) are decomposed into subwords while retaining their semantic integrity.
- **Text Normalization:** Abbreviations and shorthand notations are expanded (e.g., "CrPC" → "Code of Criminal Procedure"). Canonical representations of statutes and provisions are standardized during preprocessing itself, enabling the downstream NER model to train on consistent patterns.

3.2. Named Entity Recognition (NER)

- **Entity Categories:** The task involves recognizing entities most relevant to longitudinal legal analysis: Petitioners and Respondents • Judges • Statutes and Provisions • Courts • Organizations Labels follow the BIO (Begin–Inside–Outside) scheme to ensure consistency in span detection.
- **Model Architecture:** The Legal-BERT model is used as the base encoder. Legal-BERT is a transformer pretrained on legal corpora, making it better suited for domain-specific syntax and terminology than general-purpose BERT. Its contextual embeddings capture long-range dependencies typical of judgment text. A simple classification head (fully connected layer + softmax) is added on top to predict BIO-tag sequences.

3.3. Post-processing and Normalization

- **Canonicalization:** After NER, extracted entities are normalized into canonical forms. For example: • "S. 138 NI Act" → "Section 138, Negotiable Instruments Act" • "Sup. Ct." → "Supreme Court of India" This is implemented using dictionary lookups, abbreviation maps, and fuzzy string matching (via RapidFuzz). Unlike ontology-heavy methods, this lightweight approach is simpler to implement while still effective in unifying entity mentions.
- **Disambiguation:** Rules are applied to distinguish between ambiguous terms. For instance, "State" is treated as a government party if followed by a proper noun (State of Maharashtra), but as a generic reference if used in isolation. Context-based keyword checks within ±10 tokens help decide the interpretation.
- **Coreference Resolution (Simplified):** Instead of training a separate neural model, heuristic rules link repeated mentions like "the petitioner" or "the accused" back to the nearest named entity of that role.

	Titles	Court_Name	Cites	Cited_by	Doc_url
0	Maheshwar Mandal & Anr vs The State Of Bihar & Ors on 24 June, 2014	Patna High Court - Orders	5	9	https://indiankanoon.org/doc/154031036/
1	Sri Ashok Goenka & Anr vs Chandra Bhushan Singh & Ors on 15 September, 2009	Patna High Court - Orders	13	6	https://indiankanoon.org/doc/44290554/
2	Mt.Prabhawati & Ors vs The State Of Bihar & Ors on 24 November, 2011	Patna High Court - Orders	11	2	https://indiankanoon.org/doc/814453/
3	Md. Nizamuddin vs The State Of Bihar & Ors on 15 May, 2014	Patna High Court - Orders	39	1	https://indiankanoon.org/doc/173691638/
4	Vodafone Essar Spacetel Ltd.&A vs The State Of Bihar & Ors on 21 March, 2013	Patna High Court - Orders	12	1	https://indiankanoon.org/doc/198757445/
5	Phuljharia Devi & Anr vs The State Of Bihar & Ors on 18 September, 2009	Patna High Court - Orders	28	0	https://indiankanoon.org/doc/49275886/

Text	Doc_size	Case_Type	Court_Type	Court_Name_Normalize
IN THE HIGH COURT OF JUDICATURE AT PATNA\n Civil Writ Jurisdiction Case No.1091 of 2...	40125	Land&Property	High_Court	Patna High Court
IN THE HIGH COURT OF JUDICATURE AT PATNA\n MA No.38 of 2004\n 1.SRI ASHOK GOENKA ...	24428	Land&Property	High_Court	Patna High Court
IN THE HIGH COURT OF JUDICATURE AT PATNA\n LPA 681 of 2010\n 1.Mt. Prabhavati...	27844	Land&Property	High_Court	Patna High Court
IN THE HIGH COURT OF JUDICATURE AT PATNA\n Letters Patent Appeal...	104812	Land&Property	High_Court	Patna High Court
IN THE HIGH COURT OF JUDICATURE AT PATNA\n CIVIL WRIT JURISDICTION CASE NO.8317 OF 2009\n========...	30586	Land&Property	High_Court	Patna High Court
IN THE HIGH COURT OF JUDICATURE AT PATNA\n CWJC No.3483 of 2007\n ...	38562	Land&Property	High_Court	Patna High Court

Figure 2 Sample of the Dataset

3.4. Temporal Entity Linking

- **Entity Linking Model:** Temporal linking is performed using Sentence-BERT (SBERT), which produces dense embeddings for sentences and entity mentions. SBERT is particularly effective for measuring semantic similarity between text spans. For example: • "Justice A. Sharma" and "Justice Anil Sharma" yield embeddings with cosine similarity above 0.90, leading them to be merged into the same canonical judge entity. • Statute references like "Sec. 138 NIA" and "Section 138, Negotiable Instruments Act" are clustered together.
- **Unique Identifiers and Storage:** Each entity is assigned a globally unique identifier (GUID). A relational database (PostgreSQL/SQLite) stores entity records with attributes: case ID, entity type, normalized name, canonical ID, and date of judgment. This allows cross-document queries without the complexity of graph databases.
- **Evolution Tracking:** Temporal changes, such as organizational renaming, are resolved by comparing embeddings and linking aliases to the same GUID. This makes it possible to observe how entities evolve over decades (e.g., how the same company reappears in multiple litigations under slightly different names).

3.5. Evaluation

- **Metrics:** Evaluation is conducted at two levels: NER performance: measured using precision, recall, and F1-score under both strict (exact span/type match) and partial-span scoring. Entity linking performance: assessed with linking accuracy and threshold-based cosine similarity evaluation.
- **Baselines:** For NER: simple rule-based baselines (regex patterns for statute references). For linking: naive string-matching baseline. Comparing transformer-based approaches to these baselines demonstrates the value of Legal-BERT and SBERT.

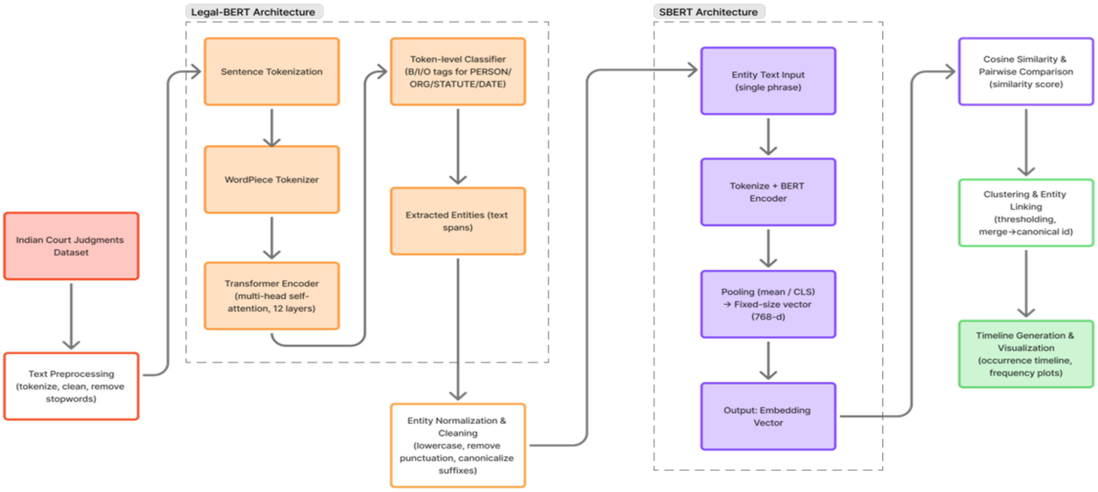

Figure 3 LETOT Model Architecture

- **Human Validation:** Legal scholars manually review a subset of outputs (200–300 cases). Cohen's is reported to measure inter-annotator agreement between system predictions and human judgment.

3.6. Visualization and Analysis

4. **Timelines:** Using Plotly or Streamlit, temporal trends are visualized. Users can view: Number of cases involving Section 138 of the NI Act per year. The caseload of a judge over a decade.

- **Entity Networks:** Co-occurrence graphs are constructed to show relationships between judges, statutes, and litigants. For example,

statutes frequently cited together are displayed as clusters.
- **Use Cases:** Legal Research: identify recurring legal provisions. Judicial Monitoring: monitor workload distribution among judges. Policy Evaluation: measure the effect of new legislation on citation frequency.

Conclusion

This study addressed a critical research gap in legal data analysis, which is the absence of systems capable of not only identifying but also tracking legal entities across multiple judicial documents over time. Prior work in this field has focused mainly on NER within a single document. So, the understanding of how these entities evolve, recur, or interrelate across years of legal activity was limited. Our framework bridges this gap by integrating Legal-BERT for entity extraction and Sentence-BERT for temporal linking, supported by normalization and visualization techniques. This contribution aims to transform static, text-heavy court judgments into dynamic, interconnected legal knowledge. Using temporal tracking of entities, LETOT can offer new opportunities for transparency, accountability, and evidence-based policy evaluation. Researchers can use it to identify recurring litigants or frequently cited provisions, legal practitioners can explore precedent patterns, and policymakers can analyze the long-term influence of laws and judicial behavior. Despite its promising results, the system can be further improved. Our findings suggest that future work can focus on expanding multilingual support for regional court judgments, integrating neural coreference resolution to improve entity linking, and exploring graph-based visualization for deeper relational insights. The use of federated learning could also en sure data privacy while expanding the scope of training across multiple jurisdictions. Performing federated learning in real world deployments is challenging. It remains as future work because it requires secure communication, strong hardware, and careful handling of sensitive legal data.

Acknowledgements

The authors received no financial support for this study. We thank all individuals who provided feedback and guidance during the research process.

References

[1]. Ajees, A. P., Manju, K., & Idicula, S. M. (2019). An improved word representation for deep learning-based NER in Indian languages. Information, 10(6), 186. https://doi.org/10.3390/info10060186

[2]. Lothritz, C., Allix, K., Veiber, L., Bissyand´e, T. F., & Klein, J. (2020). Evaluating pretrained transformer-based models on the task of fine grained named entity recognition. In D. Scott, N. Bel, & C. Zong (Eds.), Proceedings of the 28th International Conference on Computational Linguistics (pp. 3750–3760). International Committee on Computational Linguistics. https://doi.org/10.18653/v1/2020.coling-main.334

[3]. Yan, H., Deng, B., Li, X., & Qiu, X. (2019). TENER: Adapting transformer encoder for named entity recognition. arXiv. https://doi.org/ 10.48550/arXiv.1911.04474

[4]. Chalkidis, I., Malakasiotis, P., Fergadiotis, M., Aletras, N., & Androutsopoulos, I. (2020). LEGAL-BERT: The Muppets straight out of law school. arXiv preprint arXiv:2010.02559. https://arxiv.org/abs/2010.02559

[5]. Hu, K., Ou, Z., Hu, M., & Feng, J. (2019, January 12–17). Neural CRF transducers for sequence labeling. In Proceedings of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). https://doi.org/10.1109/ICASSP.2019.8682728

[6]. Reimers, N., & Gurevych, I. (2019). Sentence BERT: Sentence embeddings using Siamese BERT networks. arXiv preprint arXiv:1908.10084. https://arxiv.org/abs/1908.10084

[7]. Ma, X., & Hovy, E. (2016, March 4). End-to-end sequence labeling via bidirectional LSTM-CNNs-CRF. arXiv. https://doi.org/10.48550/arXiv. 1603.01354

[8]. Al-Zaidy, R. A., Caragea, C., & Giles, C. L. (2019). Bi-LSTM-CRF sequence labeling for keyphrase extraction from scholarly documents. In Proceedings of The Web Conference 2019 (WWW ’19). Association for Computing Machinery. https://doi.org/10.1145/3308558.3313642

[9]. Huang, Z., Xu, W., & Yu, K. (2015, August 9). Bidirectional LSTM CRFmodels for sequence tagging. arXiv. https://doi.org/10.48550/arXiv. 1508.01991

[10]. Arslan, S. (2024). Application of BiLSTM-CRF model with different embeddings for product name extraction in unstructured Turkish text. Neural Computing and Applications, 36(15), 8371–8382. https://doi.org/ 10.1007/s00521-024-09532-1

[11]. He, Z., Wang, Z., Wei, W., Feng, S., Mao, X., & Jiang, S. (2020, November 13). A survey on recent advances in sequence labeling from deep learning models [Preprint]. arXiv. https://doi.org/10.48550/arXiv. 2011.06727

[12]. Xu, H., ˇ Savelka, J., & Ashley, K. D. (2020). Using argument mining for legal text summarization. In S. Villata et al. (Eds.), Legal Knowledge and Information Systems: Frontiers in Artificial Intelligence and Applications (Vol. 334, pp. 184–193). IOS Press. https://doi.org/10.3233/FAIA200862

[13]. Moon, S., Lee, G., Chi, S., & Oh, H. (2021). Automated construction specification review with named entity recognition using natural language processing. Journal of Construction Engineering and Management, 147(1), Article 04020147. https://doi.org/10.1061/(ASCE)CO. 1943-7862.0001953

[14]. Biagioli, C., Francesconi, E., Passerini, A., Montemagni, S., & Soria, C. (2005). Automatic semantics extraction in law documents. Proceedings of the 10th International Conference on Artificial Intelligence and Law, 133–140. Association for Computing Machinery. https://doi.org/ 10.1145/1165485.1165506

[15]. Buey, M. G., Roman, C., Garrido, A. L., Bobed, C., & Mena, E. (2019). Automatic legal document analysis: Improving the results of information extraction processes using an ontology. In R. Bembenik, Ł. Skonieczny, G. Protaziuk, M. Kryszkiewicz, & H. Rybinski (Eds.), Intelligent methods and big data in industrial applications (Studies in Big Data, Vol. 40, pp. 401–417). Springer. https://doi.org/10.1007/978-3-319-77604-0 24

[16]. Dragoni, M., Villata, S., Rizzi, W., & Governatori, G. (2016). Combining NLP approaches for rule extraction from legal documents. In Proceedings of the 1st Workshop on Mining and Reasoning with Legal Texts (MIREL 2016), Sophia Antipolis, France. https://hal.science/ hal-01572443v1

[17]. Zadgaonkar, A. V., & Agrawal, A. J. (2021). An overview of information extraction techniques for legal document analysis and processing. International Journal of Electrical and Computer Engineering, 11(6), 5450–5457. https://doi.org/10.11591/ijece.v11i6.pp5450-5457

[18]. Savelka, J. (2023). Unlocking practical applications in legal domain: Evaluation of GPT for zero-shot semantic annotation of legal texts. In Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law (ICAIL 2023) (June 19–23, Braga, Portugal) (pp. 447–451). ACM. https://doi.org/10.1145/3594536.3595161

[19]. Savelka, J., & Ashley, K. D. (2023). The unreasonable effectiveness of large language models in zero-shot semantic annotation of legal texts. Frontiers in Artificial Intelligence, 6, Article 1279794. https://doi.org/ 10.3389/frai.2023.1279794

[20]. Kalamkar, P., Agarwal, A., Tiwari, A., Gupta, S., Karn, S., & Raghavan, V. (2022). Named entity recognition in Indian court judgments. arXiv. https://doi.org/10.48550/arXiv.2211.03442

[21]. Mamooler, S., Lebret, R., Massonnet, S., & Aberer, K. (2022, December). An efficient active learning pipeline for legal text classification. In Proceedings of the Natural Legal Language Processing Workshop 2022 (pp. 345–358). Association for Computational Linguistics. https: //doi.org/10.18653/v1/2022.nllp-1.32

[22]. Tuggener, D., von D¨aniken, P., Peetz, T., & Cieliebak, M. (2020). LEDGAR: A large-scale multi-label corpus for text classification of legal provisions in

contracts. In N. Calzolari, F. B´echet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, & S. Piperidis (Eds.), Proceedings of the Twelfth Language Resources and Evaluation Conference (pp. 1235–1241). European Language Resources Association. https://aclanthology.org/2020.lrec-1.155/

[23]. Santosh, T. Y. S. S., Weiss, C., & Grabmair, M. (2024). LexSumm and LexT5: Benchmarking and modeling legal summarization tasks in English [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2410.09527

[24]. Pakhale, K. (2023, September 25). Comprehensive overview of named entity recognition: Models, domain-specific applications and challenges. arXiv. https://doi.org/10.48550/arXiv.2309.14084

[25]. Peddavenkatagari, C. R. (2024). Evaluation methodologies and performance metrics in supervised NER for journal articles: A critical analysis. International Research Journal of Modernization in Engineering Technology and Science. Retrieved from https://www.researchgate.net/ publication/380068332

[26]. Bannour, N., Wajsb¨urt, P., Rance, B., Tannier, X., & N´ev´eol, A. (2022). Privacy-preserving mimic models for clinical named entity recognition in French. Journal of Biomedical Informatics, 134, 104073. https://doi. org/10.1016/j.jbi.2022.104073

[27]. Kutbi, M. (2023). Named entity recognition utilized to enhance text classification while preserving privacy. IEEE Access, 11, 117576 117581. https://doi.org/10.1109/ACCESS.2023.332 5895

[28]. Hathurusinghe, R., Nejadgholi, I., & Bolic, M. (2021). A privacy preserving approach to extraction of personal information through automatic annotation and federated learning [Preprint]. arXiv. https: //doi.org/10.48550/arXiv.2105.09198

[29]. Ge, S., Wu, F., Wu, C., Qi, T., Huang, Y., & Xie, X. (2020). FedNER: Privacy-preserving medical named entity recognition with federated learning [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2003.09288