Legal Entity Extraction: An Experimental Study of NER Approach for Legal Documents

Varsha Naik[1], Purvang Patel[2], Rajeswari Kannan[3]
Pimpri Chinchwad College of Engineering, Pune, India[1, 3]
Dr. Vishwanath Karad MIT World Peace University, Pune, India[1, 2]

***Abstract*—In legal domain Name Entity Recognition serves as the basis for subsequent stages of legal artificial intelligence. In this paper, the authors have developed a dataset for training Name Entity Recognition (NER) in the Indian legal domain. As a first step of the research methodology study is done to identify and establish more legal entities than commonly used named entities such as person, organization, location, and so on. The annotators can make use of these entities to annotate different types of legal documents. Variety of text annotation tools are in existence finding the best one is a difficult task, so authors have experimented with various tools before settling on the best one for this research work. The resulting annotations from unstructured text can be stored into a JavaScript Object Notation (JSON) format which improves data readability and manipulation simple. After annotation, the resulting dataset contains approximately 30 documents and approximately 5000 sentences. This data further used to train a spacy pre-trained pipeline to predict accurate legal name entities. The accuracy of legal names can be increased further if the pre-trained models are fine-tuned using legal texts.**

Keywords*—Named Entity Recognition; NER; legal domain; text annotation; annotation tools

I. INTRODUCTION

Artificial Intelligence (AI) has the potential to improve both the efficiency and accessibility of numerous legal processes [1]. In the current digital era, online document collections are growing rapidly. Technology and automation can help to extract information from these collections. As the amount of data continuously increasing, it is more and more necessary to access and process these data. The use of natural language processing is significant. NER, one of Natural Language Processing's (NLP) fundamental building blocks, can be used to develop AI applications in the legal domain [2]. Name entity recognition is a process of locating and classifying named entities in an unstructured text into predefined categories.

Name entity recognition is used to find a link to rigid notations in text that are related to well-known semantic classes like person, place, organization, etc. NER is used not only as a standalone tool for information extraction (IE) [3], but also in a variety of natural language processing (NLP) applications such as text understanding, information retrieval, automatic text summarization, question answering, machine translation, and knowledge base construction, and many others. Information retrieval, question-and-answer systems, machine translation, and many more applications use NER as a crucial pre-processing step [4].

To achieve high performance in NER, large amounts of knowledge in the form of feature engineering and lexicons have traditionally been required [5]. Also, there is great advancement in machine learning algorithms and deep learning algorithms in natural language processing and more specifically name entity recognition and information extraction [6]. Depending on the problem, such methods typically require a large set of manually annotated data,[5] whereas some machine learning algorithms rely on unsupervised techniques that do not require a large set of annotated data. There is an active learning-based clustering technique that is a subset of the semi-supervised technique and is used to reduce manual annotation time [7].

Annotation is a practice of adding linguistic and interpretive information to an electronic corpus of spoken or written linguistic data. Basically, annotation means adding a note to the input data. Annotation of words and characters are quite common for exactly distinctive medical specialty entities, resembling genes, proteins, and diseases [8]. In previous work, Jackson M.Steinkamp, Abhinav Sharma has annotated the unstructured clinical notes to identify the symptoms within the electronic health records. In another work related to the medical name entity recognition have prepared their dataset by annotating notes of pneumonia patients [2]. And, annotation between two words or phrases are also done for syntactic dependencies or identifying relation between two words in a sentence. For new annotation project or for doing annotation from scratch, typically includes a variety of activities including defining annotation schemas [9], developing guideline for annotations and defining entity type assembling appropriate collections of documents, and properly pre-processing those documents and create the final corpus [10].

One of the important tasks while annotation is selecting appropriate annotation tool given the large number of tools available and the lack of an up-to-date list of annotation tools and their respective pros and cons [11]. Therefore, extensive review of available tools must be done to avoid poor decision of selecting tools. Weak decision can lead to the unnecessary wastage of time of installing and converting document to the specific format for tools.

In this task, an extensive review of annotation tools for manual annotation of documents has been presented. The basic requirements for selecting the annotation tools have been defined. To gain a better understanding of name entity recognition, an ER-system for the legal domain has been created. The first step in creating a corpus of annotated judgment papers is to define relevant entities, which can mainly

be categorized into two: domain-specific named entities like legal terms, Act, legal institutes, etc., and general named entities like person, location, date, etc. An ER-system built with a spacy pre-trained model is then presented.

Following contributions were made in this paper:

- An extensive review was conducted on manual annotation tools for creating NER training corpus.
- A corpus for legal name entity recognition was created, consisting of 5000 judgment papers with 8 legal named entities.
- An Entity-Relation model was developed based on spacy pretrained model.

II. Literature Review

NER is regarded as a crucial activity in the information extraction process. While numerous studies on name entity recognition have been conducted. Several datasets have been offered over a long period of time. CoNLL' 03 [12], which was taken from a German newspaper and is regarded as a language independent NER dataset, is one of the more well- known datasets. Many datasets from many fields, including medical, law, archaeology, and many more, are afterwards proposed [13].

Early NER systems relied on rules that were created by humans. A rule-based NER system is thought to take a long time to design. Researchers have created a NER System based on a machine learning algorithm to solve this issue. They used a few learning techniques, including supervised learning, semi-supervised learning, and unsupervised training. Alex Brandsen et al. [9] have used machine learning approach for predicting name entities from Dutch Excavation reports. Not only machine learning algorithms, but also satisfactory research on NER systems using deep learning algorithms and neural networks, are being conducted. A study conducted by Franck Dernoncourt et al. [14] successfully performs NER using ANN and obtain satisfactory result out of it. Thomas AF Green et al. [15] have included a benchmark CRF-based Entity recognition model of a manually created corpus of job description and achieve accuracy of approx. 60-65 %.

While many type of research on NER is carried out using deep learning and machine learning approach. Very Few studies have been done using pre-trained model like BERT [16]. Mugisha et al. [2] have published a detailed comparison of the neuro-linguistic modelling pipeline for predicting outcomes from medical text notes using patients with pneumonia. Li, Jianfu, et al. [17] in their study, they have fine-tuned pre- trained contextual language models to support the NER task on clinical trial eligibility criteria. They have systematically explored four pre-trained contextual embedding models for biomedical domain (i.e., BioBERT, BlueBERT, PubMedBERT, and SciBERT).

Table I summarizes the literature survey conducted during this study.

TABLE I. Summary of Literature Review

Year	Paper No.	Domain of study	Comments
2019	[6]	Symptom extraction with unstructured clinical notes	To create clinically useful information extraction tools, a task definition, dataset and simple supervised NLP model were used.
2022	[2]	Outcome Prediction from medical notes	A deep comparison of natural language modelling pipelines from outcome prediction from unstructured medical text notes.
2020	[9]	NER in Archaeological domain	Developed a training dataset for name entity recognition in archaeology domain, for which Doccano tool was used for annotation.
2022	[15]	Entity recognition in job descriptions	Created a benchmark suite for entity recognition in job description which includes annotation schema, baseline model, and training of corpus.
2020	[18]	Dataset for legal name entity recognition	A dataset created for NER in German federal court decisions
2022	[19]	NER on Indian judgment paper	Created a training corpus of Indian court judgment and developed a transformer-based legal NER baseline model.

III. Research Methodology

A systematic review of literature on various annotation tools used for NER was conducted. In this paper, the standard SLR procedures as described by the authors Chitu Okoli and Kira Schabram [20] were taken under consideration. This methodology demonstrates a detailed examination of the NER system across multiple domains, as well as a manual annotation tool and various annotation methods.

A. Research Question

One of the crucial steps in a systematic review is the research question. In order to maintain focus at the start of the study, we write research questions (RQ) that will adhere to the review procedure. Table II show the lists of research questions.

B. Search Method

1) Choose keyword: Table III shows a list of keywords used in the search for the paper from the online library.

2) Inclusion and exclusion criteria: Only conference and journal papers published in English language within the last five years were considered, and any papers currently under review were excluded. Table IV show the inclusion and exclusion criteria.

IV. Review of Text Annotation Tools

The formal description of the text annotation problem and annotation tools was presented, followed by a detailed discussion of the selection criteria for annotation tools. Next, an introduction was provided on the commonly used annotation tools.

A. Annotation Tools

Data annotation tools are software used to create high-quality annotated machine learning training data such as text, images, and videos. There are wide variety of annotation tools from open-source tools that developers can modify accordingly to freeware applications that are free to use. Let us first discuss what is text annotation, what is the need of it and discuss some type of text annotation.

B. Text Annotation: Needs and Type

1) Labelling procedures: Adding labels entails putting a word in a sentence that explains its type. It can be explained using emotions, technical terms, etc. For instance, the phrase "I am satisfied with this product, it is amazing" could be given a label like "happy".

2) Adding matadata: Similar to this, relevant information can be added to the statement "Mahadevapura police have fled charge sheet against the accused alleging that he has committed an offence punishable under Section 354C of I.P.C R/w sec.66(E) of Information Technology Act" to help the learning algorithm priorities and concentrate on particular terms. One might write something like, "Mahadevapura (Location) police have fled charge sheet against the accused alleging that he has committed an offence punishable under Section 354C (Legal) of I.P.C. (Act) R/w sec.66(E) of Information Technology Act (Act)".

3) Now let us discuss in brief some of the types of data annotation: Sentiment Annotation: Sentiment annotation is nothing more than the assignment of labels to feelings like sadness, happiness, anger, positivity, negativity, and neutrality. Any activity involving sentiment analysis can benefit from sentiment annotation. (For example, in retail, facial expressions can be used to assess customer satisfaction.)

a) Intent annotation: The intent annotation also identifies the sentences but emphasises the purpose or motivation behind the statement. A message such as "I need to talk to Sam" in a customer service situation, for example, may direct the call to Sam by himself, or a message such as "I have a problem with the credit card" could direct the call to the team handling credit card issues.

b) Named entity recognition: The goal of named entity recognition (NER) is to find and categorise special expressions or predefined named entities in a sentence [21]. It is used to look up words based on what they mean, such as names of people or places. Information can be extracted using NER, together with information classification and categorization.

c) Semantic annotation: It can be also known as meaningful annotation, semantic annotation is the addition of metadata, supplementary data, or tags to text that contains concepts and entities, such as persons, places, or themes.

C. Selection of Tools

Tools that are known and have been mentioned in previous studies were listed. Google, Google Scholar, Scopus, and other online databases were searched for tools mentioned in annotation tool-related publications. There are a wide range of annotation tools available, but for this survey, the tools selected are the most widely used in any domain and meet the criteria.

There are a few requirements that have been studied and presented for an annotation tool. In this research, total 22 criteria are considered to evaluate annotating tools which are further divided into different groups such as input-output, publication, system criteria, and function. For these categories important features of tools are considered such as accessibility, usability, and cost. All these mention categories are listed in Table V along with the associated criteria.

The input-output or data criteria address the input-output format of document, schema for annotation, and input format for multi-media file. Publication criteria include the year of last publications, number of citations, and number of publications in last five years. System criteria indicate installations architecture and simplicity of installations, quality and quantity of documents, license of tools and OS support. And the last set of criteria is functional criteria which contains multimedia annotation support, support of multiple language other than English, automatic text annotation, pre-annotation support and data security.

TABLE II. LIST OF RESEARCH QUESTION

Sr. No.	Research Question
1.	Domains where Name entity recognition is used?
2.	Dataset related to Name entity recognition?
3.	Which annotation tool is used for creating corpus?
4.	What are challenges and issue faced wile manual annotating dataset using tool?
5.	What are various techniques used for annotating dataset for NER?

TABLE III. LIST OF KEYWORDS

Sr. No.	Keyword	No of Articles
1	Name Entity Recognition Corpus and Annotation tools	3
2	Name Entity Recognition Dataset and Manual Annotation	2
3	Manual Text Annotation and annotation tools	4
4	Name entity recognition and (Deep learning or Machine learning)	32
5	legal name entity recognition or medical name entity recognition	14
6	Text annotation tool or lighttag or Doccano or brat or label Studio	45
	Total	100

TABLE IV. LIST OF INCLUSION AND EXCLUSION CRITERIA

Inclusion criteria	Last five-year publication: 2017-2021
	All Open Access
	Only Conference Paper and Journal Paper
	Only considered the ER system for text dataset
Exclusion Criteria	Unpublished paper
	Literature other than English language

The features with which the tools must comply are listed below:

- It should be freely available.
- It should be a web application that can be downloaded or used online.
- It should be able to installed easily.
- It should be approachable.
- It should support multiple file format and export annotation in multiple formats.

To satisfy the availability criteria a tool must be instantly accessible, either for direct online usage (via a web user interface) or to download at the time of writing, without requiring consumers to get in touch with the developers. The availability also depends on whether the tool is free or licensed.

The tool must be a web application, which means that it must either be easily accessible online or may be downloaded and installed as a web application. The requirement that annotations be web-based ensures that annotators can focus completely on their annotation tasks without having to fight with tool installation. Manual annotation is a labor-intensive and difficult task in and of itself, and additional work may annoy the annotators and jeopardise the annotation process.

The survey requires the tool to function properly, and it is a requirement for practical experiments. A minimal set of features, as described by the criteria (as defined in Table V), should be accessible regardless of whether the tool is locally installable or accessible online for use. Therefore, there is no need to contact the developers for help because the tool should be simple to use or the documentation should be thorough enough.

Few more additional features are considered in this research other than the functionalities listed above which makes annotation process much easier such as the smallest unit of annotation (character or token), built-in domain-specific named entity extraction, and quick annotations such as keyboard shortcuts, pre-annotations, or ontology.

Some additional feature which are not compulsory for the annotation tools are listed below but they might be useful for most of the NLP based task:

- It can support multimedia.
- It can support multiple language.
- Integration with AI model for automatic annotation tools.
- Good and simple User Interface.

D. Selected Tools

In this section total eight tools are studied and selected for the research work are listed in Table VI, detailed discussion is done for the selected tools with respect to their features.

1) BRAT (Browser based rapid annotation tool): One of the most well-liked tools for manually annotating documents, it has been employed in the creation of numerous corpora. BRAT is a browser-based free online annotation tool for collaborative text annotation [22]. BRAT is not accessible online and must be installed locally. Documents are imported in the same format as the plain text file that contains the schema configuration. It was designed for rich structured annotation for a range of NLP activities. BRAT was created to enhance manual curation efforts and boost annotator productivity using NLP approaches. It is possible to highlight entities and relations as well as normalize data to pre-established terminology. It has a rich range of features such as integration with external resources such as Wikipedia, support for automatic text annotation tools, and built-in annotation comparison.

BRAT is more suitable for annotating expressions and the relationship between them, because annotating longer texts like paragraphs is really inconvenient. It only accepts text files as input documents, and text files are not presented in their original format in the user interface. Despite the fact that the last version was issued in 2012, the product is still readily accessible and well-liked in the industry. Recent upgrades include, among other things, integrating with external TM tools and embedding visualizations in HTML pages.

TABLE V. CRITERIA FOR SELECTION OF ANNOTATION TOOLS

Criteria Categories	Criteria
Input & Output (data)	Input format for document
	Input for multi-format file
	Format for annotation
	Output format for annotation
Publication	Number of citations
System Criteria	Installation Design (Web, standalone, plugin)
	Simplicity of installation
	Quality and Quantity of documentation
	Licence of Tool
	Operating System Support
	Availability (Free/ Paid)
Function	Multimedia or Multimodal Support
	Multilingual Support
	Interactive UI
	Support of Fast Annotation
	Full Text Support
	Inter-annotator agreement
	Pre-annotation Support
	Integration with external sources
	Automatic text annotation
	Annotation Relationship
	Data Security

TABLE VI. LIST OF SELECTED TOOLS

Tools	Installation	Input Format	Output Format	License
BRAT	Web	TXT	brat standoff format	CC BY3.0
Djangology	Web	DB	DB	-
Doccano	Web	TXT	JSON, CSV	MIT
GATE teamware	Web/ SA	TXT	XML, DB	GPL
Label Studio	Web/SA	TXT, JSON, CSV	CSV, JSON, CONLL	-
Lighttag	Web	TXT	JSON	-
Prodigy	SA	TXT	JSON, txt	-
UBIAI	SA	TXT, JOSN, PDF, HTML	JSON, Amazon Comprehend, Stanford CoreNLP	-

a) Doccano: Doccano is an open-source web-based annotation tool for text files only [23]. It is an open-source tool that supports a variety of job types, such as tasks involving the annotation of text sequences or text classification, which may be applied to a variety of problems, such as the annotation of text for sentiment analysis, text summarization, NER, etc. [9] It has a more modern and attractive user interface, and all configuration is done in the web user inter- face. It also generates a basic overview of tagging statistics. All of these make Doccano more beginner-friendly and user- friendly in general. It supports multiple users, but there are no additional features for collaborative annotation.

b) GATE: Gate team-ware is a web-based open-source collaborative annotation and curation tool [24] and is freely available. Gate teamware is an extension of an annotation tool GATE, which is an annotation management tool. GATE teamware offers user automatic annotation which reduces the manual annotation tool. It offers the interface which can be used to create corpus, to define annotation schema, to load pre-annotated data. As it is collaborative tool, it allows the users to monitor the annotation process i.e., number of annotated document and remaining document to be annotated. It is also use to monitor statistics like time spent on a document, inter annotator agreement.

c) Light tag: Another browser-based text labelling tool is LightTag [25], however it's not completely free. No local installation is required for annotation using lighttag. It offers a free edition with 5,000 annotations each month for its essential features. It supports working with different languages (like Arabic, Hebrew and CJK among others), document level, multi-word, nesting, relationship annotations, etc. Addition-ally, it uses machine learning to learn from active annotators and suggest possible annotations for hidden text. It assigns tasks to annotators and ensures that there is enough overlap and duplication to maintain a high degree of accuracy and consistency.

d) Prodigy: It is a paid tool, and the only free version is a demo. Prodigy is an active learning-based annotation tool that is also connected with the Spacy library. This annotation tool's active learning feature allows you to only annotate cases for which the model does not yet have an answer, greatly accelerating the annotation process. By using transfer learning technology and a more flexible approach to data gathering, you can train models of production quality with a minimal number of samples. Prodigy allows you to annotate images, videos, and audio in addition to text. When exporting your files, you can select among the JSONL, JSON, and txt formats.

e) UBIAI: UBIAI is a powerful labelling platform for training and deploying custom NLP models. UBIAI is a tool for data labelling as a service category in the technology stack [26] . It offers free and paid plans, OCR annotation tools, document classification, auto-tagging for team collaboration, and more. Widely used in the corporate world to convey important information, this is a must, especially for businesses and organizations that need to create high-quality annotations to PDFs, but difficult to edit there is. With UBIAI you can easily annotate native his PDF documents, scanned images, images, invoices or contracts in over 20 languages including Japanese, Spanish, Arabic, Russian and Hebrew can be attached. Per- form named entity recognition (NER), relationship extraction, and document classification in the same interface. Export annotations in multiple formats including Spacy, IOB, and Amazon Comprehend. Supports various input formats such as native PDF, TXT, CSV, PNG, JPG, HTML, DOCX, JSON. It also offers team management features that allow you to track progress. Measurement of text annotations, performance of assigned projects, and agreement among annotators.

f) Label studio: Label Studio is an open-source data labeller that allows you to label and explore a variety of data written in Python. You can make different entries with several data formats. You can also integrate Label Studio with machine learning models to provide label predictions (examples) or perform continuous active learning. Label Studio is also available in Enterprise and Cloud versions with additional features. Simplicity of label studio is that it has no complicated configurations, and ease of integration into Machine Learning pipelines. Label Studio can be used in different places, depending on different use-cases. It is quickly configurable for many data types. The tool gets ready in a few minutes. There is an easy way to switch between labelling texts, audios or images, or even annotating all three types at the same time. Many existing labelling frameworks accept only one data type, and it becomes tedious to learn a new app each time whereas Label Studio works with Texts, Images, Audios, HTML documents and any imaginable combination of annotation

tasks like classification, regression, tagging, spanning, pairwise comparison, object detection, segmentation and so on. After configuring what the labelling interfaces should look like, you can import your data. The web import supports multiple formats: JSON, CSV, TSV, and archives consisting of those.

V. Experiment and Results

A. Annotation of Dataset

The following section describes the premise for dataset annotation, including the defining of annotation setups, various entity types, and annotation method (Annotation guidelines).

1) Selecting suitable input documents for annotation: In order to construct a robust dataset for legal named entity recognition, a comprehensive effort was undertaken to collect a diverse range of case documents from the Indian Supreme Court and several High Courts throughout India. The documents were sourced from a multitude of publicly available repositories on the web, including the official websites of these courts and prominent legal databases such as https://www.indianka noon.org, as well as numerous other legal repositories. The dataset was made sure to represent a wide range of court cases accurately and thoroughly from various jurisdictions through a long and complex process of data collection.

2) Annotation setup: Open-source data labeller Label Studio was used as an annotation tool. After comparing the system to other tools (as previously indicated), it was discovered that this was the most straightforward, user-friendly, and effective tool for our experimentation. There are several methods for installing label studio, including installing with pip, installing with docker, and installing from source, whether you are installing it locally or in the cloud. The only need for label- studio is that Python 3.6 or later must be installed on a machine running Linux, Windows, or MacOSX. Port 8080 is expected to be open by default in Label Studio. Label Studio installation needs SQLite 3.35 or later and PostgreSQL version11.5 or above. After installing Label Studio using pip, data was uploaded and entity types were defined in the tool after the system was downloaded and launched on a local machine.

3) Entity type: The targeted entities are listed in the Table VII, along with a brief description and an example for each category. After talking with legal experts on the pertinent information that may be gleaned from court rulings, the entity kinds were established.

Fig. 1 explain sample example of document to be annotated. The highlighted part of the text indicates the name entity to be annotated. The name entities that can be extracted from above text are given in Table VIII.

TABLE VII. List of Entities of the Legal Judgments

Name Entity	Descriptions	Example
PERSON	Name of the person	Praveen Kumar Wadi, Guruanna Vedi, B.L. Gupta
LOC	Locations which include name of states, cities, villages	Pune, Haryana, Gujarat, Mumbai
DATE	Any Date mentioned in judgment	10 April, 2001
ORG	Name of organization mentioned in text apart from the court.	General Insurance Co. Ltd.
Court	Name of the court which has delivered the judgment.	Supreme Court, Andhra Pradesh High Court, Bombay High Court
LEGAL	sections, Sub-sections, articles orders etc.	Section 110-A, Section 95(2)(d)
ACT	It includes Act name in constitution	Motor Vehicles Act, IT Act, Official Secret Act, IPC
CASE_NO	It indicates the particular case no. of court judgments	C.C. No. 3286 / 2019

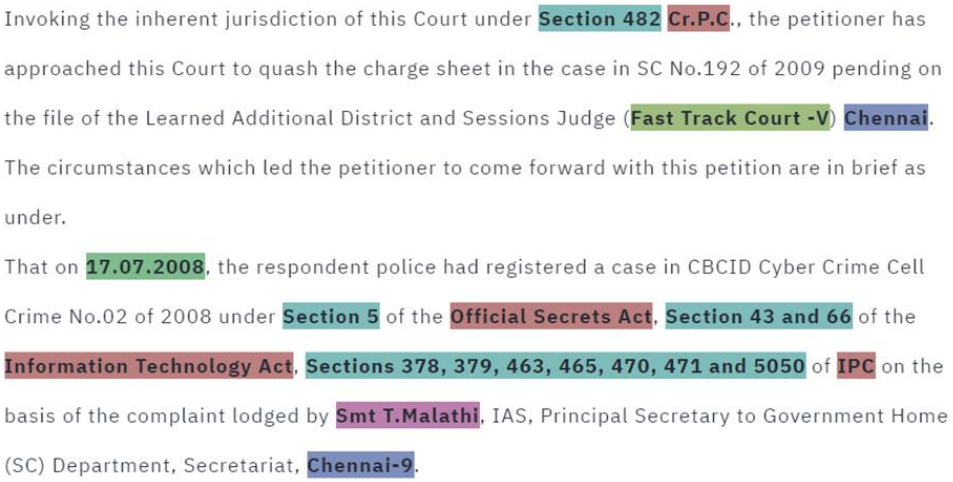

Fig. 1. Example of annotated document.

TABLE VIII. NAMED ENTITY EXTRACTED FROM SAMPLE PARAGRAPH

Name Entity	Text
ACT	CrPC, Information Technology Act, Official Secret Act, IPC
LEGAL	Section 482, Section 5, Section 43 and 66, Sections 378, 379, 463, 465, 470, 471 and 5050.
PERSON	Smt. T. Malathi
LOC	Chennai
DATE	17.07.2008
COURT	Fast track court

4) Manual annotation process: The annotation for the judgment text was done at the sentence level, therefore each judgment sentence was given separately from the annota- tion without document-level context. In the event that extra background information is required for annotation, the whole judgment text is also available. The indiankannon URL was used to obtain the whole judgment text.

To Label and annotate data we have use the open-source data labelling tool, i.e., Label Studio. After importing your data, you can start labelling and annotating your data. Fig. 2 conceptualised name entity recognition using machine learning algorithm and manual annotation.

a) Open a project in Label Studio and optionally.

b) Click Label All Tasks to start labelling.

c) Use keyboard shortcuts or your mouse to label the data and submit your annotations.

d) Follow the project instructions for labelling and deciding whether to skip tasks.

e) Click the project name to return to the data manager.

5) Annotated corpus statistics: In this paper, a dataset of annotated judgment text with seven entities has been created. A dataset of almost 5000 Indian judicial judgment sentences with seven entities has been created. The Table IX lists the number of documents, sentences, and tokens in the annotated corpus as well as other general statistics.

TABLE IX. ANNOTATED CORPUS STATISTICS

No. of Documents	30
No. of Sentences	~5196
Average no. of sentences per document	173
No. of tokens (without stop words)	63155
Annotated tokens	~5286

B. NER Model

Several well-known NER model architectures were explored to identify legal named entities in judgment papers. Initially, spacy's pre-trained NER model was used to implement Legal NER. Two of spacy's pre-trained pipelines, namely en_core_web_trf and en_core_sci_sm, were integrated with unique rules created specifically for the legal domain to improve the accuracy of predictions.

During the training phase, the model's predictions were iteratively compared to the reference annotations to calculate the gradient of the loss as shown in Fig. 3. Backpropagation was then used to determine the gradient of the weights using the gradient of the loss. This approach enabled us to determine how to adjust the weight values so that the model's predictions gradually resembled the reference labels, hence enhancing the model's accuracy.

To make sure that our Legal NER model was optimized for the needs of legal named entity identification, we used a strict and systematic methodology. Our algorithm is capable of accurately identifying many different types of legal entities, such as court names and legal terms.

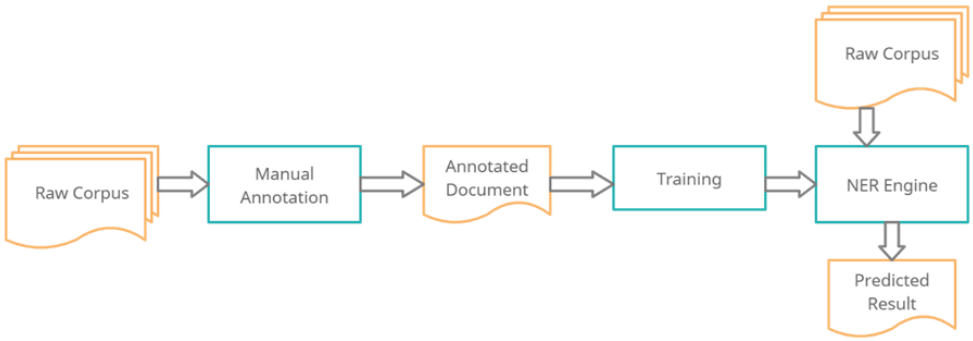

Fig. 2. Manual annotated data and NER system.

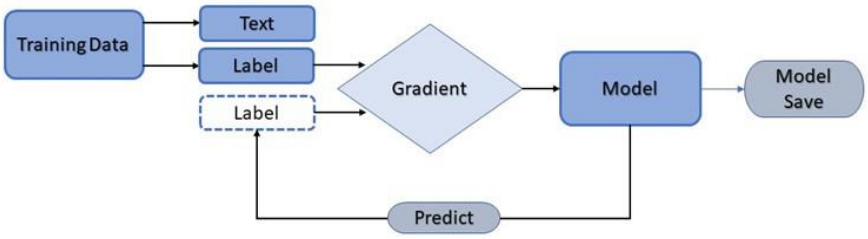

Fig. 3. Spacy pretrained pipeline.

C. Results

Various evaluation matrices were used, such as the F1 score, recall, and precision, to evaluate the model's efficiency. These metrics provide important information about how well the model can identify and categorize data points. The F1 score represents the harmonic mean of accuracy and recall, where recall represents the proportion of true positive values that the model correctly identified and precision represents the percentage of true positive values that the model correctly recognized. A variety of measurements can be utilized to better understand the model's advantages and disadvantages, which will help in deciding how to enhance its performance.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$F1\ Score = \frac{2*Precision*Recall}{Precision+Recall} \quad (3)$$

where,

TP = True Positive FP = False Positive

TN = True Negative FN = False Negative

NER model's aggregate F1 score of 0.62 indicates that the quality of our training data is higher than average. Precision, Recall, and F1scores on judgment sentences are used to assess the model. Table X displays the results of various tests and experiments.

TABLE X. RESULT OF SPACY TRAINED PIPELINE

Spacy Trained Pipeline	Precision	Recall	F1 Score
en_core_web_trf	0.6	0.41	0.48
en_core_sci_sm	0.51	0.4	0.45

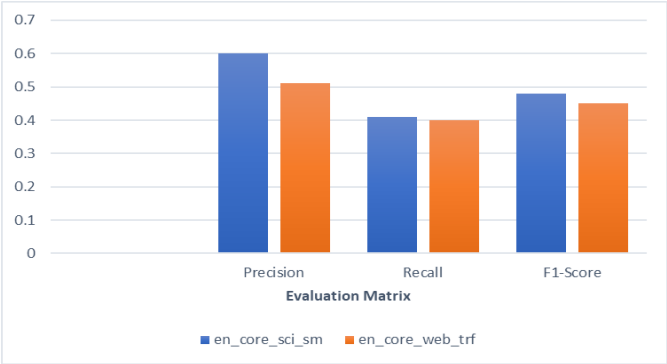

Fig. 4. Analysis of results of trained spacy models.

The Fig. 4 shows the comparison of the performance of the model on individual entities. Since the dataset is completely unbalanced, precision, recall, and F1 score have been calculated for comparison. Precision and recall are defined in terms of true positive, false positive and false negatives, whereas the F1 score is defined as the harmonic mean of precision and recall. The weighted average of precision, recall, and F1 score for spacy en_core_web_trf pipeline are 0.60, 0.41, and 0.48 respectively, and for spacy en_core_sci_sm are 0.51, 0.40, and 0.45 respectively. Good results have been obtained from the experiments and evaluations, and the Legal NER model can be a valuable tool for a variety of legal applications such as legal information retrieval, document summarization, and more.

VI. DISCUSSION

In the legal domain, NER is typically used for tasks such as document classification, contract analysis, and case law research [27]. The accuracy of NER is crucial in the legal domain, as incorrect recognition of entities can lead to incorrect legal decisions [28].

There are several challenges in NER for the legal domain compared to other domains [29]. Firstly, the language used in legal documents is often complex and technical, which might be difficult to identify with traditional NER models. Secondly, legal named entities can have multiple forms and variations, such as acronyms, abbreviations, and synonyms, requiring NER systems to have a comprehensive understanding of legal terminology. To solve this issue, NER models in the legal domain are frequently fine-tuned using massive annotated legal corpora, which can increase the accuracy of legal entity recognition [30].

Another challenge in NER in the legal domain is the presence of named entities with several mentions, such as the names of legal parties. These entities may be referred to by multiple names or titles in different places of the document, making proper identification difficult. To overcome this problem, NER models in the legal sector typically include named entity disambiguation approaches, which assist in the identification and resolution of ambiguity in named entities.

Despite these challenges, NER has proven to be a valuable tool in the legal domain. By automating the process of identifying named entities [31], NER can significantly reduce the time and effort required for legal research and analysis. This can result in increased efficiency and productivity for legal professionals, as well as improved accuracy and consistency in the analysis of legal data. Overall, NER in the legal domain is a critical tool for facilitating legal research, analysis, and decision-making. With advances in machine learning and NLP techniques [32], NER models in the legal domain are becoming more accurate and efficient, helping to make the legal process faster and more effective.

Name Entity Recognition has great potential to improve the process of legal research and analysis, but it faces significant challenges in the legal domain due to the complexity and technical nature of legal language [33]. Further development and refinement of NER systems for the legal domain will likely result in even greater benefits for legal professionals in the future. Once these entities have been extracted and tagged, they can be used for research and analysis of legal texts. Furthermore, policy-making can be informed by the knowledge gained by Legal NER. Overall, the use of LNER in legal research and text analysis can enhance legal research, inform policy decisions, and result in more efficient and fair legal systems.

VII. Conclusion and Future Work

In this paper, a corpus of Indian judgment papers is presented that is annotated with 7 distinct types of entities and can be used to identify legal named entities. In order to create the annotated dataset, a variety of annotation tools were reviewed. 30 court documents that are available publicly were manually annotated. With the dataset, a spacy model was also trained utilizing the trained NER pipelines en_core_sci_sm and en_core_web_trf. The model displays an F1-score of almost 60%, indicating that the dataset has better quality. It is believed that the dataset will be useful for additional NLP tasks on Indian judicial material, such as relationship extraction, knowledge graph modelling, extractive summarization, etc.

In terms of future work, the author will explore approaches for extending and further optimizing the dataset. They will also perform additional experiments with more recent state-of-the-art approaches. The researchers plan to produce a CSV version of the dataset, which will simplify the data format, enhance compatibility, facilitate data pre-processing, and enable data analysis.

References

[1] J. Marrero, S. Urbano, J. S. nchez Cuadrado, J. M. Morato, and G. mez Berb´ ıs, "Named entity recognition: fallacies, challenges and opportunities," *Computer Standards & Interfaces*, vol. 35, no. 5, pp. 482–489, 2013.

[2] I. Mugisha and Paik, "Comparison of Neural Language Modeling Pipelines for Outcome Prediction from Unstructured Medical Text Notes," *IEEE Access*, vol. 10, pp. 16–489, 2022.

[3] Han, Xu, Chee Keong Kwoh, and Jung-jae Kim. "Clustering based active learning for biomedical named entity recognition." In 2016 International joint conference on neural networks (IJCNN), pp. 1253-1260. IEEE, 2016.

[4] U. Neves and Leser, "A survey on annotation tools for the biomedical literature," Briefings in bioinformatics, vol. 15, no. 2, pp. 327–340, 2014.

[5] Neudecker, "An open corpus for named entity recognition in historic newspapers," Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pp. 4348–4352, 2016.

[6] J. M. Steinkamp, W. Bala, A. Sharma, and J. J. Kantrowitz, "Task definition, annotated dataset, and supervised natural language processing models for symptom extraction from unstructured clinical notes," Journal of biomedical informatics, vol. 102, pp. 103–354, 2020.

[7] J. Rodriguez, A. Diego, A. Caldwell, and Liu, "Transfer learning for entity recognition of novel classes," Proceedings of the 27th International Conference on Computational Linguistics, pp. 1974–1985, 2018.

[8] K. Bontcheva, H. Cunningham, I. Roberts, and V. Tablan, "Web-based collaborative corpus annotation: Requirements and a framework implementation New Challenges for NLP Frameworks," pp. 20–27, 2010.

[9] A. Brandsen, S. Verberne, K. Lambers, M. Wansleeben, N. Calzolari, F. B. chet, and P. Blache, "Creating a dataset for named entity recognition in the archaeology domain," The European Language Resources Association, pp. 4573–4577, 2020.

[10] Cristian Cardellino, Milagro Teruel, Laura Alonso Alemany, Serena Villata. A Low-cost, High-coverage Legal Named Entity Recognizer, Classifier and Linker. ICAIL-2017 - 16th International Conference on Artificial Intelligence and Law, Jun 2017, Londres, United Kingdom. pp.22. ffhal-01541446.

[11] S. Tripathi, H. Prakash, and Rai, "SimNER-an accurate and faster algorithm for named entity recognition," Second International Conference on Advances in Computing, Control and Communication Technology (IAC3T), pp. 115–119, 2018.

[12] E. F. Tjong, K. Sang, and F. D. Meulder, "Introduction to the CoNLL-2003 Shared Task: Language Independent Named Entity Recognition," Proceedings of the Seventh Conference on Natural Language Learning, 2003.

[13] B. Glaser, F. Waltl, and Matthes, "Named entity recognition, extraction, and linking in German legal contracts," IRIS: Internationals Rechtsinformatik Symposium, pp. 325–334, 2018.

[14] F. Dernoncourt, J. Y. Lee, and P. Szolovits, "NeuroNER: an easy-to-use program for named-entity recognition based on neural networks," 2017.

[15] T. Green, D. Maynard, and C. Lin, "Development of a benchmark corpus to support entity recognition in job descriptions," Proceedings of the Thirteenth Language Resources and Evaluation Conference, pp. 1201–1208, 2022.

[16] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, vol. 1, pp. 4171–4186, 2019.

[17] J. Li, Q. Wei, O. Ghiasvand, M. Chen, V. Lobanov, C. Weng, and H. Xu, "Study of Pre-trained Language Models for Named Entity Recognition in Clinical Trial."

[18] E. Leitner, G. Rehm, and J. Moreno-Schneider, "A dataset of German legal documents for named entity recognition," 2020.

[19] P. Kalamkar, A. Agarwal, A. Tiwari, S. Gupta, S. Karn, and V. Raghavan, "Named Entity Recognition in Indian court judgments," 2022.

[20] K. Okoli and Schabram, "A guide to conducting a systematic literature review of information systems research," 2010.

[21] S. Yadav and Bethard, "A survey on recent advances in named entity recognition from deep learning models," 2019.

[22] P. Stenetorp, S. Pyysalo, G. Topic´, T. Ohta, S. Ananiadou, and J. Tsujii, "BRAT: a web-based tool for NLP- assisted text annotation," Proceedings of the Demonstra- tions at the 13th Conference of the European Chapter, pp. 102–107, 2012.

[23] V. Sarnovsky´, N. M.-K. kova´, and Hrabovska´, "Annotated dataset for the fake news classification in Slovak language," 2020 18th International Conference on Emerging eLearning Technologies and Applications (IC- ETA), pp. 574–579, 2020.

[24] K. Bontcheva, H. Cunningham, I. Roberts, A. Roberts, V. Tablan, N. Aswani, and G. Gorrell, "GATE Teamware: a web-based, collaborative text annotation framework," Language Resources and Evaluation, vol. 47, no. 4, pp. 1007–1029, 2013.

[25] T. Perry, "Lighttag: Text annotation platform," 2021.

[26] J. B. Gillette, S. Khushal, Z. Shah, S. Tariq, Algamdi, Krstev, M. Ivan, B. Mishkovski, S. Mirchev, and G. Golubova, "Extracting Entities and Relations in Analyst Stock Ratings News," 2022 IEEE International Conference on Big Data (Big Data), pp. 3315–3323, 2022.

[27] A. Barriere and Fouret, ""May I Check Again? -A simple but efficient way to generate and use contextual dictionaries for Named Entity Recognition. Application to French Legal Texts," 2019.

[28] S. Paul, P. Goyal, and S. Ghosh, "LeSICiN: A heterogeneous graph-based approach for automatic legal statute identification from Indian legal documents," Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 11–139, 2022.

[29] I. Angelidis, M. Chalkidis, and Koubarakis, "Named Entity Recognition, Linking and Generation for Greek Legislation," JURIX, pp. 1–10, 2018.

[30] S. Paul, A. Mandal, P. Goyal, and S. Ghosh, "Pre-training transformers on Indian legal text," 2022.

[31] Chiu, Jason PC, and Eric Nichols., "Named entity recognition with bidirectional LSTM-CNNs," Transactions of the association for computational linguistics, vol. 4, pp. 357–370, 2016.

[32] S. Yadav and Bethard, "A survey on recent advances in named entity recognition from deep learning models," 2019.

[33] Lison, Pierre, Aliaksandr Hubin, Jeremy Barnes, and Samia Touileb. "Named entity recognition without labelled data: A weak supervision approach." arXiv preprint arXiv:2004.14723 (2020).