International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V72I5P134

Volume 72 Issue 5, 332-345, May 2024

Original Article

Toward Stacking Ensemble Based Bipartite Sentiment Classification of Hindi Movie Review Text

Ankita Sharma[1], Udayan Ghose[2]

[1,2]*University School of Information, Communication and Technology (USICT), Guru Gobind Singh Indraprastha University, New Delhi, India.*

[1]*Corresponding Author: ankitasharma2711@gmail.com*

Received: 16 December 2023 Revised: 25 April 2024 Accepted: 04 May 2024 Published: 26 May 2024

Abstract *- Sentiment analysis has significantly progressed in resource-rich languages like English, but research in Hindi is still advancing. Regardless of being the third most spoken language globally, Hindi faces resource limitations. However, the growing use of technology and Hindi interfaces has led to abundant Hindi text on the web, presenting opportunities for researchers to extract valuable insights. The present work aims to evaluate the effectiveness of ensemble learning methods for bipartite sentiment classification of Hindi Movie Reviews (HMRs). This area has received relatively less attention from researchers. The study involves manually creating a binary HMR dataset comprising 6,000 reviews. Preprocessing and feature extraction are performed on the collected dataset. Several individual classification models are applied to the HMRs; subsequently, the predictions from these models are combined through a hard voting ensemble approach, and finally, an integrated two-layered stacking ensemble architecture is proposed and implemented in the present work. The preprocessed dataset undergoes classification using SVM, RF, DT, and KNN models in the first classification stage. The decisions from these four classifiers are then amalgamated to build and optimize the second-level estimators SVM and MLP. Ultimately, the meta-classifier provides the final prediction for the bipartite sentiment labels. The results demonstrate that the proposed model achieves the highest performance. Furthermore, the outcomes derived from this investigation have undergone rigorous statistical assessment through the application of the Friedman statistical test. The proposed framework has achieved the most elevated ranking in both the HMR and IIT-P movie review datasets, thereby providing substantial verification of the obtained results. Notably, this study is the first instance of the application of a statistical test for supplementary validation within the realm of the Hindi Review Sentiment Classification task.*

Keywords *- Ensemble Learning, Hindi, Sentiment Analysis, Statistical Tests, Machine Learning, Movie Reviews.*

1. Introduction

Hindi, the official language of India, the country with the highest population on earth, holds the distinction of being the third most spoken language globally, with over 615 million speakers [1]. With the exponential growth of Hindi Text (HT) on the internet over the past decade, the emergence of the Unicode standard has made it a fascinating area of research, garnering attention from scholars worldwide. However, despite its status as an ancient language and extensive usage in India, Sentiment Analysis (SA) of Hindi remains relatively underexplored. In today's data-driven era, where data is often referred to as the "new oil," the ability to extract valuable insights from gathered information is crucial for the success of companies and businesses. Nowadays, it has become commonplace for people to share their opinions, experiences, and perspectives in Hindi through online reviews, blogs, and messages. Consequently, there is a pressing need to make sense of the sentiments expressed in these HT reviews. Analysing each review individually is impractical and time-consuming, particularly given individuals' increasing reliance on reviews before making purchasing decisions or determining whether to watch a movie, among other considerations. Categorizing reviews into bipartite polarity classes can provide people with a quick understanding of the overall sentiments associated with films, products, and more. SA, which falls under the umbrella of NLP, aims to recognize polarity and emotional tone in text using computational processing and text mining principles [3]. In the realm of SA, three primary approaches are commonly employed: the Machine Learning Approach (MLA) [4], the Lexicon-Based Approach (LBA), and the Hybrid Approach. Machine Learning (ML) falls under the domain of AI and revolves around training programs to learn from gathered data and enhance their performance through learned experiences. Its

primary objective is to develop programs that can implicitly enhance their outputs based on past experiences [6]. Lately, ML has witnessed widespread application in various HT mining fields, particularly in human language technology tasks, such as SA [7][8], text summarization [9][10], and more. In the realm of SA on HT, various Machine Learning Algorithms (MLAs) have been employed. However, the "no free lunch" algorithm reminds us that no single algorithm can universally solve all problems.

Thus, researchers are now exploring the development of models that integrate multiple MLAs to achieve better performance, especially in the context of Hindi text SA, which remains an area with limited exploration. Ensemble Learning Methods (ELMs) offer a promising avenue to address this, with three primary types recognized: Boosting, Bootstrap Aggregation, and Stacking. Boosting adjusts training data weights in every iteration and combines their predictions through weighted voting [11]. Bootstrap Aggregation involves training multiple weak learners using different bootstrap samples, aggregating their results, and making decisions through voting [12].

In this study, we focus on binary Sentiment Classification (SC) or SA of Hindi movie reviews and incorporate the powerful stacking ensemble technique. Stacked Generalization is a sophisticated approach that leverages the predictions of multiple weak learners as features to train new meta-estimators. It is known to outperform individual models and has the potential to yield superior results. For our specific task, bipartite or binary SC categorizes reviews into positive or negative sentiments.

While SA can be carried out at various levels, such as sentence, document, or aspect, our emphasis lies on sentence-level SC, determining whether a sentence conveys a positive or negative sentiment. For this investigation, we consider only subjective, sentence-level Hindi movie reviews. In this study, we not only present an improved and efficient stacking-based architecture for bipartite SC of HMRs, but we also validate the obtained results on two deployed datasets: HMRs and IIT-P movie reviews. The validation process involves performing statistical analysis using the Friedman statistical test. To the best of the author's knowledge, no previous work has tackled the problem of SC of Human language technology using the proposed architecture and subsequently validated the results using the statistical Friedman test. This unique approach highlights the significance and contribution of our research in addressing SC challenges in the context of HT.

The paper makes several significant contributions, which are outlined below:

- The paper focuses on the bipartite SC of Movie Reviews (MRs) written in Hindi and addresses the complexities of working with Hindi text.
- A high-quality dataset of 6,000 manually created Hindi Movie Reviews (HMRs) tailored explicitly for binary polarity classification is constructed to ensure the reliability of the data.
- Extensive analysis and preprocessing techniques are applied to the collected HMR dataset to prepare it for further analysis.
- Initially, some baseline Machine Learning Models (MLMs) are employed for the bipartite classification of HMRs, leveraging various techniques to achieve accurate results.
- Recognizing the complementary nature of different MLMs in classification tasks, this paper also combines all the classifiers and MLMs using hard or majority voting to enhance the classification performance.
- This paper introduces an integrated two-layered stacking ensemble model for bipartite SC on HMRs, which have better sentiment class predicting capability than the individual classifiers applied and standard ensemble majority voting technique, achieving encouraging classification performance on all metrics and hence strong generalization capability.
- Performance evaluation was done using accuracy, recall, precision, F1 score, and AUC-ROC score.
- A Friedman statistical test is conducted to validate the results obtained, and rank applied classifiers. To the best of the authors' knowledge, no prior research has utilized any statistical test for SC on Hindi text.

The remainder of the paper is as follows: Section 2 presents related studies on Hindi movie SA, Section 3 discusses ensemble learning and SA on HMRs, and In Section 4, materials and methods are discussed. Section 5 describes the approach and proposed methodology, Section 6 presents the experimental setup and results, and finally, Section 7 provides concluding remarks and outlines future directions for research.

2. Related Literature

The emergence of the UTF-8 standard has resulted in a significant increase in Hindi movie review data available online. Analyzing this vast amount of data is crucial as it provides valuable information to viewers, aiding them in deciding whether to watch a Bollywood movie. Additionally, it helps moviemakers understand the strengths and weaknesses of their films, enabling them to make informed decisions and forecast their success. The literature review presented below focuses specifically on MLAs in the context of SA applied to movie reviews in Hindi.

In Paper [14], the authors aim to conduct SA on MRs in Hindi. They employ RF and SVM algorithms for categorization and utilize the Hindi SWN to determine sentiment labels. They suggest the inclusion of a neutral class for future research.

In [15], the researchers propose a method called HOMS for performing document-level SA on movie data. They utilize NB and consider adjectives as polarity-bearing words for classification. Additionally, they employ negative handling techniques to improve performance. They highlight the consideration of discourse relations for future work.

Authors in [16] aim to perform SA on Twitter-fetched MRs using the Senti-lexicon algorithm. The authors utilized a separate list of sentiment words and emoticons in their analysis. They enhance the datasets and plan to explore sarcasm detection and forged review in future research. In [17], authors employ DT, NN, and regression techniques to prognosticate the box office success of movies using the SEMMA approach. They suggest the application of further text mining techniques in future studies.

Authors in [18] conduct SA on MRs in Hindi at the document level using dictionary-based approaches and MLAs such as NB, Maximum Entropy (ME), and SVM. They note that a limitation of their work is using a small dataset. In [19], the researchers focus on predicting the success class of Indian films using NN models. They plan to explore the use of unsupervised MLMs for future research.

In [20], the researchers utilize a dataset comprising 755 MRs to predict the box office performance of movies. They employ various MLAs. Among MLAs applied, the MLP algorithm demonstrates the best performance. The authors suggest that for future work, factors such as movie genre and sequel information should be considered. In [21], the box office success of Indian movies is prognosticated using MLAs, namely DT, KNN, RF, NB, and NN. The study utilizes a dataset containing only 250 reviews.

In [22], the researchers propose a backpropagation NN model to predict the box office success of movies. They gather data from movie databases and social networking sites. The proposed approach achieves an accuracy of 91%. For further work, the authors plan to incorporate additional movie categories and explore new input data attributes. In [23], the authors propose a novel approach that amalgamates SVM and NB classifiers for bipolar SA on three distinct datasets. The results indicate that the proposed method outperforms individual classifiers in all three domains. The accuracy achieved is 89.19% for the Amazon review dataset, 88.66% for the movie dataset, and 78.31% for the tweet dataset.

The objective of [24] is to introduce a hybrid method that combines weak SVCs using the boosting ensemble technique for SA on movie and hotel domain reviews. The study employs a dataset of 2,000 reviews. The results demonstrate that the boosted SVM outperforms individual SVM classifiers in terms of accuracy, achieving an accuracy of 93% specifically for MRs. The study is limited to binary SA of the reviews.

In [25], the researchers propose an ensemble approach consisting of NN and SVM classifiers for SA on social media reviews. The study utilizes a movie review dataset comprising 2,000 reviews, with an equal number of positive and negative reviews. The baseline models employed include NB, MAXENT, and SVM. The MPFS method is used for experimentation, and a genetic algorithm is employed to optimize the feature set. The results demonstrate that the proposed ensemble approach outperforms the baseline models, achieving an accuracy of 97.4%. The SVM classifier achieves the second-highest accuracy of 96.3%, followed by NB with an accuracy of 81.4%. MAXENT performs the least effectively among the models, with an accuracy of 79%. We employed three distinct approaches: LBA, MLA, and Hybrid, performed SA on Hindi movie related..

The study revealed that the hybrid approach outperformed LBA in terms of effectiveness. Additionally, the researchers discovered that DT exhibited the highest accuracy, reaching an impressive rate of 92.97%. Unfortunately, the size of the dataset used in the study was not disclosed by the author. In [27], authors introduced an HTC-SVM model explicitly designed for classifying Hindi documents. They conducted their study using a dataset comprising four Hindi documents comprising binary categories. The model they proposed achieved an outstanding level of accuracy, demonstrating a flawless performance in accurately classifying the documents.

In our previous study, we introduced a majority voting ensemble model to classify Hindi MRs into bipolar categories. Their dataset consisted of 1200 Hindi MRs. After performing preprocessing and extracting features using TF-IDF, they employed several baseline classifiers such as KNN, NB, DT, SVM, AdaBoost, LR, and RF, followed by their proposed voting ensemble method. The results demonstrated that the proposed voting model, which aggregated the predictions of multiple classifiers through majority voting, outperformed each classifier and attained a notable performance. The proposed voting ensemble model attained the highest accuracy rate of 88%. The researchers have plans to explore stacked generalization and expand dataset size.

Our study stands out in several aspects compared to previous research in the field of SA in Hindi MRs. Firstly, we addressed the need for larger datasets by utilizing significantly larger-sized datasets, i.e., the HMR dataset comprising 6000 reviews, which was a unique contribution to the existing literature. Additionally, we introduced the application of the Friedman statistical test for evaluating SA results in the Hindi language. To the best of the author's knowledge, none of the previous studies have utilized any statistical tests for SA, specifically on Hindi text. Moreover, our study went beyond dataset selection and statistical testing. We actively created our primary Hindi movie review dataset, ensuring that it met our research objectives.

In addition, we applied various state-of-the-art machine learning classifiers, along with employing techniques such as majority or hard voting, to enhance the accuracy of SC in MRs. Furthermore, we proposed a stacking ensemble method, which further aimed to improve SA performance in Hindi MRs. By combining these diverse approaches and methods, we achieved more comprehensive and accurate results in SC tasks. Overall, our study addressed the limitations of previous research and contributed significantly to the field of SA in Hindi movie reviews. The combination of larger datasets, the application of statistical tests, the creation of a primary dataset, the utilization of popular MLAs, and the introduction of a stacking ensemble method all contributed to the advancement of SA in this domain.

3. Ensemble Learning and Hindi Movie Reviews Sentiment Analysis

The heart of ML is the art of ensembling. An ensemble is a group of many classifiers' models applied together on a common dataset. A set of base or weak classifier models is used in ensemble learning and is trained and evaluated in parallel to utilize different characteristics of each classifier's pros and cons. Different base classifiers, colloquially known as weak or base learners, are combined to yield an optimal classifier in ensemble learning. This optimal classifier is called an Ensemble Classifier (EC) [29]. Weak learners can be made up of different classifiers or can be made up of the same classifiers with their hyperparameters tuned differently. EC combines the classification results of different base classifiers and outputs the best and most generalized results with enhanced classification efficiency. Usually, the results are combined through maximum voting, stacking, and averaging [30].

Maximum majority voting is the straightforward method of using different base classifiers to predict movie review sentiment polarity classes. The movie review class label prediction made by various classifiers is considered votes. The final review class label has the most votes or more than half of the votes.

The concept of the weighted average ensemble is such that a set of base MLMs are trained on the training dataset, and predictions are made on the testing dataset. The final predictions on the test dataset are obtained after averaging results, reducing total errors [31]. Stacked generalization or model stacking is an efficient method under ensemble learning in which a meta-model is intelligently trained by combining predictions from different base models. The beauty of model stacking lies in its generality. It is common to have bagging and boosting models in a stacked ensemble model. Ensemble learning is based on an old saying that unity is strength. Here, multiple ML classifiers are applied to create an improved classifier. Each classifier makes a significant contribution, and the strengths of other classifiers offset individual biases and weaknesses. It has been observed that combining different kinds of classifiers to make one classifier outperform in most cases, then using individual classifiers. Nowadays, ensemble models are employed in our day-to-day applications, such as anomaly detection, disease prediction, face recognition, person identification, behavior recognition, etc. [32]. An ensemble is a divide-and-conquer approach used to enhance the performance and accuracy of classification and regression tasks.

3.1. Exploring Sentiment Analysis of Hindi Movie Reviews

Bollywood, the Hindi cinema industry, is the second oldest movie industry globally and is renowned as one of the largest movie producers in the world [21]. Over the years, Bollywood has evolved into a multibillion-dollar industry, with significant financial investments at stake [21]. The Indian audience exhibits a profound fascination with Hindi cinema, and their engagement extends to posting reviews on various platforms after watching movies.

Furthermore, numerous reviewers actively critique movies and offer their opinions. With the advent of technology, there has been a shift towards expressing feedback about Hindi movies in Hindi itself. This has resulted in many Hindi Movie Reviews (HMRs) online. The audience's curiosity, reliance, and trust in reading Hindi movie reviews for recommendations have sparked interest among researchers in exploring this area. However, one of the challenges in developing Hindi movie review mining systems is the mixture of languages often used by viewers when posting their reviews online. This linguistic diversity poses a hurdle in extracting valuable insights from the reviews and contributes to the need for more existing systems focused on Hindi movie review analysis.

4. Materials and Methods

This section describes the dataset employed in this work.

4.1. Acquisition of HMRs and Formation of HMR Dataset

The main challenge in Hindi research lies in obtaining correctly annotated datasets. Due to a lack of accessible open-access gold-standard datasets that meet the required standards, researchers often resort to constructing their datasets. A sufficient dataset, both in terms of quality and numbers, is essential for developing an effective ML based solution that can avoid overfitting.

This study's HMR dataset consists of movie reviews written in the Devanagari script. The HMR dataset reviews were collected from well-known and reputable websites such as https://www.aajtak.in/entertainment/film-review, https://navbharattimes.indiatimes.com/movie-masti/movie-review/, https://hindi.filmibeat.com/reviews/. To ensure accurate annotations, each review was manually labeled with one of two opposing polarities - positive or negative.

Table 1. Glance of HMRs dataset

Hindi Review Text	Polarity Class
फिल्म जुग जुग जियो की दिक्कतें इसके संगीत और इसके तकनीकी टीम में हैं। {The problems of the film Jug Jug Jio lie in its music and its technical team.}	Negative
पूरी फिल्म रकुल ने अपनी दमक से चमकाई है। {Rakul has brightened the entire film with her sparkle.}	Positive
कियारा आडवाणी फिल्म दर फिल्म निखरती जा रही हैं। {Kiara Advani is getting better film by film.}	Positive
पांच साल पहले आई फिल्म जुड़वा 2 के बाद से वरुण का करियर डगमगाता रहा है। {Varun's career has been staggering since the film Judwaa 2, which came five years ago.}	Negative
माधवन का अब तक का सर्वश्रेष्ठ अभिनय। {Madhavan's best performance till date.}	Positive
फिल्म गहराइयाँ संगीत के स्तर पर बहुत निराश करती है। {The film Gharaiyan disappoints a lot on the level of music.}	Negative
फिल्म में मनोरंजन के तत्व की कमी आखिर तक खलती है। {The film lacks the element of entertainment till the end.}	Negative

Table 2. Statistics of movie review datasets employed

Dataset	Positive Reviews	Negative Reviews	Total Reviews
HMR	3,511	2,489	6,000
IIT-P Movie Review	823	530	1,353

The HMR dataset comprises 6,000 reviews in a CSV file format, featuring two columns: "Hindi review text" containing the movie reviews and "polarity class," indicating the binary polarity. To ensure the quality of annotations, two language experts independently reviewed each review in the HMR dataset to confirm the polarity label. The inter-rater agreement was assessed using Cohen's Kappa from sklearn.metrics.cohen_kappa_score, which resulted in an impressive agreement level of approximately 83%, equivalent to perfect agreement. For a glimpse of the HMR dataset along with English translations, refer to Table 1.

4.2. Dataset Deployed

Table 2 presents the dataset statistics used in the current research, consisting of the author-made HMR dataset – dataset 1 and the IIT-P movie review dataset, i.e., dataset -2. The dataset-2 is utilized for additional validation, as referenced in [28]. Initially, dataset- 2 had four sentiment labels, likely representing positive, negative, neutral, and conflict. However, for the specific task of bipartite SC undertaken in this research, only reviews with positive and negative sentiment polarities are considered. In contrast, the neutral and conflict sentiment reviews are excluded.

5. Proposed Methodology

The research methodology employed for bipartite SC on Hindi Movie Reviews (HMRs) is illustrated in Figure 1. This study aims to achieve efficient bipartite SC on HMRs. A dataset of HMRs is created, followed by preprocessing techniques to prepare the collected dataset. Subsequently, the Hindi review text is vectorized using TF-IDF to obtain relevant features. The resulting dataset, containing the extracted features, is then divided into testing and training datasets. The testing dataset comprises 25% of the data for evaluating the classifiers, while the remaining 75% is used to train the classifiers. Next, multiple MLMs are applied, along with the utilization of hard voting and a proposed stacking ensemble of classifiers. The sentiment class predictions from all the applied classifiers are evaluated and compared using various evaluation metrics to determine the best MLM for SC of HMRs. The statistical Friedman test assesses the statistical significance of the proposed solution and the applied classifiers. This test enables the ranking of classifiers to identify the top-performing ones.

5.1. Pre-Processing of HMRs

The online movie reviews written in Hindi are often informal, unstructured, and contain grammatical errors. Before applying any MLMs, it is necessary to preprocess the reviews. This preprocessing involves several steps to improve the classification performance. Firstly, the primary processing aims to reduce the dimension of the dataset, vocabulary size, and memory overhead. It involves removing generic stop words that do not carry much meaning, while movie domain-specific stop words are left as they might be relevant in the context of movie reviews. Punctuation and special characters are also removed from the text. To handle non-Hindi words, emoticons, and numbers, they are substituted with their term equivalents. This step ensures that the reviews contain only meaningful Hindi words. Negative words are left intact, as removing them would impact the model's performance. These words play a crucial role in expressing negative sentiments and should be preserved. Next, the MR sentences are tokenized, which involves breaking them down into individual word groups. This process helps in extracting relevant information from the sentences.

5.2. Hindi Review Text Vectoriztaion

It is the process of converting HT reviews into a numerical format. This work used TF-IDF to vectorize and extract features [26]. Features are information or valuable data that can be filtered out from movie reviews. The aim is to extract the relevant features and leave irrelevant features behind.

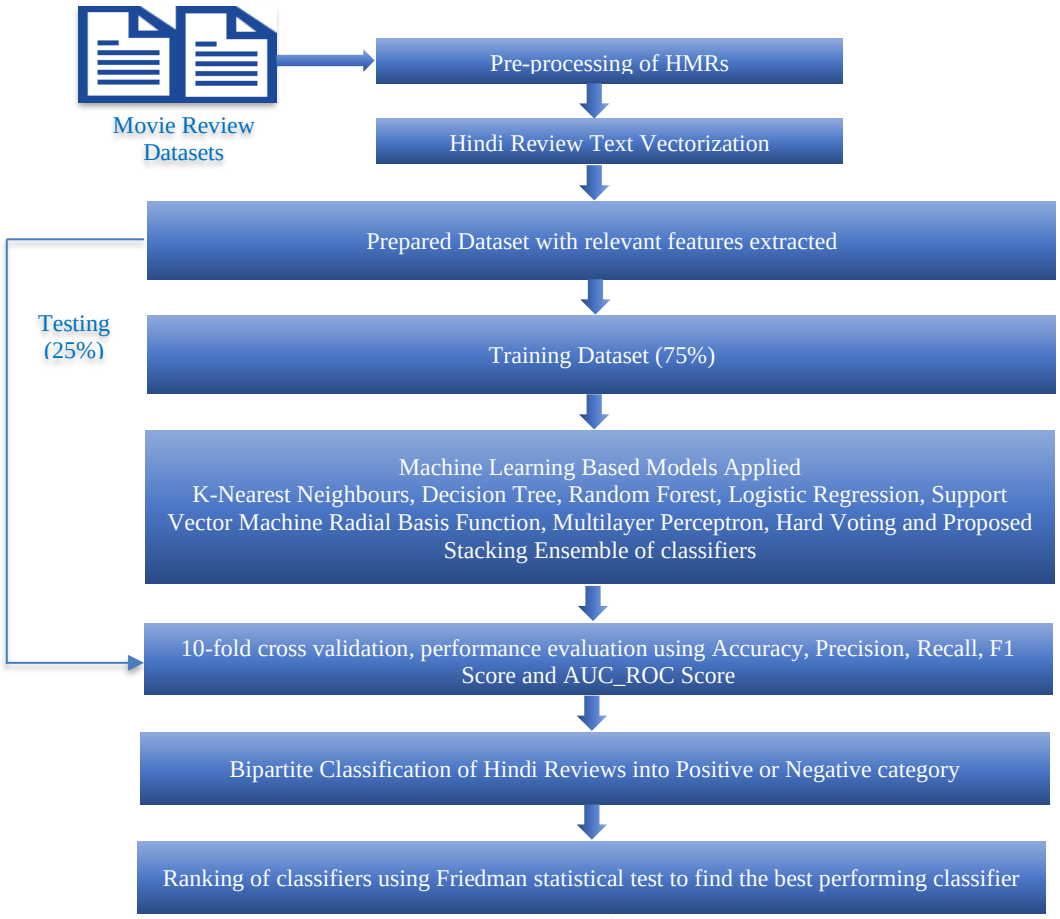

Fig. 1 Research Methodology followed for bipartite SC on HMRs

TF denotes how often a word is repeated in a movie review to the total words in a movie review document; it helps us to find valuable words in movie reviews. IDF prunes the terms or phrases which appear throughout many documents. In other words, TF-IDF assigns weights to each token based on its frequency in the review and its rarity across all reviews. This encoding scheme helps capture the importance of each word in the reviews for classification purposes.

5.3. Brief Description of Various Base Estimators, Ensemble Methods and Proposed Stacking Ensemble Model [13]

5.3.1. K Nearest Neighbors (KNN)

KNN is a non-parametric supervised MLA commonly employed for classification and regression tasks. KNN operates based on the principle of nearest neighbors or feature similarity. Unlike other algorithms, KNN is considered "lazy" because it performs no explicit training. Instead, it utilizes all available data for making predictions. The fundamental idea behind KNN is that similar data points tend to be close to each other. Therefore, when a prediction needs to be made for a new data point, KNN looks for the nearest neighbors in the feature space and assigns a label or value based on the majority of the neighbors. In Python, the KNN algorithm can be imported from the sklearn.neighbors class provided by the scikit-learn library.

5.3.2. Decision Trees (DT)

DT is a tree-like structure, using if-else conditions to represent and classify the data visually. The DT classifier can be imported from the sklearn—tree class in the scikit-learn library. The primary objective of a DT is to construct a tree that can predict the value of the target class. In our case, the sentiment class of an MR. It achieves this by employing decision rules derived from the features present in the data. One drawback of the DT classifier is its tendency to have high variance. This means that it is susceptible to the training data which is provided. Even a slight alteration in the underlying data can lead to significant changes in the resulting tree and, consequently, the predictions made by the model.

5.3.3. Random Forest (RF)

RF is a powerful ensemble method that combines the advantages of DTs with the benefits of ELMs. RF is a collection of DTs working together. It addresses the issue of overfitting often encountered with individual DTs. Internally, RF employs a technique called bagging, which enhances its

performance. As a non-parametric supervised MLA, RF is effective for classification and regression tasks. It accomplishes ensembling by aggregating decisions from multiple DTs and averaging the estimates obtained from each DT in the forest. This collective decision-making process contributes to the robustness and accuracy of RF models. To utilize RF in Python, it can be imported from the sklearn. Ensemble.RandomForestClassifier class provided by the sklearn library.

5.3.4. Support Vector Machine (SVM)

SVM is a widely used supervised MLA employed for classification tasks. The primary objective of SVM is to delineate data points into two distinct classes by utilizing a hyperplane. This hyperplane is designed in such a way that it can accurately assign new data points to one of the two classes in the future. While SVM is commonly used for linearly separable problems, it can also handle non-linearly separable problems by transforming them into linearly separable ones using a kernel function. In this case, the RBF kernel is utilized, the default kernel in the SVM classification algorithm provided by the sklearn library.

5.3.5. Logistic Regression (LR)

LR is an uncomplicated yet effective algorithm commonly employed for classification tasks. It is beneficial when the response variable, also known as the dependent variable, is categorical. In binary LR, the response variable can take one of two values, such as negative or positive. LR utilizes mathematical functions to establish relationships between dependent and independent variables, enabling the prediction of probabilities. These probabilities can then be converted into binary values for making further predictions. LR exhibits low variance, making it a reliable choice for classification tasks.

5.3.6. Multilayer Perceptron (MLP)

As its name suggests, it comprises more than one perceptron and can be regarded as a deep neural network. MLP classifier can be imported from sklearn.neural_network. In MLP, there is input, output, and hidden layer (any number of the hidden layer). The input layer receives input for, e.g., HMRs, the output layer produces a prediction of the class label from input, and the hidden layer contains the computation logic of MLP. MLP trains on HMRs and class label pairs and learns to correlate between input and output, that is, reviews and class labels. While training, weights and biases are also adjusted to minimize errors.

5.3.7. Hard Voting Ensemble

These are also called majority voting. In this, the mode of individually predicted labels by different MLMs is considered for the ultimate prediction will be the one that will get majority votes or more than half of the votes. In a voting classifier, voting by default is hard. All the classifier models applied in the present work were combined for majority voting.

5.4. Proposed Stacking Ensemble

Stacking represents a sophisticated ensemble technique that intelligently combines predictions from diverse base models to train a meta-model. Its most significant advantage lies in its adaptability, accommodating bagging, and boosting models within the ensemble. Stacking primarily aims to create a superior classifier by capitalizing on the individual strengths of multiple MLCs, effectively compensating for their individual biases and limitations. The fundamental idea behind stacking is to combine multiple weak or base learners to create a robust model with strong generalization capabilities. Stacking represents a fresh ensemble framework that employs meta-learners to blend the outcomes produced by each of these weak learners. Typically, the base learners, or weak learners, are referred to as first-level estimators, while the combinators, or meta learners, are known as second-level estimators. The underlying principle is that weak learners are trained using the initial training data. Then, the outputs from the weak learners at the first level are utilized as input features for the meta-estimator. Subsequently, a new dataset is constructed, incorporating the original sentiment polarity labels as the new labels to train the meta-estimator.

During the stacking process, each classifier plays a significant role, and their collective predictions act as input for the final estimator or meta-learner. This synergistic approach improves overall performance compared to using individual classifiers in isolation. Additionally, multilayer stacking can be implemented, involving creating several layers of weak learners before constructing the final meta-learner layer. However, it is essential to consider that incorporating numerous layers can introduce complexity to the model, necessitating a careful balance between model performance and complexity. In the proposed architecture, the first-layer and second-layer weak learners are trained using k-fold cross-validation. The step is as follows:

1. The HMR dataset HM is randomly divided into k sub-dataset {HM1, HM2, HM3,...., HMn}. Considering weak learner 1, for instance, each sub-set Subi (i =1, 2, ..., k) is verified separately, and the left-out k-1 subsets are used as training sets to obtain k prediction results. Merge into set SD1, which is the same length as HM.
2. The same operation is performed on other weak learners in the layer to obtain the SD2, SD3,..., and SDn sets.
3. The exact process is repeated from layer two estimators.
4. Merging the results of the predictions from n weak estimators, a new dataset SD = {SD1, SD2, ..., SDn} is obtained, which makes the input for the meta-learner estimator.
5. The meta-layer estimator can detect and correct the error in the prior layers and improve the overall classification performance of the model.

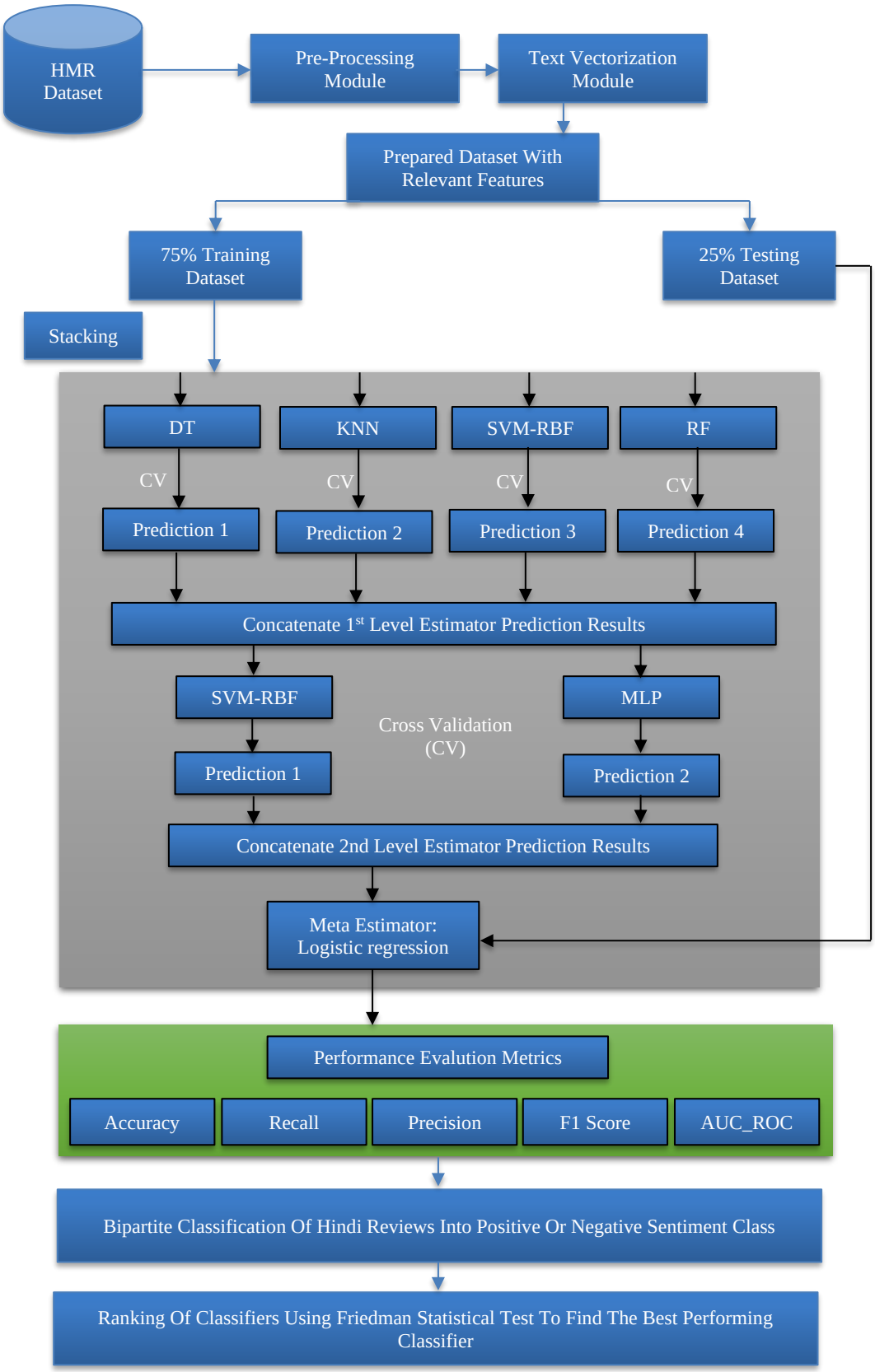

Fig. 2 Shows the proposed stacking-based architecture for bipartite SC of HMRs

This research introduces a two-stage stacked ensemble classifier with the aim of achieving improved generalization performance and prediction accuracy, given the benefits of heterogeneous ensembles. In the first classification stage, KNN, DT, SVM-RBF, and RF are employed as the base estimators or weak learners. Each of these four classifier models is trained using the entire training dataset. Subsequently, probabilistic outputs from the first stage are used as inputs in the second classification stage, which comprises SVM-RBF and MLP. Finally, the ensemble technique's output, representing sentiment polarity labels, is utilized to fit the meta-learner, which, in this case, is LR.

5.5. Statistical Tests

A statistical inferential analysis is conducted to assess the classification accuracy of several applied algorithms. The Friedman Test (FMT) is a non-parametric test specifically suited for comparing multiple classification algorithms. Compared to other non-parametric tests, FMT is more effective, making it suitable for ranking computation methods [5]. The hypothesis setup for FMT consists of the Null Hypothesis (H0) and the alternative hypothesis (H1). H0 states that all the classifiers applied are equal, while H1 proposes that the classifiers differ. By calculating the p-value through FMT, we can determine the significance of the classification results.

In hypothesis testing, the significance level, often set at 5% or 0.05, represents the probability of rejecting the null hypothesis when it is true. The researcher can confidently reject the null hypothesis if the study's results yield a probability lower than the significance level. Otherwise, the researcher may accept the null hypothesis if the probability is higher.To perform FMT, the Scipy.stats library offers the friedmanchisquare() function, simplifying the process of conducting this test. During the FMT analysis, each classification technique is individually ranked using Friedman's Rank (FIR). The classifier with the highest FIR value is considered the best performer, while the one with the lowest FIR is labelled the worst performer.

5.6. Performance Evaluation Metrics

The confusion matrix or contingency matrix gives the best interpretation of the performance of the classification algorithm by calculating the following parameters [2]:

- True Negative Rate (TNR): The classification model predicted a negative review, and the actual was also negative.
- True Positive Rate (TPR): The classification model predicted a positive review, and the actual was also positive.
- False Positive Rate (FPR): The classification model predicted a positive review, and the actual was negative.
- False Negative Rate (FNR): The classification model predicted a negative review, and the actual was positive.
- Accuracy. It is the ratio between the number of accurate sentiment class predictions to the total number of predictions.
- Precision. It is also called a positive predicted value. It is the ratio of the actual positive review observed to the total positive review predicted.
- Recall. It is also called True positive rate or sensitivity. The higher recall value denotes a more relevant result by the classifiers and relates to a low FNR rate.
- F1 score. It is also known as the F measure and is the harmonic mean among positive predicted value and true positive rate.
- AUC-ROC. It stands for the Area under curve - Receiver operating characteristics. It is an essential evaluation metric for checking classification performance, where AUC refers to the degree or measure of separability, and ROC is a type of probability curve. AUC denotes the capability of a model for distinguishing between positive and negative classes. ROC is plotted between TPR and FPR.

6. Results and Discussion

This section describes the experimental setup in detail, including datasets used, hardware and software specifications, empirical findings, and results. The Python programming language version 3.11.0 was utilized for implementing the current work on a computing device equipped with an Intel Core i7 processor and 16 GB of RAM. The experiments were conducted on an author-made HMR dataset, i.e., dataset -1, and the IIT-P movie review dataset i.e., dataset 2, was also employed for additional validation.

A test-to-train ratio of 25:75 was maintained throughout, and a 10-fold CV technique was utilized to ensure the soundness and validity of the results. This technique allows for the utilization of all reviews in the dataset for training and validation purposes. The results obtained from the 10-fold CV were averaged to generate the desired outcome. The paper focuses on the bipartite SC of movie reviews in Hindi using MLMs-based solutions. Initially, individual MLMs such as KNN, DT, RF, LR, MLP, and SVM-RBF were applied to both movie datasets.

Subsequently, a majority voting ensemble technique was employed on both datasets, followed by the application of our proposed stacking ensemble model. The performance of these models was compared using metrics such as Accuracy (Acc), Precision (Pre), Recall (Rec), F1 Score (F1S), and AUC-ROC score. Table 3 displays the evaluation results for dataset 1. SVM-RBF performed the best with an Acc of 81%, followed by LR with an Acc of 80%. As a discriminative classifier, SVM-RBF showed flexibility and power in binary classification for HMRs. Since HMRs had only two class labels, indicating a clear case of hard margin, SVM-RBF utilized a subset of training points known as support vectors.

Table 3. Evaluation results on HMRs – Dataset-1

MLM	Acc (%)	Pre (%)	Rec (%)	F1S (%)
KNN	0.67	0.70	0.67	0.67
DT	0.74	0.76	0.74	0.73
RF	0.78	0.78	0.78	0.78
LR	0.80	0.79	0.82	0.82
SVM	0.81	0.81	0.81	0.80
MLP	0.79	0.80	0.79	0.79
Hard Voting	0.54	0.51	0.54	0.51
Proposed	**0.83**	**0.82**	**0.83**	**0.82**

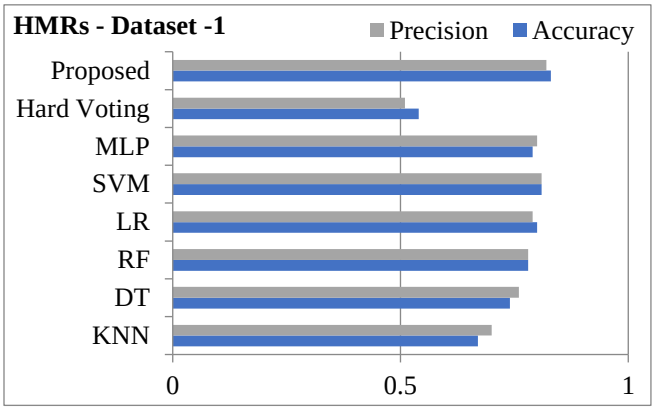

Fig. 3 Shows Accuracy and Precision evaluation results on dataset -1

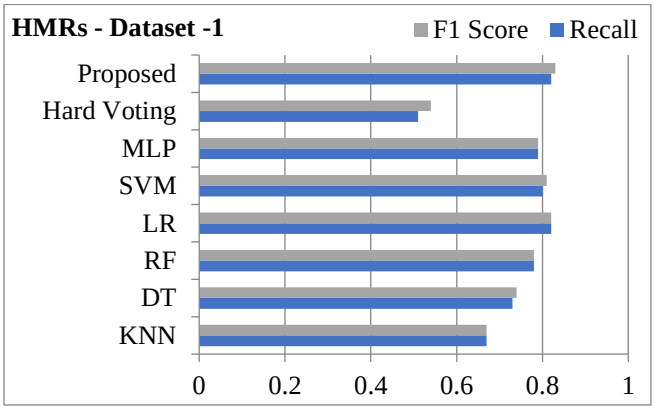

Fig. 4 Shows Recall and F1 score evaluation results on dataset -1

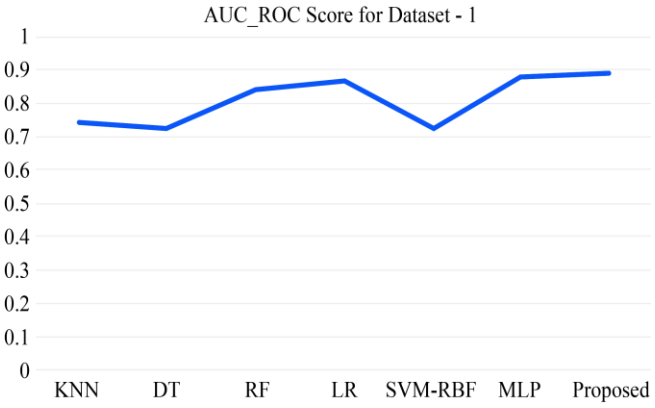

Fig. 5 Shows ROC_AUC based evaluation results on dataset-1

Table 4. Mean ranks were obtained by applying Friedman's Test on various evaluation metrics on dataset - 1

Classifier	Mean Rank
KNN	1.75
Hard Voting	1.75
DT	2.75
MLP	5.00
SVM-RBF	5.50
LR	5.75
RF	5.75
Proposed	**7.75**

Table 5. FMT ranks summary on dataset-1

Total N	4
Chi-Square	23.782
Degree of Freedom (df)	7
Asymptotic Significance	0.001

Table 6. Hypothesis test summary on dataset-1

Null Hypothesis	Test	Sig. a,b	Decision
The distributions of KNN, DT, RF, LR, SVM-RBF, MLP, HARD_VOTING and PROPOSED are the same.	Related-Samples Friedman's Two-Way Analysis of Variance by Ranks	.001	Reject the H0. Asymptotic Significance is displayed.

Table 7. Evaluation results on IIT-P movie review - dataset - 2

MLM	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
KNN	0.53	0.72	0.53	0.50
DT	0.71	0.71	0.71	0.68
RF	0.73	0.73	0.73	0.73
LR	0.73	0.72	0.73	0.72
SVM	0.73	0.74	0.73	0.70
MLP	0.73	0.72	0.73	0.72
Hard Voting	0.56	0.54	0.55	0.55
Proposed	**0.77**	**0.78**	**0.77**	**0.76**

This observation suggests that SVM performs well with Hindi text. MLP achieved the highest accuracy among the classifiers, leveraging its strong function approximation capabilities for classification and prognostication problems. MLP excels at learning complex non-linear functions with reasonable accuracy. The Acc, Pre, Rec, and F1S evaluation results for dataset 1 are presented in Figures 3 and 4. Referring to Figure 5, MLP obtained the highest AUC-ROC score of 0.88 for dataset 1, followed by LR with a score of 0.87. Similarly, for dataset 2 (refer to Table 7), SVM-RBF attained the highest accuracy of 73% among the individual classifiers. Regarding AUC_ROC scores for dataset 2 (refer to Figure 8), SVM-RBF had the highest score, followed by LR. AUC_ROC is a critical performance metric that reflects the classifier's ability to accurately predict positive and

negative Hindi reviews at various threshold settings. Despite this, the Acc obtained for the SC task of MRs in Hindi still needed to be improved. The hard voting ensemble technique was again applied.The MRs datasets were divided into 75:25, i.e., training and testing datasets for hard voting. KNN, DT, SVM-RBF, RF, LR, and MLP were trained on the training dataset, and P1, P2, P, P4, P5, and P6 represented the predictions made on the testing dataset by KNN, DT, SVM-RBF, RF, LR, and MLP, respectively.

The final prediction was based on the mode of individually predicted class labels by different MLMs. For dataset 1, an Acc of 54%, Pre of 51%, Rec of 54%, and F1S of 51% were obtained. Similarly, for dataset 2, an Acc of 56%, Pre of 54%, Rec of 55%, and F1S of 55% were achieved. However, the results still needed to be satisfactory. The next step involved applying the proposed stacking model to both datasets to evaluate its effectiveness.

The concept of stacking relies on combining multiple classification models at different levels to enhance predictive potentiality. Stacked models employ a layered approach to make predictions, with two-layered models being commonly used due to their complexity. The proposed stacking ensemble model, as illustrated in Figure 2, follows a two-layer architecture. In the first layer, estimators such as DT, KNN, SVM-RBF, and RF are utilized, and their predictions serve as features for the second-layer estimators. The second layer consists of SVM-RBF and MLP. SVM-RBF, being a widely used classifier, is employed in both layers based on the literature review. Introducing multiple classification techniques at each stage allows for the verification and validation of polarity class predictions for Hindi reviews generated by existing methods. Ultimately, the predictions from the second layer are fed to the final estimator, LR, which makes the ultimate movie review label prediction. Promising results were obtained for both movie review datasets.

The results obtained on various evaluation metrics for Dataset 1 and Dataset 2 are provided in Tables 3 and 7, respectively. It is evident that the proposed model outperformed all other classifiers and demonstrated improved performance for both datasets. The smaller size of Dataset 2 resulted in a slightly lower Acc compared to Dataset 1. Based on the research findings, they indicated that neither baseline classifiers nor a majority voting approach achieved satisfactory results independently. Consequently, the idea arose to amalgamate the advantages of both methods into the outlined architecture. It is noteworthy that no prior studies have tackled binary SA of Hindi MRs utilizing this proposed architecture. The unique characteristics of this architecture encompass heightened Acc, straightforward implementation, and enhanced performance. Moreover, it requires fewer computational resources and proves especially adept at addressing overfitting concerns, thereby bolstering the research findings.

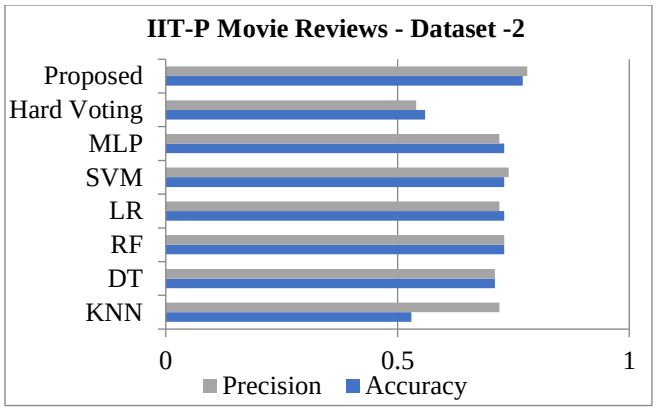

Fig. 6 Shows accuracy and precision evaluation results on dataset -2

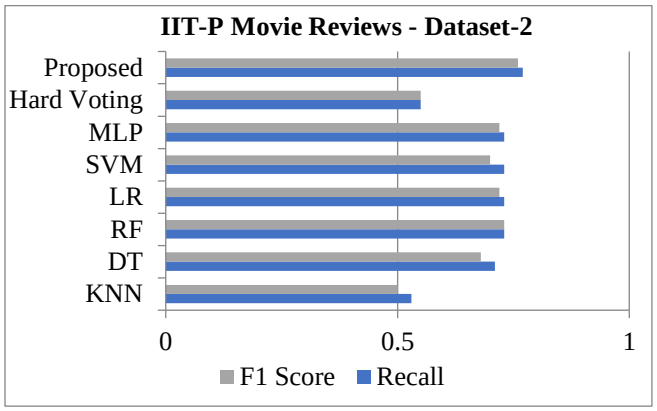

Fig.7 Shows F1score and recall evaluation results on dataset -2

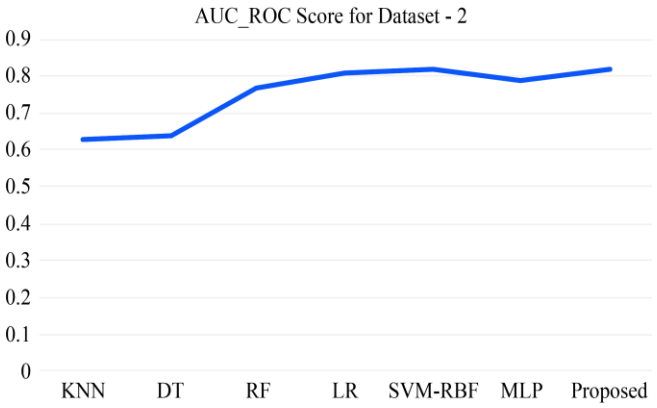

Fig.8 Shows ROC_AUC based evaluation results on dataset-2

Table 8. Mean rank obtained on applying riedman's test on various evaluation metrics on dataset - 2

Classifier	Mean Rank
KNN	2.00
Hard Voting	1.00
DT	3.00
MLP	5.25
SVM-RBF	6.50
LR	6.75
RF	4.00
Proposed	**7.50**

Table 9. FMT ranks summary on dataset – 2

Total N	4
Chi-Square	26.417
Degree of Freedom (df)	7
Asymptotic Significance	<.001

Table 10. Hypothesis test summary on dataset - 2

Null Hypothesis	Test	Sig. [a,b]	Decision
The distributions of KNN, DT, RF, LR, SVM-RBF, MLP, HARD_VOTING and PROPOSED are the same.	Related-Samples Friedman's Two-Way Analysis of Variance by Ranks	.001	Reject the H0. Asymptotic Significance is displayed.

When working with problems we aim to solve using MLAs, evaluating and comparing different classifiers is essential to determine which performs better. However, this task can be challenging since it is only sometimes obvious which classifier is superior. To address this issue and ascertain the effectiveness of our proposed architecture compared to individual classifiers applied, we conduct the Friedman Test (FMT). This statistical test allows us to determine whether there is a significant difference in performance between the classifiers or if they perform similarly. In our experiment, we obtained results for two different datasets. For dataset 1, our proposed architecture showed a 2% improvement in Acc compared to the best-performing individual classifier within that dataset group. Similarly, dataset 2 had a 4% Acc improvement with our proposed architecture compared to the best-performing individual classifier within that dataset group.

The FMT assesses these improvements and variations to determine if they are statistically significant, thus helping us draw meaningful conclusions about the effectiveness of the proposed architecture compared to other classifiers. Using this test, we can gain more confidence in our analysis and identify the best-performing approach for our problem. An FMT was conducted to evaluate the performance of applied MLMs and the proposed architecture for Hindi movie review text classification. The FMT is a non-parametric test used to determine statistical differences in classifier accuracy during sentiment classification of Hindi text. The FMT results for Dataset 1 are presented in Tables 5 and 6, while the results for Dataset 2 are shown in Tables 9 and 10. The results indicate that the asymptotic significance for both datasets was 0.001, which is lower than the commonly used significance level of 5% or 0.05. Thus, the null hypothesis is rejected, signifying a significant difference among the applied classifiers. The degrees of freedom (df) were seven, corresponding to the number of classification approaches used minus one.

Consequently, accepting the alternative hypothesis (H1), it can be concluded that a significant difference exists among the performance of the various classification algorithms and approaches utilized in this study. Furthermore, the individual ranking of each classification algorithm based on FMT's rank (FIR) reveals that our proposed stacking solution achieved the highest rank. At the same time, KNN and hard voting exhibited the lowest performance. This pattern is consistent for dataset 2, as shown in Table 8, where the proposed solution obtains the highest rank, and hard voting and KNN are the poorest performers. The superior rank achieved by the proposed stacking model in both datasets supports the validity of the obtained results. Notably, to the best of the author's knowledge, this is the first work to apply the non-parametric FMT for statistically evaluating the SC of Hindi MRs.

7. Conclusion and Future Works

The past decade has witnessed a remarkable surge in HT over the internet, driven by the development of the Unicode standard, making it an intriguing research domain that has captured the attention of researchers worldwide. Despite being an ancient language, SA in Hindi still needs more research. Recent studies have demonstrated that combining different classifiers yields superior classification performance and more generalized results on datasets than using individual classifiers alone. This has led to the adoption of ensemble classifiers for SC of HMRs. This study investigates various MLMs, including KNN, DT, RF, LR, MLP, and SVM-RBF, for bipartite polarity classification of sentiments expressed in HMRs and IIT-P movie review datasets.

Furthermore, hard voting of all classifiers deployed is applied to both datasets. Lastly, we proposed and used a stacking architecture for bipartite SC. Based on the empirical results obtained, our proposed stacked ensemble model outperforms all other classifiers and voting ensembles, demonstrating commendable performance on both datasets.

The application of FMT ranks the proposed stacked model highest in both datasets, thus rejecting the null hypothesis and indicating the superiority of the stacking ensemble approach. To the best of the author's knowledge, this is the first work to apply non-parametric FMT for statistically evaluating the SC of Hindi MRs. Additionally, the study highlights the limited availability of movie review datasets in Hindi, emphasizing the need for more extensive and diverse HMR datasets.

As part of future work, the authors aim to augment the HMR datasets by incorporating more reviews, including additional fine-grained emotion classes alongside positive and negative classes, and making them freely accessible for further research by fellow scholars. Moreover, the

researchers propose extending their work to encompass other resource-poor Indian languages beyond Hindi. The proposed model holds the potential for online deployment, facilitating the bipartite review classification of Hindi review text. This study will inspire and encourage further research in this field, driving advancements in SA and its application to HMRs.

Acknowledgements

The research conducted has received support from the Guru Gobind Singh Indraprastha University through the Indraprastha Research Fellowship (IPRF), as indicated by the award letter with reference number GGSIPU/DRC/2022/1247.

References

[1] Dhanashree S. Kulkarni, and Sunil S. Rodd, "Sentiment Analysis in Hindi—A Survey on the State-of-the-Art Techniques," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 21, no. 1, pp. 1-46, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Katarzyna Stąpor, "Evaluation of Classifiers: Current Methods and Future Research Directions," *Annals of Computer Science and Information*, vol. 12, pp. 37-40, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[3] Abdalsamad Keramatfar, and Hossein Amirkhani, "Bibliometrics of Sentiment Analysis Literature," *Journal of Information Science*, vol. 45, no. 1, pp. 3-15, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[4] Gazi Imtiyaz Ahmad, and Jimmy Singla, "Machine Learning Techniques for Sentiment Analysis of Indian Languages," *International Journal of Recent Technology and Engineering*, vol. 8, no. 11, pp. 3630-3636, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[5] Anshul Bhatia, Anuradha Chug, and Amit Prakash Singh, "Statistical Analysis of Machine Learning Techniques for Predicting Powdery Mildew Disease in Tomato Plants," *International Journal of Intelligent Engineering Informatics*, vol. 9, no. 1, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[6] Sujata Rani, and Parteek Kumar, "A Journey of Indian Languages over Sentiment Analysis: A Systematic Review," *Artificial Intelligence Review*, vol. 52, pp. 1415-1462, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[7] Aditya Joshi et al., "Towards Sub-Word Level Compositions for Sentiment Analysis of Hindi-English Code-Mixed Text," *Proceedings of COLING the 26th International Conference on Computational Linguistics: Technical Papers*, Osaka, Japan, pp. 2482-2491, 2016. [Google Scholar] [Publisher Link]

[8] Md Shad Akhtar, Asif Ekbal, and Pushpak Bhattacharyya, "Aspect Based Sentiment Analysis in Hindi: Resource Creation and Evaluation," *Proceedings of the 10th International Conference on Language Resources and Evaluation*, Portorož, Slovenia, pp. 2703-2709, 2016. [Google Scholar] [Publisher Link]

[9] Sandhya Singh et al., "Comparing Recurrent and Convolutional Architectures for English-Hindi Neural Machine Translation," *Proceedings of the 4th Workshop on Asian Translation*, Taipei, Taiwan, pp. 167-170, 2017. [Google Scholar] [Publisher Link]

[10] Pradeepika Verma, Sukomal Pal, and Hari Om, "A Comparative Analysis on Hindi and English Extractive Text Summarization," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 18, no. 3, pp. 1-39, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[11] Lior Rokach, "Ensemble-Based Classifiers," *Artificial Intelligence Review*, vol. 33, pp. 1-39, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[12] Youwei Wang, Jiangchun Liu, and Lizhou Feng, "Text Length Considered Adaptive Bagging Ensemble Learning Algorithm for Text Classification," *Multimedia Tools Applications*, vol. 82, pp. 27681-27706, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[13] M. Thangaraj, and M. Sivakami, "Text Classification Techniques: A Literature Review," *Interdisciplinary Journal of Information, Knowledge, and Management*, vol. 13, pp. 117-135, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[14] Charu Nanda, Mohit Dua, and Garima Nanda, "Sentiment Analysis of Movie Reviews in Hindi Language Using Machine Learning," *2018 International Conference on Communication and Signal Processing*, Chennai, India, pp. 1069-1072, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[15] Vandana Jha et al., "HOMS: Hindi Opinion Mining System," *2015 IEEE 2nd International Conference on Recent Trends in Information Systems*, Kolkata, India, pp. 366-371, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[16] Deebha Mumtaz, and Bindiya Ahuja, "Sentiment Analysis of Movie Review Data Using Senti-Lexicon Algorithm," *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology*, Bangalore, India, pp. 592-597, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[17] Marta Galvao, and Roberto Henriques, "Forecasting Model of a Movie's Profitability," *2018 13th Iberian Conference on Information Systems and Technologies*, Caceres, Spain, pp. 1-6, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[18] Vandana Jha et al., "Sentiment Analysis in a Resource Scarce Language: Hindi," *International Journal of Scientific and Engineering Research*, vol. 7, no. 9, pp. 968-980, 2016. [Google Scholar] [Publisher Link]

[19] Arundeep Kaur, and A.P. Nidhi, "Predicting Movie Success Using Neural Network," *International Journal of Science and Research*, vol. 2, no. 9, pp. 69-71, 2013. [Google Scholar] [Publisher Link]

[20] Nahid Quader, Md. Osman Gani, and Dipankar Chaki, "Performance Evaluation of Seven Machine Learning Classification Techniques

for Movie Box Office Success Prediction," *2017 3rd International Conference on Electrical Information and Communication Technology*, Khulna, Bangladesh, pp. 1-6, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[21] Ashutosh Kanitkar, "Bollywood Movie Success Prediction Using Machine Learning Algorithms," *2018 3rd International Conference on Circuits, Control, Communication and Computing*, Bangalore, India, pp. 1-4, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[22] Travis Ginmu Rhee, and Farhana Zulkernine, "Predicting Movie Box Office Profitability: A Neural Network Approach," *2016 15th IEEE International Conference on Machine Learning and Applications*, Anaheim, CA, USA, pp. 665-670, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[23] Konstantinas Korovkinas, Paulius Danenas, and Gintautas Garsva, "SVM and Naïve Bayes Classification Ensemble Method for Sentiment Analysis," *Baltic Journal of Modern Computing*, vol. 5, no. 4, pp. 398-409, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[24] Anuj Sharma, and Shubhamoy Dey, "A Boosted SVM based Ensemble Classifier for Sentiment Analysis of Online Reviews," *ACM SIGAPP Applied Computing Review*, vol. 13, no. 4, pp. 43-52, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[25] Savita Sangam, and Subhash Shinde, "Sentiment Classification of Social Media Reviews Using an Ensemble Classifier," *Indonesian Journal Electrical Engineering Computer Science*, vol. 16, no. 1, pp. 355-363, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[26] Amira M. Gaber, Mohamed Nour El-Din, and Hanan Moussa, "SMAD: Text Classification of Arabic Social Media Dataset for News Sources," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 10, pp. 1-9, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[27] Shalini Puri, and Satya Prakash Singh, "An Efficient Hindi Text Classification Model Using SVM," *Computing and Network Sustainability*, pp. 227-237, 2019. [CrossRef] [GoogleScholar] [Publishers link]

[28] Md Shad Akhtar et al., "A Hybrid Deep Learning Architecture for Sentiment Analysis," *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, Osaka, Japan, pp. 482-493, 2016. [Google Scholar] [Publisher Link]

[29] Grégoire Mesnil et al., "Ensemble of Generative and Discriminative Techniques for Sentiment Analysis of Movie Reviews," *Arxiv*, pp. 1-5, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[30] Muhammad Usman et al., "Urdu Text Classification Using Majority Voting," *International Journal of Advanced Computer Science and Applications (IJCSA)*, vol. 7, no. 8, pp. 265-273, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[31] K. Sarkar, "Heterogeneous Classifier Ensemble for Sentiment Analysis of Bengali and Hindi Tweets," *Sādhanā*, vol. 45, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[32] Nikunj C. Oza, and Kagan Tumer, "Classifier Ensembles: Select Real-World Applications," *Information Fusion*, vol. 9, no. 1, pp. 4-20, 2008. [CrossRef] [Google Scholar] [Publisher Link]