Indian Supreme Court Judgement Dataset for Prediction Models

Dr. N. Sivaranjani

Department of Computer Science and Business System,Sri Sairam Engineering College.
Sivaranjani.csbs@sairam.edu.in

Karthik R

Department of Computer Science and Business System, Sri Sairam Engineering College.
Sec22cb013@sairamtap.edu.in

Jaya Sancith

Department of Computer Science and Business System, Sri Sairam Engineering College.
Sec22cb042@sairamtap.edu.in

Siva P

Department of Computer Science and Business System, Sri Sairam Engineering College.
Sec22cb081@sairamtap.edu.in

Abstract. The development of legal records requires extra time and their absurd length raises the need for programmed legal record handling frameworks. One of the handling steps is to recognize the essence of the reports expressions for legitimate case records. A computerized framework that could help a legal professional in anticipating the result of a case would help facilitate the legal interaction. For such a framework to be essentially valuable, expectations by the framework ought to be reasonable. To advance exploration in growing such a framework, this paper presents the Indian Legal Judgment Dataset (ILJD). ILJD is a huge corpus of 35k Indian Supreme Legal disputes commented on with unique court choices. A part of the corpus (a different test set) is commented on with best quality level clarifications by lawful specialists.

Keywords: Legal judgement Prediction, Machine Learning, Legal Dataset, Supreme Court of India, chi^2.

1. Introduction

Significantly in countries with exceptional populations such as India, there is countless forthcoming overabundance of legitimate cases that hinder the legal cycle [1]. The overabundance is because of various variables, including the inaccessibility of appointed judges. In this way, a framework fit for helping a legal professional by predicting the result of a new ongoing case is probably going to be helpful for facilitating the legal process. However, unless it is adequately described in terms of how people understand the legal cycle, a computerised prediction framework is inappropriate for use in the legal field. Consequently, it's critical to make the concept clear. In other words, we like to forecast both the formal outcome of a legal issue and the process by which that outcome will be reached. In this paper, we created a dataset ILJD that could be used in machine learning algorithms trying to predict judgment of a new case. We have taken the raw dataset from IndianKannon which contain around 30k cases with judgments. ILJD is the collection of case proceedings from the Supreme Court of India containing original case decisions. Any machine learning algorithm aiming to predict the final decision of a legal case given with all the information of the case like facts, law and arguments made in the courtroom, ILJD can be used. Each of the cases in the ILJD is annotated with labels. For example, if any case file contains the details about land dispute and petitioner has won the case against the defendant, then this case will be labelled as Civil, petitioner won. Though the legislative provision is subjected to change overtime, it might mislead the reader that the usefulness of the task is limited. However, the legal canons of how to apply a given law to a given fact of the case stays constant over the period.

Preparing a judgment document for prediction is quite difficult than the other text classification task for the following reason: 1. Legal documents are unstructured, long, contain noisy data and wordy [2]. So, identifying the direct easy way of extracting and using the facts and arguments involves a lot of effort. 2. The models used in domain-based dictionary court cases are pre-trained and commonly available texts are useless in such documents. As a result, standard models are needed to be modified to the legal domain for the proposed ones Judgment prediction in court cases. 3. Third, the interpretation of the prediction in legal documents is significant. Understanding the facts, following the arguments, and applying the rules of law is very challenging as it requires principles to make the final decision. Our contribution towards this paper is

1. Developed a new corpus called Indian Legal Judgement Dataset (ILJD), can be used for prediction of the judgments labelled with category of the legal case. We Perform Comprehensive Case Studies on Corpus Differences in Prediction and Interpretation by Legal Experts, Calculation refers to the challenges in data modelling.
2. We introduced a way of extracting the features from the legal documents that contribute more to the final decision. Using chi2 the important features from the accepted and rejected cases and labeled accordingly.
3. We also compare the labelling accuracy of the model against legal experts, showing the difference of viewpoint given the facts of the case.

The idea/ motive behind IJLD creation is to make labelling the legal dataset process automatic and help research to reduce the initial work. ILJD is just the initial step towards legal prediction, however this area of research requires a lot more research to be done to incorporate the unexpected consequences that occur during model development.

The paper is organized in following ways section II says about some of the similar work carried out in legal judgment prediction, section III explains about creation of ILJD (Indian Legal Judgment Dataset) to be used in prediction model followed by tasks done by the legal experts in section IV. Section V explains the performance of the model and finally the conclusion.

2. Related work

There has been broad exploration on lawful domain text, and different corpora and assignments have been proposed e.g., summarization [3,4], prior case retrieval [5], judgment prediction [6], crime classification [7] and catchphase extraction [8]. The legal case prediction and creation of the dataset are explained in [9,10,11,12]. The description of the case proceeding usually contains interesting aspects like statutes, legal argument happened during the trial and lower court decision [13] there are some similar works done past like [14] which creates the judgment dataset for UK court removing the final decision given by the UK court and used machine learning algorithms for labelling. ILJD also does the similar thing with a large corpus of data with 34,816 case proceedings. As far as Indian law is concerned it has common law, but the decisions are not strictly based on the law applicable to the case. With the judges having the cautiousness to decipher their form of the legal terms as appropriate to the current case; this can at times settle on the decision process biased. ILJD doesn't concentrate on cases like civil, criminal, business etc. addresses the cases that are publicly available on the website. A similar work by [12] creates a Chinese legal dataset for prediction called Chinese AI law challenge dataset considering only 2.68 million criminal cases from the Supreme Court of China. ECHR by [9] created a legal prediction dataset from the European Court of Human Rights consisting of 11,478 cases. It only considers cases from human rights violations. ILJD diverges from the current LJP corpora, where principally the civil cases are thought of. However, the proposed corpus centres around Indian cases, our examination reveals that the language utilized in the cases is very challenging to process computationally and gives a great platform for creating practical legal proceedings understandable framework.

3. Indian Legal Judgment Dataset

In this paper, we present the INDIAN LEGAL JUDGMENT DATASET (ILJD), an assortment of case procedures (in the English language) from the Supreme Court of India (SCI). For a case documented at the SCI, a decision ("accepted" v/s "rejected") is taken between the litigant/candidate versus the respondent by a judge while considering the realities of the case, administering by lower Court(s), if any, contentions, rules, and points of reference. For each case recorded in the Supreme Court of India (SCI), the judge (or then again, a bench) settles on whether the claim(s) recorded by the litigant/applicant against the respondent ought to be "accepted" or "rejected". The choice is comparative with the litigant. In ILJD, every case is labelled with the original judgment given by the SCI judge. Notwithstanding the ground truth choice, a different test set reports are commented on (by legal professionals) with clarifications that influenced the decision.

3.1 Creation of ILJD

a. Data Mining

This section of the paper includes the initial work done before labelling the dataset, the training the model and performing the analysis of the algorithm. This initial work involves scraping the data from the source website and creating the dataset with the data, followed by cleaning the skewed data. (Since this section includes most of process performed in python language, the reader can skip if not interested) *b. Scraping the data*

In python many web scraping libraries are available to directly scrap any data from a website directly, URLLIB library can be used directly to scrape data from a website. The dataset is taken from https://indiankanoon.org/browse/supremecourt/ where URLLIB library meant of no use at all, as the library can scrap data only from the single scripted websites i.e. client scripted websites, while these shows error while we are trying to scrap from the duel scripted website (i.e. dynamic and static websites). Since error 401 always accrued, such that we need to act as a web browser which can help us in interpreting server-side script since it is a dynamic website. To clear the unauthorize error produced by the server we have used selenium library in python along with the chrome web driver and interpreted as a web browser and we successfully accessed the website and the data from the website. In-order to extract all the 76 years data from 1950-2020, the process is divided using a for loop, each loop contained 5 years and each year data is saved in the text format, we were able to achieve this using file handling method. Each loops information is also noted using common print functions. To monitor the count of the cases being extracted.

c. Tools used for Scrapping

- Selenium web browser creating library
- Beautiful soup to extract text from the website
- Chrome web-driver

3.2 Cleaning the Dataset

Since every dataset comes skewed, and this paper focuses on creating dataset to deal with the forecasting of a text-based data, it is encountered that many data values that are completely unusual and can also have a negative impact on the models. To perform the data analysis, all the special characters from the data and the '\n' values from the data, and many more such type of data values are removed using regular expressions (library) and replace (python-inbuilt) function. we have used the pandas indexing method to visualize each case, and personally observe what are the small tweaks can be done in the cases, such that we can get the output of the cases, cleaned from special characters.

Special characters

Replacing '\n' with "spaces"

Replacing suspicious 'L….J….' with "spaces" etc.

Tools used for Data Cleaning

a. Pandas library
b. Regular Expressions
c. Replace (built-in function of python)

We mined all the open access SCI case procedures from the year 1950 to April 2020 from the site: https://indiankanoon.org/browse/supremecourt/.

Case procedures are unstructured in nature and have various formats across different decades and sizes, have spelling mistakes (since these are composed during the court hearing), making pre-process a difficult task. We utilized regular expression to eliminate the noisy text and meta-data (e.g., introductory parts of the report containing case number, judge name, dates, and other meta data) from the legal proceeding. Practically speaking, as pointed by the legal professionals, the judge choosing the case also other meta data impact an ultimate choice. In SCI case legal proceedings, the judgment is written at the end of the proceeding. These end section(s) straightforwardly expressing the judgment have been erased from the proceeding in ILJD since that is what we mean to predict. Each case's judgment mark has been retrieved from the erased last portion of the proceeding utilizing regular expression. Another challenge associated with the legal proceedings is the presence of cases with different petitions where, in a particular case, different petitions have been documented by the litigant prompting various judgment. Thus, we partitioned ILJD archives into two sets. The initially set, called $\text{Judgment}_{\text{SAME}}$, either have reports where there is a individual request (and, along these lines, a single choice) or different petitions, however the judgment are something very similar across all the petitions. Another set, called $\text{judgment}_{\text{DIFFERENT}}$, is a superset of $\text{Judgment}_{\text{SAME}}$ and has numerous requests leading to various judgment. Anticipating various judgment for cases with different requests is altogether challenging. During the labelling and training we assumed the labelled as accepted even if single appeal was accepted in a particular case having multiple petitions.

3.3 Feature Engineering and Labeling by ILJD

The Feature Engineering pipeline comprised of 4 major steps, where each step produces a new feature, Each Feature resemble the pipeline of any judiciary when picking up a case.

The features extracted are:

Case type (i.e., criminal, or civil)

Lower court (i.e., state of high court)

Appeal case status (i.e., Leave Granted or Dismissed)
The case outcome (i.e., winning or losing)

Library Used

a. Regular Expressions
b. Yake (Feature Extracting library)
c. Matplotlib (Visualizing library)

Phrase Modelling

Cast type extraction: we used the sections suffix as a key to extract the case-type, since IPC (Indian Penal Code) and CPC (Civil Procedure Code).These two are the most commonly used in this paper as suffix to evaluate the two different types of the cases, civil code and criminal code, but there are also some other cases where the CPC, was not mentioned clearly, we considered those cases also as the CPC, as the income tax case files are written with just a section name without any CPC in their suffix. But there are a smaller number of cases with such a non-disclosure suffix value.
The classification of the case type can be done without any machine learning algorithm if the case is already handled by any lower court, since the case dealing section is already mentioned and we can obtain the target values with a simple regular expression, module usage. But there are other types of cases which are DER (Draft Enforcing Rule), these are the certain rules which are enforced by any lawyer on a particular company (we can also state them as the bylaws of any firm), these DER type of cases are also considered under the civil cases. But there is an easy catch about this DER case, every DER case is approved and successfully win, because it is a firms document

3.3.1 Yake feature extracting model

Initially we tried to extract the features from the YAKE feature extraction method, YAKE is a light-weight unsupervised automatic keyword extraction method which rests on text statistical features extracted from single documents to select the most important keywords of a text, the yake model gave us the best results, when it is considered in a single document, as all the important phrases and tokens from a case document were obtained. Each token obtained in each case, is considered as a feature using the yake, model, and using this model we obtained the top 500 features from each case.

Customized Phrase Model

The phrase modelling is another technique that we used to extract the accepted cases and the rejected cases, from the dataset, this technique required me to extract more than 100 phrases from each category to ensure that all the cases are evaluated perfectly. After every case documentation finished the final verdict is about the case outcome is given in the last lines of every case, these meaning full verdict phrases are extracted.

3.3.2 Visualizing the Frequency of the models

Fig 1 shows The Initial Idea is to extract the phrases based on the frequencies of the tokens in each, accepted or rejected cases, but after visualizing the frequency distribution of both the accepted and rejected cases, we could understand that frequency distribution cannot help in separating the cases.
This means we cannot use the best frequency tokens, to separate the cases from one other.

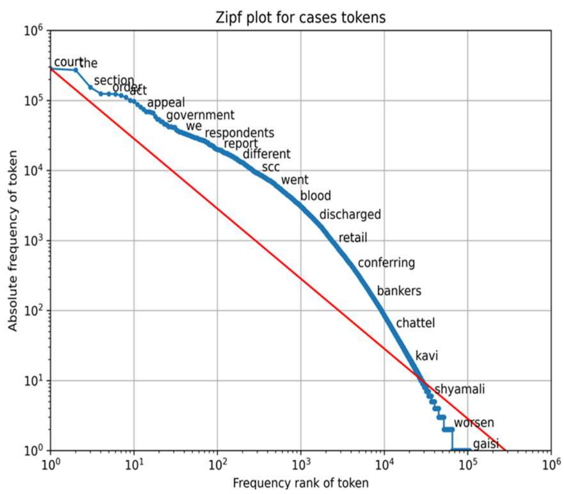
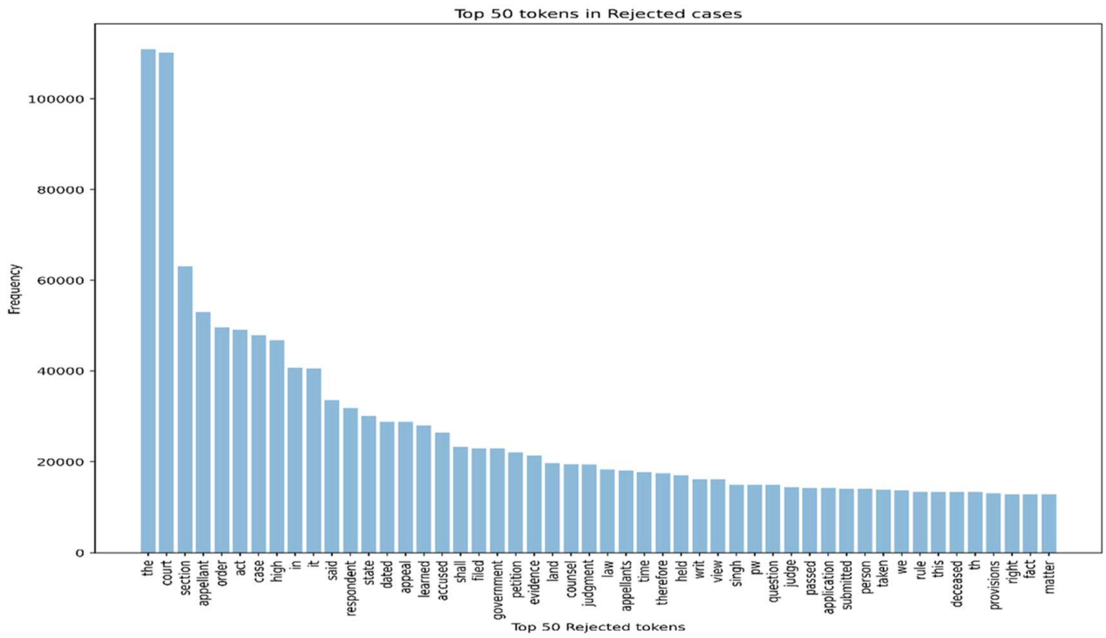


Fig. 1. Zips plot for the cases tokens to check the frequency of the tokens in the whole corpus.



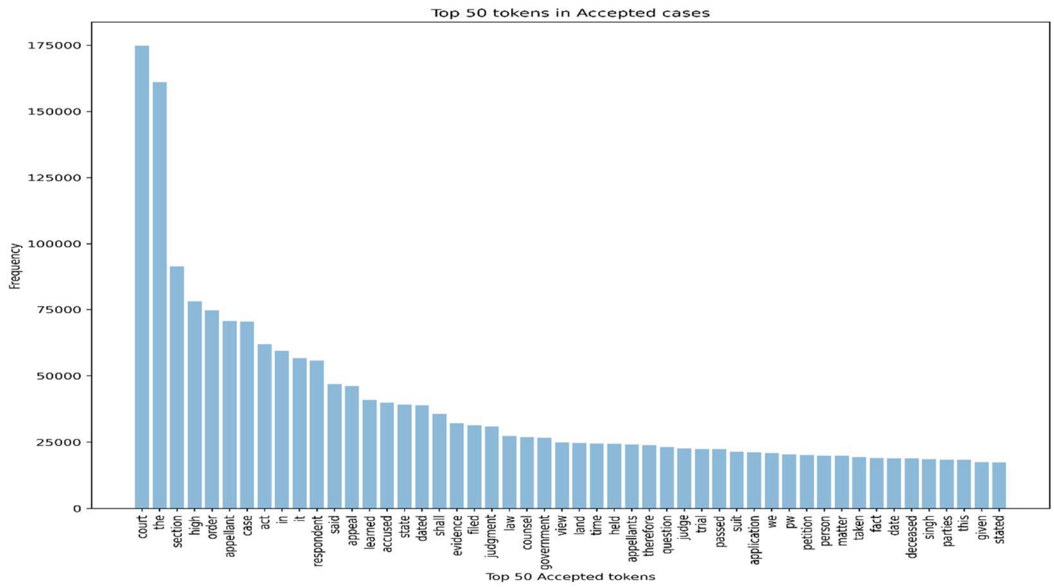


Fig 2. The frequency distributions of all the tokens in the accepted and rejected cases, this is to visualize the best tokens, which could help in separating the accepted cases from the rejected cases.

3.3.2 Using the Chi^2 score to identify the important tokens

The chi square score, analysis of the phrases, to extract the tokens or phrases with the lack of the independence in a document, as the chi2 score gives each term the score based on the lack of independence of a word in a document.

The chi^2 analysis, gave us the phrases, or words we were expecting from the documents, the chi2 analysis, was further made and, we extracted all the meaning full features which can separate the documents, precisely.

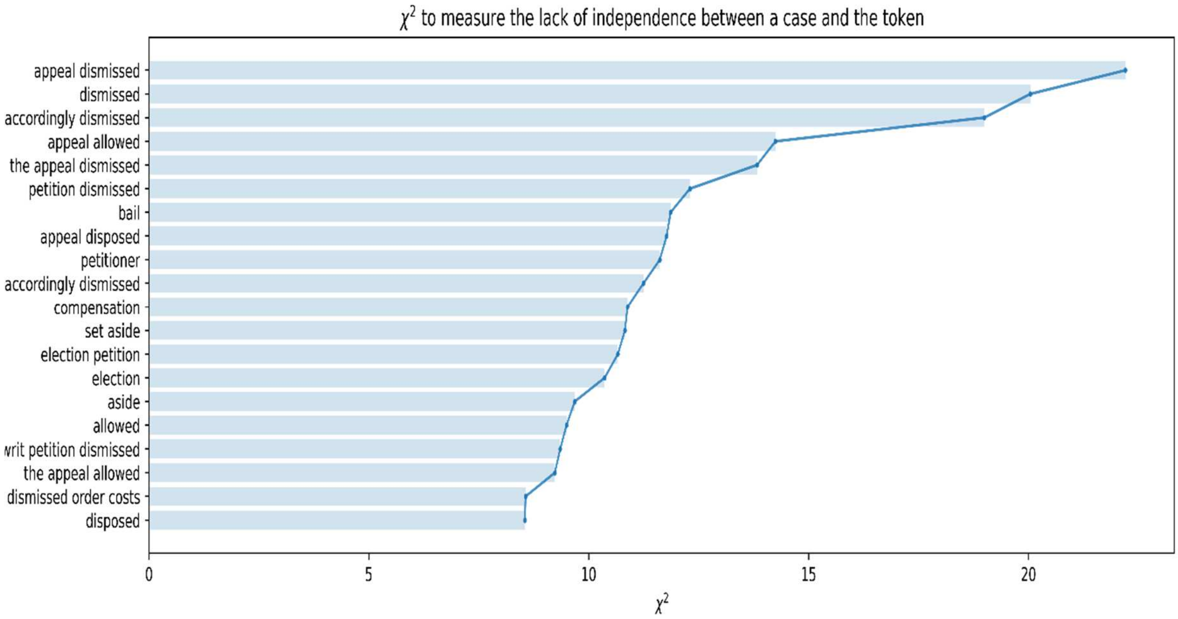


Fig. 3. Important tokens extracted by chi² algorithm

Fig 3 shows the graph obtained using the chi² analysis of the words. This Graph contains the words which are completely related to a particular type of document. This gave us the best token, "appeal dismissed". It must have come from a dismissed case, and we are given a word "appeal allowed", It should have come from the appeal allowed or accepted case.

After using this phrase with the conditional statements if…else. we were not much satisfied with the results, cause this phrase modelling, was able to produce values for a total of 7000 cases only among the 16140 cases, so we again used the Regular Expression Technique, to classify the cases as the accepted cases or rejected cases. The phrases are mainly considered from the chi² output, which could completely classify the documents. It is also observed that imbalance among the classified case. The outcome was: Accepted cases percentage: 62.89% and Dismissed case percentage: 37.11%. So, it is understood that this difference in the case types percentage, could affect any training algorithm that we pass in it, which makes the task more challenging.

3.4 Temporal Aspect

The corpus is arbitrarily isolated into train, approval, and test sets, with the limitation that approval and test sets ought to be adjusted with respect to the judgment. The division into train, validation, and test set was not based on specific time or specific law. The dataset is not in view of any worldly thought or definition since the framework's true intention isn't intended to be restricted to a specific law(s), nor zeroed in on a specific time of time. In actuality, the point is to recognize important features of decisions articulated in connection to different regulation by various judges and across different temporal stages, to have the option to utilize the extracted features to interpret the decision-making process and effectively anticipate the idea articulated by the court given the facts and other influencing factors. While there would be a level of bias included, given the attitude, and understanding of the judge, such difference results in different judgment given by the two different judges for the similar case happened across many years. The focus is to create a framework that would be similarly fruitful in anticipating the result of a judgment given the law that had been stylish twenty years back, as it would in connection to the law that is right now by and by. The dataset created must be applicable to any model that predicts the case outcome. The efficiency of the model is tested by applying it to cases from years back, as to cases from a recent time. Though the proceedings are in different format across various decades, it should be converted to same common format to able to use by any model. If the dataset is not generalized and temporally independent then performance of any efficient model will be degraded because the law might change or updated by the time the final model is ready to be used, the decisions that would hence be delivered by the court may be as not the same as one articulated today, as the last option would vary from one articulated in the 20th century. Not recognizing time as an element during information test decision, accordingly, appears to be the reasonable advance for this situation, particularly given the remarkable rate at which Law or ACT is getting changed today, as well as the speedy development of innovative turn of events.

4. Annotations by Legal Experts

For our situation, the legal experts and legal practitioners were employed to predict the outcome of the given case. Random documents are taken from the test dataset and given the group of 5 legal experts and requested to predict the outcome of the legal document and also the features that influenced the judgment. Each case document was explained by every one of the 5 experts (in disengagement) utilizing the WebAnno structure [15]. They were also asked to rank the features according to their impact of the feature in decision-making. Higher the ranker indicates that feature contributes more to the decision – making. Although documents were explained with explanations arranged by rank, we didn't have a comparative automated model to do this. From the AI viewpoint, this is an exceptionally difficult assignment, and to the best of our information, none of the state-of-art models can do this. Explanation of clarifications is an extremely particular, tedious, what's more relentless exertion. In the current version of ILJD we give clarification comments only to part of the test dataset for the purpose of prediction. Indeed, even a small part of the dataset is capable enough to highlight the difference in annotation done by the expert and the model. All things considered, we intend to proceed to develop the corpus by adding greater reasonableness comments and different sorts of explanations. Moreover, we intend to increase the corpus size.

4.1 Integrity and Bias

While making the corpus, we found a way all potential ways to relieve any dispositions that might sneak in. We have not made a particular decision concerning a particular law or any category of cases, i.e., the examining of cases was completely arbitrary. As clarified before, we took care of the temporal perspective. Significantly, the names of the judge(s), appellants, applicants, and so on, were anonymized in the legal proceedings so that no intrinsic bias sneaks in. The anonymization regarding judge names is essential as legal experts brought up that an appointed authority's personality can sometimes impact the case outcome. It is important that as per the legal experts if we had not done likewise, we might have had higher prediction accuracy.

4.2 Annotation Analysis

A detailed analysis of case outcome prediction and the ranking of features are done. With the help from the legal experts, the complexity of the task and possible difference between the prediction between the legal experts and ILJD is studied.

4.2.1 Labelling Accuracy

The labelling accuracy of the ILJD is compared with the labelling prediction done the legal experts. And the accuracy of legal experts is evaluated with respect to original judgment given by the SCI judges. Results are shown in the table 1. The precision shows that no annotator concurs with the unique judgment in every one of the cases. Though the accuracy of the labelling done by the legal experts are high, they still can't achieve 100% accuracy since no experts accept the original judgment given the SCI judge. This potentially portrays the subjectivity in the lawful area with respect to direction. The bias aspect such as sentiment and emotional analysis are observed during the decision-making process. To sum up, the review demonstrated that the sources of chaos are primarily because of contrasts in linguistic understanding (by the annotators) of the legitimate language given for the legal proceedings.

4.2.2 Inter-Annotator Agreements

For the quantitative analysis, the accuracy of labels given by each of the legal experts is compared with each of the legal expert's labelling accuracy. Table 1 shows the labelling accuracy of each of the legal experts. Table 2 shows the highest value between the experts' 1-5 and 3- 4. The evaluation of labels given by the expert explains the complexity and bias of the task. Legal proceedings are in general long, wordy, and hence challenging to summarize.

Table 1 Legal Expert's Accuracy in labelling

Expert	Accuracy
Expert 1	92.34%
Expert 2	89.45%
Expert 3	97.68%
Expert 4	97.37%
Expert 5	92.52%

Table 2 Comparison of labelling accuracy of each of the legal experts

	Expert 1	Expert 2	Expert 3	Expert 4	Expert 5
Expert 1	100	86.3	93.2	91.4	86.4
Expert 2	93.1	100	92.4	88.6	91.3
Expert 3	93.1	93.4	100	92.3	94.7
Expert 4	86.3	88.5	94.6	100	89.2
Expert 5	85.3	92.7	94.3	92.7	100

5. Performance Analysis

This section explains the performance comparison of ILJD created using regular expression and manual dataset creation done by the legal experts. The comparison of accuracy among experts is already given in the table 2 and even experts can't achieve 100% accuracy the ambiguity arises though the same facts. Table 3 explains the accuracy achieved by experts and the ILJD. The performance is evaluated in two tasks. (i) ability to label the legal proceedings correctly and (ii) identification of features or sentences that influenced the judge during decision making. The results show ILJD is able perform the task with high accuracy than legal experts. This is because of the difference in thinking among humans. However expert 5 has the close match with the ILJD than other experts.

Table 3 Comparison of ILJD with Legal Experts

	ILJD	Expert 1	Expert 2	Expert 3	Expert 4	Expert 5
Labelling Accuracy	94.26%	93.24%	90.56%	91.73%	91.47%	93.78%
Identification of Important Features	93.12%	91.28%	89.45%	92.34%	92.12%	92.31%

6. Conclusion

This paper explains about the method of preparing the legal prediction dataset that can be used in any machine learning model for prediction of judgment. The dataset is not just labelled with accepted/ rejected and win/loss but also identifies the important token that influenced the decision. The performance analysis and results demonstrated the close match in accuracy value between the model and legal experts and difficulty in dataset creation from a computational point of view. We trust that the corpus and the assignment would give a difficult and fascinating asset for the Legal NLP scientists. For future work, we might want to expand our work to all case types and to seek after more exploration in such a manner to completely comprehend the unanticipated social ramifications of such models.

Reference

1. Sivaranjani, N., Jayabharathy, J., Teja, P.C., "Predicting the supreme court decision on appeal cases using hierarchical convolutional neural network", , 2021.

2. Sivaranjani, N., Jayabharathy, J., "A Broad View of Automation in Legal Prediction Technology" Proceedings of the 3rd International Conference on Electronics and Communication and Aerospace Technology, ICECA 2019, pp. 180–185, 8822019. 2019.

3. Vu Tran, Minh Le Nguyen, and Ken Satoh, "Building Legal Case Retrieval Systems with Lexical Matching and Summarization Using a Pre-Trained Phrase Scoring Model". In Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law, ICAIL '19, page 275–282, New York, NY, USA. Association for Computing Machinery, 2019.

4. Paheli Bhattacharya, Kaustubh Hiware, Subham Rajgaria, Nilay Pochhi, Kripabandhu Ghosh, and Saptarshi Ghosh, "A comparative study of summarization algorithms applied to legal case judgments", *In European Conference on Information Retrieval, pages 413–428. Springer*, 2019a.

5. Peter Jackson, Khalid Al-Kofahi, Alex Tyrrell, and Arun Vachher., "Information extraction from case law and retrieval of prior cases". *Artificial Intelligence, 150(1-2):239–290*. 2003

6. Haoxi Zhong, Yuzhong Wang, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. "Iteratively Questioning and Answering for Interpretable Legal Judgment Prediction", *Proceedings of the AAAI Conference on Artificial Intelligence, 34(01):1250–1257*. 2020.

7. Pengfei Wang, Yu Fan, Shuzi Niu, Ze Yang, Yongfeng Zhang, and Jiafeng Guo, "Hierarchical Matching Network for Crime Classification". *In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019, pages 325–334*. 2019

8. Filippo Galgani, Paul Compton, and Achim Hoffmann, "Towards automatic generation of catch phrases for legal case reports", *In International Conference on Intelligent Text Processing and Computational Linguistics, pages 414–425.* Springer. 2012.

9. Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. "Neural Legal Judgment Prediction in English". *In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4317–4323, Florence, Italy. Association for Computational Linguistics*. 2019.

10. Haoxi Zhong, Yuzhong Wang, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. Iteratively Questioning and Answering for Interpretable Legal Judgment Prediction. Proceedings of the AAAI Conference on Artificial Intelligence, 34(01):1250–1257

11. Wenmian Yang, Weijia Jia, Xiaojie Zhou, and Yutao Luo. "Legal Judgment Prediction via Multi-Perspective Bi-Feedback Network." *In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019, pages 4085– 4091*. 2019a.

12. Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, et al. Cail2018: "A large-scale legal dataset for judgment prediction". *arXiv preprint arXiv:1807.02478*, 2018.

13. Paheli Bhattacharya, Shounak Paul, Kripabandhu Ghosh, Saptarshi Ghosh, and Adam Wyner. "Identification of Rhetorical Roles of Sentences in Indian Legal Judgments". *In Legal Knowledge and Information Systems - JURIX 2019, volume 322 of Frontiers in Artificial Intelligence and Applications, pages 3–12*. IOS Press, 2019b.

14. Benjamin Strickson and Beatriz de la Iglesia, "Legal Judgement Prediction for UK Courts".

ICISS 2020: The 3rd International Conference on Information Science and System, Cambridge, UK, March 19-22, 2020, pages 204–209, 2020.

15. Richard Eckart de Castilho, Eva Mujdricza-Maydt, Seid Muhie Yimam, Silvana Hartmann, Iryna Gurevych, Anette Frank, and Chris Biemann, "A web-based tool for the integrated annotation of semantic and syntactic structures", *In Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities (LT4DH), pages 76–84,* 2016.