Sentence Modelling with Convolutional Neural Networks for Enhancing Natural Language Understanding: A Comprehensive Exploration

Harsh Kumar Saha[1], Priyanka Dubey[2], Vibhor Srivastava[3]

[1,2,3]Department of Computer Science, Amity University Haryana, Gurgaon

Abstract

The paper presents a novel approach to implementing Convolutional Neural Networks (CNN) for Semantic Sentence Modelling, a significant advancement in Natural Language Processing. This model categorizes sentiments in text as Positive, Negative, or Neutral, improving the accuracy and efficiency of Sentiment Analysis Systems. Its applications include Opinion Mining, Paraphrase Detection, and Discourse Analysis. The paper emphasizes the need for sophisticated tools to understand and analyse sentiments in various languages, highlighting the potential of dedicated CNN models.

Keywords: Convolutional Neural Network (CNN), Natural Language Processing, Lexicons, Deep Belief Network (DBN), BERT (Bidirectional Encoder Representation from Transformers), Lemmatization, Hyperparameter, Word Sense Disambiguation (WSD), Corpus, Tokenization.

Introduction

During this digital age, Internet has surfaced as the main source of information and communication for users worldwide, communicating their views and feelings across various platforms. The increase in user data has led to rise of Sentiment Analysis, a sub-field of Natural Language Processing. It has gained global acceptance not just among researchers but also amongst the business groups, governments, and corporations who develop it to monitor public opinion and helps in policymaking.[3,5]

The paper addresses the challenges faced during Sentiment Analysis by executing a specialized and dedicated Convolutional Neural Network (CNN) specifically for Semantic Sentence Modelling. This approach is expected to increase accuracy and efficiency of Sentiment Analysis System by providing more detailed and segmented insights in the various applications.[5]

It delves into various tools and techniques, applications and algorithms being used while conducting Sentiment Analysis which showcases the comparative data for easy understanding. Explores the most frequently used techniques for Sentiment Analysis with various machine learning, transformer learning, and lexicon-based approaches.[11] The purpose is to conduct a broader exploration on Sentiment Analysis through various viewpoints, including various research components related to Sentiment Analysis.

The main aim is to contribute through the ongoing research in Sentiment Analysis and carry forward further studies in the same direction, stressing on importance of more advanced tools and techniques to better understand and analyse sentiments in multiple languages. The introduction of the dedicated CNN model point towards a huge step forward in Sentiment Analysis capable of improving its performance.[1]

The paper provides an inclusive overview of contemporary opinion over mining literature, covering on how to extract text features from opinions in noise or uncertainty, characterize data of opinions, and classify them. Here we examine various techniques for adaptive characteristic-based Lexicons for

Sentiment Classification and how a dynamic lexicon can be automated for updating and provide more precise grading for context-related concepts.[4]

Furthermore, the paper explores the challenges associated with the system for different languages. It discusses how the meaning and orientation of many words can change depending on the context and domain in which they are engaged, making Sentiment Analysis a complex task. It also highlights the need for more sophisticated tools and techniques to better understand and analyse sentiments.[7]

The comprehensive introduction does the detailed search and analysis that follows in to provide a clear roadmap for the reader and outlining the key topics that will be covered and the arguments that will be made.

Literature Review

Sentiment analysis is an important task in natural language processing (NLP) and has attracted widespread attention due to its diverse applications in various fields, such as social media monitoring, market intelligence, and customer feedback analysis.[1]

This section details the theoretical foundations, techniques, and challenges associated with sentiment analysis:

1. **Vocabulary-based approach:** Vocabulary-based sentiment analysis is based on a sentiment vocabulary or dictionary that contains a list of words that are interpreted in their corresponding sentiment categories (positive, negative, neutral, etc)[1]

   These lexicons are often hand-curated or generated using automated methods.[8]

   The algorithm then assigns an emotional rating to the document based on the occurrence and distribution of the words in the dictionary.[1]

   Although lexical-based approaches are simple, they can suffer from language differences, ambiguity, and context dependence.
2. **Rule-based systems:** Rule-based sentiment analysis systems use predefined linguistic rules to identify sentiment patterns and expressions in text[1]. These rules may include syntactic structures, grammatical patterns, or semantic rules that apply specifically to the expression of emotions.[1] Rule-based systems offer greater flexibility and interpretability compared to dictionary-based approaches. However, it requires extensive manual rule development and may not translate well to different domains or languages.[1]
3. **Supervised learning methods:** Supervised learning algorithms such as support vector machines (SVMs), Naive Bayes, and neural networks divide text into sentiment categories (positive, negative, neutral, etc) based on labelled training data.[1] Learn how to categorize. These models use features extracted from the text, such as word frequencies, N-grams, and word embeddings, to make predictions.[8] Supervised learning techniques have shown promising results in sentiment analysis tasks, especially when trained on large-scale annotated datasets.[1]
4. **Unsupervised Learning Approaches:** Unsupervised learning techniques, including clustering algorithms such as K-means and topic modelling techniques such as Latent Dirichlet Allocation (LDA), can identify sentiment patterns and topics inherent in unlabelled text data.[1] The purpose is to discover these approaches are useful for exploratory analysis and identifying latent sentiment structures within large datasets.[1]
5. **Deep Learning Paradigms:** Recent advances in sentiment analysis are based on deep learning techniques, including recurrent neural networks (RNNs), convolutional neural networks (CNNs), and

transformer architectures such as BERT (Bidirectional Encoder Representation from Transformers). It has been driven by learning technology. These models excel at capturing complex dependencies and contextual information within text, resulting in state-of-the-art performance on sentiment analysis tasks.

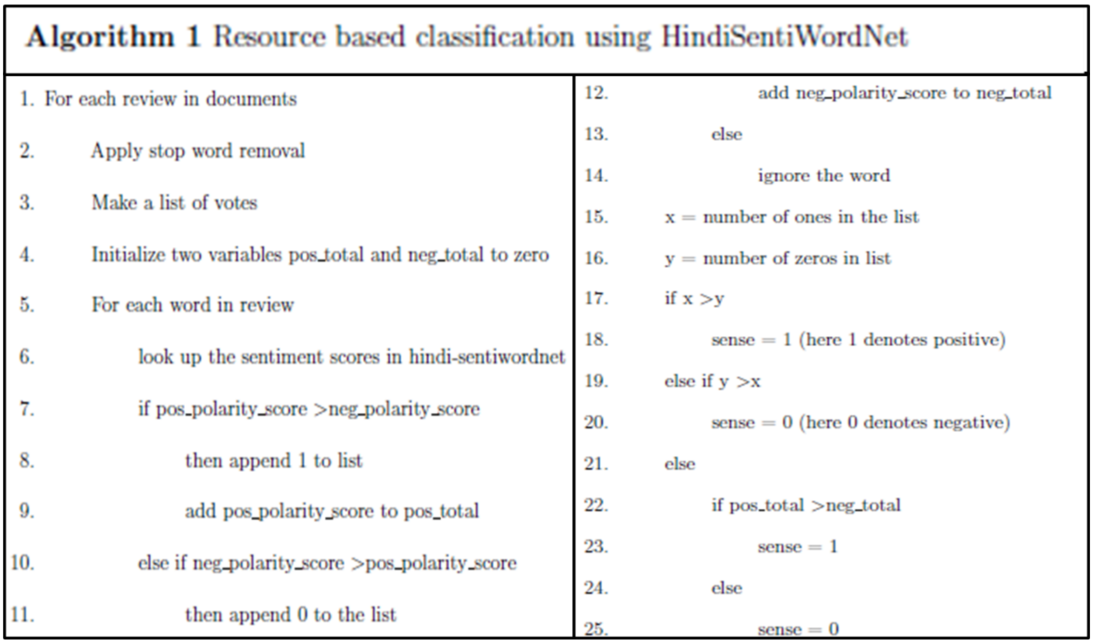
6. **Multilingual and Multilingual Sentiment Analysis:** Distributing multilingual content over the Internet requires the development of robust multilingual and multilingual sentiment analysis methods.[1] Techniques such as cross-linguistic emotion transfer learning, language adaptation, and code-switching detection have been developed to address the challenges of sentiment analysis in various linguistic contexts.[1]
7. **Challenges and Future Directions:** Despite advances in sentiment analysis, several challenges remain, such as sarcasm detection, emotional ambiguity, and fine-grained sentiment analysis.

   Addressing these challenges requires interdisciplinary research efforts that integrate linguistic insights, domain knowledge, and advanced machine learning techniques.[7] Future research directions in sentiment analysis include exploring multimodal sentiment analysis, dynamic sentiment analysis, and domain-specific sentiment analysis tailored to specific applications and industries.[5] Sentiment analysis is a complex task involving a wide range of methods and techniques. By leveraging the theoretical foundations and advances in NLP, machine learning, and deep learning, sentiment analysis has great potential to derive valuable insights from text data in a variety of domains and languages.[1]

Methodology

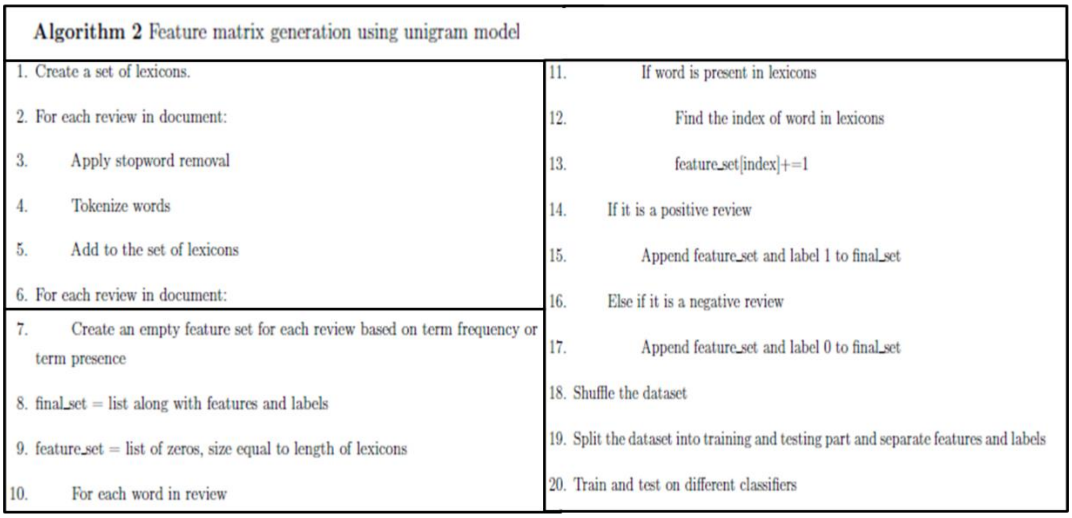
In this section, we examined the dataset and models used for review pre-processing, the algorithms utilized for feature set development, and the various classifiers used.

1. **Dataset:** The dataset used in this study is made up of textual data gathered from various platforms, including social media, online reviews, and pertinent textual sources.[11] It includes a balanced distribution of positive and negative sentiment-labelled samples to ensure a thorough sentiment analysis.[11]
2. **Data Pre-processing:** Before sentiment analysis, the raw textual data is pre-processed to assure consistency and improve analysis accuracy.[6] The pre-processing includes:
   a. Tokenization: Separating text into distinct tokens or words.
   b. Eliminating common stop words that do not convey substantial mood.
   c. Lemmatization is the process of reducing words to their basic form to increase vocabulary and generalization.
3. **Resource-Based Classification:** This method includes using external sentiment lexicons or dictionaries to classify text based on the presence or absence of sentiment-bearing terms.[1] This method uses predefined lists of positive and negative terms to provide sentiment scores to papers.



Algorithm 1 Resource based classification using HindiSentiWordNet

```
For each review in documents
    Apply stop word removal
    Make a list of votes
    Initialize two variables pos_total and neg_total to zero
    For each word in review
        look up the sentiment scores in hindi-sentiwordnet
        if pos_polarity_score >neg_polarity_score
            then append 1 to list
            add pos_polarity_score to pos_total
        else if neg_polarity_score >pos_polarity_score
            then append 0 to the list
            add neg_polarity_score to neg_total
        else
            ignore the word
    x = number of ones in the list
    y = number of zeros in list
    if x >y
        sense = 1 (here 1 denotes positive)
    else if y >x
        sense = 0 (here 0 denotes negative)
    else
        if pos_total >neg_total
            sense = 1
        else
            sense = 0
```



Algorithm 2 Feature matrix generation using unigram model

```
Create a set of lexicons.
For each review in document:
    Apply stopword removal
    Tokenize words
    Add to the set of lexicons
For each review in document:
    Create an empty feature set for each review based on term frequency or term presence
final_set = list along with features and labels
feature_set = list of zeros, size equal to length of lexicons
    For each word in review
        If word is present in lexicons
            Find the index of word in lexicons
            feature_set[index]+=1
    If it is a positive review
        Append feature_set and label 1 to final_set
    Else if it is a negative review
        Append feature_set and label 0 to final_set
Shuffle the dataset
Split the dataset into training and testing part and separate features and labels
Train and test on different classifiers
```

4. **In-language classification:** In-language classification focuses on building machine learning models for sentiment analysis using the dataset's language.[9] This strategy entails creating classification models utilizing labelled data in the target language to accurately classify sentiment.[10]
5. **Machine Translation-based Semantic Analysis:** Machine translation-based semantic analysis seeks to improve sentiment analysis by translating material into a shared language for analysis. This method entails converting text data into a standardized language using machine translation algorithms before performing sentiment analysis.[10]
6. **Feature Matrix Generation:** Feature matrix generation entails converting pre-processed textual data into numerical feature vectors for use in machine learning models.[12] Two popular methods for feature matrix generation are:
   a. The TF-IDF Algorithm assigns weight to terms based on their frequency in the document and across the dataset.[2]
   b. The Unigram Model represents text data by using single words as features, without considering word order.[9]
7. **Classification:** Different classification methods are used for sentiment analysis, such as:

a. Deep Neural Network (DNN) is a model with numerous layers that can learn complex patterns from data.

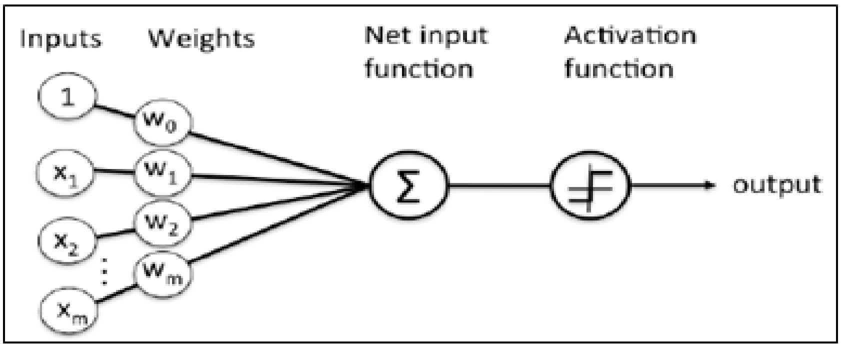


Figure 1: how a single node looks like a Neural Network

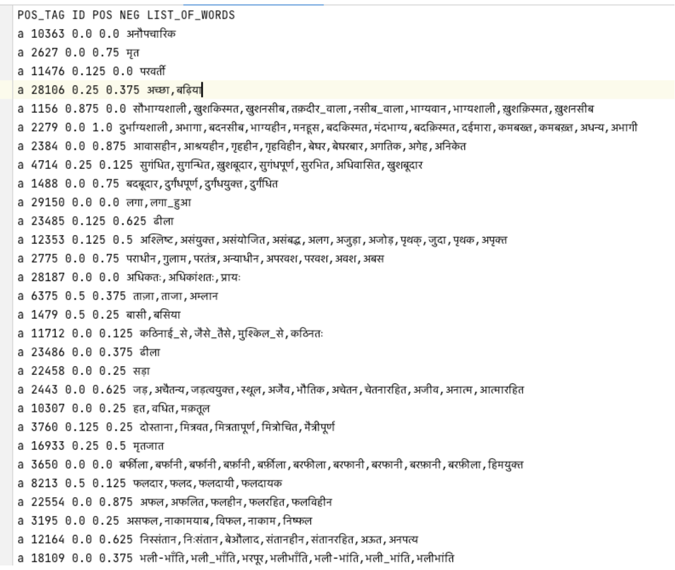
The Deep Belief Network (DBN) is a probabilistic model that learns data hierarchies.

A. Naive Bayes is a probabilistic classifier that applies Bayes' theorem and assumes feature. independence.[2]
B. Logistic Regression is a linear model used for binary classification tasks.[8]
C. Support Vector Machine (SVM) is a supervised learning technique that uses hyperplanes t separate classes in high-dimensional space.[8]
D. A Decision Tree is a tree-like model with nodes representing feature-based decisions.[8]

Experimental Setup

The experimental setup describes the steps and configurations used to do sentiment analysis with the given code. This section covers data preparation, model training, assessment measures, and any other procedures used for analysis.

1. **Data Preparation:** The first stage in the experimental setup is preparing the dataset for sentiment analysis. This entails gathering relevant textual data from a variety of platforms and ensuring a fair distribution of positive and negative sentiment-labelled samples. The dataset is pre-processed to clean and standardize the text, which includes tokenization, stop word removal, and lemmatization.



```
POS_TAG ID POS NEG LIST_OF_WORDS
a 10363 0.0 0.0 अनौपचारिक
a 2627 0.0 0.75 मृत
a 11476 0.125 0.0 परवर्ती
a 28106 0.25 0.375 अच्छा,बढ़िया
a 1156 0.875 0.0 सौभाग्यशाली,खुशकिस्मत,खुशनसीब,तक़दीर_वाला,नसीब_वाला,भाग्यवान,भाग्यशाली,ख़ुशक़िस्मत,ख़ुशनसीब
a 2279 0.0 1.0 दुर्भाग्यशाली,अभागा,बदनसीब,भाग्यहीन,मनहूस,बदकिस्मत,मंदभाग्य,बदक़िस्मत,दईमारा,कमबख्त,कमबख़्त,अधन्य,अभागी
a 2384 0.0 0.875 आवासहीन,आश्रयहीन,गृहहीन,गृहविहीन,बेघर,बेघरबार,अगतिक,अगेह,अनिकेत
a 4714 0.25 0.125 सुगंधित,सुगन्धित,ख़ुशबूदार,सुगंधपूर्ण,सुरभित,अधिवासित,खुशबूदार
a 1488 0.0 0.75 बदबूदार,दुर्गंधपूर्ण,दुर्गंधयुक्त,दुर्गंधित
a 29150 0.0 0.0 लगा,लगा_हुआ
a 23485 0.125 0.625 ढीला
a 12353 0.125 0.5 अश्लिष्ट,असंयुक्त,असंयोजित,असंबद्ध,अलग,अजुड़ा,अजोड़,पृथक्,जुदा,पृथक,अपृक्त
a 2775 0.0 0.75 पराधीन,गुलाम,परतंत्र,अन्याधीन,अपरवश,परवश,अवश,अबस
a 28187 0.0 0.0 अधिकतः,अधिकांशतः,प्रायः
a 6375 0.5 0.375 ताज़ा,ताजा,अम्लान
a 1479 0.5 0.25 बासी,बसिया
a 11712 0.0 0.125 कठिनाई_से,जैसे_तैसे,मुश्किल_से,कठिनतः
a 23486 0.0 0.375 ढीला
a 22458 0.0 0.25 सड़ा
a 2443 0.0 0.625 जड़,अचैतन्य,जड़त्वयुक्त,स्थूल,अजैव,भौतिक,अचेतन,चेतनारहित,अजीव,अनात्म,आत्मारहित
a 10307 0.0 0.25 हत,वधित,मक़तूल
a 3760 0.125 0.25 दोस्ताना,मित्रवत,मित्रतापूर्ण,मित्रोचित,मैत्रीपूर्ण
a 16933 0.25 0.5 मृतजात
a 3650 0.0 0.0 बर्फीला,बर्फानी,बर्फानी,बर्फ़ानी,बर्फ़ीला,बरफीला,बरफानी,बरफानी,बरफ़ानी,बरफ़ीला,हिमयुक्त
a 8213 0.5 0.125 फलदार,फलद,फलदायी,फलदायक
a 22554 0.0 0.875 अफल,अफलित,फलहीन,फलरहित,फलविहीन
a 3195 0.0 0.25 असफल,नाकामयाब,विफल,नाकाम,निष्फल
a 12164 0.0 0.625 निस्संतान,निःसंतान,बेऔलाद,संतानहीन,संतानरहित,अऊत,अनपत्य
a 18109 0.0 0.375 भली-भाँति,भली_भाँति,भरपूर,भलीभाँति,भली-भांति,भली_भांति,भलीभांति
```

Figure 2: Data Prepared

2. **Model Training:** Once the dataset has been pre-processed, the sentiment analysis model is trained using supervised learning methods. The pre-processed textual data is converted into feature vectors using techniques like TF-IDF or unigram representation. To predict sentiment labels, a classification model, such as the Deep Belief Network (DBN) or other algorithms, is trained using feature vectors.
3. **Evaluation metrics:** The sentiment analysis model's performance is assessed using a range of metrics. These measurements include accuracy, precision, recall, and the F1-score, which indicate the model's ability to reliably identify sentiment labels. To ensure an unbiased performance assessment, the evaluation employs a distinct test dataset.
4. **Hyperparameter Tuning:** Hyperparameter tuning improves the performance of sentiment analysis models. This requires gradually changing hyperparameters such as learning rate, batch size, and network architecture to discover the best configuration. Hyperparameter tweaking can be done using grid or random search methods.
5. **Cross-validation:** To assure the sentiment analysis model's resilience, cross-validation techniques are used. This entails dividing the dataset into numerous folds and training the model using various combinations of training and validation sets. Cross-validation evaluates the model's performance across multiple subsets of data and reduces overfitting.

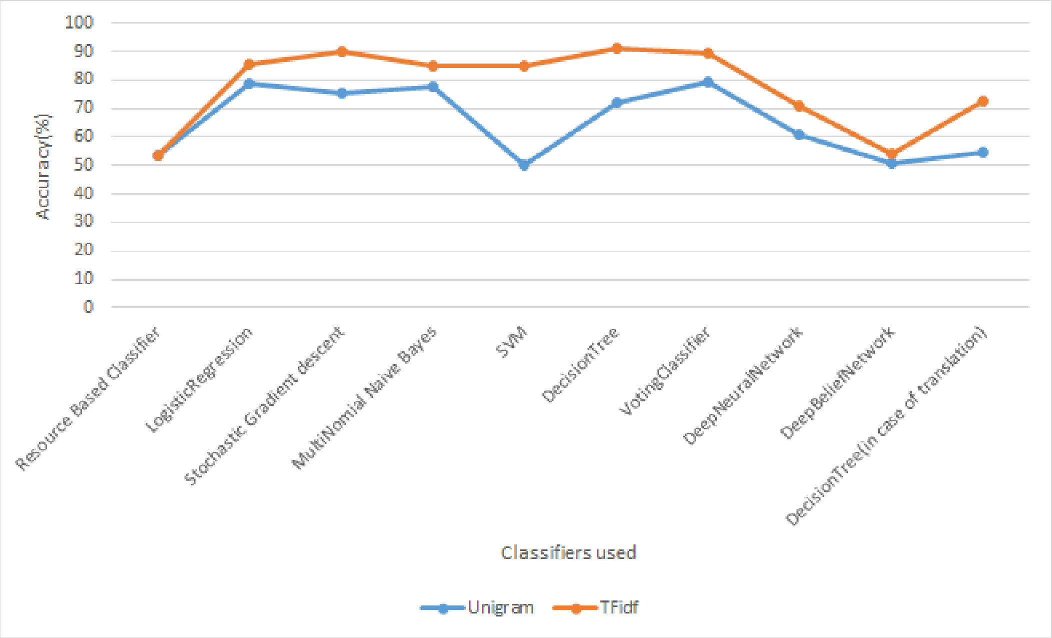


Figure 3: Comparison of accuracy by different classifiers

6. **Results Analysis:** After model training and assessment, the sentiment analysis experiments are examined. The performance measures, which are represented by graphs or charts, are analysed to determine the success of the sentiment analysis model. The findings from the analysis are presented in light of the study objectives and consequences.

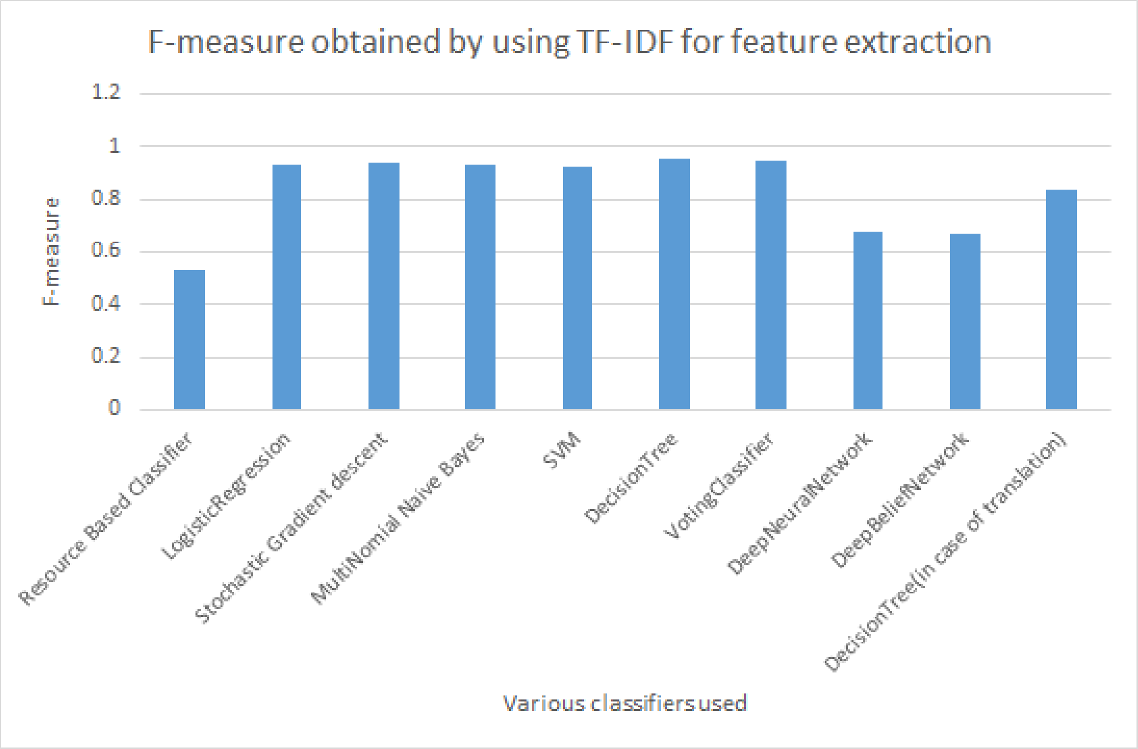


Figure 4: F-measure graph for different approaches with different models

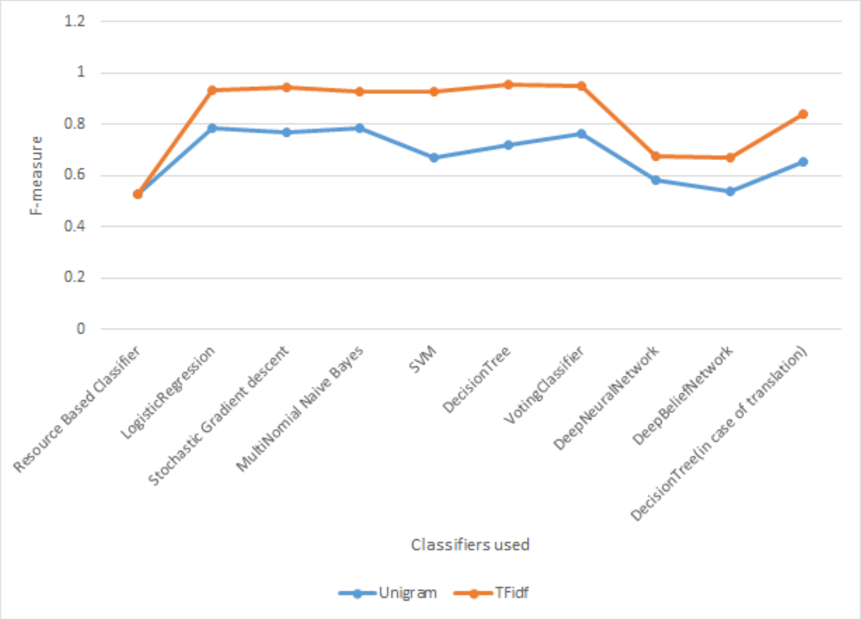


Figure 5: Comparison of F-measure obtained using different models by different classifiers.

Conclusion and Future Work

The study looked into three different methods for sentiment analysis in Hindi text. Our first strategy used a majority-based classifier trained on Hindi SentiWordNet,[1] whereas the second entailed building a model on an annotated English corpus and translating Hindi documents into English for analysis.[2] Finally, our third strategy focused on developing a Hindi-specific classifier model utilizing the same training corpus.[3] Our findings indisputably proved the superiority of the third technique, emphasizing the need of using an annotated corpus in the original language to achieve optimal sentiment analysis performance.[3] Furthermore, in the third approach, TF-IDF outperformed the unigram model, highlighting the importance of adopting proper approaches for accurate sentiment analysis.[9]

Looking ahead, our findings indicate various areas for additional investigation and improvement of sentiment analysis algorithms in Hindi literature. One interesting strategy is to include Word Sense Disambiguation (WSD) methods with resource-based sentiment analysis to refine and increase accuracy.[10] Furthermore, improving Hindi SentiWordNet's coverage by enriching the lexicon with more terms and nuanced sentiment annotations may result in increased sentiment analysis accuracy and depth.[9] Furthermore, introducing negation rules into our models might improve their capacity to detect sentiment subtleties in complicated language situations, hence increasing overall efficacy.

Furthermore, our findings highlight the usefulness of language-specific techniques to sentiment analysis, particularly in resource-constrained languages such as Hindi. We can produce more accurate and nuanced sentiment analysis findings by combining native language resources with specialized approaches. These findings open the door for improved sentiment analysis techniques and a better comprehension of human emotions and attitudes in multilingual settings.

References

1. "Neural Networks Overview.": https://deeplearning4j.org/neuralnet-overviewdefine.
2. "Restricted Boltzmann Machine.": https://en.wikipedia.org/wiki/RestrictedBoltzmannmachine.
3. Arora, P. (2013). "Sentiment Analysis for Hindi Language." MS by Research in Computer Science.
4. Bakliwal, A., Arora, P., & Varma, V. (2012). "Hindi Subjective Lexicon: A Lexical Resource for Hindi Polarity Classification." In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC), pp. 1189-1196.
5. Bansal, N., Ahmed, U. Z., & Mukherjee, A. (2013). "Sentiment Analysis in Hindi." Department of Computer Science and Engineering, Indian Institute of Technology, Kanpur, India, 1-10.
6. Joshi, A., Balamurali, A., & Bhattacharyya, P. (2010). "A Fall-back Strategy for Sentiment Analysis in Hindi: A Case Study." Proceedings of the 8th ICON.
7. Mittal, N., Agarwal, B., Chouhan, G., Pareek, P., & Bania, N. (2013). "Discourse Based Sentiment Analysis for Hindi Reviews." In International Conference on Pattern Recognition and Machine Intelligence, Springer, pp. 720-725.
8. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). "Scikit-learn: Machine Learning in Python." Journal of Machine Learning Research, 12, 2825-2830.
9. Dhanashree Gajanan Kulkarni, & Sunil F. Rodd. (2022). Word Sense Disambiguation for Lexicon-based Sentiment Analysis in Hindi. Webology, 19(1), 592–600.
10. Dhanashree Gajanan Kulkarni, & Sunil F. Rodd. (2022). Word Sense Disambiguation for Lexicon-based Sentiment Analysis in Hindi. Webology, 19(1), 592–600.
11. Kush Shrivastava, & Shishir Kumar. (2020). A Sentiment Analysis System for the Hindi Language by Integrating Gated Recurrent Unit with Genetic Algorithm. The International Arab Journal of Information Technology, 17, 954–964.
12. Mohammed Arshad Ansari. (2019). SENTIMENT ANALYSIS OF MIXED CODE FOR THE TRANSLITERATED HINDI AND MARATHI TEXTS. Zenodo (Cern European Organization for Nuclear Research).