Unnatural Language Processors at SemEval-2022 Task 6B: Legal Named Entity Recognition (NER)

Maria Valentini
University of Colorado
Boulder

Junwoo Jang
University of Colorado
Boulder

Khaled Hossain
University of Colorado
Boulder

[Maria.Valentini, Junwoo.Jang, Khaled.Hossain]@colorado.edu

Abstract

Named Entity Recognition (NER) is a common task undertaken in the field of Natural Language Processing. Named Entities (NEs) in a body of text are usually organizations, people, or other such entities that have proper names. The procedure of identifying and categorizing Named Entities from a text is called Named Entity Recognition. This paper discusses the development of a Legal Named Entity Recognition model trained on a dataset of Indian court cases, created for the identification of Named Entities in legal documents similar to those used in the model's training. The Named Entities in the documents generally include attorneys, courts, litigators, plaintiffs, dates, and other key entities involved in law. The results of the experiments described in this paper leave some questions unanswered, but they clearly indicate the ability of machine learning models trained on domain-specific data to perform nearly as well as NER models working with generalized data to identify standard Named Entities.

1 Introduction

In countries with large and growing populations such as India, the number of legal cases that are awaiting conclusion has increased by a great amount. Although it would be difficult to completely automate the judicial pipeline, it could potentially be very beneficial if some tasks could be done with the use of Natural Language Processing (NLP) to help legal practitioners and make the system faster and more efficient. However, legal texts are not the same as many other texts that have been used to train NER models in the past. This makes it difficult to apply existing NLP model architectures and techniques directly, necessitating the development of legal domain-specific models and techniques. In this paper, we examine the effectiveness of a modified spaCy model trained on bodies of legal texts to perform in the NLP task of Named Entity Recognition for Indian legal documents.

The dataset used to train our model contained 14,444 Indian court judgment sentences and 2,126 judgment preambles, annotated with 14 legally named entities. The process of extracting Named Entities from court judgment texts is one that is important to develop, both for its own sake and because it opens up the opportunity for legal practitioners to use it for more tasks in the future like relation extraction, coreference resolution, and knowledge graph creation. The research described in this paper aims to answer two main research questions: (1) Does the addition of a rule-based entity tagger improve the performance of a spaCy pipeline in the task of Named Entity Recognition on Indian legal documents? (2) Can a spaCy model be optimized and trained on a sample of legal documents to perform similarly to the level of standard Named Entity Recognition models? [1]

2 Background

This project is being completed as part of LegalEval 2023, one of the 12 Natural Language Processing tasks that make up the 2023 SemEval Competition. The LegalEval task focuses specifically on NLP's applications in a legal setting, exploring its abilities to tackle three relevant tasks in the area: (A) Rhetorical Role Prediction, (B) Legal Named Entity Recognition, and (C) Court Judgement Prediction with Explanation. In this paper, we explore an approach for performing LegalEval's Task B or Legal Named Entity Recognition. This approach uses a model which implements Python's spaCy framework and combines a trans-

[1] https://github.com/Khaled2049/CSCI-5832

former model with a rule-based tagger. NER has performed as a successful tool when parsing ordinary documents for standard entities, but the different entities involved and the language used in legal documents requires the development of a specifically designed model. The results show that the use of a machine learning model trained on domain-specific data can perform effectively in the field of Legal NER, even in comparison to standard NER applications. They also demonstrate potentially promising future results for our modified spaCy pipeline with a rule-based entity tagger added as a pipe.

2.1 Motivation

Named Entity Recognition (NER) is a problem commonly explored in the field of Natural Language Processing, taking in a body of text and identifying its "named entities," or the objects it contains that have proper names (e.g., cities, people, etc.). The process of NER is usually comprised of two stages - differentiating the proper nouns from a body of text and putting them into their respective categories. Models that can perform well at NER have a wide variety of applications in different fields such as medicine, education, and law, but these models must be developed and fine-tuned specifically for each field in order to perform most effectively. Because of this need for a specialty within NLP models in order to perform well in different domains, this work proposes a Named Entity Recognition model developed specifically for the field of Law, making use of features pertaining to legal data to maximize the effectiveness and accuracy of the model. The data for this study was voluntarily collected by Law students from multiple Indian law universities and classified into 14 pre-defined rhetorical roles in Indian court judgments.

2.2 Related Work

The authors of the paper "Named entity recognition without gazetteers," [7] present results on the named entity recognition without the usage of gazetteers- which are lists of names of people, organizations, locations, and other named entities. They created a model that only uses rule-based grammar and statistical models. Their results showed that small gazetteers with well-known names are adequate to create a well enough model for named entity recognition. Looking at the results from this paper we decided to work on our model without using any gazetteers.

In recent years, there has been growing interest in the use of NLP techniques for analyzing legal texts such as court judgments. Previous research on NER in legal texts focuses on various types of named entities including people, organizations, locations, and dates. For example, the paper "Named-Entity Recognition for Legal Documents" [11] presents a model that can correctly recognize entities like "court name," "date of judgment," "petitioner," "respondent," "name of the judge," and "acts" from any legal document and give output to the user. The first step used in their method is data preparation. Once that has been completed, the NER model uses four convolutional neural networks with residual connections for encoding, and all words are converted to feature vectors. Finally, a multi-layer perceptron is used to map the feature vectors to 128 dimensions. It updates the weights and gives the result. The model had a precision score of 72.88% and an F-1 score of 59.31%.

In another paper, titled "Fine-Grained Named Entity Recognition in Legal Documents," [4] author Acosta discusses the development of a system that was able to recognize persons and locations. The dataset used consisted of 66,723 sentences and 2,157,048 tokens, and the results consisted of court decisions with 19 fine-grained classes. They also utilized sklearn-crfsuite and UKPLab-BilSTM to generate the class. After observing the results, they concluded that the BiLSTMs had superior performance in comparison to the CRFs. This BiLSTM model with character embeddings had an F1 score of 95.46% for coarse-grained classes.

The authors of the paper "Named Entity Recognition in Indian court judgments" [3] worked on developing a model to predict entities from judgment sentences and preambles using a transformer-based architecture. They used multiple transformer models to achieve their results and used the Roberta-Base model for fine-tuning. Their results showed that transition-based parsers greatly improved the model's accuracy in the task of Named Entity Recognition in legal documents. The authors showed that the transformer with a transition-based parser and the Roberta base transformer model performed best, with a precision of 92%, recall of 90%, and F1 score of 91.1%.

3 Model Overview

The baseline model [3] for the experiment was trained to predict entities from judgment sentences and preambles using a transformer-based architecture and a transition-based dependency parser on top of the transformer model. The architecture for the parser was Roberta-Base model, for which a fine-tuned model was also tested. During training, early stopping was used to select the best hyperparameters for the model, such as a batch size of 256 and a maximum training step of 40000 with an Adam optimizer. Our model uses many similar techniques, but it is composed of a spaCy pipeline which uses a modified version of spaCy's built-in NER tagger trained on legal data and an additional rule-based entity tagger to improve the model's performance.

3.1 SpaCy Model Setup

The basis of the model developed in this research is a spaCy pipeline that is trained on a dataset of Indian legal documents. After the data was prepared and all irregularities removed, the model was created using spaCy v3 and used a transformer architecture, similar to the baseline model. The NER tagger also used a transition-based parser and transformer to parse through all of the training data and allow the model to learn and improve effectively from this training. We removed each of the labels that were present in spaCy's default NER tagger and replaced them with the labels that corresponded with the Indian legal documents used to train it.

Every pipe in the pipeline was disabled aside from the NER tagger, as the other pipes (e.g., Attribute Ruler, Lemmatizer, etc.) were unnecessary for the task being attempted and made the model train more slowly than it would with them disabled. After all other pipes had been disabled, we added an additional pipe to classify some entities using a rule-based method, which was included to aid in the identification and classification of some entities that followed specific rules or patterns, such as dates. Finally, the model was used to add BILUO labeling to the entity labels, with B indicating Beginning, I indicating Inside, L indicating Last, U indicating a single unit, and O indicating other or outside (anything that does not fall into one of the categories of Named Entities).

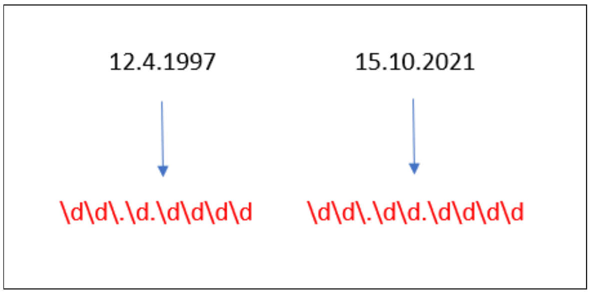

Figure 1: Example of RegEx pattern used for rule-based pipe

3.2 Rule-Based Pipe

As previously mentioned, the rule-based pipe was added to the spaCy pipeline in order to improve the model's ability to ascertain the identity of some entities which could be generalized with patterns or rules. One such entity type was date, for which each standard mention of the entity followed either the regular expression pattern `"\d\d\.\d.\d\d\d\d"` or `"\d\d\.\d\d.\d\d\d\d"`. A visualization of this pattern can be seen in Figure 1. This rule-based tagger was added in attempt to make the model's identification and classification abilities for specific types of entities improve, as it is able to automatically find and classify all data entities that followed a specific pattern.

4 Experiments

Numerous experiments have been performed to test the model described above, all of which were performed in a Jupyter Notebook on Google Colab. The model proposed in this research was trained on a standard corpus made up of Indian legal documents with specified and annotated Named Entities. Several experiments tested the model's ability to perform Named Entity Recognition on legal documents in comparison to other similar models. Also studied in the process of experimentation was what modifications could be made to the model to optimize its performance, evaluating the best model setups and features and using the information to adjust the model accordingly.

After the model had been created and optimized, it could then be tested on a separate dataset which was set aside at the beginning for development. The model's predictions on the development dataset could then be compared with each document's actual Named Entities and evaluated using precision, recall and F1 metrics.

4.1 Metrics

The primary metrics that were used to test the effectiveness of each model iteration on the development dataset were precision, recall, and F1 score. These metrics were used because of their widespread use in the evaluation of machine learning models; they provide a standardized method for understanding the performance of a model and comparing it to other models addressing the same or similar tasks. In addition to the standard set of machine learning metrics used to evaluate models, several other metrics were used in the process of optimizing the model, including the amount of time needed to train the model and the model's loss progression.

5 Results

The results of the experiments performed demonstrated the effectiveness of our modified spaCy model when used for the task of Named Entity Recognition on Indian legal documents. The model developed in this research was not able to produce full results, meaning that at the present our best model is still the baseline model. This leaves our research questions somewhat unanswered, as it is not currently clear whether the addition of a rule-based pipe improves the performance of the built in spaCy NER model on its own. As our current results do not fully address the questions proposed in our research, further development of the models and more result collection will establish the eventual answers to our two questions listed in the introduction. The baseline model alone, however, was still able to produce very promising results using a similar architecture to ours but without the addition of the rule-based classifier.

5.1 Metrics Results

After developing our model and performing the series of experiments described above, we were unable to make our model perform better in the previously established metrics than the performance of the original baseline model. As such, our best metrics as of this point are those which we achieved using the baseline Roberta model. Using the Roberta base and a transition-based parser, this model yielded a precision score of 92.0%, a recall score of 90.2%, and an F1 score of 91.1%. The scores in these metrics that our model is able to attain once full results are available will be made available in a later version of this paper. A complete illustration of these metrics can be found by referencing Table 1.

5.2 Comparison

The authors of the paper "Named entity recognition in the Romanian legal domain" [9] created a named entity model in the Romanian legal domain. Their system used Recurrent neural networks based on BiLSTM cells with the last layer being a CRF layer. The best F1 score that their model was able to achieve was 90.36%. The F1 score for our baseline model was 91.1%, which showed improvement from the model cited previously. The BiLSTM model discussed in the paper "Fine-Grained Named Entity Recognition in Legal Documents" [4] had the best F1 score with a value of 95.46%, outperforming our baseline model by over 4%. In regard to our second research question, which asked if domain-specific NER models such as those used in the legal domain could perform at a level near the performance of standard NER models, a survey of recent NER models showed that domain-specific models are indeed able to perform similarly when trained with domain-specific data. The data chosen for comparison was from the paper "A Simple Semi-supervised Algorithm For Named Entity Recognition," [**empty citation**] which achieved an F1 score of 94.2% and demonstrated a decreased performance in comparison to some of the most recent NER models in the legal domain.

6 Conclusion

Investigated in this research were two primary research questions: (1) Does the addition of a rule-based entity tagger improve the performance of a spaCy pipeline in the task of Named Entity Recognition on Indian legal documents? (2) Can a spaCy model be optimized and trained on a sample of legal documents to perform similarly to the level of standard Named Entity Recognition models? This led to the development of a spaCy pipeline optimized for accuracy with built-in pipes aside from the NER tagger disabled and an added pipe that would classify some entities using a rule-based method. The model was trained on a dataset of Indian legal documents and fine-tuned to maximize its performance. It was then tested on a separate

Metric	Baseline	Romanian	Fine-Grained	Standard NER
Precision	0.920	N/A	0.953	0.954
Recall	0.902	N/A	0.956	0.933
F1 Score	0.911	0.904	0.955	0.942

Table 1: Results of each of the evaluation metrics for our model and several others also attempting NER, both in the legal domain (Romanian, Fine-Grained) and in the standard NER domain (Standard NER).

development dataset to be evaluated for metrics such as precision, recall, and F1 score. The results of this experiment show that we are still unable to fully answer the first question without further investigation, but our second question seems to have a fairly clear answer. With the baseline model demonstrating that a model constructed in a manner similar to ours and trained on a sample of legal documents is able to perform very close to the level of standard Named Entity Recognition models.

References

[1] Asif Ekbal and Sivaji Bandyopadhyay. "A Hidden Markov Model Based Named Entity Recognition System: Bengali and Hindi as Case Studies". In: *Pattern Recognition and Machine Intelligence*. Ed. by Ashish Ghosh, Rajat K. De, and Sankar K. Pal. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 545–552. ISBN: 978-3-540-77046-6.

[2] Anil K Jain and Martin HC Law. "Data clustering: A user's dilemma". In: *International conference on pattern recognition and machine intelligence*. Springer. 2005, pp. 1–10.

[3] Prathamesh Kalamkar et al. "Named Entity Recognition in Indian court judgments". In: *arXiv preprint arXiv:2211.03442* (2022).

[4] Elena Leitner, Georg Rehm, and Julian Moreno-Schneider. "Fine-Grained Named Entity Recognition in Legal Documents". In: *Semantic Systems. The Power of AI and Knowledge Graphs*. Ed. by Maribel Acosta et al. Cham: Springer International Publishing, 2019, pp. 272–287. ISBN: 978-3-030-33220-4.

[5] Wenhui Liao and Sriharsha Veeramachaneni. "A Simple Semi-supervised Algorithm For Named Entity Recognition". In: (2021), pp. 469–479.

[6] Wenhui Liao and Sriharsha Veeramachaneni. "A simple semi-supervised algorithm for named entity recognition". In: *Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing*. 2009, pp. 58–65.

[7] Andrei Mikheev, Marc Moens, and Claire Grover. "Named entity recognition without gazetteers". In: *Ninth Conference of the European Chapter of the Association for Computational Linguistics*. 1999, pp. 1–8.

[8] Jade Moillic and Roxane Bois. *spaCy matchers guidelines*. `https://medium.com/besedo-engineering/spacy-matchers-guidelines-c38207f884d7`. 2022.

[9] Vasile Păiș et al. "Named entity recognition in the Romanian legal domain". In: *Proceedings of the Natural Legal Language Processing Workshop 2021*. 2021, pp. 9–18.

[10] Eleni Partalidou et al. "Design and implementation of an open source Greek POS Tagger and Entity Recognizer using spaCy". In: *IEEE/WIC/ACM International Conference on Web Intelligence*. 2019, pp. 337–341.

[11] Harsh Vardhan, Nitish Surana, and B. K. Tripathy. "Named-Entity Recognition for Legal Documents". In: *Advanced Machine Learning Technologies and Applications*. Ed. by Aboul Ella Hassanien, Roheet Bhatnagar, and Ashraf Darwish. Singapore: Springer Singapore, 2021, pp. 469–479. ISBN: 978-981-15-3383-9.

7 Appendix

The Git repository containing the code that was used in this project can be accessed using the following link: `https://github.com/Khaled2049/CSCI-5832`.