ISSN 1840-4855
e-ISSN 2233-0046

Original Scientific Article
http://dx.doi.org/10.70102/afts.2025.1833.747

A ROBUST MACHINE LEARNING-BASED ENSEMBLE LEARNING FRAMEWORK FOR HATE SPEECH DETECTION IN LOW-RESOURCE SOCIAL MEDIA TEXT

Husnain Saleem[1*], Muhammad Javed[2], Kiran Hanif[3], Asad Ullah[4], Muhammad Usman Ghani[5], Muhammad Waqas[6], Muhammad Ali Khan[7], Sheraz Ali Hassan[8]

[1]Faculty of Computing, Gomal Research Institute of Computing (GRIC), Gomal University, D.I. Khan, K.P.K, Pakistan. e-mail: jilani.husnain@yahoo.com, orcid: https://orcid.org/0009-0001-7513-1086
[2]Faculty of Computing, Gomal Research Institute of Computing (GRIC), Gomal University, D.I. Khan, K.P.K, Pakistan. e-mail: javed_gomal@gu.edu.pk, orcid: https://orcid.org/0000-0001-6884-6641
[3]Faculty of Computing, Gomal Research Institute of Computing (GRIC), Gomal University, D.I. Khan, K.P.K, Pakistan. e-mail: safdarkiran3@gmail.com, orcid: https://orcid.org/0009-0000-7573-7515
[4]Faculty of Computing, Gomal Research Institute of Computing (GRIC), Gomal University, D.I. Khan, K.P.K, Pakistan. e-mail: asadullahpushia@gmail.com, orcid: https://orcid.org/0009-0006-1504-9642
[5]Faculty of Computing, Gomal Research Institute of Computing (GRIC), Gomal University, D.I. Khan, K.P.K, Pakistan. e-mail: usmanicit@gmail.com, orcid: https://orcid.org/0009-0004-2138-6886
[6]Faculty of Computing, Gomal Research Institute of Computing (GRIC), Gomal University, D.I. Khan, K.P.K, Pakistan. e-mail: alijijer@gmail.com, orcid: https://orcid.org/0009-0006-9489-8836
[7]Faculty of Computing, Gomal Research Institute of Computing (GRIC), Gomal University, D.I. Khan, K.P.K, Pakistan. e-mail: malikwaqasofficial1@gmail.com, orcid: https://orcid.org/0009-0000-7039-3290
[8]Faculty of Computing, Gomal Research Institute of Computing (GRIC), Gomal University, D.I. Khan, K.P.K, Pakistan. e-mail: sheeraz8082@gmail.com, orcid: https://orcid.org/0009-0007-1285-6198

Received: August 23, 2025; Revised: October 01, 2025; Accepted: November 18, 2025; Published: December 20, 2025

SUMMARY

The low-resource social media text i.e., Urdu tweets containing hate speech are identified with the help of a machine learning-based ensemble approach. The dataset used for this study consisted of 8,800 tweets and half of them were labeled as Hateful and the other half as No-Hate. In preprocessing, we took into account the features of Urdu normalizing the characters, eliminating frequent words, and filtering the punctuation. TF-IDF was used to extract features based on unigrams and bigrams and the number of terms was restricted to 5,000. At first, Logistic Regression, Multinomial Naive Bayes, and Support Vector Classifier were chosen as the base learners and the Logistic Regression was used again as meta-learner

in the last layer of the ensemble. The training data consisted of 80% and the rest, 20%, data was used to test the performance of models. Compared to other baseline ensemble approaches and classifiers including Random Forest, Gradient Boosting, AdaBoost, Bagging, Soft Voting, and Hard Voting, our proposedmachine learning based-stacking ensemble approach achieved a high accuracy of 86.53%, precision of 85.45%, and recall of 86.96% and F1-score of 86.20%. The research indicates that the machine learning-based stacking ensemble approach plays a vital role in the identification of hate speech in Urdu Tweets.

Key words: *machine learning, stacking ensemble approach, TF-IDF, hate speech detection, urdu tweets.*

INTRODUCTION

The necessity to identify the hate speech has become significant in the context of research due to the increased level of negative content that is being shared on the Internet. The social and psychological consequences of hate speech, which implies abusive, discriminatory, or offensive speech directed to a person or a group of people based on some of their features, such as race, religion, gender, or nationality, are considerable [1]. There has been a tremendous rise of such content on the social media platforms like twitter and it is important to have automated detection to reduce the negative effect. Hate speech detection is, however, a complex process, especially in low-resource languages, which are overwhelmed by the lack of annotated data, pre-trained models, and computing resources [24]. Hate speech is any type of statement that is intended to oppress, insult, or discriminate against people based on who they are for example: their racial background, religion, ethnicity, gender or nationality. It incites negativity and causes a rift in our society, resulting at times with tension or worse violence. Social media has made it easier to spread hate speech and more difficult to rein it in, affecting people on a broad scale. This problem is exacerbated in languages that have less digital resources such as Urdu, which suffers from lack of tools and datasets.

Most other languages are not well-represented in the study area, even though significant progress has been achieved in the detection of hate speech in the most popular languages, such as English, Spanish, and Arabic [3][25]. Urdu is a language with a writing system, morphological complexities, and the lack of annotated datasets that present particular challenges to natural language processing (NLP) tasks, having more than 230 million speakers, most of who live in Pakistan and India [5]. The other factor associated with Urdu and Urdu-English code-switching is a high degree of linguistic variability that makes the task of text classification even more problematic [2] [26].

Deep Learning (DL) and Machine learning (ML) models in particular, have been quite successful in the context of text classification, in terms of sentiment analysis, spam classification, and hate speech [7][27]. However, because of their potential, the deep learning methods typically require large amounts of labeled data, which is not accessible to languages like Urdu. Additionally, linguistics of such languages is often too complicated to be fully encompassed by the traditional machine learning models [9] (Figure 1).

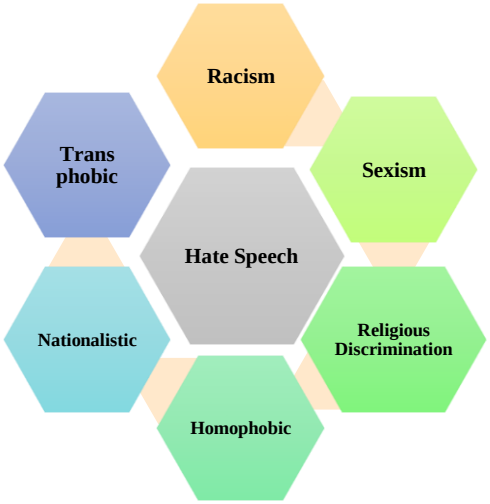

Figure 1. Different forms of hate speech

Ensemble Learning for Hate Speech Detection

Ensemble learning is a form of machine learning involving the combination of several models for improved performance in terms of accuracy and robustness. Rather than depending on one, it synthesizes the powers of multiple algorithms to give an output. Common ensemble methods include bagging, boosting, and stacking, each enhancing performance uniquely. Ensemble learning often performs better compared to individual performances since it reduces the error variance and minimizes the overfitting problems. It is hence a great technique in handling challenging tasks such as text classification and hate speech detection. The ensemble learning methods as Algorithms combining multiple machine learning models with the goal of improving predictive performance have been particularly effective in a wide range of NLP tasks. Machine learning Based Stacking Classifiers (wherein several base models are trained and their output is combined by a meta-learner) have been shown to outperform individual classifiers in sentiment analysis, spam detection and hate speech classification [28][11]. Stacking enables the use of the complementary strengths of the various classifiers in order to generate stronger and more accurate predictions.

In ensemble learning, the approach has demonstrated to be of significant promise in the context of hate speech detection. [29] used ensemble classifiers to detect abusive language on social media, and it showed significant improvement compared with single-model methods. Along the same lines, the most recent literature on hate speech detection in other low-resource languages has experimented with ensemble techniques to represent an effort to mitigate the weaknesses of individual models [13][14][4]. Nonetheless, ensemble methods of hate speech detection in Urdu are still lacking subscripts, which make it necessary to investigate methods addressing such issues as Urdu text.

Challenges in Urdu Hate Speech Detection

Hate speech detection in Urdu is highly challenging because of its rich morphology, complicated script, and diverse writing styles. This is further complicated by the model training process because of the limited amounts of available annotated datasets and linguistic resources. Code-switching between Urdu and Roman Urdu further makes text processing difficult. The other obstacles in proper classification include spelling variants, slang expressions, and context-sensitive expressions. These are indications that better NLP tools and larger high-quality datasets are necessary in Urdu for improving the detection performance. There are certain challenges that make detection of hate speech in Urdu more challenging than languages that have more resources. To begin with, the morphological complexity of Urdu means that along with its script variations (e.g., the difference between "ى" and "ي"), it is more complex to tokenize and normalize [15]. Urdu also employs a range of colloquialism, slang, and transliterated words particularly in the social media where language mixing between languages like Urdu and English is common, a process called code-switching [6]. These complexities demand advanced preprocessing methods to provide the textual data with proper cleaning and preparation of the data to be used by the machine learning algorithms.

Secondly, training robust models becomes even more challenging due to the unavailability of large annotated datasets in the detection of hate speech in Urdu. Although some progress has been made in constructing datasets of languages such as English and Arabic, the amount of Urdu datasets remains low and it is difficult to train high-performing models [16]. Moreover, the nuances of finding hate speech in such a language as Urdu, where the manifestation of hatred might be different than in languages with a larger body of research, require the need to create more specific solutions.

Role of Classifiers in Hate Speech Detection

This study detects hate speech in Urdu tweets using a machine learning based stacking ensemble approach. The combination of Multinomial Naive Bayes (MNB), Support Vector Classifier (SVC), and Logistic Regression (LR), were used as base learners and Logistic Regression was used as the meta-learner. The specific role of each classifier was to model the data and to use its specific mathematical properties to increase predictive accuracy and generalizability.

Logistic Regression is the linear classification algorithm, making the probability of a binary event a model using a logistic sigmoid. It approximates the probability of the target variable as given the input features. The model is as follows:

$$P(y = 1 \mid x) = \sigma(w^T x + b) = 1 / (1 + e^{-(w^T x + b)}) \quad (1)$$

Where x is the input feature vector, w is the weight vector, b is the bias term and sigmoid activation function is asσ.Being a base learner, Logistic Regression encapsulates linear relationships among input attributes and the target label. When used as a meta-learner in the stacking ensemble, it combines the predictions of the base classifiers to come up with a final prediction. Suppose the outputs of the base classifiers are $h1(x), h2(x), h3(x)$, then the meta-level prediction is:

$$P(y = 1 \mid h1, h2, h3) = \sigma(w1\, h1 + w2\, h2 + w3\, h3 + b) \quad (2)$$

This Meta level modeling assists the ensemble in the learning of how to put together strengths of the individual base learners to enhance robustness and accuracy.

Another base learner included in the ensemble is Multinomial Naive Bayes (MNB), a probabilistic classifier that is perfectly suitable when it comes to text classification. It presumes conditional independence of features and it provides model of the likelihood of data input under each class. Consider a document $x = (x1, x2, \ldots, xn)$, where each xi represents the number of times the word$i$, is used in the document, the posterior probability of class y is calculated as:

$$P(y \mid x) \propto P(y) \prod P(x_i|y)^{x_i} \quad (3)$$

In this case, $P(y)$ is the prior probability of the class y and $P(x_i \mid y)$ is the likelihood of word i, given class y in most cases estimated with maximum likelihood with Laplace smoothing:

$$P(x_i \mid y) = (N_{i,y} + \alpha) / (N_y + \alpha V) \quad (4)$$

$N_{i,y}$Is the multiplicity of occurrence of word i in y documents, N_y is number of words in a y document, V is the size of vocabulary and alpha is smoothing parameter.MNB is successful in sparse feature space with high-dimensionality and it can learn word-frequency-based differences in hate speech.

Support Vector Classifier is also a strong discriminative model, and its aim to identify the best hyperplane taking into consideration the performance of the model and maximizing between the classes. Provided that we have a set of training cases(x_i, y_i), of which the$y_i$is -1 or $+1$, the SVC is an optimization problem of the following type:

$$min\left(1/2 \left|\left|w\right|\right|^2 + C \sum \xi_i\right);\ subjecttoy_i(w^T x_i + b) \geq 1 - \xi_i\ ;\ \xi_i \geq 0 \quad (5)$$

In this case, ξ_i are slack variables to enable soft margin classification and $C$is a regularization parameter that trades off margin maximization and classification error. When the data cannot be linearly separated, the data may be projected to higher-dimensional spaces using kernel functions where a linear separator can perform better. SVC has a special ability to fit complex, non-linear decision boundaries on textual data which is usually important in subtle tasks such as hate speech detection.

Stacking an ensemble of such diverse models enables the system to learn both linear and non-linear patterns, probabilistic distributions of words, and decision margins and thus offer a more complete and accurate classification system. The meta-learner integrates the advantages of all the base learners, decreasing the bias and the variance of the final predictions.

RELATED WORK

The rising popularity of hate speech and abusive materials in social media has led to a new upsurge in studies dedicated to automatic detection of materials on the low-resource language including Urdu. Some recent research has endeavored to achieve success by examining a broad range of designs of machine and deep learning (ML/DL) models and transformers that are effective in identifying and classifying offensive and harmful messages in the Urdu tweets.

A hybrid network composed of CNN and LSTM based on deep learning was used by [17] to identify hate speech in tweets written in Urdu. Their methodology attained an accuracy of about 81%, and it proves that deep neural models can be applied to partial and sequential patterns in Urdu text. A similar attempt has been made by [18] where the problem of abusive and threatening language has been tackled on the basis of supervised machine learning methods deployed, such as Support Vector Machines (SVMs) with different sets of handcrafted features. The research based on the FIRE 2021 Urdu dataset achieved an accuracy of 83.18% that allowed concluding that traditional ML models could be used and were effective in combination with feature set design.[19] concentrated on the Urdu-based multi-class sentiment analysis with an approach based on a transformer. They used a multilingual BERT (m-BERT) architecture and tested it against Multilingual Sentiment Corpus [6]. It demonstrated that the pre-trained multilingual Language models could be used to take care of sentiment and text hate classification in the Low-Resource Languages like Urdu with an F1-score of 81.49% based on system performance [12].[20] proposed UHated, an Roberta-based hate speech detection model with transfer learning that was built on capabilities of transformer models [8]. They built a hand-annotated collection of 7, 800 Urdu tweets and attained a macro F1-score of 82% and out-performed the standard ML and DL models, demonstrating the benefits of contextual embeddings to support classification tasks that require fine-grained information. As [21] suggested, they developed a modeling framework to detect abusive language in the Urdu tweets, which cover both classical and deep learning algorithms. They had 3,500 hand-marked tweets with the best model, an SVM with character trigrams, scoring 82.68% F1-score. This investigation affirmed once again that, although deep models are in fashion, the classical algorithms can be competitive with rich feature engineering. Recently, [22] introduced a large-scale cyberbullying detection system that is specific to Urdu tweets [10]. To collect their data, they have gathered 12,759 annotated tweets, which belong to one of the categories of abuse (insults, profanity, threats, etc.). They found in their experiments that this fastText-based deep learning model was more efficient than other classifiers and the F1-score calculated was above 83%. This paper emphasized the importance of fine-grained classification and usefulness of embedding-based models when representing semantic relations.

Indicative of the amount of progress than can be made concerning the detection of offensive languages and hate speech analysis in Urdu language is the fact that most of these studies recorded a performance of between 81% and 83.5%. Though the traditional machine learning and transformer-based approaches have been promising, the current areas of concern are the lack of good-quality labeled data, code mixing content, and the linguistic diversity of the Urdu language.

RESEARCH METHODOLOGY

The section explains how a Machine Learning-Based Stacking ensemble approach was created and tested to identify Hate Speech in Urdu tweets. The primary goal of the work is to assess the efficiency of an ensemble Machine learning-Based Stacking ensemble approach when it comes to classifying tweets in Urdu as Hate or No-Hate. The process has several significant stages in the methodology: Data preprocessing, feature extraction, model construction, and performance evaluation. The entire stages are described below in details (Figure 2).

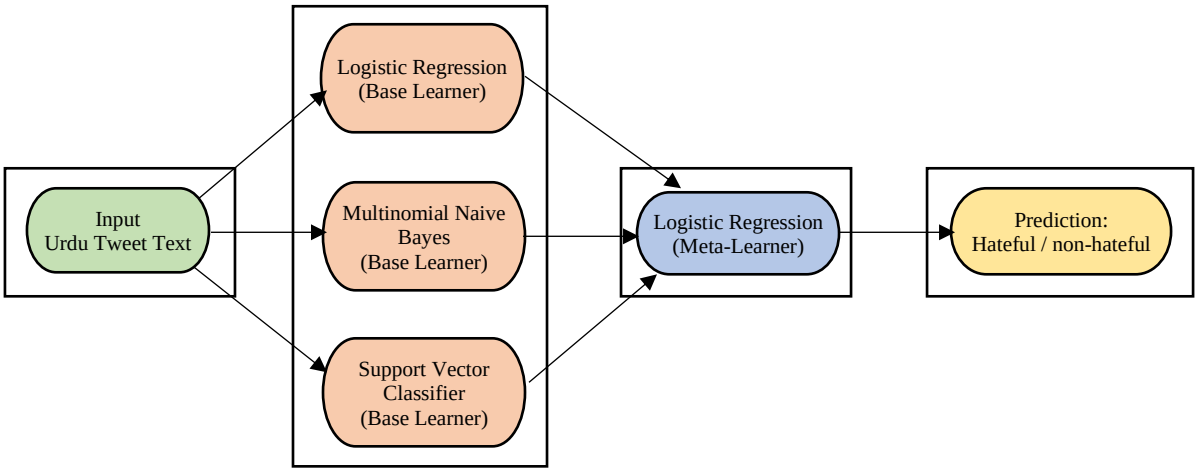

Figure 2. Proposed ensemble approach

Dataset Overview

Data used inthis study is composed of 8,800 manually categorized (Hate or No-Hate) Urdu Tweets. The tweets that contain hate speech are called Hate label and the tweets that contain no hate speech are called No-Hate label. The data is balanced as they have 4,400 tweets in each category that is why it could be used to train and test a text classification model. The main columns of the data include Text (Urdu) which contains the actual text of the tweet in Urdu and Label wherein, the tweet is either hateful or not (Figure 3).

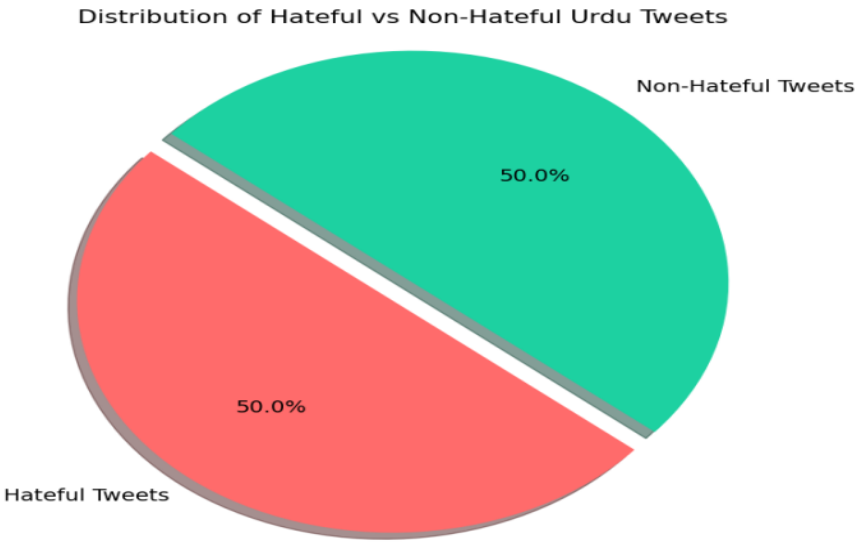

Figure 3. Distribution of urdu tweets

Data Preprocessing

Text data preprocessing is an important process in the preparation of the dataset to be used by machine learning models. The dataset is in Urdu and so certain preprocessing methods were adopted to address the language peculiarities. The initial step was normalization of the text, in which variants of some Urdu characters (e.g. "ی" and "ی", or "ھ" and "ہ") were regularized to a common form.

Then, irrelevant characters, including English letters and numerical digits, were eliminated, which do not add any significant value to the classification of hate speech in Urdu language. Also, the punctuation marks such as commas, periods, and question marks were eliminated as they do not have any meaningful value in the classification task.

The elimination of stopwords was another significant action. Stopwords are the frequent words in Urdu, which do not add any useful information to differentiate hate speech, like(ہے is),(کا of), and(تو then). These stopwords were removed to make the dataset less dimensional and to make sure that the model will concentrate on the more informative words in the text.

Label Encoding

To make the text ready to be used in machine learning models, the categorical labels (Hate and No-Hate) were transformed into numbers. It was achieved with the help of a Label Encoder, a scikit-learn library tool. In this encoding procedure, the No-Hate label was coded 0 and the Hate label was coded 1. This conversion is important since machine learning algorithms normally operate on numeric data and not categorical data.

Text Feature Extraction

The text data were transformed into feature vectors in the numerical form by TF-IDF (Term Frequency-Inverse Document Frequency) vectorization to be readable by machine learning algorithms. This process converts the text into a matrix of numbers with each number indicating the significance of a specific word or phrase in the text in relation to the occurrence of the term in the whole data.

TF-IDF Vectorizer applied in this research was set to take into account the unigrams (single words) and bigrams (two consecutive words) in the text. This is important because it captures the context in which words are used, which could be essential to the tasks such as the identification of hate speech. The TF-IDF transformation was limited to 5000 features (this permitted to decrease the dimension of the data and to improve the performance of the models by focusing on the most informative words).

Model Construction: Machine Learning-Based Stacking Ensemble Approach

It fulfilled the classification task with the help of a Machine learning Based Stacking Classifier. Stacking is a type of ensemble learning, in which multiple base models are employed and a final estimator is applied to refine the predictions of the base estimators further to improve the accuracy of overall estimator. The Machine learning Based Stacking Classifier has three base models that have been chosen in this paper:

- Logistic Regression (LR) is a linear classifier, which works well in such binary classification tasks.
- A probabilistic model that is effective to use on text classification tasks is Multinomial Naive Bayes (NB).
- Support Vector Classifier (SVC), a powerful model which performs well in high dimensional space, which is the case in text classification.
- The final estimator was used in the combination of the predictions of the three base models and it was a Logistic Regression model in this case. The final estimator is intended to improve the output of the base models further, which enhances the performance of the classifier as a whole.

Train-Test Split

The data was divided into two: training and test set. The ratio of splitting was set at 80:20 whereby 80 percent of the data will be utilized to train the model and the remaining 20 percent will be left to test. This methodology will guarantee that the model will be tested on new data that it has never seen before giving a more realistic test of performance.

Model Evaluation

The performance of models was evaluated using different standard metrics, which includes:

- Accuracy is applied in order to determine the percentage of the correctly classified cases.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (6)$$

- Precision, the percentage of true positives (correctly predicted "Hate" labels) of all the instances classified as "Hate".

$$Precision = \frac{TP}{(TP + FP)} \quad (7)$$

- Recall is the percentage of true positives of all actual instances of Hate in the data set.

$$Recall = \frac{TP}{(TP + FN)} \quad (8)$$

- F1 Score is the harmonic mean of precision and recall, and it is a balanced performance measure, especially in the case of class imbalances.

$$F1 - score = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \quad (9)$$

The outcome of the classification was also displayed as a confusion matrix in addition to the aforementioned metrics. Confusion matrix shows the number of true positives, false positives, true negatives and false negatives, which can be used to know where the model has mis performed.

RESULTS AND DISCUSSION

Machine learning-Based Stacking ensemble approach was trained on 80 percent of the data and tested on the remaining 20 percent of the data (test set of 1,760 tweets). The model has shown high accuracy on all the major performance scores, which shows that it can be used to discern between Hate and No-Hate tweets in Urdu. In particular, it was able to attain an accuracy of 86.53% or in other words; about 87 out of 100 predictions were accurate. The precision of 85.45% indicates a good capability to accurately detect Hate tweets and few false positives. The recall of 86.96% indicates that the model managed to identify most of the actual hate speech cases, and the F1 score of 86.20% proves the balance of the model performance in terms of precision and recall as shown in Figure 4.

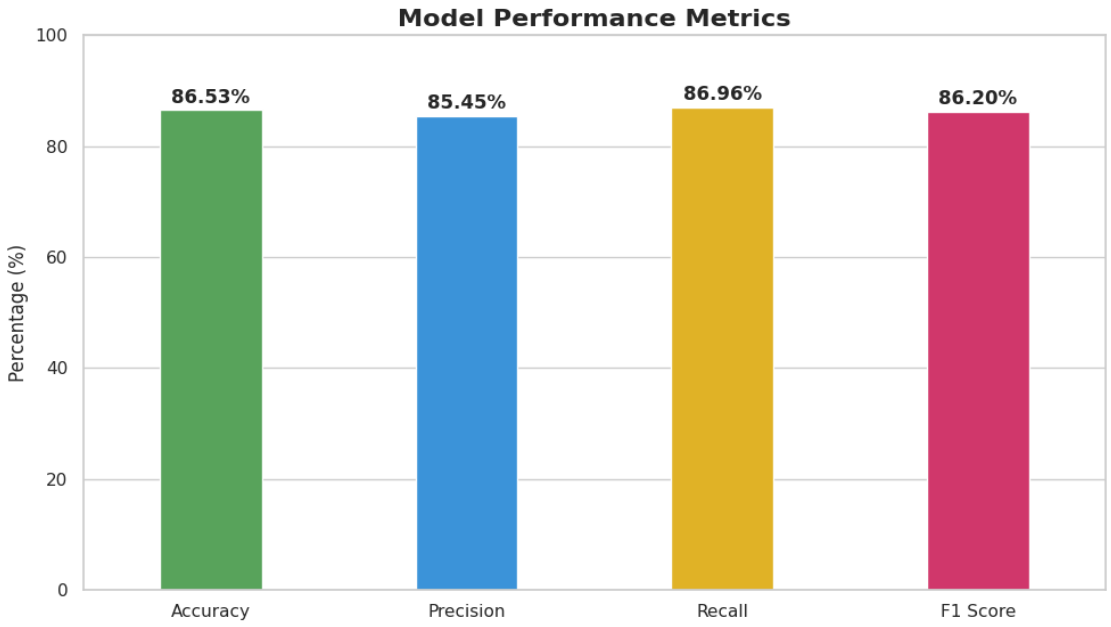

Figure 4: Results of Machine Learning-Based Stacking Ensemble Approach

Table 1. Performance comparison of classifiers

Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Random Forest	81.87	82.92	78.73	80.77
Gradient Boosting	74.77	69.18	86.25	76.78
AdaBoost	63.07	58.14	84.37	68.84
Bagging	81.19	81.48	79.08	80.26
Soft Voting	82.27	81.52	81.90	81.71
Hard Voting	81.19	81.18	79.55	80.36
Machine Learning-Based Stacking Ensemble Approach	**86.53**	**85.45**	**86.96**	**86.20**

To have a comparative idea of the performance of the other ensemble classifiers, the performance of some other ensemble classifiers was also tested, as shown in Table 1 and Figure 5. The Random Forest classifier achieved an accuracy of 81.87%, a precision of 82.92%, a recall of 78.73%, and an F1-score of 80.77%. Gradient Boosting is able to achieve an accuracy of 74.77 percent, precision of 69.18 percent, recall of 86.25 percent and F1 Score of 76.78 percent. AdaBoost did not perform very well as the accuracy achieved was only 63.07% while the precision was 58.14%. The findings of bagging classifier were calculated to be 81.19%, 81.48%, 79.08%, 80.26% for accuracy, precision, recall and f1 score. The Soft Voting ensemble was shown to yield slightly better results than other ensembles. It achieved 82.27% accuracy, alongside 81.52% precision, 81.90% recall, and 81.71% F1-score. In contrast, the Hard Voting approach yielded 81.19 percent accuracy, 81.18 percent precision, 79.55 percent recall, and 80.36 percent F1-score. Evidence in these comparisons indicates that the Machine Learning-Based Stacking ensemble approach is a high-performance one, as it performed better than other established ensemble methods in all significant measures.

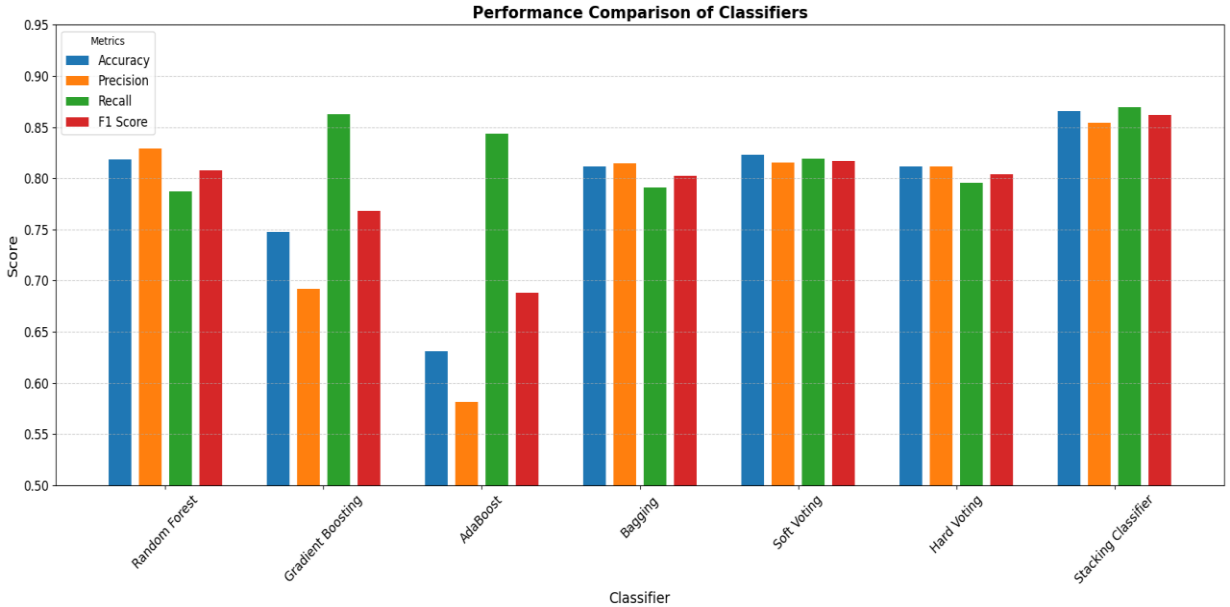

Figure 5. Visual summary of different evaluation metrics for the different classifiers

Moreover, the confusion matrix also indicated the robustness of the model since there were numerous correctly classified instances under both categories. The matrix showed that the model was efficient in reducing false positives (identifying No-Hate tweets as Hate) and false negatives (failing to identify actual Hate tweets), which further proves the validity of the classifier.

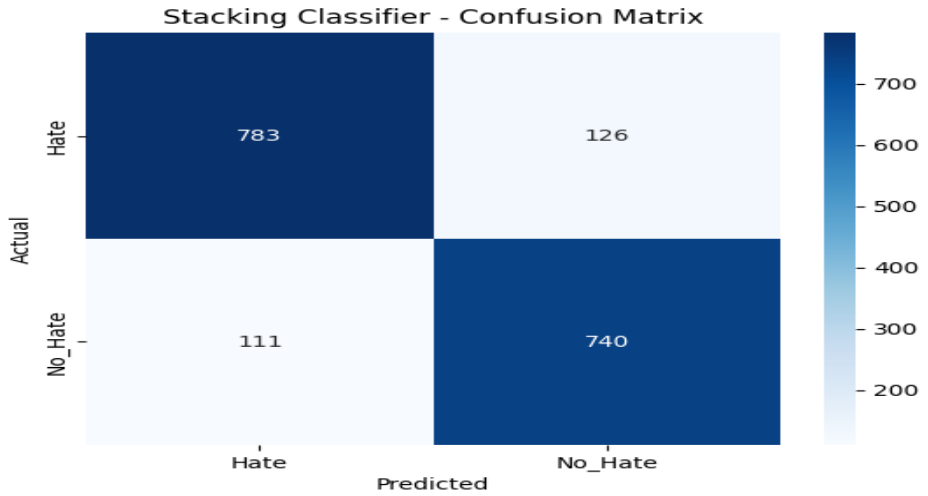

Figure 6. Confusion matrix showing the distribution of correct and incorrect classifications for hate and no-hate tweets

Figure 6's classification report gave us an in-depth understanding of how the model performed per the classes. The Hate and No-Hate classes had high scores for precision, recall, and F1-score. This indicates that the model operates equally for both languages and is independent of the classes. This is important while working with sensitive data like hate speech detection. The results of the evaluation indicate that the ensemble-based stacking classifier using machine learning is the appropriate and efficient method for detecting hate speech in the Urdu tweets.

COMPARATIVE ANALYSIS

This section contains the comparisons and contrasts of the proposed research study with the available alternative methods and techniques of hate speech identification in the Urdu tweets.

Table 2. Proposed study comparison with existing studies

Studies	**Domain**	**Techniques Used**	**Dataset**	**Accuracy, Precision, Recall and F1-score**
[17]	Hate Speech Detection	BiLSTM	Urdu tweets	82%
[18]	Hate Speech Detection	Support Vector Machine	FIRE 2021 Urdu dataset	83.18%
[19]	Hate Speech Detection	m-BERT	Multilingual Sentiment Corpus	81.49%
[20]	Hate Speech Detection	RoBERTa Embedding	7,800 Urdu tweets	82%
[21]	Hate Speech Detection	SVM with character trigrams	3,500 manually annotated Urdu tweets.	82.68%
[22]	Cyber bullying Detection	fastText Deep Learning Model	12,428 Urdu tweets	84.2%
[23]	Hate Speech Detection	Different ML & DL Models	18,426 Tweets	LR achieved 0.8073 and CNN achieved 0.7786
Proposed	**Hate Speech Detection**	**Machine Learning Based Stacking Ensemble Approach**	**Urdu Tweets**	**86.53%**

As shown in Table 2, our proposed Machine Learning-Based Stacking Ensemble Approach outperforms other baseline models and techniques for hate speech identification in Urdu tweets.

CONCLUSION

In the present research, the Machine learning Based Stacking Ensemble Approach performed better than various other ensemble approaches and classifiers (Random Forest, Gradient Boosting, AdaBoost, Bagging, Soft Voting, and Hard Voting) to identify hate speech in Urdu tweets. With such a high accuracy of 86.53%, the Machine learning Based Stacking Classifier demonstrated a high precision (85.45%), recall (86.96%), and a balanced F1-score (86.20%), which is why it is a highly reliable model to identify hate speech in Urdu tweets. Compared to other classifiers, the Machine learning Based Stacking ensemble approach has surpassed all baseline models and approaches in all the major metrics, especially precision and recall, which proves its robustness in reducing false positives and false negatives. These results were also confirmed by the confusion matrix, which revealed that the model successfully identified Hate and No-Hate tweets with few errors. In addition, the classification report showed that the performance of both classes was balanced showing that the model is unbiased against one class over another which is important in fair and ethical content moderation. In short, we can say that the Machine learning Based Stacking ensemble was a reliable approach in identifying hate speech in the Urdu language which is a low-resource language and has its linguistic peculiarities. The model presented can be taken to be the solution to the real life problem of controlling of objectionable content. The results of the study showed that ensemble learning methods have the ability to solve complex problems like hate speech detection, and they geared to future efforts in this respect.

FUTURE WORKS

Even though the study yielded some positive outcomes with The Stacking ensemble model for hate speech detection in Urdu tweets, there are still endless ways to improve the model performance and more on-range application. Some of this future work is discussed next. A future research direction could include investigating deep learning methods which have proven very successful in natural language processing tasks. Transformer, RNN, LSTM–these are a few examples of text classification models that use deep learning. The text classification models are useful as it captures the complex semantic patterns that are presented in the text. Using models particularly the transformer-based models like BERT (Bidirectional Encoder Representations from Transformers) are likely to bring significant performance boosts to the classification system. The performance of classification system can also be improved by data augmentation of the dataset. The existing data, although it ensures a decent sample size, may be increased with the inclusion of additional sources of Urdu text. Also, to increase the current amount of data and, therefore, the generalization capacity of the model, synthetic data generation techniques like back-translation or paraphrasing might be used.

Moreover, it is possible to optimize the text preprocessing pipeline. Although the existing pipeline covers the simple tasks of stopword removal, punctuation and other characters, the more advanced methods may be applied to cover the intricacies of the Urdu language. It may involve using certain techniques such as named entity recognition (NER), dependency parsing and sentiment analysis to better express context and sentiment in the text. In addition, further enhancement of the model can be done to make it even stronger at a real-life application level through correction in code-switching scenario i.e., namely when the users mix language use like Urdu and the English language.

Another potential field is transfer learning. It is likely that fine-tuning large pre-trained language models like BERT or multilingual BERT (mBERT) on Urdu will yield significant improvements as such pre-trained models exist. The model will be more equipped to detect limited trends in Urdu text that may not be adequately captured in a small domain-specific dataset by leveraging what it learns on high volumes of data in other similar languages. It would also be interesting to create multilingual models that can detect hate speech in multiple languages at once. The model can identify hate speech in different languages, allowing it to be more flexible to use. In areas with varying linguistic sceneries, it can be implemented easily.

Lastly, another direction that can be taken as future work would be real-time hate speech detection, in which the model created would be implemented into social media or other internet environments to detect and report hate speech in real-time. This would involve making the system faster, scalable and

real time so that it can process a lot of data efficiently. In short, despite the fact that the current study forms a good basis of detecting hate speech in Urdu, there are numerous avenues of further optimization and improvement of the model using deep learning, data augmentation, advanced preprocessing, transfer learning, multilingual models and real time deployment. Such initiatives may lead to the development of better and precise mechanisms of addressing online hate speech.

DECLARATIONS

Funding: Not applicable.

Conflict of Interest: No Conflict of Interest.

Ethical Approval: No Required.

REFERENCES

[1] Vidgen B, Yasseri T. Detecting weak and strong Islamophobic hate speech on social media. Journal of Information Technology & Politics. 2020 Jan 2;17(1):66-78. https://doi.org/10.1080/19331681.2019.1702607

[2] Imomova U, Fayzullayeva D, Turdibayev D, Gulomjonova N, Kenjaev B, Shadyeva N, Yarashova N, Zaynutdinova D. A critical discourse analysis of linguistic framing in climate change skepticism across media and political narratives. International journal of aquatic research and environmental studies. 2025;5:121-31. https://doi.org/10.70102/IJARES/V5I1/5-1-12

[3] Founta A, Djouvas C, Chatzakou D, Leontiadis I, Blackburn J, Stringhini G, Vakali A, Sirivianos M, Kourtellis N. Large scale crowdsourcing and characterization of twitter abusive behavior. In Proceedings of the international AAAI conference on web and social media 2018 Jun 15 (Vol. 12, No. 1). https://doi.org/10.1609/icwsm.v12i1.14991

[4] Nayak P, Mathur D. Evaluating the impact of social media algorithms on information dissemination. International Academic Journal of Innovative Research. 2021;8(2):21–4. https://doi.org/10.71086/IAJIR/V8I2/IAJIR0812

[5] Khan L, Amjad A, Ashraf N, Chang HT, Gelbukh A. Urdu sentiment analysis with deep learning methods. IEEE access. 2021 Jun 28;9:97803-12. https://doi.org/10.1109/ACCESS.2021.3093078

[6] Khan AR, Karim A, Sajjad H, Kamiran F, Xu J. A clustering framework for lexical normalization of Roman Urdu. Natural Language Engineering. 2022 Jan;28(1):93-123. https://doi.org/10.1017/S1351324920000285

[7] Zhang L, Wang S, Liu B. Deep learning for sentiment analysis: A survey. Wiley interdisciplinary reviews: data mining and knowledge discovery. 2018 Jul;8(4):e1253. https://doi.org/10.1002/widm.1253

[8] Nirosha G, Velmani RD. Raspberry Pi based Sign to speech conversion system for mute community. InIOP Conference Series: Materials Science and Engineering 2020 Dec 1 (Vol. 981, No. 4, p. 042005). IOP Publishing. https://doi.org/10.1088/1757-899X/981/4/042005

[9] Östling R, Tiedemann J. Neural machine translation for low-resource languages. arXiv preprint arXiv:1708.05729. 2017 Aug 18. https://doi.org/10.48550/arXiv.1708.05729

[10] Prabu K, Sudhakar P. An automated intrusion detection and prevention model for enhanced network security and threat assessment. International Journal of Computer Networks and Applications. 2023 Aug;10(4):621. https://doi.org/10.22247/ijcna/2023/223316

[11] Wolpert DH. Stacked generalization. Neural networks. 1992 Jan 1;5(2):241-59. https://doi.org/10.1016/S0893-6080(05)80023-1

[12] Dharmireddi S, Mahdi HM, Rajendran M, Suryasa IW, Soy A. Artificial Intelligence-Driven Natural language processing for the futuristic Language Processing. In2025 International Conference on Computational Innovations and Engineering Sustainability (ICCIES) 2025 Apr 24 (pp. 1-6). IEEE. https://doi.org/10.1109/ICCIES63851.2025.11033144

[13] MacAvaney S, Yao HR, Yang E, Russell K, Goharian N, Frieder O. Hate speech detection: Challenges and solutions. PloS one. 2019 Aug 20;14(8):e0221152.

[14] Mim SJ, Mahmud T, Ali MH, Aziz MT. Stacking ensemble framework for hate speech detection in bangla videos. In2024 IEEE International Conference on Computing, Applications and Systems (COMPAS) 2024 Sep 25 (pp. 1-7). IEEE.

[15] Daud A, Khan W, Che D. Urdu language processing: a survey. Artificial Intelligence Review. 2017 Mar;47(3):279-311.

[16] Bilal M, Khan A, Jan S, Musa S. Context-aware deep learning model for detection of roman urdu hate speech on social media platform. IEEE Access. 2022 Oct 21;10:121133-51. https://doi.org/10.1109/ACCESS.2022.3216375

[17] Khan MM, Shahzad K, Malik MK. Hate speech detection in roman urdu. ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP). 2021 Mar 9;20(1):1-9. https://doi.org/10.1145/3414524

[18] Humayoun M. Abusive and threatening language detection in Urdu using supervised machine learning and feature combinations. arXiv preprint arXiv:2204.03062. 2022 Apr 6. https://doi.org/10.48550/arXiv.2204.03062

[19] Khan L, Amjad A, Ashraf N, Chang HT. Multi-class sentiment analysis of urdu text using multilingual BERT. Scientific Reports. 2022 Mar 31;12(1):5436. https://doi.org/10.1038/s41598-022-09381-9

[20] Arshad MU, Ali R, Beg MO, Shahzad W. UHated: hate speech detection in Urdu language using transfer learning. Language Resources and Evaluation. 2023 Jun;57(2):713-32. https://doi.org/10.1007/s10579-023-09642-7

[21] Amjad M, Ashraf N, Sidorov G, Zhila A, Chanona-Hernandez L, Gelbukh A. Automatic abusive language detection in Urdu tweets. Acta Polytechnica Hungarica. 2021;8860. https://doi.org/10.12700/APH.19.10.2022.10.9

[22] Adeeba F, Yousuf MI, Anwer I, Tariq SU, Ashfaq A, Naqeeb M. Addressing cyberbullying in Urdu tweets: a comprehensive dataset and detection system. PeerJ Computer Science. 2024 Apr 29;10:e1963. https://doi.org/10.7717/peerj-cs.1963

[23] Saleem H. Performance Assessment of ML and DL Models in Detecting Hate Speech from Mixed English–Roman Urdu Text with Small-Scale Datasets. Advances in Artificial Intelligence and Machine Learning. 2025; 5 (2): 220. InWorkshop on Speech and Language Technologies for Dravidian Languages 2023 (Vol. 244, p. 249).

[24] Santosh TY, Aravind KV. Hate speech detection in hindi-english code-mixed social media text. InProceedings of the ACM India joint international conference on data science and management of data 2019 Jan 3 (pp. 310-313). https://doi.org/10.1145/3297001.3297048

[25] Al-Hassan A, Al-Dossari H. Detection of hate speech in Arabic tweets using deep learning. Multimedia systems. 2022 Dec;28(6):1963-74. https://doi.org/10.1007/s00530-020-00742-w

[26] Khan AR, Karim A, Sajjad H, Kamiran F, Xu J. A clustering framework for lexical normalization of Roman Urdu. Natural Language Engineering. 2022 Jan;28(1):93-123. https://doi.org/10.1017/S1351324920000285

[27] Pitsilis GK, Ramampiaro H, Langseth H. Effective hate-speech detection in Twitter data using recurrent neural networks. Applied Intelligence. 2018; 48:4730–42. https://doi.org/10.1007/s10489-018-1242-y

[28] Hansen LK, Salamon P. Neural network ensembles. IEEE transactions on pattern analysis and machine intelligence. 2002 Aug 6;12(10):993-1001. https://doi.org/10.1109/34.58871

[29] Founta A, Djouvas C, Chatzakou D, Leontiadis I, Blackburn J, Stringhini G, Vakali A, Sirivianos M, Kourtellis N. Large scale crowdsourcing and characterization of twitter abusive behavior. In Proceedings of the international AAAI conference on web and social media 2018 Jun 15 (Vol. 12, No. 1). https://doi.org/10.1609/icwsm.v12i1.14991