Improving Efficiency and Accuracy of Criminal Case Management of Supreme Court for Predicting Judgment and Penalty with Machine Learning

Girma Assefa Woyessa[1], Teklu Urgessa[2], T.Gopi Krishna[3], Mohamed Abdeldaiem Mahboub[4]

[1,2,3]Adama Science and Technology University, Department of Computer Science and Engineering
School of Electrical Engineering and Computing, Adama, Ethiopia
[4]Department of Information Systems, Faculty of Information Technology,
University of Tripoli, Libya

***Abstract*—** This study explores the use of machine learning to predict judicial decisions in criminal cases from the Oromia Supreme Court. A dataset of 1638 cases was collected and pre-processed, and various ML models were applied with different feature extraction techniques. The Random Forest model with TF-IDF features achieved the highest accuracy for judgment prediction (98.5%), while the Support Vector Machine model with TF-IDF features performed best for penalty prediction (79.68%). Legal experts confirmed the model's effectiveness with a 77.5% accuracy rate. This study highlights the potential of ML for predicting judicial outcomes in criminal cases and recommends further exploration for potential implementation in court systems.

Keywords— Forecast Legal Verdict, Ethiopian Criminal Code Procedures, Penal Legislation, Automated Learning, Synthetic Minority Oversampling Technique.

I. INTRODUCTION

Historically, courts served to resolve disputes impartially. While early legal systems focused on customary law, codified laws and written constitutions emerged over time, leading to the modern concept of judicial decision-making. This process involves legal analysis based on established laws, dispute resolution through reasoned arguments, and verdict issuance. Ethiopia's legal system has evolved through various stages, transitioning from customary law to codified laws starting in 1931. The current constitution establishes a federal system with a diverse legal landscape encompassing various areas like labor, criminal, and family law [1-3]. Criminal law, specifically, defines and punishes offenses, impacting lives and liberties. Ethiopia operates two parallel court structures: one for the federal government and one for each of its ten regional states, including Oromia. Oromia's judicial system comprises Supreme, higher, and first instance courts, with the Supreme Court (OSC) ensuring regional justice consistency. While OSC [4-7] currently relies on manual processes and employs over 2,500 judges, diverse perspectives and potential biases can influence outcomes. Additionally, the court receives appeals from 18 zones, leading to potential overburdening and delays. Technological advancements, particularly in Natural Language Processing (NLP)[8-10] and Machine Learning (ML) [10-11], offer promising solutions to these challenges. Researchers have explored ML for predicting legal outcomes, primarily focusing on judgment (guilty/not guilty) without incorporating penalties, which are crucial aspects of complete judicial decisions. Existing approaches often rely on manual extraction of factors from legal materials, limiting their accuracy and scalability. This study tackles these constraints by utilizing machine learning (ML) to anticipate judicial decisions in the Oromia Supreme Court (OSC), focusing on two dimensions: judgment (accusation) and penalty [12]. The objective is to enhance decision-making, reduce verdict delivery time, and mitigate bias. Through harnessing ML's data analysis capabilities, our aim is to assist non-lawyers, lawyers, and judges in comprehending legal proceedings and elevating the quality of their work. To our knowledge, this marks the inaugural endeavor to predict judicial decisions in Ethiopia specifically for the Oromia Supreme Court.

II. RELATED WORK

Despite promising possibilities, NLP and ML solutions for the legal field are mostly in the testing phase and rarely used in real courts [13]. Additionally, concerns remain regarding ML's ability to fully explain its predictions within the legal domain [14]. Feature extraction in legal documents presents a significant challenge, often requiring legal expertise. This section reviews relevant research in the legal domain using ML techniques related to judicial decision prediction, both globally and locally. Locally, Eskinder M. [15] presented a predictive model for active and pending cases in the Ethiopian Federal Supreme Court. This model focused solely on predicting the time it takes for cases to be decided, not the actual judicial decision itself. Despite not directly predicting judicial outcomes, the study employed an Artificial Neural Network (ANN) model with 9 inputs and 33,000 records, achieving 94.4% accuracy. This represents the only related research conducted within Ethiopia.

Table 1. Review of Existing Research

S/No	Focus	Author	Gap	Technique
1	Case processing time	[15]	Cannot predict judicial decisions, only time span	ANN 94.4% accuracy
2	Judgment prediction	[16]	Small data size, 2 classes (violation/no violation), no penalty prediction, trained per article, ignores court procedure	SVM 87.4% accuracy
3	Penalty prediction	[17]	Only 2 classes, no penalty/verdict prediction	SVM 78.3% accuracy
4	Predicting both judgment and penalty	[18]	No penalty/verdict prediction, only 2 classes (affirmed/reversed), BOW models	SVM 78.3% accuracy
5	Providing informed predictions of Supreme Court decisions	[19]	Small data size, binary classification (acquittal/conviction), features extracted manually	CART 92.5% accuracy
6	implemented a MLN-based method for predicting judicial decisions in divorce cases	[20]	Limited to binary class, no penalty/verdict prediction	Markov logic network 85.6% accuracy

III. METHODOLOGIES

This research adopts a quantitative research design to investigate the application of machine learning for predicting judicial decisions in Oromia Supreme Court. The selected design allows us to identify the relationship between legal documents and judicial outcomes (verdict and penalty) using quantitative data analysis methods.[21].

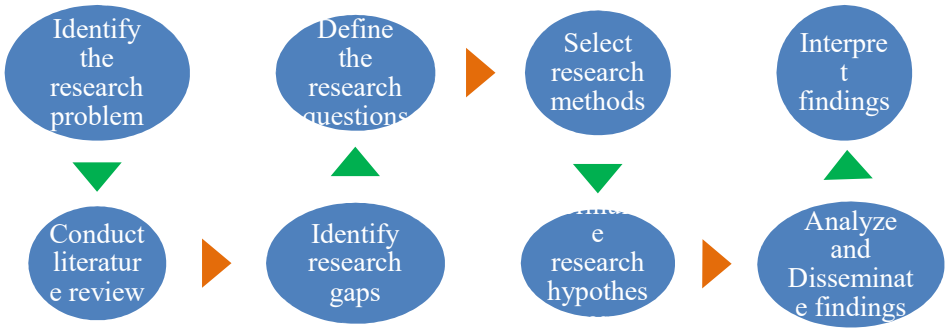

Figure 1. Design Process for PJD Research

1.1. Creating Dataset

Three stages were involved in dataset construction for predicting OSC judgments, namely:

a. Building a dataset of criminal cases through document collection and selection
b. Preparing, filtering, consolidating, and transforming data from images or scanned documents containing text into a unified dataset file.
c. Save the dataset.

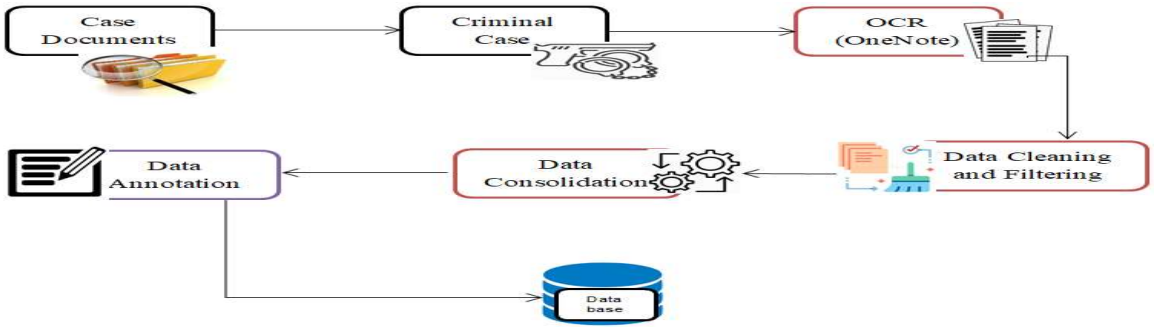

Figure 2. Dataset Construction Methodology

1.2. Data Source

This study draws upon data collected from a closed case at the Oromia Supreme Court (OSC), a vast regional court serving a population exceeding 35 million and handling numerous legal matters. The OSC receives appeals from various regional zones and also initiates new cases at the regional level. After verdicts, closed cases are stored in the archive with rulings from either the cassation or regular court divisions. These cases have been digitized through scanning, creating a comprehensive digital archive. Our research focuses specifically on the judgments rendered by the Oromia Supreme Court. In total, over 8,000 case documents were collected, encompassing a diverse range of legal cases, including civil, criminal, labor, and mixed (tax and torture) matters.

Table 2. Distribution of Initial OSC Case Collection

#	Type of Case	# of case	Description
1	Civil case	3000	Not selected
2	Criminal case	2000	Selected
3	Labor case	800	Not selected
4	Others (mixed)	2500	Not selected
5	Total	8300	

1.3. Building the Research Data Set

- The first step involved classifying the case documents, separating them into two categories: "other" and "criminal." Criminal cases related to murder and injury was identified by examining the nature of the accusation mentioned on the cover page. These relevant documents were then each assigned their own individual folder for further processing. Next, we transformed "fact or use text" and "decision text" within the selected documents into a standardized format. This involved converting any non-standard text, such as images or PDF text files, into plain text for easier analysis. Incomplete entries due to missing information:
- Entries with inadequate formatting.
- Redundant entries.
- Cases that are not of a criminal nature

While 2,000 criminal case documents were initially acquired from the OSC (as shown in Table 3), only 1,638 were ultimately usable for our analysis. This reduction was due to several factors. Some documents lacked complete information, others were not scannable or readable by OCR software, and a final group contained duplicates or cases involving multiple offenses. The final, cleaned dataset was saved in a structured tabular format using Excel software.

Table 3. Data Filtering and Preparation Summary

#	Category of Offense	Number of Issues	Type of Case
1	Physical Harm	803	Offender
2	Homicide	839	Offender
3	**Total**	**1642**	

1.4. Extracting Key Features from Case Documents

Among various text feature extraction techniques, our study utilized the following two [22].

Word Bag: The word bag approach, a fundamental method for transforming tokens into a feature set [23], builds a vocabulary by collecting all unique words from the corpus. For example, consider two documents

Document 1: Impacting the Name Ajjeese Dhokate

Document 2: Impacting Uleedhaan Rukutee with Harkaa Cabse

The word bag method begins by identifying all unique words across the entire corpus to create a vocabulary. In the case of our two documents, the unique words extracted are: {ajjeese, dhokate, uleedhaan, rukutee, harkaa, cabse}. These words represent the vocabulary for our analysis. The BOW technique then creates a vector representation of each document by

marking the presence of each vocabulary word. A value of 0 signifies the absence of the word, while a value of 1 indicates its presence. To illustrate this representation, the word bag approach employs a table where each row corresponds to a document, and each column corresponds to a word in the vocabulary. The subsequent Table 4. depict the vector representations of our two documents:

Table 4. Example of word bag Feature Representation

Word Repository	*ajjeese*	*dhokate*	*uleedhaan*	*rukutee*	*harkaa*	*cabse*
Doc 1	1	1	0	0	0	0
Doc 2	0	0	1	1	1	1

Ultimately, we transformed the provided text into vectors as follows:

```
ajjeesee dhokatee=[1 1]
uleedhaane rukuteee harkaaa cabsee=[0 0 1 1]
```

TF-IDF, a simple method for analyzing text, calculates word importance. It considers how often a word appears in a document (term frequency) and how rare it is across all documents (inverse document frequency). The product of these two values gives each word a score, reflecting its local relevance and global rarity. This makes TF-IDF effective for converting text into a format usable by machines, particularly for legal text classification and research.

IV. PROPOSED MODEL ARCHITECTURE

The proposed solution encompasses methodologies for preparing datasets, diverse ML algorithms, and NLP techniques employed in constructing the PJD model. Additionally, model evaluation techniques are incorporated into the study, employing SVM [25], NB [26], and RF [27] machine learning algorithms, we created a predictive model designed to handle binary and multiclass classification, along with addressing imbalanced data. Assessment of these models involves employing stratified 10-fold cross-validation techniques and classification metrics. Stratified k-fold cross-validation is employed to maintain an imbalanced class distribution in each fold, aligning it with the distribution in a comprehensive training set [28]. The selection of the optimal model is based on the accuracy scores. The chosen judicial decision model is then utilized to develop a prototype capable of receiving new textual inputs and predicting the judgment and penalty associated with the input text. The proposed model architecture is presented to elucidate the research flow.

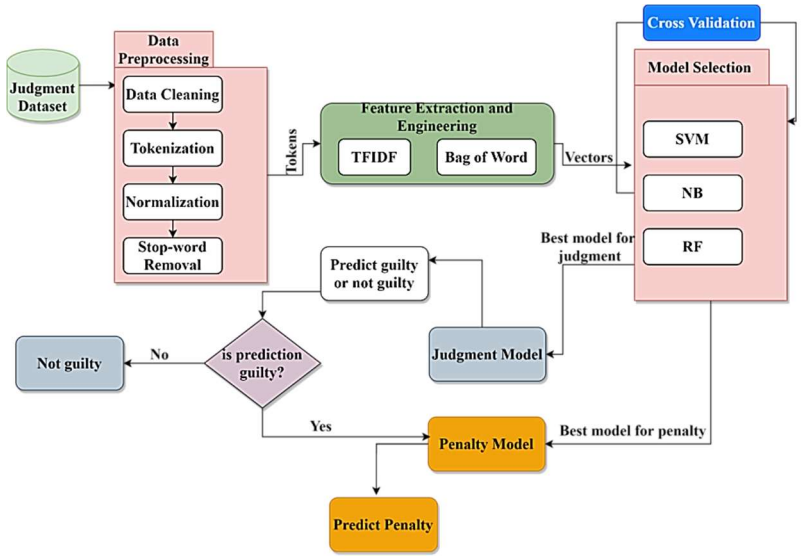

Figure 3. Judicial decision prediction model architecture

This study introduces both binary class classification and multiclass classification. Initially, the proposed model distinguishes between the defendant's guilt and innocence. In the case of a guilty verdict, the individual is then assigned a penalty corresponding to the committed crime. Figure 3 shows a single, combined model for both judgment and penalty prediction, aiming for simpler architecture.

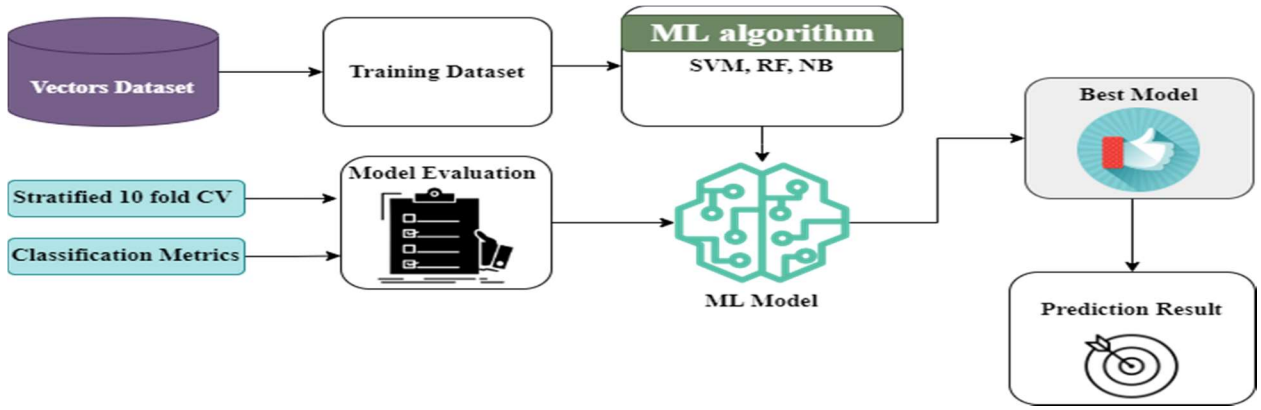

Figure 4. Model training diagram

The prototype model was developed using a separate tool. After training, the best model is deployed on a web server. Users interact with the model through an HTTP interface. The server receives requests and forwards them to the model, which then generates a response based on the user's query.

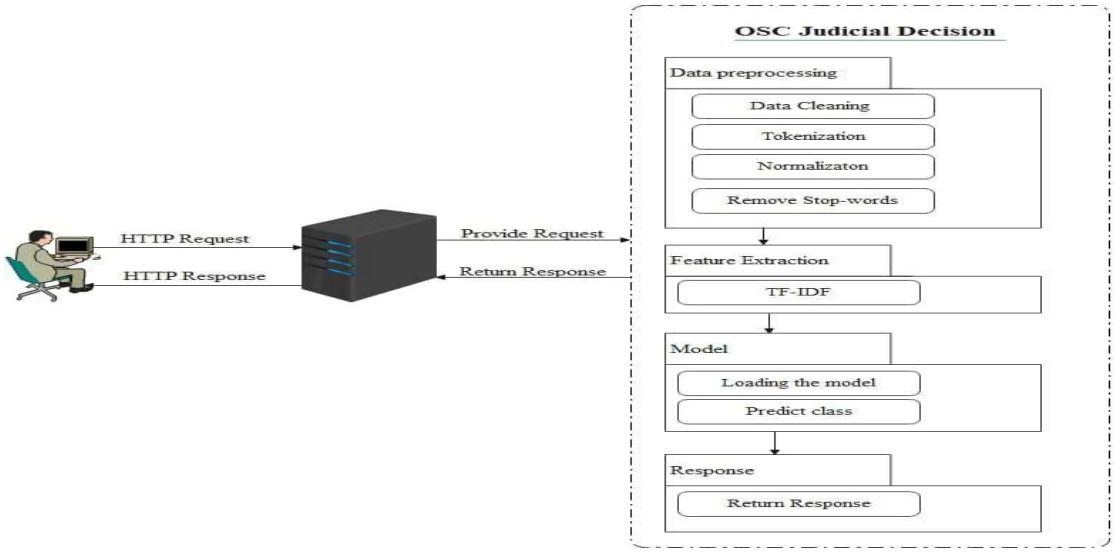

Figure 5. Prototype of judicial decision

In our experiments, we compiled and rearranged the feature labels into datasets. The dataset includes 1736 criminal case documents decided by the Oromia Supreme Court, with a specific focus on cases related to murder and bodily injury. The subsequent Table 5 illustrates the different features and their corresponding descriptions.

Table 5. Feature Characteristics

No	*characteristics*	*The distinct label of the attributes*	*Explanation*
1	Legal Statute	Kewwata	It is a constitutional provision dedicated to prosecuting a criminal based on the committed offenses.
2	Accusation	himata	It encompasses a declaration of the crimes committed by the defendant against the plaintiff.
3	Acknowledgment	wakkatera	It includes the defendant's admission or denial of the alleged crime
3	The prosecutor's witness	raggasisera	Evidence, whether written or testimonial, substantiating the committed crime
4	Defense's Testimony	Ittisa_raga	Evidence, whether written or testimonial, presented in defense of the accused.
5	Verdict	Murte	This section entails the determination of the defendant's culpability, deciding whether they are guilty or not guilty.
6	Mitigation of punishment	YA_salphisu	The defendant will present mitigating factors under Art 82/1/A as the hearing approaches, seeking a reduced sentence.
7	The idea of increasing punishment	YA_cimsu	The concept of escalating the severity of punishment.
8	Penalty phase	Gulanta	It marks the phase of administering punishment, commencing with its initiation and concluding with its termination
9	Sanction/Penalty	adabbii	Upon the court's determination of the accused person's guilt, the subsequent implementation of the punishment follows.

V. RESULTS AND DISCUSSIONS

This section analyzes using machine learning to predict judicial decisions in the Oromia Supreme Court. The data has 1638 judgments (classified as "guilty" or "not guilty") and 868 penalties (with 38 different classes). The analysis explores the results and discusses the approach.

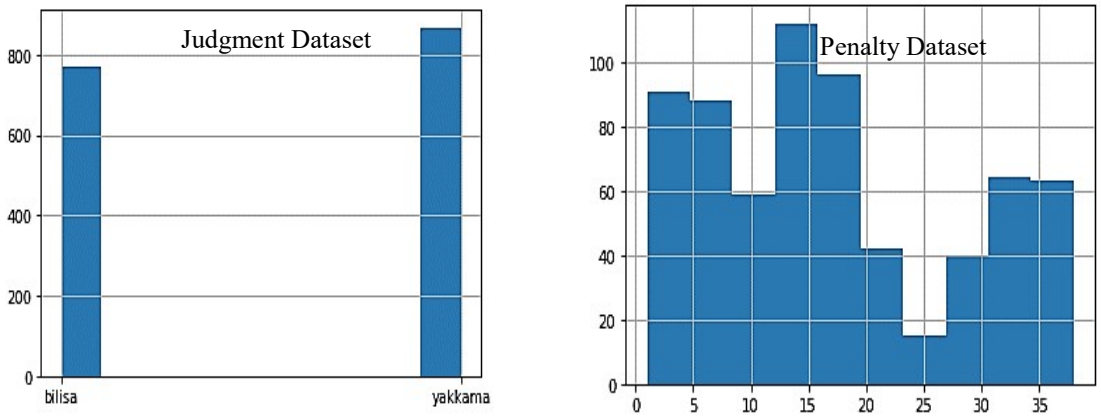

Figure 6. Dataset distribution in each class of judgment and penalty dataset

This study predicted judicial decisions in the Oromia Supreme Court using machine learning, splitting the data into judgr Dataset of Judicial Decisions ınd evaluation. The judgment data has 16 Dataset of Penalties :lasses (guilty and not guilty), while the penalty data has 868 instances with 38 classes.

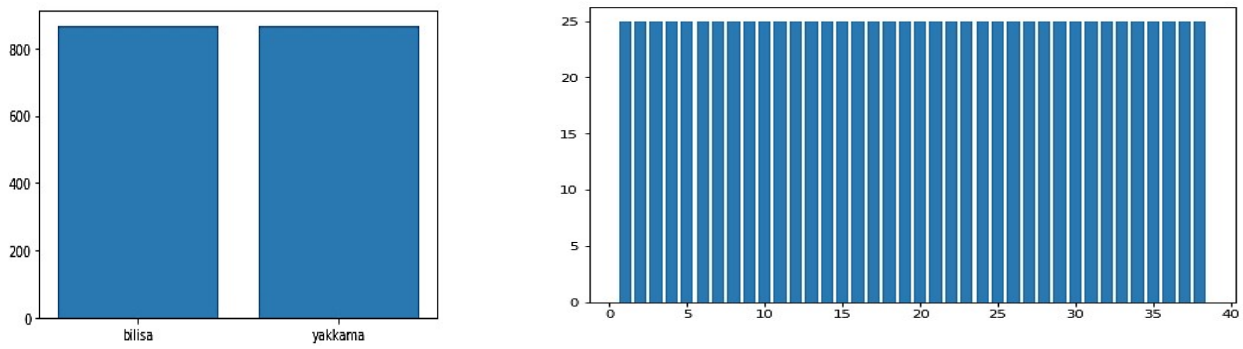

Figure 7. Distribution of Datasets after SMOTE Application

5.1. Judgment Model Evaluation Results

Three machine learning models (SVM, NB, and RF) were trained on a binary dataset to predict guilty/not guilty verdicts. Their performance was evaluated using 10-fold cross-validation and various metrics like precision, recall, F1-score, and confusion matrix. The Table 6. summarizes the average accuracy of these models.

Table 6. Average Accuracy in Three Models with Stratified 10-Fold Cross-Validation

Extraction of Features	*SVM Model 10-Fold Average Accuracy (Mean) Percentage*	*Random Forest Model 10-Fold Average Accuracy (Mean) Percentage*	*Naïve Bayes Model 10-Fold Average Accuracy (Mean) Percentage*
TF-IDF	94.41	96.02	93.02
BOW	93.25	93.95	93.78
Unigram	70.45	68.44	79.36
Bigram	84.26	82.32	81.34
Trigram	91.24	92.54	83.92

Table 6. shows average accuracy scores of SVM, NB, and RF models using five different feature extraction methods. SVM with TF-IDF achieved the highest mean accuracy of 94.41%, followed by SVM with BOW at 93.25%. The remaining feature extractions—unigram, bigram, and trigram—produced 70.45%, 84.26%, and 91.24%, respectively, with the SVM model. For NB, various experiments were conducted with different Naïve Bayes algorithms, and the results displayed in the table indicate that the BOW feature extraction achieved the highest average accuracy at 93.78%. However, TF-IDF, unigram, bigram, and trigram feature extractions also yielded average accuracies of 93.02%, 79.36%, 81.34%, and 83.92%, respectively, with the Naïve Bayes model.

The Random Forest model achieved the highest accuracy across different feature extraction methods. Notably, it reached 96.02% and 93.95% average accuracy with TF-IDF and BOW, respectively.

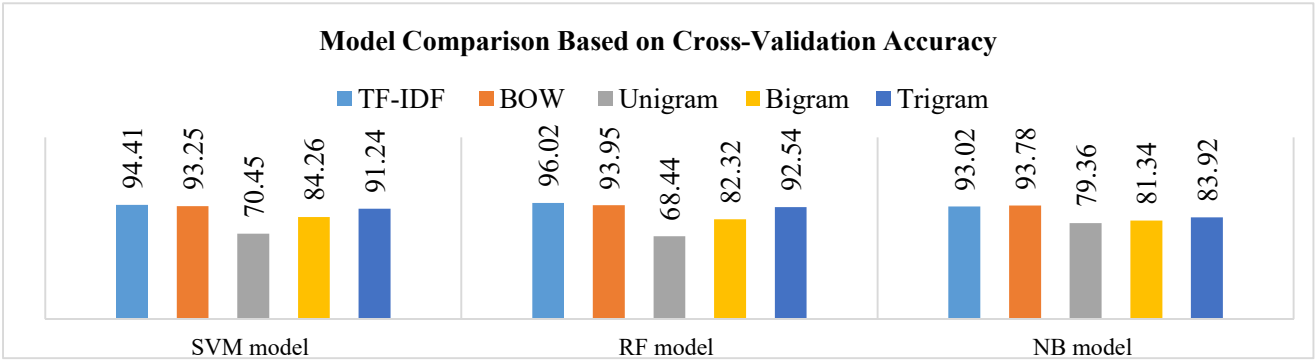

Figure 8. Comparative Analysis of Three Models Using Stratified 10-Fold Cross-Validation Average Accuracy

TF-IDF and BOW feature extractions consistently achieved the highest accuracy across all three models, as shown in Table 7 and Figure 8. The chart below illustrates the feature extraction methods that achieved superior accuracy compared to the alternatives across the three models.

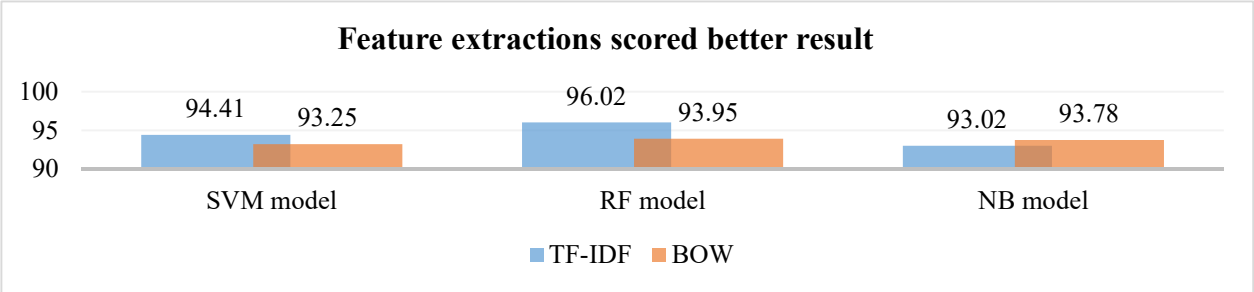

Figure 9. Feature extraction yielded superior results across the three models

5.2. Outcomes of Hyperparameter Tuning

Table 6. presents results with default parameters. Subsequent tuning focused on the best-performing models (shown in Figure 10) using grid search to optimize parameters.

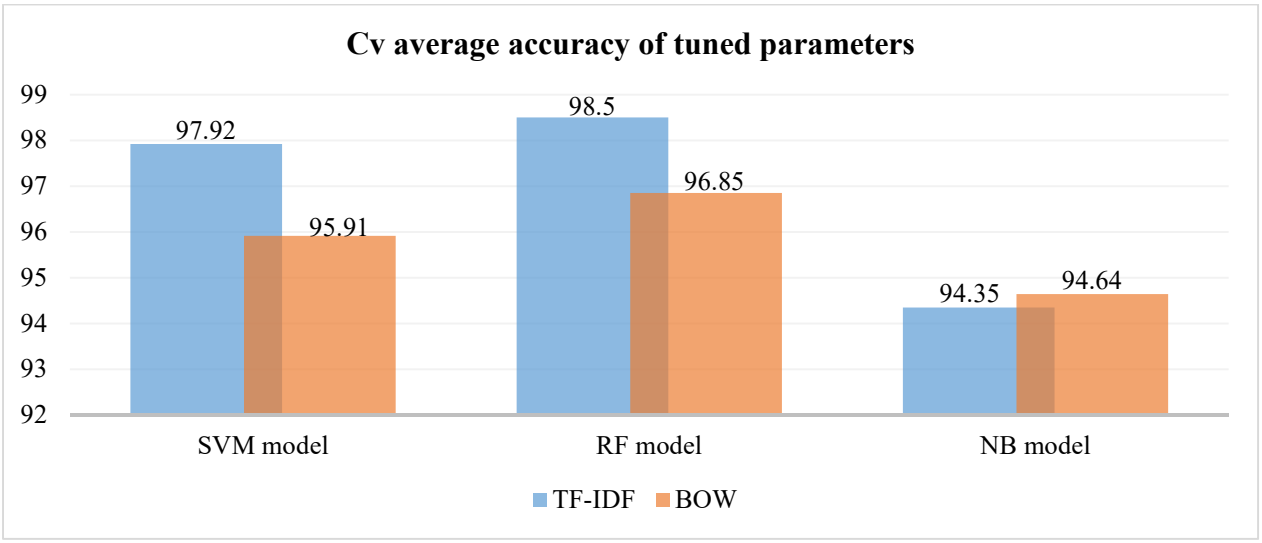

Figure 10. Outcomes of Hyperparameter Tuning for Three Models with Chosen Feature Extraction

After parameter tuning, SVM, RF, and NB models achieved significantly higher accuracy scores with TF-IDF feature extraction: 97.92%, 98.50%, and 94.35% respectively. This emphasizes the importance of parameter tuning for optimal performance, particularly for the judgment model.

5.3. Classification Metrics Results

Apart from the stratified ten-fold cross-validation score, the study employs performance evaluation metrics for the model, including Precision (P), Recall (R), and F1-score (F1). The outcomes of these metrics are detailed in the subsequent table.

Table 7. Classification Metrics Outcome

Feature Analysis	*SVM Model Percentage*			*Random Forest Model Percentage*			*Naïve Bayes Model Percentage*		
	P	***R***	***F1***	***P***	***R***	***F1***	***P***	***R***	***F1***
TF-IDF	96	98	97	98	98	**98**	94	95	94
WB	96	97	**96**	96	97	97	95	95	**95**

TF-IDF feature extraction yielded higher F1-scores than WB for all models. Specifically, RF achieved 98% F1-score with TF-IDF, followed by SVM at 97% and NB at 94%.

5.4 Penalty Model Evaluation Results

A multiclass dataset was used to predict criminal punishments. Models were trained, tested, and evaluated using 10-fold stratified cross-validation. Various metrics, including accuracy, precision, recall, F1-score, and confusion matrix, were used. Table 8 shows the mean accuracy results for each model.

Table 8. Average Accuracy in Three Models with 10-Fold Stratified Cross-Validation

Feature Representation	*SVM Model 10-Fold Average Accuracy (Mean) Percentage*	*Random Forest Model 10-Fold Average Accuracy (Mean) Percentage*	*Naïve Bayes Model 10-Fold Average Accuracy (Mean) Percentage*
TF-IDF	77.98	74.28	61.89
WB	72.66	73.95	70.42
Unigram	59.89	60.48	56.81
Bigram	61.67	64.72	58.29
Trigram	71.92	72.33	59.97

Using TF-IDF, SVM and RF models achieved the highest accuracy for predicting legal judgments (78% and 74%, respectively), exceeding the accuracy of NB with BOW (70%). All models performed worse with other feature extraction methods (unigram, bigram, trigram). TF-IDF and BOW consistently provided the best results.

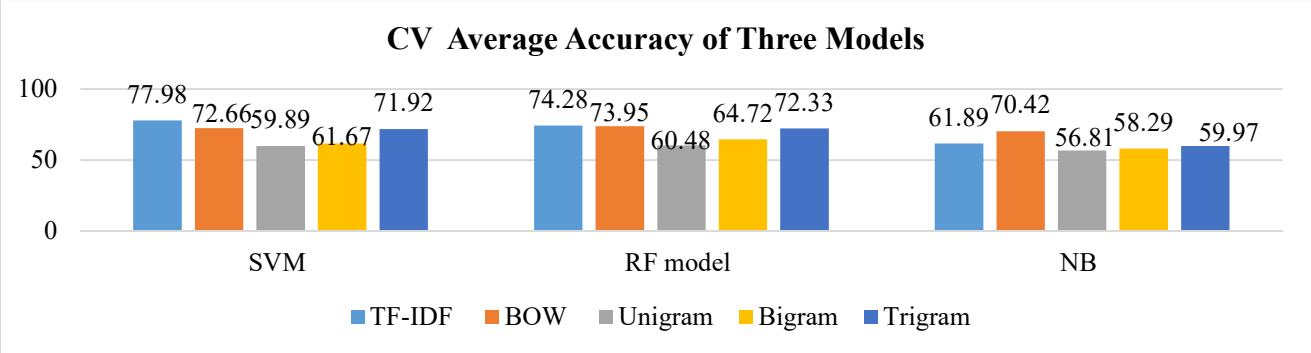

Figure 10. Comparative Analysis of Three Models for Penalty Prediction Using Stratified 10-Fold Cross-Validation Average Accuracy

5.5 Hyper-parameter Tuning Results

After tuning hyperparameters with grid search, the models showed improved accuracy. For TF-IDF feature extraction, SVM achieved 79.68% accuracy, RF 77.37%, and NB 68.22%. For BOW, SVM reached 76.95%, RF 75.87%, and NB 70.44%. These results highlight the importance of hyperparameter tuning for further performance optimization.

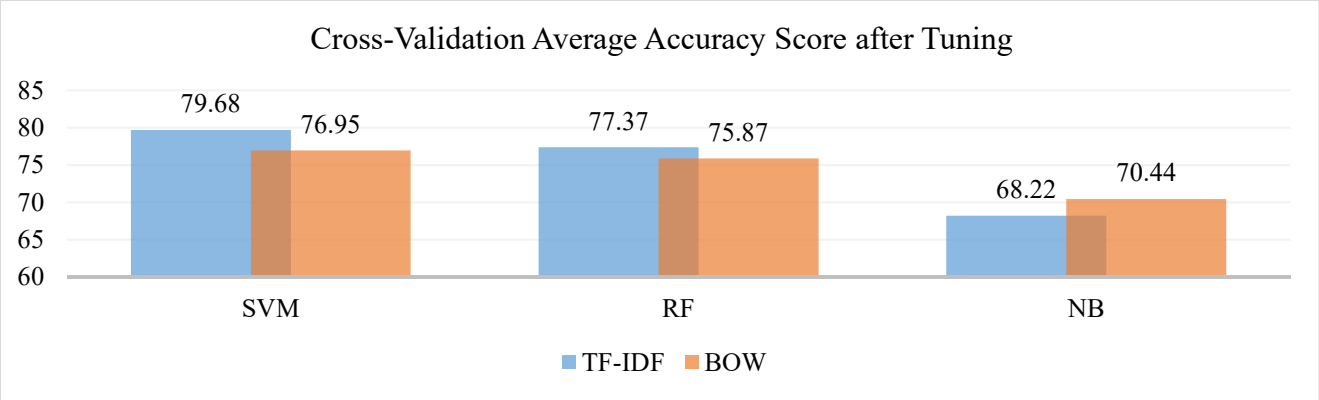

Figure 11. Outcome of Hyperparameters on TF-IDF and WB

5.6. Result of Classification Metrics

The proposed model was assessed using classification metrics (Precision, Recall, and F1-score), and the results are presented in Table 9.

Table 9. Model Performance Metrics

Feature Extraction	*SVM model in %*			*RF model in %*			*NB Model in %*		
	P	***R***	***F1***	***P***	***R***	***F1***	***P***	***R***	***F1***
TF-IDF	78	80	79	79	76	77	64	69	66
BOW	75	77	76	74	76	75	66	70	68

The SVM model achieved the best F1-score (77%) using TF-IDF feature extraction. Overall, SVM outperformed other models. Legal experts evaluated the model's accuracy, focusing on correctly predicted cases.

$$Correctness = \frac{total\ number\ of\ cases \text{ accurately predicted}}{total\ number\ of\ cases \text{ provided for the model}} * 100 \qquad (3)$$

Based on this formula, the performance of the judicial decision prediction model has been calculated.

Table 10. Human Evaluation Results for Model Performance

Quantity of Individuals	***Overall Count of Entered Cases***	***Total Count of Correctly Predicted Cases***	***The Overall Count of Incorrectly Predicted Cases***	***Accuracy Percentage***
OSC law experts (2)	20	15	5	75%
High court law experts (2)	14	12	2	85.71%
First court law experts (2)	6	4	2	66.6%
Total	40	31	9	**77.5%**

Table 11. Model Comparison with Previous Studies

Author	*Research Approach*		*Number of Instances in the Dataset*	*Objective*		*A model with the highest accuracy percentage result*
	Model	*Feature Extraction*		*Predict Judgment*	*Predict Penalty*	
[15]	Only SVM	N-gram	584	Yes	No	SVM with an accuracy rate of 79%
[16]	Only SVM	TF-IDF	3132	Yes	No	SVM with 75 % of accuracy
[17]	CART, KNN, LR, RF, and Bagging	Not clearly put	86	Yes	No	CART with an accuracy rate of 91.86%
Proposed model	**SVM, RF, and NB**	**N-gram, TF_IDF, BOW**	**1638**	**Yes**	**Yes**	**Random Forest achieved a 96.67% accuracy for judgment, and SVM achieved a 77.48% cross-validation accuracy for penalty using TF-IDF.**

VI. CONCLUSION

This research explored using machine learning and natural language processing to predict judicial decisions and penalties based on textual data. By analyzing two distinct aspects – judgment (accusation) and penalty – the research achieved promising results. Specifically, the Random Forest (RF) model demonstrated strong performance in predicting judgments, while the Support Vector Machine (SVM) model proved effective for penalty prediction. Both models were optimized using tuned parameters and TF-IDF feature extraction. Beyond automated evaluation based on classification metrics; the study prioritized human evaluation by law experts. Through evaluations conducted with 40 new cases, the proposed model achieved an impressive 77.5% accuracy, further validating its efficacy in real-world settings. This research demonstrates the potential of AI-powered systems to assist in judicial processes by providing informed predictions and facilitating informed decision-making. Future research could explore incorporating additional factors and refining the models for even greater accuracy and practical applications.

References

[1] Murphy, E. F., & Plucknett, T. F. T. (1957). A Concise History of the Common Law. *The American Journal of Legal History*, *1*(3), 259. https://doi.org/10.2307/844567

[2] Duncan, M. (2021). *history of the judiciary*. Online. https://www.judiciary.uk/about-the-judiciary/history-of-the-judiciary/

[3] Randall Lesaffer, J. A. (2009). *European legal history : a cultural and political perspective*. Cambridge University Press.

[4] Pound, R. (1923). The Theory of Judicial Decision . III . A Theory of Judicial Decision for Today Author (s): Roscoe Pound Source : Harvard Law Review , Vol . 36 , No . 8 (Jun ., 1923), pp . 940-959 Published by : The Harvard Law Review Association Stable URL : http://. *Havard Law Review*, *36*(8), 940–959.

[5] Abate, T. (2014). *Introduction to Law and the Ethiopian Legal System*. *272*, 640.

[6] Oromia Supreme Court. (2019). *Oromia Courts Mission, Objective and Values*. https://oromiacourt.org/en/oromia-courts-mission-objective-and-values.

[7] Greenleaf, G. (1989). Legal expert systems – robot lawyers? *Computers and Law Newsletter*, *2*, 21–24.

[8] Lawlor, R. C. (1963). What Computers Can Do: Analysis and Prediction of Judicial Decisions. *American Bar Association Journal*, *49*(4), 337–344.

[9] Li, J., Zhang, G., Yan, H., Yu, L., & Meng, T. (2018). A Markov logic networks based method to predict judicial decisions of divorce cases. *Proceedings - 3rd IEEE International Conference on Smart Cloud, SmartCloud 2018*, *1*, 129–132. https://doi.org/10.1109/SmartCloud.2018.00029

[10] Strickson, B., & De La Iglesia, B. (2020). Legal Judgement Prediction for UK Courts. *ACM International Conference Proceeding Series*, 204–209. https://doi.org/10.1145/3388176.3388183
[11] Marr, B. (2018). *How AI And Machine Learning Are Transforming Law Firms And The Legal Sector*. Forbes. https://www.forbes.com/sites/bernardmarr/2018/05/23/how-ai-and-machine-learning-are-transforming-law-firms-and-the-legal-sector/?sh=5ba1091832c3
[12] Visentin, A., Nardotto, A., & Osullivan, B. (2019). Predicting judicial decisions: A statistically rigorous approach and a new ensemble classifier. *Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI*, *2019-Novem*, 1820–1824. https://doi.org/10.1109/ICTAI.2019.00275
[13] Kedia, A., & Rasu, M. (2020). *Hands-On - Python Natural Language Processing* (1st ed.). Packt Publishing Ltd.
[14] Suleymanov, E., & Ugur, A. (2019). The role of machine learning in the legal domain: A systematic literature review. *International Journal of Law and Information Technology, 27(1), 50-70*
[15] Eskinder Mesfin. (2009). *Application of Multilayer Feed Forward Artificial Neural Network Perceptron in Prediction of Court Case's Time Span: The Case of Federal Supreme Courts* (Vol. 2009, Issue 75). Addis Abeba University: Unpublished Master's Thesis.
[16] Aletras, N., Tsarapatsanis, D., Preoţiuc-Pietro, D., & Lampos, V. (2016). Predicting judicial decisions of the European court of human rights: A natural language processing perspective. *PeerJ Computer Science*, *2016*(10), 1–19. https://doi.org/10.7717/peerj-cs.93
[17] Medvedeva, M., Vols, M., & Wieling, M. (2020). Using machine learning to predict decisions of the European Court of Human Rights. *Artificial Intelligence and Law*, *28*(2), 237–266. https://doi.org/10.1007/s10506-019-09255-y
[18] Virtucio, M. B. L., Aborot, J. A., Abonita, J. K. C., Avinante, R. S., Copino, R. J. B., Neverida, M. P., Osiana, V. O., Peramo, E. C., Syjuco, J. G., & Tan, G. B. A. (2018). Predicting Decisions of the Philippine Supreme Court Using Natural Language Processing and Machine Learning. *Proceedings - International Computer Software and Applications Conference*, *2*(July 2018), 130–135. https://doi.org/10.1109/COMPSAC.2018.10348
[19] Shaikh, R. A., Sahu, T. P., & Anand, V. (2020). Predicting Outcomes of Legal Cases based on Legal Factors using Classifiers. *Procedia Computer Science*, *167*(2019), 2393–2402. https://doi.org/10.1016/j.procs.2020.03.292
[20] Li, J., Zhang, G., Yan, H., Yu, L., & Meng, T. (2018). A Markov logic networks based method to predict judicial decisions of divorce cases. *Proceedings - 3rd IEEE International Conference on Smart Cloud, SmartCloud 2018*, *1*, 129–132. https://doi.org/10.1109/SmartCloud.2018.00029
[21] Kamiri, J., & Mariga, G. (2021). Research Methods in Machine Learning: A Content Analysis. *International Journal of Computer and Information Technology(2279-0764)*, *10*(2), 78–91. https://doi.org/10.24203/ijcit.v10i2.79
[22] Eklund, M. (2018). Comparing Feature Extraction Methods and Effects of Pre-Processing Methods for Multi-Label Classification of Textual Data. *Degree Project Computer Science and Engineering*, 11.
[23] Muskan Kothari. (2020). Feature Extraction Techniques – NLP. https://www.geeksforgeeks.org/feature-extraction-techniques-nlp/
[24] Karniol-tambour, O. (2013). *Learning Multi-Label Topic Classification of News Articles*. 1–6. http://cs229.stanford.edu/proj2013/ChaseGenainKarniolTambour-LearningMulti-LabelTopicClassificationofNewsArticles.pdf
[25] Adrian Erasmus, "Introduction to Support Vector Machines, http://www.svms.org/introduction.html," vol. 2011, no. January 1st, p. 1, 2010, [Online]. Available: http://www.svms.org/introduction.html.
[26] Zhang, H., & Zhao, X. (2018). Naive Bayes Classifier for Imbalanced Datasets. Journal of Computer Science and Technology, 33(5), 1022–1037. https://doi.org/10.1007/s11390-018-1854-4)
[27] Chen, Y., Li, Y., & Zhang, Y. (2018). An Improved Random Forest Algorithm for Imbalanced Classification. IEEE Access, 6, 64820–64829. https://doi.org/10.1109/access.2018.2878506)
[28] Hussain Mujtaba. (2020). Types of Cross Validation. https://www.mygreatlearning.com/blog/cross-validation/
[29] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*(Sept. 28), 321–357.https://arxiv.org/pdf/1106.1813.pdf%0Ahttp://www.snopes.com/horrors/insects/telamonia.asp.