



ANALYSIS AND CLASSIFICATION OF HINDI TEXT USING NATURAL LANGUAGE PROCESSING TECHNIQUES

Thesis submitted for the award of the degree of

Doctor of Philosophy

Submitted By

**Dhanashree Sagar Kulkarni
USN: 2GI17PEA02**

Research Centre

Gogte Institute of Technology, Belagavi

Under the Guidance of

**Prof. Dr. S. F. Rodd
Professor, Department of CSE,
Gogte Institute of Technology, Belagavi**

June 2022

**Department of CSE, Gogte Institute of Technology Research Centre,
Belagavi, Karnataka, 590008**

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI



GOGTE INSTITUTE OF TECHNOLOGY
RESEARCH CENTER
UDYAMBAG, BELGAUM – 590008

Certificate

This is to certify that the research work which is being presented in the thesis entitled **“ANALYSIS AND CLASSIFICATION OF HINDI TEXT USING NATURAL LANGUAGE PROCESSING TECHNIQUES”** in partial fulfillment of the requirements for the award of the Degree of Doctor of Philosophy is carried out by **Mrs. Dhanashree Sagar Kulkarni** during a period from March 2017 to May 2022 and is submitted to Department of Computer Science and Engineering, Gogte Institute of Technology, Research Center of Visvesvaraya Technological University, Belagavi, Karnataka, 590018. The work satisfies the academic requirements in respect of the research work for the said degree.

Signature of the Guide

Signature of the HOD

Head of The Department
Computer Science & Engineering
Gogte Institute of Technology
Belagavi 590 008, India

Signature of the Principal

PRINCIPAL
Karnatak Law Society's
Gogte Institute of Technology
Udyambag, BELAGAVI-590008





CANDIDATE'S DECLARATION


I hereby certify that the work which is being presented in the thesis entitled "**ANALYSIS AND CLASSIFICATION OF HINDI TEXT USING NATURAL LANGUAGE PROCESSING TECHNIQUES**" in partial fulfillment of the requirements for the award of the Degree of Doctor of Philosophy and submitted to Department of Computer Science and Engineering, Gogte Institute of Technology, Research Center of Visvesvaraya Technological University, Belagavi, Karnataka, 590018 is an authentic record of my own work carried out during a period from March 2017 to May 2022 under the supervision of **Dr. S. F. Rodd**, Professor, Department of Computer Science and Engineering, Gogte Institute of Technology, Belagavi, Karnataka, 590008, India.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other Institute / University.

Candidate's Signature


(Mrs. DHANASHREE SAGAR KULKARNI)

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.


Guide's Signature
(Dr. S. F. RODD)

Date: 13/07/2022

The Ph.D. Viva-Voce Examination of **Mrs. DHANASHREE SAGAR KULKARNI**, Research Scholar, has been held on.....

Signature of Supervisor (s)

Signature of External Examiner

**© VISVESVARAYA TECHNOLOGICAL UNIVERSITY,
BELAGAVI, INDIA 2022
ALL RIGHTS RESERVED**

Acknowledgements

First and foremost, praises and thanks to Shri Gajanan, the almighty for showering his blessings on me and providing me the strength to complete this work. I take this opportunity to acknowledge the support and help extended by my well-wishers, guide and family.

I express my deepest gratitude to my Guide Prof. Dr. Sunil F. Rodd for his constant support and encouragement that has helped me take this work to an acceptable level. I am really honored to have a supervisor like him, who is immensely concerned about the progress of the work. He responded to the queries and questions promptly. His untiring guidance was instrumental in getting the work completed. He advised broader research areas, insisted on exhaustive survey of the literature and fine-tuning the research gaps. His fruitful suggestions, constructive recommendations, knowledge and patience kept me going during difficult times.

I would like to thank Principal of Gogte Institute of Technology Dr. J. K. Kittur and Dr. Vijay Rajpurohit, Head of CSE dept for giving me an opportunity to work at GIT research Centre. I wish to thank Prof. Dr. Santosh Deshpande, VTU, Belagavi and Prof. Dr. Rashmi Jogdand for critical review of my work during progress review meetings and giving valuable suggestions that made the work to progress in the right direction. I would like to thank Dr. S. S. Sannakki and all faculty members of CSE department, GIT for their support during my visits to GIT.

I sincerely thank Dr. Spoorti Patil, Director of SAEF and Dr Anand Deshpande, Principal and Director, Angadi Institute of Technology and Management, Belagavi for their support. I also wish to thank all colleagues at CSE Department of AITM for support and encouragement.

A heartfelt thankyou to my husband, my parents, my in-laws and my adorable son who have provided me immense moral support. I could not have completed this work without their help and support.

Finally, I thank all who were instrumental and supportive to me in carrying out the research work.

DHANASHREE SAGAR KULKARNI

ABSTRACT

The progression of Internet has given rise to blogs, forums and social networking platforms which produce a large amount of user information on regular basis. Mining this user-generated data and trying to find out likings, interests, opinions and sentiments of the user is turning out to be fascinating for researchers. The introduction of Unicode standards in the recent years has resulted into development of webpages in Indian Languages especially Hindi. Hindi language is ranked as the fourth most spoken language in the world. The number of users and contributors on social media in Hindi language is surging. There is a need to adequately mine the Hindi data that is generated and perform analysis so that it could be beneficial especially to Government organizations and product-based companies. Sentiment Analysis (SA) is an evolving area of research in the arena of natural language processing (NLP) and is mainly concerned with analyzing and classifying feelings and emotions from a given text. Sentiment analysis in Indian Languages is one of the major challenges in the area of NLP. When the task of performing sentiment analysis on an Indian language is considered, it is very difficult. There are different challenges that need to be addressed before analyzing the opinions and categorizing them into various classifications. Firstly, the data on the web is unstructured data collected from different sources and many-a-times tends to be noisy as users have a habit of using abbreviations, emoticons, slangs and spelling variations. Also, there are comparatively less tools, annotated corpora and resources in Hindi Language that can be used as a part of Sentiment analysis which makes the task more tough.

Lexicon based approaches, machine-learning approaches and deep learning methods are being used to achieve SA in Hindi language. The major concern in Lexicon based approaches is its inefficiency. Word sense ambiguity, Morphological variations, spelling variations, Negations are some of the challenges of Natural Language Processing that are the reasons resulting for inefficiency in sentiment analysis applications and need to be effectively addressed so as to enhance the performance of the lexicon-based approaches. In comparison, Machine Learning approaches are generally more accurate than lexicon-based approaches but require a large labelled training dataset. Also, machine learning approaches perform differently on various domains. Taking into consideration the several pros and cons of both the approaches, exploring the possibility of combining Lexicon and Machine Learning approaches and evaluating the Hybrid approach for sentiment analysis in Hindi

can turn out to be beneficial. Further, the effect of linguistic properties of Hindi text such as length of the message (wordcount) and size of training dataset on the classification accuracy needs to be significantly investigated.

This research aims at proposing an enhanced Lexicon model and a hybrid model for performing sentiment analysis in Hindi Language. The simple lexicon-based approach for Hindi language makes use of Hindi SentiWordNet but suffers from various limitations like absence of lexicons, word sense ambiguity, morphological variations, low accuracy due to presence of diminishers and negators etc. An enhanced lexicon-based algorithm is proposed which performs morphological handling, provides a Graph based Lesk approach for solving the word sense ambiguity problem and enhances the accuracy of the sentiment analysis system by proposing a rule-based method for handling intensifiers, diminishers, conjunctions and negators. Further a hybrid model is built by combining the enhanced lexicon model along with machine learning approach and its performance is evaluated for efficiency and effectiveness. The built hybrid method is tested on various datasets as well as for multiclass classification and its effectiveness is assessed by considering performance parameters such as Accuracy, Recall, Precision, and also F1-score. The effect of linguistic properties of Hindi text mainly word-count and size of training dataset on accuracy is investigated and its statistical significance is presented.

Table of Contents

Acknowledgements	i
ABSTRACT	ii
List of Abbreviations	vii
List of Figures.....	ix
List of Tables	xi
Chapter 1 Introduction and Statement of the problem.....	1
1.1 Introduction	1
1.2 Sentiment Analysis	2
1.3 Need of Sentiment Analysis in Indian languages	2
1.4 Sentiment Analysis in Hindi Language	3
1.5 Challenges in Hindi Sentiment Analysis	4
1.6 Levels of Sentiment Analysis	8
1.7 Statement of the problem.....	8
1.8 Organization of the Thesis.....	9
Chapter 2 Literature Survey.....	11
2.1 Introduction	11
2.2 Lexicon Approach	13
2.3 Machine Learning Approach	16
2.3.1 Feature extraction methods in Machine Learning	20
2.4 Deep Learning Technique	22
2.5 Hybrid Approach	26
2.6 Levels of Analysis	27
2.7 Addressing the Negation Problem.....	28
2.8 Addressing other issues	30
2.9 Research gaps identified	31
2.10 Summary.....	33
Chapter 3 Investigating the Lexicon based Technique	35

3.1 Introduction	35
3.2 Resources Used.....	36
3.2.1 Hindi SentiWordNet (HSWN)	36
3.2.2 Hindi WordNet	36
3.2.3 Movie Review Dataset	36
3.2.4 Multidomain Dataset	37
3.2.5 Evaluation measures	37
3.3 Simple Lexicon HSWN algorithm	38
3.4 Comparison with Machine Learning Classifiers	40
3.4.1 Comparison of Simple Lexicon with Machine Learning classifiers	46
3.5 Summary.....	48
Chapter 4 Enhancing the Lexicon-based Sentiment Analysis Method.....	49
4.1 Introduction	49
4.2 Morphological Handling.....	49
4.3 Word Sense Disambiguation	51
4.4 Conjunction Handling.....	57
4.5 Modifier Handling	58
4.6 Summary.....	62
Chapter 5 Building a Hybrid Model and evaluating its efficiency and effectiveness	63
5.1 Introduction	63
5.2 Proposed Hybrid Model.....	64
5.2.1 Building the Hybrid Model	64
5.2.2 Checking Effectiveness of the hybrid model	65
5.2.3 Results of Single domain dataset.....	65
5.2.4 Results of Multi domain dataset	66
5.3 Hybrid Approach for Emotion Classification.....	68
5.3.1 BHAAV dataset.....	68
5.3.2 Creation of Emotion Lexicon	69

5.3.3 Time complexity of Hybrid model	73
5.4 Summary.....	75
Chapter 6	76
Investigating the impact of length of messages on sentiment classification accuracy	76
6.1 Impact of Wordcount.....	76
6.2 Statistical Significance	83
6.3 Summary.....	84
Chapter 7 Conclusion and Scope for Future Work.....	86
7.1 Conclusion	86
7.2 Scope for future work.....	87
Appendix A Basics of Python.....	89
Features of Python	89
Appendix B	91
References.....	93
PAPER PUBLICATIONS OUT OF RESEARCH WORK	101
SCOPUS INDEXED JOURNALS	101
Conferences	101

List of Abbreviations

Acronym	Description
SA	Sentiment Analysis
NLP	Natural Language Processing
SVO	Subject Verb Object
POS	Part-of-Speech
LTRC	Language Technologies Research Centre
HSWN	Hindi SentiWordNet
AI	Artificial Intelligence
SVM	Support Vector Machines
MNB	Multinomial Naïve Bayes
DT	Decision Tree classifier
RF	Random Forest classifier
NN	Neural Networks
ME	Maximum Entropy
HSL	Hindi Subjective Lexicon
CSPL	Context Specific Polarity Lexicon
PMI	Pointwise Mutual Information
SAIL	Sentiment Analysis in Indian Languages
HOMS	Hindi Opinion Mining System
CRF	Conditional Random Fields
TF-IDF	Term frequency -Inverse Document Frequency
TF-ICF	Term Frequency-Inverse Corpus Frequency
RBFNN	Radial Basis Function Neural Network
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Networks
LSTM	Long Short-Term Memory Networks
GPU	Graphics Processing Unit
SMT	Statistical Machine Translation
HEOT	Hindi English Offensive Tweet
MOO	Multi Objective Optimization
CBOW	Continuous bag of words
SG	Skip-Gram

WSD	Word sense Dismbiguation
NER	Named Entity Recognition
GBMLA	Graph-based Modified Lesk Approach
NRC	National Research Council

List of Figures

Figure. 1.1 Levels of Sentiment Analysis.....	8
Figure. 2.1 Techniques used in Hindi Sentiment Analysis.....	11
Figure. 2.2 Concept of Machine Learning.....	17
Figure. 2.3 Machine Learning and Deep Learning flow	22
Figure. 2.4 A simple Convolutional Neural Network	24
Figure. 2.5 A simple Recurrent Neural Network.....	24
Figure. 3.1 Lexicon based Technique.....	35
Figure. 3.2 A Simple Lexicon Algorithm.....	38
Figure. 3.3 Output of using Simple Lexicon HSWN for Movie review dataset	39
Figure. 3.4 Output of using Simple Lexicon HSWN for Multidomain dataset.....	39
Figure. 3.5 Steps in building a Machine Learning model	40
Figure. 3.6 Decision Trees.....	41
Figure. 3.7 Random Forest Classifier.....	44
Figure. 3.8 Accuracy Comparison of Simple HSWN approach with various ML Classifiers on 1200 movie review dataset	46
Figure. 3.9 Accuracy Comparison of Simple HSWN approach with various ML Classifiers on 4000 movie review dataset	46
Figure. 4.1 HSWN specific morphological Handling	49
Figure. 4.2 Comparison of HSWN specific morphological Handling with Simple HSWN	50
Figure. 4.3 Comparison of WSD Results: First Sense and Averaging.....	51
Figure. 4.4 Comparison of WSD Results: First Sense and Averaging.....	52
Figure. 4.5 Example of Graph based WSD.....	53
Figure. 4.7 Comparison of GBMLA with first sense and averaging techniques.....	55
Figure. 4.8 Output of Enhanced Lexicon Model	55
Figure. 4.9 Conjunction Handling	56
Figure. 4.10 Rule based Algorithm for handling Modifiers	58

Figure. 4.11 Final Enhanced Lexicon Algorithm.....	59
Figure. 4.12 Simple Lexicon vs Enhanced Lexicon Model.....	60
Figure. 5.1 Proposed Hybrid Model	63
Figure. 5.2 Accuracy obtained for unclassified reviews for single domain dataset...65	
Figure. 5.3 Evaluating various classifiers as a part of Hybrid model for single domain dataset.....	66
Figure. 5.4 Accuracy obtained for unclassified reviews for multidomain dataset....66	
Figure. 5.5 Evaluating various classifiers as a part of Hybrid model for multidomain dataset.....	6
6Figure.5.6 Accuracy obtained for unclassified reviews for BHAAV dataset.....	71
Figure.5.7 Evaluation of classifiers for Hybrid model for BHAAV dataset.....	71
Figure.5.8 Training Dataset Size Vs Time taken for training.....	73
Figure.6.1 Steps and Approaches used for investigation of impact of wordcount	76
Figure.6.2 Effect of wordcount on Accuracy of enhanced Lexicon model for Single Domain dataset.....	77
Figure.6.3 Effect of wordcount on Recall, Precision and F-score of enhanced Lexicon model for Single Domain dataset.....	78
Figure.6.4 Effect of wordcount on Accuracy of enhanced Lexicon model for Multidomain dataset.....	78
Figure.6.5 Effect of wordcount on Recall, Precision and F-score of enhanced Lexicon model for Multidomain dataset.....	79
Figure.6.6 Effect of wordcount on Accuracy of Hybrid Model for Single domain dataset.....	80.
Figure.6.7 Effect of wordcount on Recall, Precision and F-score of Hybrid Model for Single domain dataset.....	80
Figure.6.8 Effect of wordcount on Accuracy of Hybrid Model for Multidomain dataset.....	81
Figure.6.9 Effect of wordcount on Recall, Precision and F-score of Hybrid Model for Multidomain dataset.....	81

List of Tables

Table 1.1 Variations in Word Order	5
Table 3.1 Accuracy Comparison results of Classifiers	46
Table 4.1 Comparison of Simple Lexicon and Enhanced Lexicon model	61
Table 5.1 Sentences in BHAAV dataset	69
Table 5.2 Performance of BHAAV dataset in [78].....	69
Table 5.3 Total number of Lexicons in the built Emotion Lexicon.....	70
Table 5.4 Comparison of results of BHAAV dataset in [78] with Hybrid model	73
Table 5.5 Comparison of Multinomial Naïve Bayes and Random Forest classifier for all three datasets	75
Table 6.1 Results of the statistical significance	84

Chapter 1

Introduction and Statement of the problem

1.1 Introduction

The technological developments that have happened in the recent years enabled people to connect with one another by means of social networking. The evolution of Internet has given rise to blogs, forums and social networking platforms which generate a large amount of user information on regular basis[1]. Facebook, Twitter, Google are some of the social platforms which have millions of users and they generate Terabytes of data per day.

Mining the large amount of user-generated data and trying to find out patterns, interests, opinions and sentiments of the user is turning out to be fascinating for researchers. Users tend to give feedbacks, reviews and recommendations on different products. Product managers are always interested in understanding customer emotions and psychology of customers so that they can make appropriate changes to the product or processes and enhance the value of the brand and attain customer satisfaction. When customers give product feedbacks and opinions on blogs, review channel and social platforms, mining this web generated data helps different companies and organizations to establish and maintain their profitability and reputation.

Opinionating text has contributed to an emerging research area in the domain of text analytics which is referred to as Opinion mining or Sentiment analysis. Extracting relevant opinions from the web generated information is a tough task. The process of mining information from different forms of text such as blogs, reviews, tweets etc. and classifying them under different classes such as positive class, negative class and neutral class based on the polarity is referred to as Sentiment Analysis.

The following sections describe and explain some of the concepts which form the basis of discussion and help to make the further chapters easier to understand.

1.2 Sentiment Analysis

Sentiment Analysis (SA) is an emerging area of research in the area of natural language processing (NLP) which deals with analyzing and classifying feelings and emotions from given text. It is sometimes referred to as emotion mining and it incorporates different techniques related to text analytics and computational linguistics so as to extract relevant polarity and subjective information. (Source: Wikipedia). Sentiments mainly relate to emotions pertaining to something that showcases what a person feels.

SA can be applied to various situations and can be used for a variety of reasons. The commonly used SA application is ecommerce. Customers can register their purchasing and product quality experiences on internet sites and also give ratings or comments on the website. These customer reviews and recommendations for products can be viewed by people who want to buy the product. Reviews and feedbacks regarding a product form an important component for a customer and helps in the identification of a new product invention potential and to meet product functional needs. Another application of SA is Brand Reputation Management which manages a firm's brand image in the market. It allows businesses to better maintain and develop their company reputation. Advertising, public opinion and other factors influence it. Sentiment analysis assists in making the customer's choice of a company's brand, goods, or services. Sentiment classification relating to public opinion can also be used to evaluate government performance.

Analyzing sentiments and performing accurate sentiment classification is a challenging task. Recent research is directed towards multiclass classifications and fine-grained sentiment analysis.

1.3 Need of Sentiment Analysis in Indian languages

India is called a country with diversity. It is called a multilingual country where people speak 22 different official languages belonging to various families such as Indo-Aryan, Dravidian, Nishada, Kirta etc. Majority of the people speak Indo Aryan languages like Hindi, Marathi, Konkani, Urdu, Bengali etc. while rest of them speak Dravidian languages like Kannada, Telugu, Tamil, Malayalam etc. Santali is the official language of Nishada and Manipuri and Bodo are languages concerning Kirta or Sino-Tibetan family [1].

As a result of the Indian government's Digital India project, there was a tremendous expansion in usage of Internet and India turned out to be one amongst the largest online internet market. India also witnessed a new generation of users using the Internet who chose

to use Internet in their native language. A study by KPMG, India and Google was done in 2017 which stated that there were around 23 crores internet users who preferred to use internet in Indian languages whereas internet users of English language were around 17 crores [2].

A large number of Search Engines, Social Networking Platforms, Apps, and majority of Government Websites are now accessible in Indian Languages, resulting in a massive amount of data being transmitted over the Internet. This has permitted researchers to investigate the NLP study field for Indian languages and has thus presented Sentiment Analysis (SA) on Indian Languages as one of the prominent research areas of NLP.

1.4 Sentiment Analysis in Hindi Language

Internet has evolved and has seen a noteworthy progress by development of websites, portals, blogs, forums and discussion groups. People are very much interested in writing their opinions and giving their feedbacks on various categories such as products, restaurants, books, movies, events etc. In recent years, it has been observed that people are sharing their opinions in the language in which they seem to be more comfortable. In most of the cases, it is their native language. Hindi forms the native language of majority of people in India.

Hindi language is one amongst the two official languages of India and is spoken by almost about 80 crores of people in India. The 2011 census recorded that 57.09% of the total population in India speak Hindi. (Source: Simple English Wikipedia).

The introduction of Unicode standards has resulted into development of various Hindi websites. Some of the well-known websites that provides the content in Hindi and which have been accessed by users are bhaskar.com, jagran.com, hinkhoj.com, webdunia.com, raftaar.in, acchhibatein.com, hindividya.com etc. Along with these websites, the number of blogs, forums and reviews where users communicate in Hindi are also increasing rapidly. Analyzing the views expressed by the users on websites, blogs and forums and mining the sentiments can be of sheer importance for decision making and reputation management. While analyzing the text, the nature of the text that is under consideration, can be either subjective wherein the text can hold an opinion or it can be objective in which case the text may not have any opinion. A sentence. such as,

“मिशन मंगल एक बहुत अच्छी फिल्म है “

[Mission Mangal is a very good movie]

Gives an opinion about the movie Mission Mangal. So, it is a subjective sentence whereas “बारिश हो रही है”

[It's raining]

is an objective sentence which does not imply any sentiment as to how does the person feel when it is raining. It just expresses some general information and does not give any view about the opinion of the person. Hence the sentence is termed as objective.

In case of subjective text, there are 3 main classifications that can be considered for the text namely positive, negative or neutral. The sentence mentioned above “मिशन मंगल एक बहुत अच्छी फिल्म है” gives a positive opinion of the movie whereas a sentence like “इस मोबाइल की बैटरी जल्दी खराब होती है.”

[The battery of this mobile gets depleted very quickly]

portrays a negative opinion about the mobile. Neutral statements are the ones which have a subject but do not portray whether it is positive sentiment or a negative sentiment.

In case of discussion forums, depending on the views and sentiments expressed by the users, it is possible to get a sense of the subject that is being discussed. Thus, performing Sentiment analysis for Hindi language is aimed at categorizing text based on sentiments and emotions that each entity will invoke.

Hindi is resource scarce language due to which Sentiment Analysis for Hindi has its unique set of challenges and issues that needs to be addressed carefully. Generally, the number of tools, annotated corpora, and other resource materials for scarce resource languages is limited, or are still in development. Hence SA in Hindi is not only interesting but also very challenging.

1.5 Challenges in Hindi Sentiment Analysis

Performing sentiment analysis considering Indian languages is usually difficult. There are different challenges that need to be addressed before analyzing the opinions and categorizing them into various classifications. Firstly, the data on the web is unstructured data collected from different sources and comprising of various formats such as text, image, audio etc. Secondly, the data tends to be noisy as users have a habit of using abbreviations, emoticons, slangs and spelling variations. This makes the analysis more difficult. Some of the challenges that need to be dealt with while performing SA in Hindi Language are as follows:

Free Word Order: Every sentential formation in a language includes subject (S), verb(V) and object (O). English language has a fixed SVO pattern. But in case of Hindi language, it has a free word order. There is as such no fixed pattern for Subject, verb and object as shown in Table 1.1. The order of words and their occurrence contribute immensely in sentiment classification.

Table 1.1 Variations in Word Order

Hindi Sentence	Word Order
मैं एक प्रयोग कर रहा हूँ I'm doing an experiment	Subject Object Verb (SOV)
सीता ने खाया सेब Sita ate apple	Subject Verb Object (SVO)
राजेश को रोहित ने पीटा Rajesh was beaten up by Rohit	Object Subject Verb (OSV)

Insufficient Resources: Hindi is a language which is scarce in terms of resources when dealing with problem of analyzing sentiments. There are very less tools, resources and annotated datasets which can be used for analysis. There are as such very less Part-of-Speech (POS) taggers, stemmers and morphological analyzers and the ones which are available are not that efficient as compared to English languages. This makes the sentiment analysis task much more difficult.

Word Sense Disambiguation: There are some words in Hindi language whose meaning and polarity may change depending on the context in which it is being used. Word sense disambiguation deals with words that have multiple meanings and tries to sense the appropriate meaning of the word depending on its usage. In the below given example both sentences use the word सोने but the context in which it is used is totally different.

Eg: सोने का भाव बढ़ गया है

[Gold price has increased]

वह सोने की कोशिश कर रहा था लेकिन नींद नहीं आ रही थी

[He was trying to sleep but he couldn't]

Morphological and Spelling Variations: Hindi being a morphologically rich language, there exists many variations of a word which have the same root word. For example, जाएगी,

जा रहा है, जा रही है, जाएगा (she will go, he is going, she is going, he will go) are all variations of the root word जा (go). Further, the same word can be written with different spellings which makes it more difficult for analysis. For example, चाहंगा and चाहंगा are both valid spellings.

Types of Sentences: To perform Sentiment analysis in Hindi accurately, there is a need to identify different types of sentences that occur in Hindi language like comparative sentences, conditional sentences, sarcastic sentences or ironic sentences, sentences containing idioms etc. अगर, जब तक, केवल, परंतु, लेकिन (if, till then, only, but) are some of the conditional connectives which can occur and tend to change the polarity of the sentences. “Sarcasm” is somewhat related to taunting or giving an unpleasant expression or remark through irony. To handle sarcastic sentences, more semantic and contextual information would be required. Hindi language also allows use of paired words such as कच्चा-पक्का (soft hard), छेड़-छाड़ (tease) in sentences that are totally contradictory in terms of their meaning as well as polarity thus making sentiment analysis much more complex.

Sentiment Analysis research in case of English language has been going on for more than a decade now. There are effective resources available which are making the task much easier. But in case of Hindi Language, the same does not hold true. There are comparatively less resources in Hindi language and so it is termed as a resource scarce language. Some of the resources that help in analysis of Hindi text are:

Lexical resources –SentiWordNet is one type of lexical resource which mainly involves words along with their synset-ids and their respective scores. Lexical resources are very crucial in taking the decisions of polarity during sentiment analysis. Hindi SentiWordNet and Hindi Subjective Lexical are the two main lexical resources which are available.

Linguistic resources –Preprocessing of data is a major part of text mining for which different linguistic resources are required like Part-of-Speech (POS) tagger, Morphological Analyzer, Shallow Parser, Stemmer, etc. POS tagger is generally utilized for labelling every term or word with its corresponding part-of-speech such as adjective, verb, noun, adverb, etc. Stemmer is a tool which is used during the preprocessing to convert or reduce the terms to their base or root form. Few POS-taggers and Stemmers are available which are used in

preprocessing of data for Hindi language. Occasionally a Morphological Analyzer may be utilized which along with the base form, also describes features such as singular or plural form of the term, gender etc. A linguistic resource called Shallow Parser may also be used which for a given sentence, tries to achieve the morphological structure, tokenization, POS tagging etc. LTRC Shallow Parser is a commonly used parser in case of Hindi language.

Datasets –A dataset which is well annotated is the elementary requirement for performing sentiment analysis. Researchers mostly tend to develop the datasets on their own by mining the information from different sources such as websites, reviews, blogs, discussion forums etc. and then performing the annotation by taking help of some experts of that language. Some of the investigators use translation services such as Google Translate, Shabdanjali , Shabdkosh etc. on existing English datasets, translate it into Hindi and verify it from the language experts. Many researchers have also used English resources like WordNet, Subjectivity Word list, OpenOffice thesaurus, English-Hindi WordNet linking etc. in creation of the datasets.

There are some well-known sites which are used by researchers for mining the information and develop datasets. Some of them are listed below:

<https://hindi.webdunia.com/>

<http://dir.hinkhoj.com/>

<https://www.bbc.com/hindi>

<https://www.jagran.com/>

<http://www.virarjun.com/>

<https://www.raftaar.in/>

Though there are resources available for Hindi language, they are inadequate and they do not match the efficiency of resources available in English language. There is a significant amount of improvisation that needs to be done in terms of resources if analysis has to be carried out efficiently.

1.6 Levels of Sentiment Analysis

There are three levels at which SA can be accomplished namely Aspect Level, Sentence Level and Document Level as shown in Figure 1.

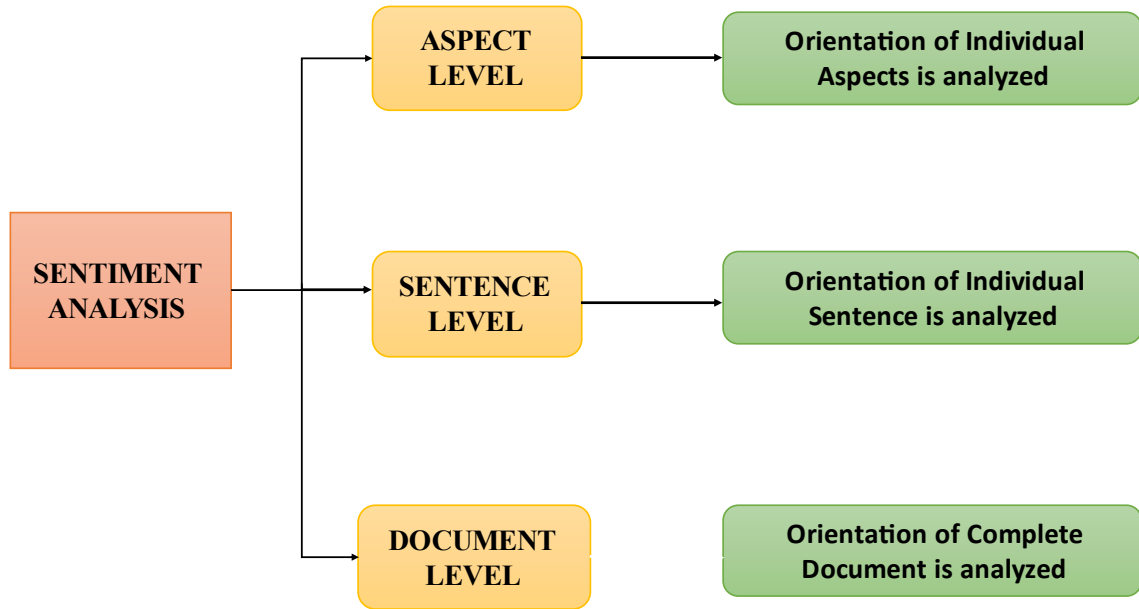


Figure 1.1: Levels of Sentiment Analysis

In case of Document level Sentiment analysis, the opinions which are stated as a part of the complete document are classified either as positive opinion or negative opinion. Sentence Level Sentiment analysis is similarly a higher-level analysis wherein a sentence is considered and its polarity is determined. When analysis at sentential level is performed, handling diverse forms of sentences such as conditional sentences, sarcastic sentences, comparative sentences etc. and then obtaining the polarity turns out to be a difficult task. Aspect Level analysis is fine grained level analysis where the aspects for which the opinion has been given is extracted and then the polarity of that specific aspect is identified. A substantial amount of research in sentiment analysis has been accomplished at document level and at sentence level.

1.7 Statement of the problem

Sentiment analysis in Indian Languages is one of the major challenges in the area of NLP. There are different approaches that are being used for Hindi sentiment analysis and Lexicon approach is one of them. The major concern in Lexicon based approaches is its inefficiency.

Word sense ambiguity, Morphological variations, spelling variations, Negations are some of the challenges of NLP that are the reasons resulting in inefficiency of sentiment analysis applications and need to be effectively addressed so as to improve the performance of the lexicon-based approaches. In comparison, Machine Learning approaches are generally more accurate than lexicon-based approaches but they require a large labelled training dataset. Also, machine learning approaches perform differently on data of various domains whereas, Lexicon approach shows consistent performance when considered across different domains. Taking into consideration the various pros and cons of both the approaches, exploring the possibility of combining Lexicon and Machine Learning methods and evaluating the Hybrid method for Hindi Sentiment analysis can turn out to be beneficial. The advantages of building a hybrid system have not been explored substantially for sentiment analysis in Hindi. Even, the effect of linguistic properties of Hindi text mainly the length of the message (word-count) on the classification performance needs to be significantly investigated.

In this context, the problem may be stated as:

To develop an enhanced lexicon-based hybrid model for sentiment analysis in Hindi that produces highly accurate results and is capable of analyzing the Hindi text in an effective manner.

This problem may be sub-divided into following sub-problems.

- **To investigate the use of Lexicon based technique in performing sentiment analysis in Hindi language**
- **To enhance and improve the lexicon-based sentiment analysis method**
- **To build a hybrid model and evaluate its efficiency and effectiveness**
- **To investigate the impact of length of messages on sentiment classification accuracy**

1.8 Organization of the Thesis

Chapter 1 Discusses the general background of Sentiment Analysis as the research area and explains its importance and applications.

Chapter 2 Gives an exhaustive literature survey involving detailed description of the research, methods, techniques and frameworks used by different researchers for performing Sentiment analysis in Hindi along with their limitations.

Chapter 3 Discusses about the investigation of the lexicon approach and compares its performance with various machine learning classifiers. The chapter also discusses performance of the Lexicon approach taking into consideration different parameters such as Accuracy, Recall, Precision, and F-score and assesses the reasons behind inefficiency of lexicon approach.

Chapter 4 Proposes an Enhanced Lexicon model approach by handling Morphological Variations, presents a Graph based Modified Lesk Approach to solve the problem of Word sense ambiguity and discusses a rule-based scoring algorithm for handling modifiers and conjunctions. The accuracy of the enhanced lexicon model is assessed.

Chapter 5 Describes the building of hybrid approach by combining the Enhanced Lexicon model and the machine learning classifiers. The hybrid approach is evaluated for efficiency on various datasets and also assessed for multiclass emotion classification.

Chapter 6 Discusses the investigation of the impact of length of messages and its effect on the parameter accuracy. The results are obtained and its statistical significance is also verified.

Chapter 6 Conclusion are drawn based on the experimental results along with mention of the future scope for further research in area of SA.

Chapter 2

Literature Survey

2.1 Introduction

SA has been a targeted area of research from the last one decade. A substantial amount of work has been done for the English Language and now the focus is on regional languages. There are various approaches that are being used to analyze sentiments or reviews in Hindi language. These methods can be classified as Lexicon based method, Machine Learning (ML) methods, Deep Learning based methods and Hybrid method as shown in Figure 2.1

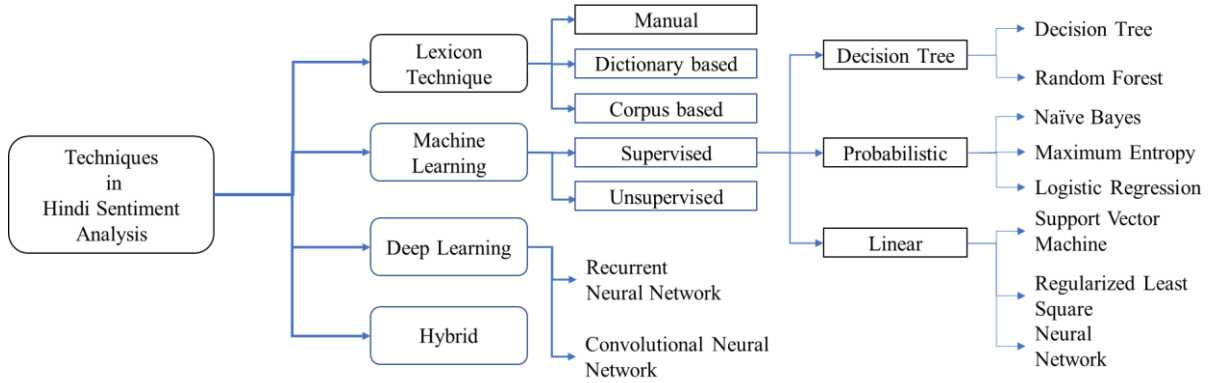


Figure 2.1: Techniques used for Sentiment Analysis in Hindi

Lexicon based technique is mainly dependent on a dictionary that comprises of a set of words and their corresponding scores. These scores will be summed up for categorizing the sentence as either positive or negative. When relevant data is extracted from Hindi textual reviews, preprocessing steps such as tokenization, stop word removal are performed and the dictionary or built lexicon is referred to get the actual polarity of the word. Lexicon techniques are further divided into manual techniques, dictionary-based techniques and corpus-based techniques. Manual construction approach, as the name suggests, utilizes some language experts to assign the polarities manually. But to construct the lexicon manually, it is very tough and time consuming too. Hence manual construction is generally, one of the less used approaches. Dictionary based approach involves creating a dictionary of sentiment words with polarity assigned to each word at initial stage. By adding synonyms and antonyms from various sources, the list can go on increasing further. Corpus based approach consists of huge corpus from which syntactic patterns are extracted to produce

sentiment words[12]. Sentiment words which have content specific polarities are found by using the corpus-based approach. This approach is further divided into two types as follows:

- Statistical technique: When an observation is made about two words being repeated together in the same context, then the chances of the word having same orientation are high. This basis is used for statistical technique and seed opinion words can be determined by this technique [13].
- Semantic Approach: is based on an approach in which words are semantically associated and allocated same sentiment values.

Another approach used for SA is the Machine Learning approach. Machine learning is an emerging field of Artificial Intelligence (AI) that focuses on generating a system which learns from experience and improves itself [32]. This method is constructed on the basis of feature extraction and study of the supplied data. Variety of ML algorithms can be utilized under this approach. In order to achieve sentiment analysis, classification can be done by using supervised learning method or an unsupervised learning method [33]. Supervised learning approach utilizes labeled data, whereas in the case of Unsupervised technique, the data is unlabeled. Naive Bayes (NB), Maximum Entropy (ME), Decision Tree (DT), Support Vector Machines (SVM), Multinomial Naive Bayes (MNB), etc. are various classification algorithms which can be utilized to train the model. The model can further be used to build predictions and help in categorizing the documents into positive class, negative class or neutral class.

An approach which is related to the field of AI and forms a subfield of Machine Learning is the Deep Learning approach which is gaining popularity in the recent years. This technique considers learning from experience and repeatedly doing the task and improving the output by performance tuning. The deep learning approaches can perform exceedingly well when the amount of training data provided is very high so that it gets sufficient information to learn the hidden semantics.

Hybrid approach is one which involves a combination of two or more approaches. The aim of any hybrid approach, when used for SA application is to combine the advantages of both the techniques and try to provide a better result. Research done in case of hybrid methods is significantly less as compared to other methods [8]. While making use of hybrid techniques for sentiment analysis, it was found out by researchers that combination of various approaches does not only combine its benefits but also its limitations which is why

hybrid methods have not been utilized more on Hindi sentiment analysis. Keeping this in mind, hybrid systems must be designed and used such that the sentiment analysis systems perform efficiently.

A survey related to opinion mining in Hindi Language was presented by Richa Sharma et al. [6] in 2014 highlighting the existing work in Hindi language and primarily concentrating on supervised and unsupervised approaches used for SA. A survey on journey of mainly Indian languages over SA was presented by Sujata Rani et al.[8] wherein the periodical evolution in the arena of sentiment analysis was shown for all Indian language families i.e., Arya, Dravidian family, kirata and Nishada from 2010 to 2017. The authors highlighted about the resources available such as annotated datasets, linguistic resources in various Indian languages for SA and focused on overall depiction of sentiment analysis in Indian languages. Sentiment analysis techniques used in different Indian languages were explained.

A detailed survey of Sentiment Analysis in Indo-Aryan, Tibeto-Burman and Dravidian Language Families was presented by Jasleen Kaur et al. [16] considering different languages such as Hindi, Marathi, Urdu, Bengali, Punjabi etc. from the Indo-Aryan group, Telugu belonging to the Dravidian group and specifically Manipuri belonging to the Tibeto-Burman group. [17] and [18] included surveys on SA in Hindi language with some literature review concerning the techniques and the research done for SA in this language.

2.2 Lexicon Approach

In lexicon-based technique, sentiment analysis is achieved by executing a certain set of rules. Hence, it is also referred to as rule-based approach. This analysis is based on words and the polarity given to each of them. In a set of words, each word is assigned a score, may be positive or negative based on its nature which is referred to as Lexicon. The scores of those words appearing in the input text are added together to reveal the combined score[12]. The final score maybe positive, negative or neutral which will be the nature of the text.

Following are the steps involved in the lexicon-based technique:

- Extracting data from Hindi text reviews [Data extraction]
- Tokenization and removal of Stop word [Preprocessing]

- Positive and negative score as result after using the lexicon on the processed data
[Run the algorithm]

Hindi language provides a lexical resource known as Hindi SentiWordNet (HSWN) which can be used to find the sentiment polarity. HSWN was built by IIT Bombay using an English Lexical source named as SentiWordNet and English-Hindi WordNet Linking [14]. Synsets with positive, negative and objective scores are included in the SentiWordNet. When a matching synset was found between the English Lexical source and the English-Hindi WordNet, a corresponding score was entered in HSWN. HSWN obtained polarity on each term in the document and ultimate polarity was computed based on voting.

Das et al. generated sentiment lexicons using an automated process and also semi-automatically using SentiWordNet for Bengali, Telugu and Hindi [11]. The researchers have used different approaches like Bilingual Dictionary Based, WordNet Based, antonym generation method and a Corpus based method. These approaches take into consideration language specific or culture specific words.

Bakliwal et al. [10] created a Hindi Subjective Lexicon (HSL), which used Hindi WordNet to implement a lexicon with polarity scores primarily for adjectives and adverbs. This was built by using WordNet and Breadth First graph traversal method. The initial seed list used to build the WordNet helped in lexicon generation in HSL. Word sense disambiguation was not performed because of non-availability of information on most frequently used senses. The results could have been improved by the use of morphological analyzer to lessen the missing number of adjectives and adverbs.

Mittal et al. devised a technique to improve and upsurge the coverage in HSWN by including more opinion words [19]. In this method, it started with negation handling and discourse relation, followed by sentiment classification of reviews in Hindi. The accuracy was much higher in this case as compared to existing HSWN. But there were limitations in terms of usage as it was used only for movie review corpus.

Arora et al.[20] proposed an algorithm to build a subjective lexicon consisting of a pre-defined seedlist and the preferred language WordNet, wherein the words were the nodes and they were linked to each other through their antonyms and synonyms. This method also could not handle Word Sense Disambiguation as it relied on the predefined seed list. One more limitation to this method was the coverage of number of adjectives and adverbs. Appropriate handling of morphological variations and multiple spellings could have been done. This shows there is a huge scope to improve and further enhance its capabilities.

To create a Hindi Senti-Lexicon covering Verb, Adjective, Adverb and Noun, a bootstrap method to find out polar words was suggested by Sharma et al.[21], which used WordNet of Hindi language and multi module rule-based sentiment analysis system. There were two seed lists in this approach; positive and negative. From the seed list, for every index word, all the senses were extracted. The sense was discarded if it found any of the words belonging to opposite direction or the word was placed in the lexicon along with its orientation. The polarity orientation was decided by Multi-Module Sentiment Analysis System which considered a window of size 3.

Lexicon-based approach and machine learning techniques were implemented on Hindi movie reviews by Jha et al. [22]. This methodology utilized extracted adjectives as sentiments words for classification. Accuracy was improved by applying negation rules. Discourse relations could have been handled and other POS types might have been considered for better accuracy.

For hotel reviews and movie reviews in Hindi, Mishra et al. [23] developed a context specific polarity lexicon (CSPL) resource. Four different variations were built for experimentation and results were compared with HSWN. Performance of synonyms extension could have been improved and antonyms extension could have been included to improve the CSPL resource.

Modi et al.[24] designed a rule-based approach named Part of speech tagging system. Corpus matching was utilized, when known words were tagged whereas grammar rules were utilized when unknown words were tagged. Jha et al.[25] addressed the issue of multiple domain sentiment classification by developing a sentiment aware dictionary which consisted of Hindi adjectives, verbs, adverbs and nouns. The target domain of classification was product reviews. But the problem aroused when the same word was utilized in a different manner in two separate domains and thus needed to be resolved for future use.

Hussaini et al. [26] implemented a score-based sentiment analysis system considering book reviews. Though, the system could handle word sense disambiguation and morphological variants, resolving issues of discourse relations and expanding the lexicon would have increased the accuracy of the system.

An algorithm was designed by Yakshi Sharma et al. [27] which used subjective lexicon method to analyze sentiment of Hindi tweets on topics such as “JAIHIND” and “World cup 2015”. In this approach, SentiWordNet comprising of adverbs and adjectives was constructed, along with development of a scoring scheme. The outcomes were compared

with Unigram presence approach and the results suggested that if the coverage of lexicon could be improved it could increase the performance of the system.

Jha et al. [28] proposed a subjectivity lexicon called the Hindi Subjectivity Analysis System (HSAS) which was generated by using two methods namely OpinionFinder and Seedlist expanding algorithm. OpinionFinder is an English subjectivity lexicon and used for translation too. Seedlist expanding algorithm, as the name suggests is an algorithm to expand the seed list and uses Pointwise Mutual Information (PMI) score for calculation. The use of Seedlist expanding algorithm led to expanding the seedlist from 60 to 4320 subjective words. The probability that a particular word is associated with an opinion is given by PMI calculation [29]. And to have an appropriate PMI score, the corpus files have to be expanded.

As per Garg et al. [30], sentiment analysis was performed on the tweets related to the “Mann Ki Baat” by Prime minister Mr. Narendra Modi using a lexicon-based method which utilized Hindi SentiWordNet and obtained an accuracy of 84.2%. A frequency-based approach was proposed further which increased the accuracy to 85.4% from 84.2%.

2.3 Machine Learning Approach

Machine Learning relates to the field of Artificial Intelligence (AI) wherein a system is trained and made to learn from experience so that it can improve itself. Any machine learning algorithm is fed with the training data from which it learns and then produces a model that is tried with the test data as shown in Figure 2.2. The model then gives predictions or classifications. When machine learning approach is used in the application of Sentiment Analysis, the most essential task is that of feature engineering which includes feature extraction and feature selection. Lemmas, Part-of-speech tags, Unigrams, bigrams, N-grams form some of the major feature sets which are commonly used in the Sentiment analysis process. Performing feature extraction in an efficient way can lead to improvement in accuracy as well as help in speeding up the training process.

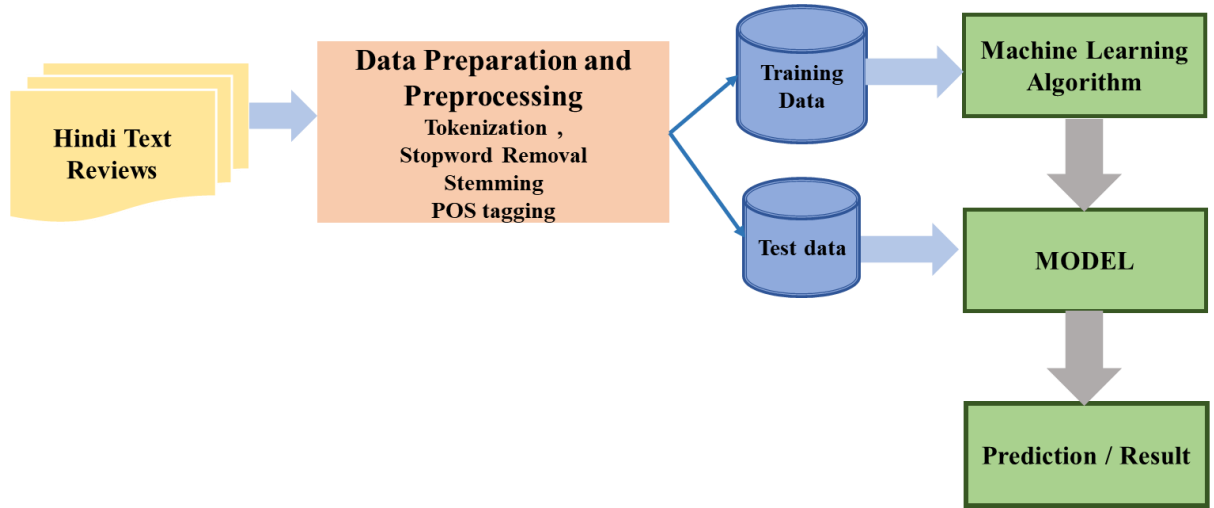


Figure 2.2. Concept of Machine Learning

Prasad et al. [34] contributed to Sentiment Analysis for Indian Languages (SAIL) Shared Task in 2015. In SAIL, sentiment detection was performed on tweets of Hindi language and classification was done into three classes either positive tweets, negative tweets or neutral tweets. Firstly, pre-processing of tweets was done, followed by creation of the model and then it was trained utilizing Decision Tree algorithm C4.5. To choose the feature, Information gain ratio was used. Implementation was done using J48. It was noticed that data processing in Hindi had to be done carefully as the unconstrained run performance degraded due to stop words. The accuracy in this case could be substantially improved by Negation handling and handling Hashtags, special characters, words and Emoticons.

As part of the SAIL task, Sarkar et al. [35] performed sentiment analysis of Hindi tweets using Multinomial Naïve Bayes Classifier. When the results were compared to results of SVM under unconstrained and constrained mode, Multinomial Naïve Bayes outperformed those of SVM. Another member of SAIL Shared Task, Sachin Kumar et al. [36] created statistical and binary features whose mapping was done to random Fourier feature space. These were utilized on a regularized least square supervised ML technique. The enhancement of language specific and sentiment features could have led to improvement in accuracy.

Ayush Kumar et al. [37] who was a participant of SAIL 2015, performed classification of Indian tweets based on their polarities by considering sentence level co-occurrences and expanding the Indian sentiment lexicon using a Distributional Thesaurus (DTs). Constrained run was performed by utilizing model which was trained by incorporating a SVM classifier on n-gram word features along with SentiWordNet features. For the proposed method, by creating an in-domain lexicon and by improving the dataset, accuracy

could have been improved. Shriya Se et al. as a part of SAIL task used Naïve Bayes supervised machine learning algorithm for classification of tweets into positive, negative and neutral labels. The results suggest classifiers work better in constrained Environment as compared to an unconstrained environment [38].

Hindi Opinion Mining System (HOMS) for movie reviews was built by Vandana Jha et al. [39]. In this system, Naïve Bayes classifier was used to perform document level opinion mining. The documents were classified on the basis of probability of presence of the class with Unigrams, and Best words with bigram chi-square word features. [39]. By using text mining approach, noise could have been minimized which could bring improvement in the system. Jha et al. [25] used machine learning techniques like Maximum Entropy, Naive Bayes Classifier and Support Vector Machine techniques to analyze sentiment in Hindi Language for movie reviews. Result shown for the classifiers were comparative before preprocessing and after preprocessing for both Unigrams and Bigram features. Even though, the system performance was good, larger data sets need to be tested.

In order to predict the outcomes of the elections held in 2016, Sentiment analysis was performed on tweets related to 2016 general elections by Sharma et al. [40]. The researchers used Naïve Bayes and SVM to analyze the tweets, so that people's sentiment and interest in political parties could be understood. The results by both the classifiers predicted a win for BJP. But, Emoticons, the most important feature while dealing with tweets was not considered for the research work.

Akhtar et al. in 2016 addressed the problem related to aspect-based SA for Hindi language. The work related to this was reported in two papers; "Resource creation and evaluation" [41] and "Category detection and Sentiment classification" [42]. A review dataset was created for 12 domains, followed by identifying aspects terms and sentiment in each of the reviews [41]. Conditional Random Fields (CRF) was used to train the model using various features like chunk information, local context, prefix and suffix information, POS tags etc. SVM algorithm was used to find out the opinion around every aspect term. The features were not domain specific which might have caused an effect on the results of the model. In [42], predefined aspect categories namely movies, electronics, travels and mobile apps were mentioned. Classifiers used were Naïve Bayes, Decision tree and SVM. Aspect category detection was regarded as a multi-label classification issue. Decision tree performed better in travels domain and also in movies domain. Naive Bayes classifier was better in electronics domain as well as in mobile apps domain. h

LibSVM was used to handle the issue of sarcasm detection in case of Hindi SA by Desai et al. [43]. LibSVM, a multiclass classifier was utilized to categorize the sentences into five classes. A lot of domain knowledge and semantic information was required for identifying sarcastic sentences correctly. Bafna et al. [44] used decision tree classifier, SVM classifier, Neural Network classifier and Naïve Bayes classifier to categorize Hindi poems into classes such as Bhajan, Updesh-Geet, Baal-geet and Desh Bhakti. Classification was done on a corpus that included 697 poems and the outcomes were better for SVM as compared to other algorithms.

With the growing development in web content, performance analysis of mixed code was the need of hour as websites started having lot of reviews and feedbacks in English and Hindi mixed code. But the analysis of such mixed code data is hard as generally this data is noisy and necessitates cleaning, POS tagging and appropriate language identification. Khandelwal et al. [45] tried studying the gender in English-Hindi mixed data with interesting results that tweets written by females had more punctuations and hashtags than males. They used Kernel SVM, Naïve Bayes and Random Forest Classifiers for experimentation. Results can have been improved by using POS tags for annotating the dataset.

SVM classifier and Random Forest classifier was used in order to detect irony in mixed code of languages Hindi and English by Deepanshu Vijay et al. [46]. SVM outperformed Random Forest Classifier when the results of irony detection were compared. The results could have been certainly improved by annotating the corpus with POS tags. Ravi Kumar et al. [47] classified sentiment of Hinglish text from news and Facebook comments by using a radial basis function Neural Network for the purpose of classification. In this approach, the results were presented by using evaluation metrics such as specificity and sensitivity. The results could have been improved by using a sentence parser to understand the relationship between various parts of speech.

To experiment with Hindi movie reviews, Random Forest and Support Vector machine was used by Charu Nanda et al. [48]. Mukesh Yadav et al. [49] tested diverse domains such as tourism, technology, current affairs, movie etc. by making use of neural network on Hindi mixture words. This testing worked at different levels of sentiment analysis and showed comparative results. It also showed significant difference in accuracy output when translation and without translation data were presented. For further work, various training algorithms and transfer function might be used for experimentation. Multiclass classification of poems based on Ras was performed by utilizing POS features and

emotional features was performed by Kaushika Pal et al. [50] by using SVM and Naïve Bayes. For better results, morphological variations need to be handled, along with increasing the dataset to a higher value.

Substantial research has been done using ML and there will be a huge scope in the years to come. In Machine learning techniques, smaller datasets tend to produce bias results and hence larger dataset need to be trained for better results. A detailed review of data and carefully selected features are required to train the model for better results. Along with these, time required for training should also be considered as evaluation parameter.

2.3.1 Feature extraction methods in Machine Learning

Feature engineering is a vital task of machine learning used for sentiment analysis as algorithms in ML are majorly reliant on the mined features. Due to this, extraction of features and their selection are very crucial tasks. Feature extraction is carried out by reducing the number of features by creating new ones. However, in feature selection, the number of features is reduced by removing the least important ones [51]. Feature sets such as Ngrams, Trigrams, Bigrams, Unigrams and features like lemmas and POS tags which are language dependent can be used for extraction in the process of sentiment analysis. Unigram includes all individual words, whereas Bigram consist of each pair of the words. Construction of N-grams is done by combination of n-consecutive words. Character N-grams refers to a sequential arrangement of n characters and Word N-grams refers to a sequence of n words.

Consider a sentence “यह फिल्म बहुत अच्छी है”

[This movie is very good]

Unigrams would be “फिल्म”, “बहुत”, “अच्छी”, “है”, whereas bigrams would be “यह फिल्म”, “फिल्म बहुत”, “बहुत अच्छी” “अच्छी है”. It is observed that as the order increases, the representation of the context can be done more effectively.

Usage of N-grams is in the order of 1 to 5, along with part of speech tags namely JJ (adjective), RB (adverb), NN (noun), and VB (verb). These can be utilized in the process of feature extraction and thus would help in finding the appropriate sentiment.

In the sentence “यह फिल्म अच्छी है”

The POS tags are marked as DT: “यह”, NN: “फिल्म”, JJ: “अच्छी” and VB: “है”

POS tags can help in the process of sentiment analysis and therefore needs to be extracted.

There are few tweet specific features which can be extracted during Sentiment Analysis in Hindi. Majority times, Hashtags, punctuations such as exclamation marks or question marks, Intensifiers, Emoticons, etc. play a vital part in expression of a sentiment and thus helps in the investigation. Tweet specific features are also discussed in [52].

Aspect-based sentiment analysis considers the context and aspect term to be very significant. Therefore, along with features such as bigrams, trigrams, prefix and suffix, the local context and the aspect term also can be considered. In order to classify poems on the basis of different rasas such as Shanta, Raudra, Bhayanaka, Hasya, Veera, POS based features and emotional features are used [50]. In this analysis, 9 classes were used, with every class representing one emotional feature. For emotional feature adjective, nouns and adverbs have to be considered.

To analyze Hindi sentiment Analysis, various feature selection methods are used. Some of them are listed below:

1. Chi square test: For measurement of independence between two random variables Chi square test is used.
2. Gain ratio: Gain ratio is defined as variation in Information gain in order to decrease bias.
3. Information gain: Evaluation of gain of each variable with regard to the target variable is done in this case [51].
4. Correlation method: This method works on the principal of linearity between two variables. If two variables are said to be linearly dependent then, they have high correlation, which means they will have a similar consequence on the target variable and one of them can be dropped.

Chi square method is the widely applied method for feature selection [52][53][54].

Chi-square, Information Gain, Gain Ratio, t-statistics, and correlation on the Document Term Matrix were used as feature selection methods in [47]. Comparatively, better results were obtained when 50 features were used in combination with Term Frequency-Inverse Document frequency (TF-IDF), Gain Ratio and Radial Basis Function Neural Network (RBFNN).

TF-IDF, Pointwise Mutual Information (PMI), term frequency, term presence, term frequency- inverse corpus frequency (TF-ICF), semantic orientation score etc. are different

feature representation techniques that can be utilized for learning suitable features. Very commonly used statistical measure for representation of features is TF-IDF as it gives relative importance to every term of the entire document. Another commonly used technique for feature representation is PMI[29].

One of the key roles of machine learning method is Feature engineering. Employing feature extraction along with providing handcrafted features to the machine learning process, will result in reduction of redundant data that in-turn will speed up learning leading to an improved performance.

2.4 Deep Learning Technique

Deep Learning is based on the technique of Neural Networks [55]. It is one of the sub fields of ML and AI that is inspired from the human brain. Deep Learning is based on the idea of learning from experience and performing the task repetitively and improving the output by tuning the tasks every time. As the name suggests, this technique has numerous layers that permits learning. The technique shows excellent performance in various applications such as speech recognition, pattern recognition, stock market analysis, computer vision, handwriting recognition, cancer detection, etc.[56]. In this method, a neural network is formed as the data is passed through various layers. Each layer tries to learn from the data provided to it. If appropriate training data is provided, then this technique can execute well compared to other ML techniques. One of the main benefits of this approach is that hidden semantics can be learnt from large unlabeled data corpus[57]. Lexical and structural features can be captured in this approach.

The main difference between ML approaches and deep learning approaches is that the deep learning approaches try to learn the features from the data with no human intervention required whereas in ML approaches, it is essential to provide handcrafted features to the classification algorithm as shown in Figure 2.3.

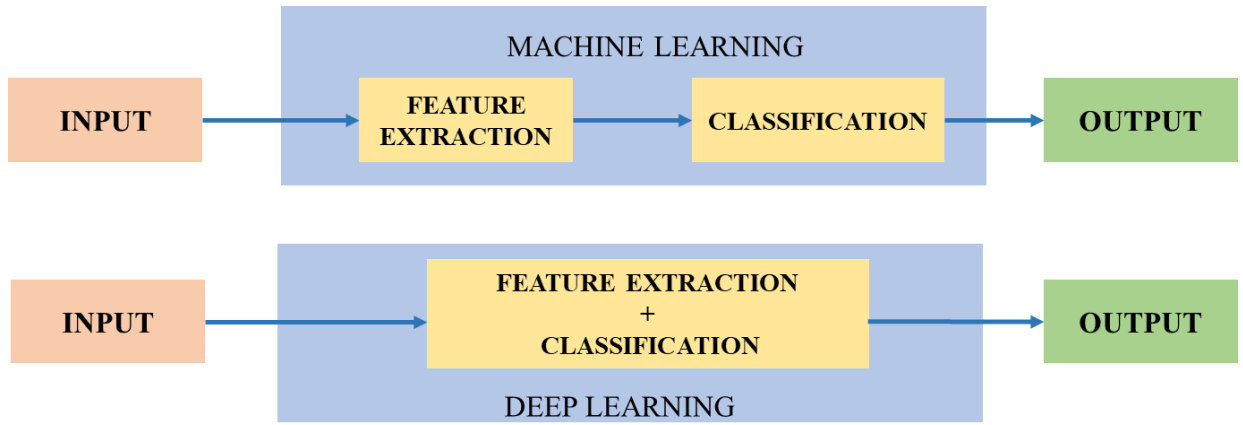


Figure 2.3. Machine Learning and Deep Learning flow

The deep learning networks which are used predominantly are Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). In CNNs, images are taken as inputs, which are then passed through various layers as follows:

1. Convolutional layer: In this layer, features are extracted by using filters and a feature map is generated as output to the pooling layer
2. Max pooling layer: In this layer, the data obtained from convolutional layer is aggregated and is known for reducing the representation.
3. Connected layer: This is the last layer from which the output is generated.

A simple CNN is shown in Figure 2.4. RNNs consist of several layers and the output is fed back to the input [72]. As RNNs contain memory, they can store previously calculated information and use it to make predictions. RNNs have different types such as Bidirectional, Deep Bidirectional and Long Short-Term Memory (LSTM) Network. In case of Bidirectional RNN, both preceding and following elements in sequence are considered. Compared to others, higher learning capacity is provided by deep Bidirectional RNN. A simple RNN is shown in Figure 2.5.

A different version of RNNs are LSTMs which have been used in applications of Sentiment Analysis in recent years. To achieve sentiment analysis by means of deep neural network, it requires an effective word embedding. Word embedding is a type of file that contains vector representations of words and phrases with comparable words and depictions in a predetermined vector space. One of the sources of word embedding for Hindi Language is FastText.

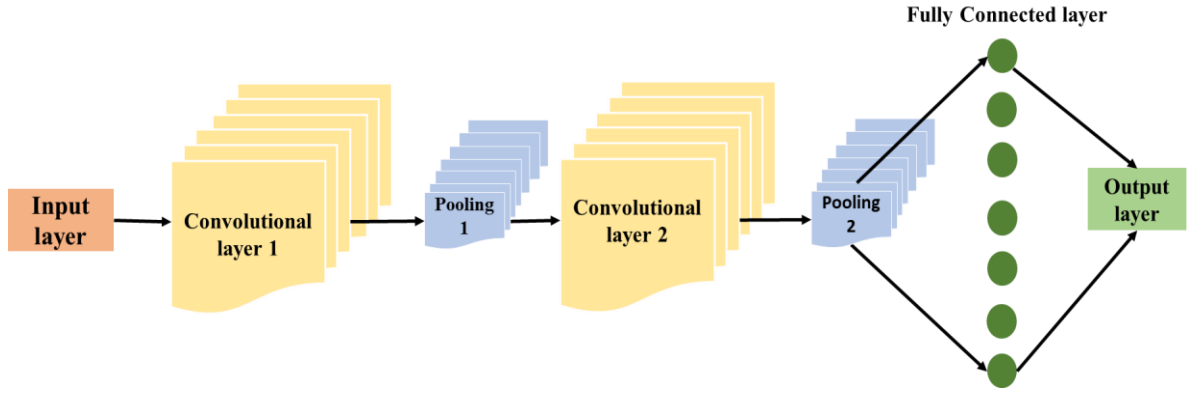


Figure 2.4. A simple convolutional neural network

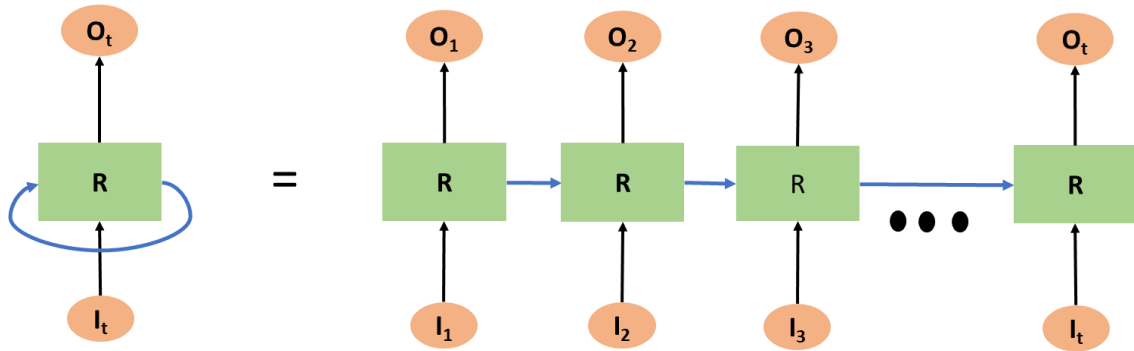


Figure 2.5. A simple Recurrent neural network

Deep learning approach generally includes word embedding which plays a crucial role in solving numerous issues. One of the major challenges faced is to create an effective word embedding for languages that are data sparse and ones which are resource poor languages.

For the deep learning networks, human intervention is less as it learns from its own errors. For effective performance of this approach, enormous amount of data is required as input. In other words, as data increases, Deep learning technique performance increases. Since, Deep learning technique uses complex calculations such as matrix multiplication procedures, mainly Graphical Processing Units (GPUs) are required.

Sujata Rani et al. [60] obtained exceptional results for performing sentiment analysis in Hindi on movie reviews using CNNs which could have been investigated on other domains. In CNN, one half of the dataset was used for training and other half was used for testing. Various parameter settings were used for experimentation for all the CNN models and they were found to be better than baseline Machine Learning algorithms. A CNN model which

contained two convolutional layers turned out to be the best in comparison with other models and obtained almost 95% accuracy.

The issue of data sparsity was solved by Akhtar et al. [59] wherein the author proposed the use of a bilingual word embedding which learned through parallel corpus. Training of the bilingual embedding was done on a product review corpus of Amazon which consisted of parallel sentences in both languages English and Hindi and was developed using Statistical Machine Translation (SMT) by means of tool Moses. Skip-Gram word2vec model was used to provide semantic understanding of English as well as Hindi word embedding and Bi-lingual skip-gram model was used to create an embedding for both Hindi and English. This technique gave highest results showing dominance of Deep learning methods. Besides this, few other significant characteristics had been identified. The neutral class could not execute properly and classification could not be done properly when the polar information was out of the context window. Identification and classification of implicit sentiment could not be done properly.

Considering Hindi English mixed code, deep Learning approach was done by Mathur et al. [61], Sane et al. [62] and Singh et al. [63]. Offensive tweets were detected by Mathur et al. [61] using CNN approach. This approach consisted of a novel dataset called the HEOT dataset and the classification problem was handled using transfer learning. The results which were successful were shown by comparing HEOT with transfer learning and without transfer learning. The problem of humor detection was solved using CNN and attention based bidirectional LSTM by Sane et al. [62]. To create Bilingual word embeddings, Word2vec and fastest skip-gram model were used. The results could be further improved by using aligned multilingual word embedding. SA for Hindi English mixed tweets was performed by Singh et al. [63] based on transfer learning.

Santosh et al. [64] designed two different models namely LSTM and Hierarchical LSTM model in order to perform hate speech detection and the outcomes were compared with Random Forest classifier and SVM. Accuracy was found to be better for SVM classifier. But, the best recall and F-score was obtained for Hierarchical LSTM model. As per the requirement for hate speech detection, by recording more semantic information, the results could have been improved.

Singhal et al. [65] solved the problem of multilingual sentiment classification using the concept of deep learning on tourism reviews and Hindi movies reviews. In this approach, the publicly accessible word embeddings and polarities of words from English were

borrowed or exploited and sentiment classification was performed using CNN. This approach performed well as compared to SVM and other CNN baselines. Some unknown words were handled effectively using this approach. Binary classification was used by this approach, which could be extended to multiclass sentiment classification.

Experimentation of 39 diverse models which were related to LSTM, CNN and RNN was done on SAIL 2015 dataset by Bhargava et al. [66]. CNN and a mixture of LSTM and RNN had greater success rates in comparison with other models. One of the major observations was that as complexity of text increased, accuracy decreased.

Akhtar et al. [82] proposed an architecture based on Bidirectional Long short term Memory Network which solves dual issues of aspect-based sentiment analysis viz., aspect term extraction and aspect sentiment classification. Three different models were proposed by making use of BiLSTM network. The outcomes were substantially better as compared to the reference outcomes of SemEval 2016 and the results presented by Akhtar in 2016. The results could have been improved by handling sarcastic phrases and metaphoric phrases in the review.

Akhtar et al. [83] developed a unified multitask framework for aspect-based Sentiment analysis which consisted of 2 frameworks namely joint modeling and End-to-End mechanism. The approach was built on the notion of multitask learning. A CNN framework and BiLSTM with attention for aspect term extraction and classification had been utilized in the approach. LSTM method was also used by Siddhartha Mukherjee [84] for evaluating English Hindi combination code considering both word and character features.

2.5 Hybrid Approach

A Hybrid approach aggregates two or more methods or approaches together and tries to make the system more accurate. Akhtar et al. [67] built a hybrid approach by using Convolutional Neural Network (CNN), Multi Objective Optimization (MOO) framework and SVM. CNN was used to build features whereas MOO for optimization and classification of sentiments was done using SVM classifier. In this approach, sentiment augmented optimized vector was fed as the important feature for training which was obtained by increasing the ‘sentiment-embedded vector’ from CNN and some optimized features which have been gained from MOO framework. Evaluation of this system was done by SemEval2014 on English twitter database to produce domain and language ability.

The proposed system outperformed all other models, but couldn't give an appropriate output when conflict was seen in the review or when a particular opinion word was missing.

Madan Gopal et al. [68] used an ensemble of LSTM model and multinomial Naïve Bayes (MNB) for SA of mixed code containing both Hindi and English. LSTM showed better results for those sentences which had longer length whereas Multinomial Naïve Bayes model was used to analyze some rare keywords and slangs. Thus, the results produced by this hybrid approach were better.

Garg et al. [69] explained Multiclass classification of Hindi statements by considering intensities of the views or opinions and proposing a hybrid system taking neural network and integrating it with fuzzy to emulate human thinking in undefined circumstances. Promising outcomes were obtained, but the results could have been enhanced by involving language specific constructs in order to analyze the sentiments in Hindi. Cyberbullying detection was executed by Tarwani et al. [70] on a mixed dataset containing Hindi and English. Along with using N-grams features, various classifiers were used such as Passive Aggressive Classifier, Logistic Regression, SVM, Decision tree and MNB to form a hybrid model after which results were obtained by voting classifier. 80% accuracy was achieved but the results could have been better by increasing the size of training set.

As compared to other approaches, limited research is seen in use of Hybrid approach. Researchers found that when techniques are mixed, along with the benefits, the limitations also may increase which has to be taken into consideration for effective performance of the system. This could be the reason of less research using hybrid approach.

2.6 Levels of Analysis

Sentiment analysis can be done at Aspect Level, Document Level and at Sentence Level. Substantial research at document level and sentence level is done for Hindi sentiment analysis as compared to aspect level. Researchers have used lexicon approach for sentiment analysis at document level [22, 26, and 35]. Research has been done considering only positive and negative classes [25] as well as considering neutral class along with positives and negatives for sentence level sentiment analysis [30, 31, 32, 48, 92 and 93].

Studies on machine learning techniques to perform sentiment analysis at document level is done and outputs are generated in namely 3 classes positive, negative and neutral

[26,50,58,94]. Whereas one study showed results being generated using 2 classes namely positive and negative [101].

Aspect category detection and classification of sentiments was done by Akhtar et al., by considering a fourth class which was named as conflict class along with the three classes namely positive, negative and neutral classes. Problems using conflict class was explained and the way to solve it by adding more instances was also explained [52]. Positive and Neutral class accuracy was good compared to accuracy of the Negative class.

Use of deep learning techniques at the sentence level as well as at the aspect level has been studied. [73] described the data sparsity problem in case of word representation for aspect-based SA. This issue of data sparsity was solved by use of LSTM deep learning architecture with positive, negative, neutral and conflict classes.

Studies by researchers showed use of sentence level sentiment analysis taking into consideration the three classes i.e., neutral, negative and positive [89, 95]. In [88], Shad Akhtar et al. developed a novel hybrid deep learning architecture with the use of CNN and Support vector Machine. Sentiment analysis was performed at sentence level as well as aspect level achieving accuracy of 57.43% and 65.96% respectively. Fourth class conflict was also considered for generating the outputs.

One of the researchers represented document level sentiment analysis with the combination of HSWN with LM classifier [96]. In this approach, the document was classified into various classes by considering ontology extracting sentiments from those respective classes in the form of neutral, positive or negative classes.

A lot of research has been done using three classes viz; neutral, positive and negative. By use of additional classes such as mildly positive, mildly negative, extremely positive, extremely negative, etc., the research can be further extended to finer level of granularity. The research can be further extended to sentiment analysis based on intensities and multiclass classification. At every level of analysis, hybrid and deep learning approaches can be used as they are still in growing stage with a lot of scope in future.

2.7 Addressing the Negation Problem

Negation plays an important in textual analysis and is mainly used to reverse the polarity of the statement. Thus, negation handling becomes a significant task, while performing sentiment analysis. This is usually carried out by surveying a window of size n and then

the polarities of every word that fall within the window are reversed. For negation, words like न, नहीं, ना etc. (No, Not) are used in Hindi language.

In lexicon-based techniques, a substantial progress has been done in addressing the negation handling problem. Negation handling by Bakliwal et al. [10] was done by identifying those words marked by the parser and swapping their polarities by use of sliding window of 6 words. This approach revealed an improvement of almost 3% in the classification accuracy. Different rules for negation handling were proposed by Mittal et al. [19] that could be applied in forward and backward direction depending on the sentence structure. Rules proposed reversed all the words in the window by use of ‘!’ symbol before every word till the sentence end or conjunction or delimiter is met.

Supervised and unsupervised approach was used to classify the results into positive and negative, after which negation handling was done considering window size of 3 in [39]. Arora et al. [20] showed a 3% improvement in accuracy after performing negation handling by use of sliding window of 6 words and polarities of those adjectives were swapped which fell under that particular range. Different rules for different negation cases such as single negation word in a sentence or if negation word and conjunction exist or when multiple ‘न’ in the sentence exists were provided by Namita Mittal et al. [71]. Ansari et al. [72] proposed that if negation is a preposition, the adjectives and adverbs were inverted. Garg et al. [73] performed negation handling by reversal of polarity, when there is a difference of 1 between the position of the negation word and sentiment word. This led to increase of accuracy from 51.4% to 74.7%. Considering words with negative implications was used for handling negation. This was done by Dalal et al. [74], but accuracy could not be improved to a great extent. The scope of negation could not be fixed because of presence of some linguistic features such as conjunctions, POS of negation etc. Some diminisher negations reduced the polarities of other words.

Consider the sentence:

Hindi: वे पुस्तके शायद ही कभी उबाऊ हो |

Meaning in English: Those books are rarely boring.

Diminishers are words like शायद, शायद ही कभी (rarely in English) which reduces the impact of negative words. Diminishers are to be handled in a different way than handling syntactic negations. Maintaining such negations can become difficult and can also increase the complexity of the sentences.

If negations are properly handled, it can significantly improve the performance of sentiment analysis. Majority of the work done on negation handling is by using appropriate rules and window size consideration typically between 3 to 6. Some more algorithms and methods need to be researched so as to perform negation handling efficiently. Handling intensifiers and diminisher negations can also be thought of as a scope of research.

2.8 Addressing other issues

Some of relevant issues concerning Sentiment analysis in Hindi Language are handling conjunctions, word sense disambiguation, discourse connectors, contextual variances etc. Conjunctions are used for connecting two clauses or two sentences such as पर, मगर, एवं, और (but, and). These conjunctions can have an impact on the final polarities. The words that are used to bind two phrases or sentences together are referred to as Discourse connector words. Particular Rules for handling discourse connectors were presented by Mittal et al. [19] which helped in increasing the accuracy from 78.39% to 80.21%. They divided conjunctions into two categories called Conj-after in which the part after the conjunction was preferred and Conj_infer in which it took into consideration conjunctions that drew inference.

There are Linguistic constructs like connectors and conditionals that can have an impact on the polarity of the statement. Discourse relations were handled by Namita Mittal et al. [71] using the HSWN approach that comprised inferential conjunctions such as इसलिए, कुल मिलाकर (Therefore, Altogether). Garg et al. [73] accomplished handling of conjunctions by postulating effective rules which is dependent on the location where the conjunction term occurred and improved the accuracy from 74.7% to 83.3%. Pandey et al. [75] solved this problem by postulating suitable rules and then finding out the correct polarity related with that statement.

Word Sense Disambiguation (WSD) is a well-known problem of NLP which refers to the task of allocating the utmost suitable sense to the term within a certain context. WSD forms a foremost challenge when lexicon-based method is being used as a precise sense of word that needs to be looked up in the Hindi SentiWordNet. The problem of word sense disambiguation was solved by Hussaini et al. [26] by means of score-based method improving the accuracy from 74.1% to 82.7%. Word sense disambiguation problem was resolved by Archana Kumari et al.[76] by creating a word embedding using Continuous bag of words (CBOW) and Skip-Gram (SG) model achieving an accuracy of 52%. Named

Entity Recognition (NER) can be a problem for resource poor languages. NER for Hindi was solved by Jain et al. [77] by developing an association rule mining algorithm and testing it on a corpus of news dataset. A context specific lexicon for reviews in Hindi was presented by Mishra et al.[23] and a solution was projected for contextual variances. The method built a polarity lexicon with context sensitivity that showed much better accuracy than HSWN.

In case of Deep learning techniques good word embedding is required so as to learn the hidden features efficiently. If the representation of a word is not present, it leads to data sparsity problem. Data sparsity problem was solved by Akhtar et al. [59] in case of aspect-based sentiment classification by using a Long Short-Term Memory (LSTM) based architecture on top of bilingual word embeddings

Deepanshu et al. approached the irony problem by using a supervised method for a Hindi English code-mixed dataset[46]. Different features were considered as irony indicators such as laugh words, intensifiers, punctuations, character, emoticons and word Ngrams. SVM and Random Forest classifiers were used and some prominent features that were incorporated were average word length, number of characters in the tweet etc.

Emotion analysis is also a very challenging problem of NLP that was approached by Yaman Kumar et al.[78] by classifying the sentences into five different categories namely anger, suspense, joy, neutral and sad. BHAAV meaning emotions was the text corpus that was developed. and various supervised models for classification were used. Offensive tweet detection problem in Hindi was considered by Puneet Mathur et al.[61] where they used the concept of transfer learning and employed convolutional neural network. Vikas Kumar Jhaa et al. [79] also considered the same offensive tweet detection problem by making use of a FastText classifier and a grid search method obtaining an accuracy of 92.2%. Insult detection problem was considered by Dalal et al.[74] by considering various features like n-grams, skip-grams and using SVM and Logistic Regression classifiers which helped in achieving almost 87% accuracy. The problem of humor detection was addressed by Sane et al.[62] by considering an English Hindi code mixed dataset and using attention based biLSTM model which achieved 73.6% accuracy. Santosh et al. [64] performed hate speech detection using LSTM models on a Hindi English code-mixed text giving good precision and recall results.

2.9 Research gaps identified

The exhaustive literature survey performed has helped in identifying the relevant research gaps in the area of sentiment analysis in Hindi and establishing the prominent objectives of this research. From the exhaustive literature survey done the following research gaps were identified:

1. Critically Analyze Lexicon based approaches and identify the key reasons for their inefficiency:

The Lexicon based approach needs to be investigated and analyzed thoroughly to find out what may be the reasons for its ineffectiveness. Once the key reasons for lower accuracy are analyzed, measures can be taken to solve the relevant issues and improve the effectiveness of the lexicon-based system

2. To Investigate the use of Word Sense Disambiguation methods effectively to enhance the performance of the lexicon- based approaches.

Word sense disambiguation is a well-known problem of NLP but an investigation needs to be done so as to understand whether solving the problem of WSD has an effect on the accuracy of the lexicon-based system. Instead of using averaging or first sense approach, a novel WSD approach must be devised to effectively improve the accuracy of the system

3. To Explore the problem of Negation Handling based on the study of Hindi linguistics.

Handling Negations is a major problem to be solved in case of Sentiment analysis. An appropriate rule-based method which takes into consideration Hindi linguistics and aims at handling negations efficiently can be devised which can help in increasing the accuracy of Hindi sentiment analysis

4. To Investigate the use of Intensifier and Diminisher handling to enhance the efficiency.

Intensifiers and Diminishers are a part of language Linguistics that may or may not change the polarity of the statements. Their usage can be analyzed and their effect on polarity can be determined to find out whether accuracy is affected by the presence of Intensifiers and Diminishers

Explore the possibility of combining Lexicon and Machine Learning approaches and evaluate this Hybrid approach for sentiment analysis in Hindi.

Lexicon approaches can be combined with Machine learning approaches to find out if a hybrid model can perform better in analyzing sentiments in Hindi. The hybrid model can be evaluated for different datasets and can be explored further for analysis of multiclass classification

5. Evaluate the impact of data properties of Hindi text such as length of the message (word-count) and size of training dataset on the classification performance.

Data properties such as word count or size of the training dataset may affect the performance of the system. An analysis of the same may be done and the impact of such properties may be evaluated along with defining its statistical significance.

2.10 Summary

Various lexicon-based techniques have been predominantly used in case of Hindi sentiment analysis such as Hindi SentiWordNet based, dictionary based, corpus based. In case of Lexicon classification, the text data is compared with the lexicons of the dictionary whose polarities are known beforehand and orientation of the sentiment is found out. There is no need of training in case of Lexicon techniques but the major concern is its coverage [115]. There is constantly a need to expand the lexicons and to build an efficient system, there is a need to solve the various issues of NLP like negation, discourse relations, Word sense Disambiguation, morphological variations etc. Research on word sense disambiguation for lexicon approach needs to be investigated more. Machine Learning Techniques are many-a-times used for performing sentiment analysis in Hindi language and are being used on numerous domains such as Movie reviews, product reviews, Election prediction, Facebook comments etc. The predominantly used ML techniques are Naïve Bayes, Support Vector Machines and Random Forest. There are variations of SVM which have been utilized such as CRF with SVM, SVM with radial basis function etc. Maximum Entropy, Logistic Regression need to be explored more. Ensemble learning of classifiers can also be investigated. ML methods perform exceedingly well only when trained with a larger labelled dataset. Even though the outputs received as a part of Machine which is why it is required to upsurge the size of datasets and evaluate the efficiency. ML methods also rely

more on feature representation and on the handcrafted features. TF-IDF, Chi-square statistics are the commonly used feature methods but there is a need to exploit other methods like Gain Ratio, Information Gain, t-statistics in order to improve performance. Researchers have to ensure that proper selection and representation techniques have been utilized for extracting relevant features.

Hybrid techniques are the ones which are least used techniques in Hindi sentiment analysis but when used they can turn out to be a powerful one. Multiclass classification using hybrid techniques can be researched and predominant issues like Sarcasm detection and Emotion analysis can be tried. A huge scope of research exists on evaluating a hybrid technique which tries to combine lexicon approach with the other approaches. Hybrid techniques may have some adverse effects when they are combined since the disadvantages of two approaches also are being brought together. Hence, a full systematic study is required before building a hybrid model.

Chapter 3

Investigating the Lexicon based Technique

3.1 Introduction

Lexicon based method is an approach wherein there exists a sentiment lexicon with a set of words along with their scores which aim at defining the polarity of the word, whether it is positive or negative. All the scores of the words that contribute to the sentence are taken and investigated and the combined score is calculated by summing the scores which then will reveal the polarity of the sentence. Figure 3.1 shows a typical flowchart for the lexicon-based technique. As shown, preprocessing is done on the extracted Hindi sentences which is performed so as to achieve dimensionality reduction. Data preprocessing is the first crucial step that deals with preparing the raw data and making it appropriate for the model for which it is being used.

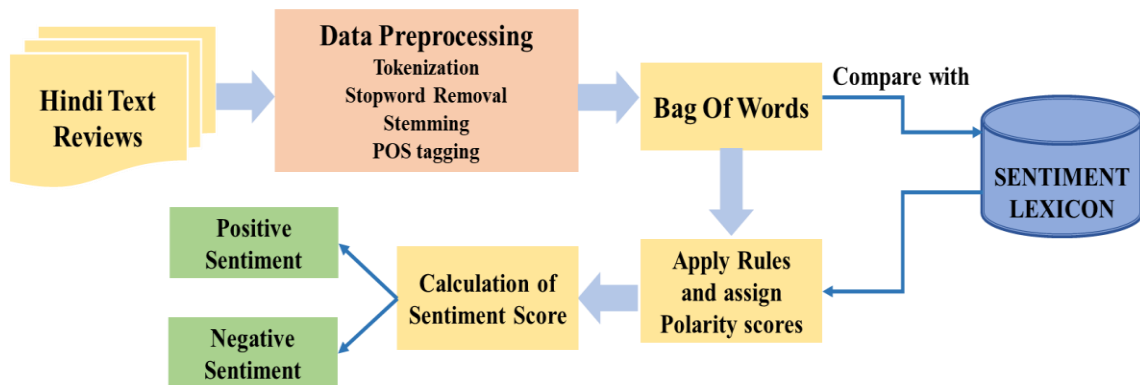


Figure 3.1 Lexicon based technique

The various steps involved in data preprocessing are:

1. Tokenization
2. Stop words removal
3. Stemming
4. Lemmatization

Tokenization deals with splitting the sentence into individual words. Stop words are the most common words used in the language that have no significance as such in the analysis and can be removed. Hence Stop-word removal process is done as a part of preprocessing. Stemming is generally a heuristic method of converting a particular word to its base form.

Lemmatization is similar to stemming but it reduces the word to a specific one that exists in the language. Part of Speech (POS) tagging is also done many-a times as the next step in Preprocessing, which gives each word its corresponding part of speech such as adjective, adverb, verb, noun etc. POS tagging can be very useful when used as a feature extraction method. Therefore, this process is generally used as a part of preprocessing. After preprocessing is performed, the next step is creating a Bag of words Model which is nothing but a collection of unigrams. These unigrams or words in a sentence are then compared against the sentiment Lexicon or the dictionary which will return the respective scores.

3.2 Resources Used

3.2.1 Hindi SentiWordNet (HSWN)

Hindi SentiWordNet (HSWN) is a well-known lexicon dictionary that was developed by IIT Bombay [14]. The dictionary was built taking into consideration two different resources namely the English SentiWordNet and the English Hindi WordNet Linking. The lexicon contains different Hindi words with their positive and negative scores. The words may be of different parts of speech such as Adjective, Verb, Adverb or a noun. Along with the POS_TAG, positive and negative score, a synset id is also included for the words in the HSWN. When the HSWN lexicon is used, it assigns a score to the word but does not consider the sense in which the word is being used.

3.2.2 Hindi WordNet

Hindi WordNet is one of the most significant and informative resource available in Hindi language. This resource was developed way back in 2002. The resource provides syntactic and semantic relations between various words thus defining how various words are related to one another. For every word its related information such as synonym, antonym, hypernym, hyponym is included in the Hindi WordNet.

3.2.3 Movie Review Dataset

A 1000 sentences movie review Dataset is used for this research which consists of Movie reviews taken from a Hindi review site named jagran.com combined with movie review dataset from IIT Bombay. The dataset totally has 500 positive reviews and 500 negative reviews.

3.2.4 Multidomain Dataset

A 4000 sentences dataset pertaining to different domains such as Movie reviews dataset, Product domain dataset from IIT Patna and tourism domain dataset from IIT Bombay are used for checking the efficiency of the proposed approaches.

3.2.5 Evaluation measures

Accuracy:

Accuracy is the calculation of ratio of the level to which the proposed method is able to guess correctly what are the true observations and false observations out of all total observations. The maximum value of accuracy can be 1.0. As the number of false positives and false negatives goes on decreasing the accuracy value goes on increasing.

$$Accuracy = \frac{(True\ Positives + True\ Negatives)}{(True\ Positives + True\ Negatives + False\ Positives + False\ Negatives)} \quad (3.1)$$

Precision:

Precision is defined as the fraction of positive observations that are appropriately predicted to the total number of true positive observations and false positive observations. High precision is obtained when false positives is less. The maximum value of precision can be 1.0.

$$Precision = \frac{True\ Positives}{(True\ Positives + False\ Positives)} \quad (3.2)$$

Recall:

Recall is defined as the fraction of positive observations that are appropriately predicted to the total number of true positives and false negatives observations. High recall is obtained when the value of false negatives is less. The maximum value of recall is said to be 1.0.

$$\frac{True\ Positives}{(True\ Positives + False\ Negatives)} \quad (3.3)$$

F-score:

F1-score is used to provide a balance amongst Recall and Precision since it is observed that when Precision increases, Recall decreases and vice versa. Thus, it is referred to as weighted average of Precision and Recall. The Optimal value that can be achieved by F1-score is 1.0.

$$F1 = \frac{2 * (Precision * Recall)}{(Precision + Recall)} \quad (3.4)$$

3.3 Simple Lexicon HSWN algorithm

The simple Lexicon algorithm makes use of the Hindi SentiWordNet as the lookup Dictionary and then calculates the polarity of the sentence. Figure. 3.2 shows pseudo code of the Simple Lexicon HSWN algorithm. According to this algorithm, preprocessing is the first step to be performed which includes stop word removal, stemming and POS tagging.

A voting list is maintained and two variables are initialized for storing the positive and negative score. Each word in the sentence is considered and its corresponding positive and negative score is extracted from the HSWN. Depending on the score which is greater either 1 for positive or 0 for negative is appended to the voting list. Also, if positive score is greater than the negative score then the positive score will be added to the positive total else negative score will be added to the negative total. At the end, whichever count is greater in the voting list is selected. If equal, then the positive total and negative total are compared and the corresponding polarity is given as the output polarity of the sentence. The algorithm just ignores the word which is not present in the HSWN. Therefore, many-a-times due to absence of lexicons in the HSWN, score of a particular word cannot be found and this affects the resultant polarity. Some promising techniques need to be implemented using which the HSWN can be utilized to its maximum and the resource can turn out be a useful one.

ALGORITHM 1: ALGORITHM FOR SIMPLE LEXICON APPROACH USING HSWN

***Input:** Sentence S containing n number of words*

***Output:** polarity either positive or negative*

- 1 *Perform stop word removal, stemming and POS tagging*
- 2 ***Initialization of List:** $VotingList = \{\}$*
- 3 ***Initialization of variables:** assign zero to variable pos and neg*
- 4 ***Initialization of variables:** assign zero to variable i*
- 5 ***while** ($i < n$) **do***
- 6 $W = \text{word}(i)$ // Read the word in the sentence S one by one
- 7 **if** (W is in HSWN)
- 8 $Pos_score = \text{Read the positive score from HSWN}$
- 9 $Neg_Score = \text{Read the negative score from HSWN}$
- 10 **If** ($Pos_score > Neg_Score$) **then**
- 11 $VotingList.append(1)$
- 12 $pos = pos + Pos_score$ // add positive score from HSWN to pos
- 13 **else if** ($Neg_Score > Pos_score$) **then**
- 14 $VotingList.append(0)$
- 15 $neg = neg + Neg_Score$ // add negative score from HSWN to neg
- 16 $a = \text{number of ones in the VotingList}$
- 17 $b = \text{number of zeroes in the VotingList}$
- 18 **if** ($a > b$) **then**
- 19 $polarity = 1$ (positive)
- 20 **else if** $b > a$ **then**
- 21 $Polarity = 0$ (negative)
- 22 **else if** ($pos > neg$) **then**
- 23 $polarity = 1$ (positive)
- 24 **else if** ($neg > pos$) **then**
- 25 $Polarity = 0$ (negative)

Figure 3.2 Simple Lexicon Algorithm

Using the lexicon algorithm as depicted in Figure 3.2, implementation was done on the single domain Movie review dataset and the results in terms of accuracy, precision, recall and f-score were obtained as shown in Figure 3.3. When the analysis of the results was done, it was seen that scores of some words though present in HSWN could not be extracted due to morphological variations.

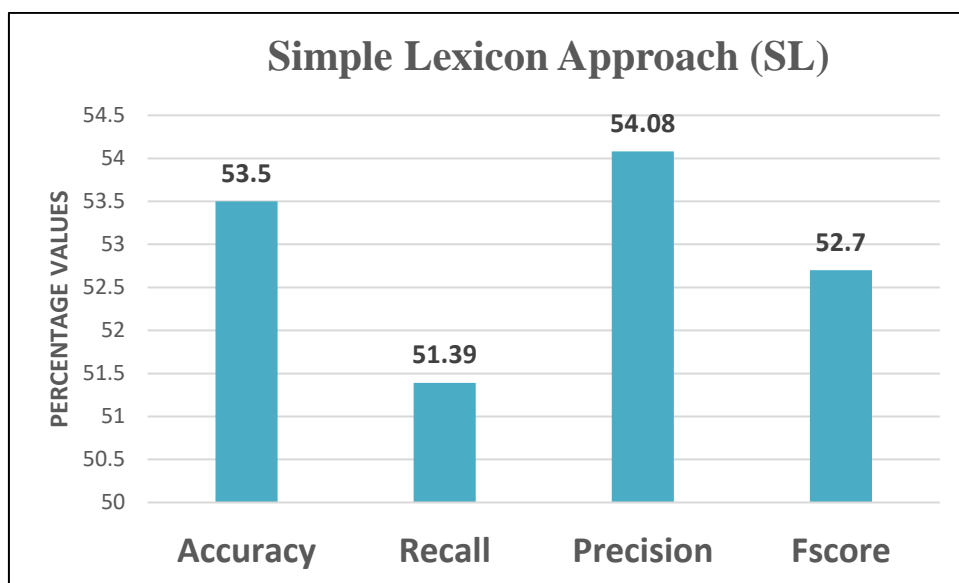


Figure 3.3: Output of using Simple Lexicon HSWN for Movie review dataset

The lexicon algorithm was also tested on Multidomain review dataset and the corresponding results were obtained as shown in Figure 3.4. Accuracy obtained was around 61%. The Simple Lexicon approach provided good results on product domain dataset and hence the overall multidomain output was better.

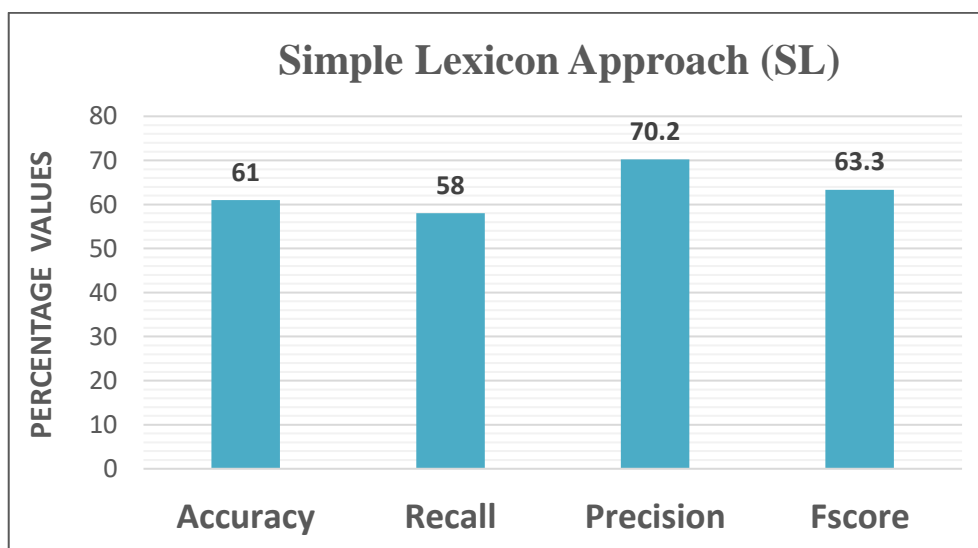


Figure 3.4: Output of using Simple Lexicon HSWN for Multidomain dataset

3.4 Comparison with Machine Learning Classifiers

Different Machine Learning classifiers have been used by researchers for performing sentiment analysis of Hindi Language. The steps to be performed in building and evaluating a Machine Learning model is as shown in Figure 3.5. Data preprocessing is done on the

input Hindi text review sentences and the data is divided as training and test data. Every machine Learning algorithm learns from the training data and builds a model to which the test data is given as input. The model then accordingly gives the prediction output.

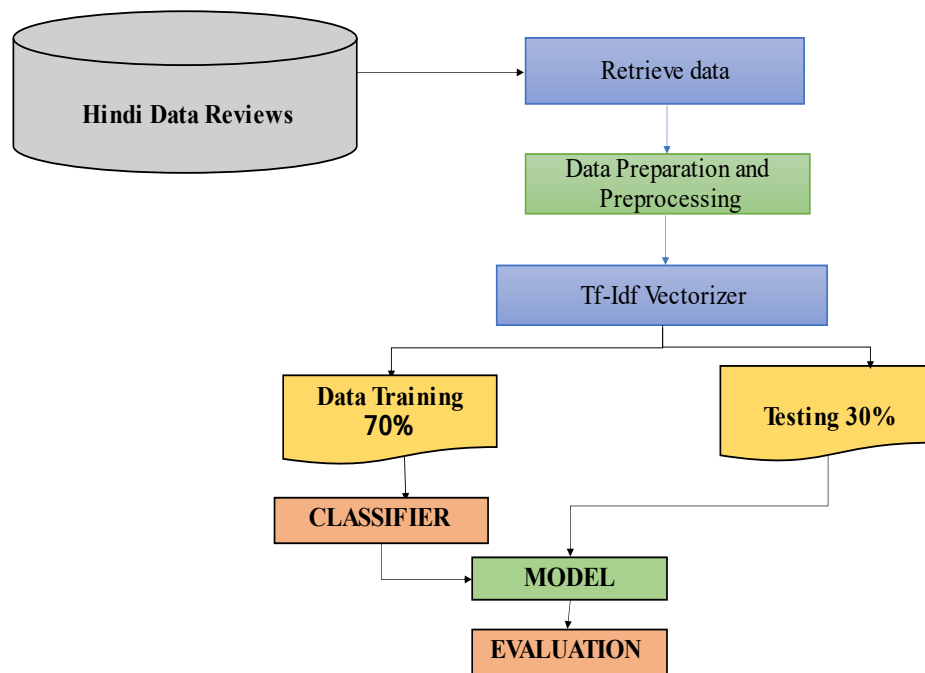


Figure 3.5. Steps in building a Machine Learning model

The review textual data was preprocessed to eliminate any irrelevant information. A very vital preprocessing step is feature extraction which comprises of building feature vectors from textual data to enable learning. Here, the significant features were extracted and the textual data was converted to a numerical format. The stop words were removed and training and testing datasets were created. It is possible to represent text as vectors of identifiers. This is generally called as vector space model. For tasks related to task categorization, TF-IDF is considered to be the most effective function for indexing a term. The term frequency method is basically used to assign weights to those terms which have relatively more importance compared to other words and hence, the value increases depending on whether the occurrence of the word is frequent in the document. But according to the term frequency method, the terms which are not of that importance such as का, है, पर, को, (of, is, on, the) also get a higher frequency due to which there is a need to weigh down these terms. Hence, an IDF method is combined with Term Frequency method to moderate the weight of terms that are seen frequently and try to intensify the importance of relevant terms.

Thus, $TF-IDF = TF * IDF$ Where

TF = frequency of the term t / total number of words in the text

IDF = \log (total number of documents / (number of documents with term t in it))

The various machine Learning classifiers used for experimentation and testing on the 4000 reviews multidomain dataset are explained below:

- **Decision tree:** Decision tree is a familiar classifier that is known to build a tree like structure by taking rules into consideration, which can be effortlessly understood. A decision tree fundamentally utilizes the divide and conquer technique for recursively partitioning the instance space. The start node is named as the root node which is further split into external nodes and the internal nodes [38]. External nodes are called as the end nodes or leaf nodes as shown in Figure 3.6.

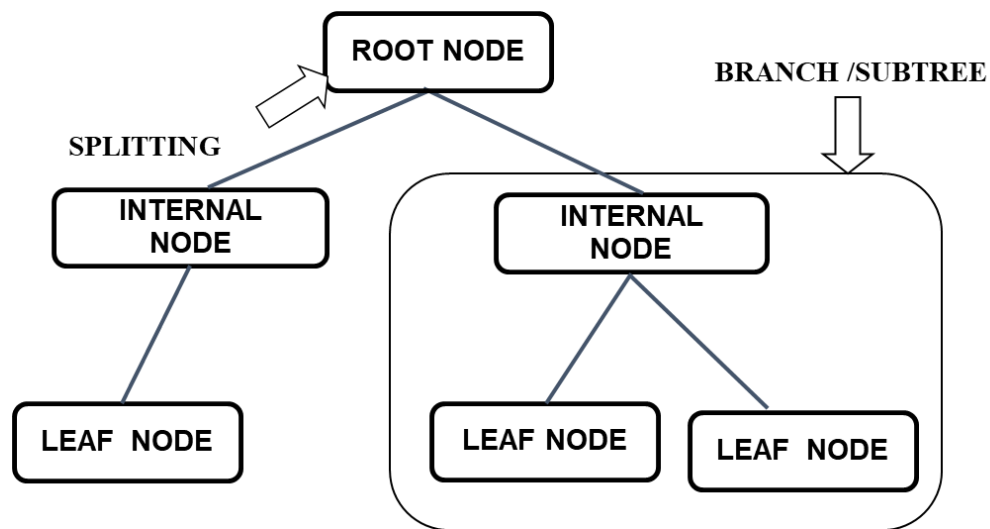


Figure 3.6. Decision Trees

The decision tree classifier works by classifying the instances from the root of the tree until a leaf node is reached. The working methodology of decision tree is as below:

Step 1: Create a Root node R

Step 2: If T_{set} are positive, then return a single-node tree, with label as +

Step 3: If T_{set} are negative, then return a single node tree, with label as –

Step 4: If A is empty, return Root, with label = T_a

Step 5: Otherwise Begin

- I_a = the attribute in A that best classifies instances
- Root = I_a
- For each possible value a_i , of A

Add a new branch below Root to test if A equals a_i and create subset E_v

d. If E_v is empty

Then beneath the branch add a node with label = T_a

e. Else

beneath this new branch add the subtree created by recursively calling the algorithm with $T_{set}=a_i$ and $A=A-I_a$

End

Step 6: Return Root

- **K-Nearest Neighbor:** K-nearest neighbor is a supervised learning method that considers the similarity between the test data and the available cases and categorizes the test data into one of the categories which is very much similar. The working methodology of KNN is as below:

Step 1: Firstly, the number value K of the neighbors is selected

Step 2: Then the Euclidean distance of K number of neighbors is calculated

Step 3: Amongst the K nearest neighbors which are obtained from calculated Euclidean distance., the number of the data points in every class are counted.

Step 4: The new data points are allocated to that group for which the quantity of the neighbor is maximum.

- **Neural Network** is a type of machine learning technique which generally includes layers such as an input layer, an output layer and many middle layers called as the hidden layers. A layer is mostly a set of neurons that are accountable for capturing the input, processing it through some activation functions and producing an output. According to the problem in hand and the type of data, the activation function that is to be used is fixed. One such type of neural network is Multilayer Perceptron that uses back propagation learning algorithm and helps in solving non linearly separable problems [40]. The algorithm uses feed-forward network training and the working methodology is as given below:

Step 1. For every training pattern:

- a) First the inputs to the network are applied.
- b) The output for each neuron is calculated from the input layer, considering the hidden layers, to the output layer. Also, the error at the outputs is calculated.
- c) The output error is used in order to compute the error signals for pre-output layers.

- d) The error signals are then used to compute the weight adjustments and the corresponding weight adjustments are applied.

Step 2. The network performance is periodically evaluated

Step.3. Once training is complete, the review is tested to find the polarity.

- **Support Vector Machine** is a technique which is used for classification of data using measurable learning hypothesis [5]. This classification approach is predominantly constructed on maximizing the margin standing between the examples and the separation hyper-plane. The data items are considered as points in the n dimensional space and the ones which are nearest to the hyperplane are critical deciding elements called as support vectors. In case of SVM method, each feature value is taken as coordinate value and classification is done by finding out the hyper-plane that clearly distinguishes the classes [41]. SVM thus aims at solving the given problem by determining the maximum marginal hyperplane. The working methodology of Support vector Classifier is as follows:

Step 1: Split the given data set into two set of data items having diverse class labels assigned to them.

Step 2: Add them to the support vector set.

Step 3: Loop the divided n data items

- a) If a data item is not allocated to any of the class labels, then add it to support vector set.
- b) Break if inadequate data items are found.

Step 4: Train by means of the derived Support vector classifier model and test so as to validate over the data items that are not labelled.

Step 5: End

- **Random Forest Classifier** is a well-known ensemble classifier dealing with construction of multiple decision trees. It makes use of bagging and feature randomness and creates an ensemble for reducing the problem of overfitting that may be seen in the case of simple decision trees. The predicted class given by each decision tree is taken and the end result of the random forest classifier is determined which will be the class with the most votes [39][40]. Figure 3.8 depicts the random forest classifier.

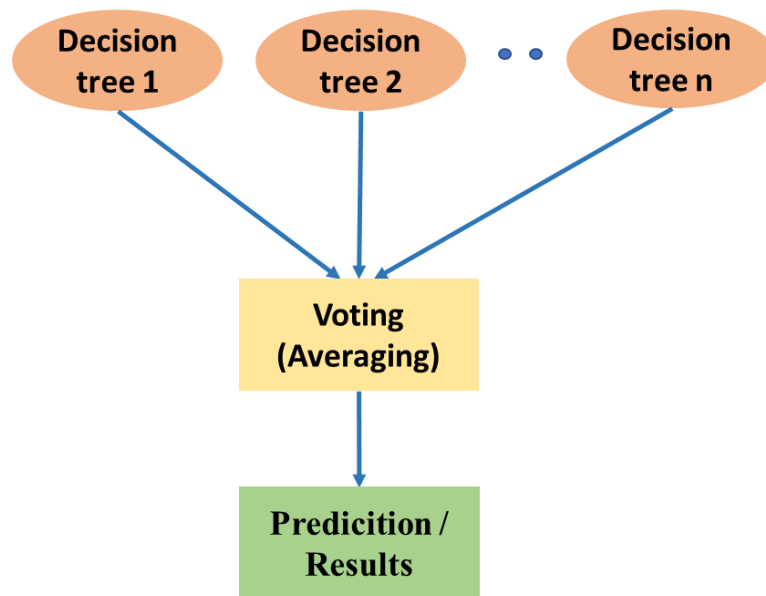


Figure 3.7. Random Forest Classifier

The working methodology of Random Forest classification is as follows:

Step 1: Construct n number of tree bootstrap samples from the original data.

Step 2. Considering every bootstrap sample, build an unpruned classification tree, with the adjustment that at every node, instead of choosing the best split, perform random sampling of the predictors and select the best split from among those variables.

Step 3. Predict new data by performing aggregation of the predictions of the n trees which can be majority votes in case of classification.

- **Logistic Regression** is a classifier which belongs to the probabilistic classifier type which considers one or more independent variables in order to determine the outcome of a dependent variable. In majority of the cases the dependent variable is said to have a binary outcome meaning if the probability in one of the cases is P , then the probability of the other case is always $1 - P$.
- **Naïve Bayes** is another example from the probabilistic classifiers group that depends on the Bayes theorem. Given a feature vector, this machine learning algorithm aims at finding the probability of allocating every class to that feature vector and then selecting the class which has maximum probability [41]. **Multinomial Naïve Bayes** is a version of Naïve Bayes that makes use of multinomial distribution for every feature and generally considers word count.

3.4.1 Comparison of Simple Lexicon with Machine Learning classifiers

The output of the simple HSWN approach is compared with the various machine learning classifiers on the single domain movie review dataset and the multidomain review dataset. Table 3.1 shows the comparison of accuracy of classifiers for single and multidomain dataset.

Table 3.1 Accuracy Comparison results of Classifiers

Classifier	Single domain dataset	Multidomain dataset
Simple HSWN Approach	53	61
Decision tree	89.4	74.6
KNN Classifier	88.3	71.8
Neural network	91.1	70.9
Support Vector	89.5	76.6
Random forest	89.2	76.8
Logistic regression	85.04	69.8
Multinomial Naive Bayes	84.83	72.8

Simple Lexicon HSWN approach shows less accuracy for both the datasets. Output was significantly better on multidomain dataset than single domain as it could find the respective domain lexicons in the HSWN. Machine Learning approaches performed very well on single movie review domain dataset since training was done better pertaining to a single domain.

Figure 3.9 shows accuracy comparison of HSWN approach with different machine learning classifiers on movie review dataset. The output shows that Random Forest, Support Vector, decision Tree and Neural Networks perform very well when compared to other classifiers. There is a difference of more than 30% in accuracy as seen between simple Lexicon HSWN approach and the machine learning classifiers.

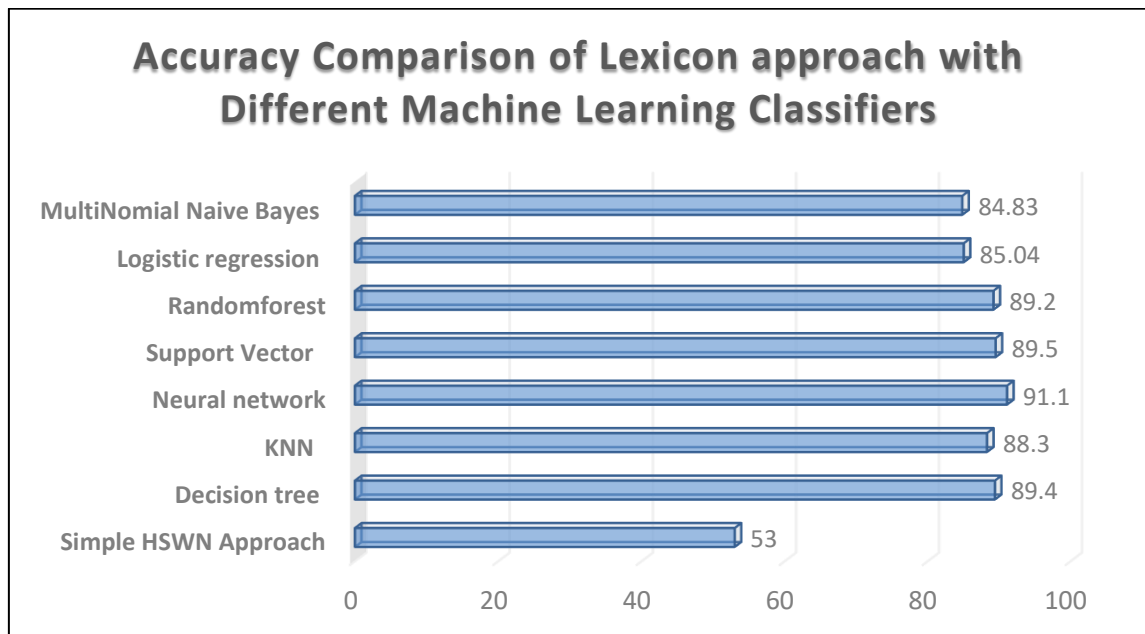


Figure 3.8. Accuracy Comparison of Simple HSWN approach with various ML Classifiers on 1200 movie review dataset

Figure 3.10 shows accuracy comparison of Simple HSWN approach with various machine learning classifiers on Multidomain review dataset. The output clearly shows that Random Forest and Support Vector Classifiers are significantly better when compared to other classifiers. The results show that there is more than 10% difference in accuracy as seen between simple Lexicon HSWN approach and the machine learning classifiers.

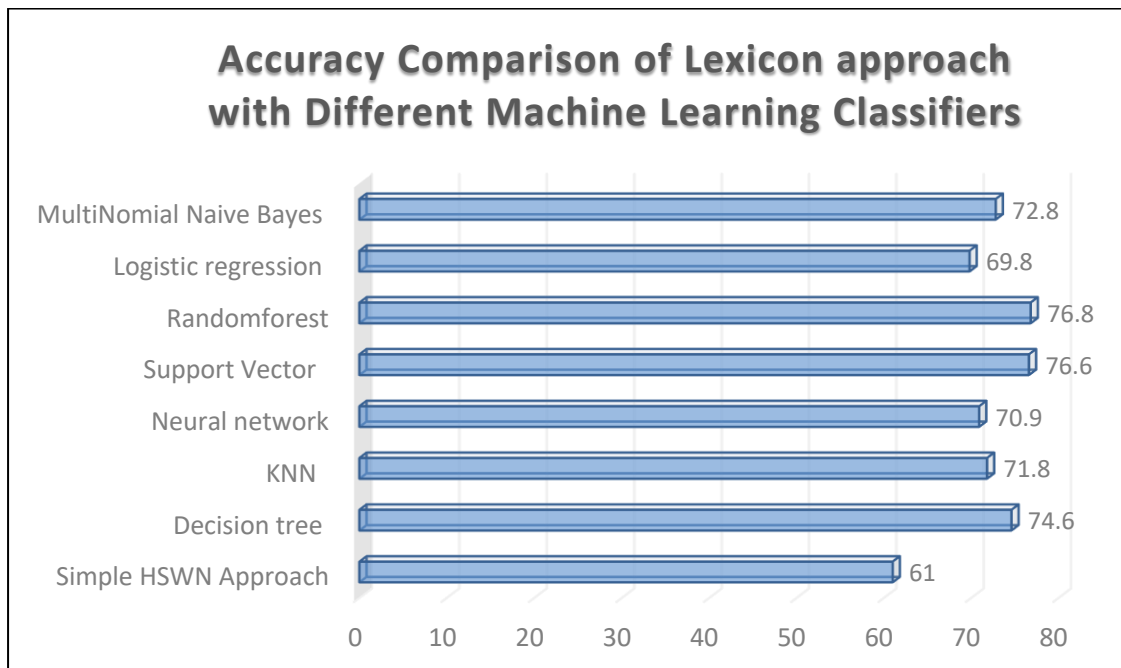


Figure 3.9. Accuracy Comparison of Simple HSWN approach with various ML Classifiers on 4000 Multidomain review dataset

Both the comparison outputs clearly show that the simple Lexicon HSWN approach needs a lot of improvement to perform better than or become at par with Machine Learning approaches. The Hindi SentiWordNet needs to be used effectively and the problem of absence of lexicons in HSWN needs to be handled. Further, problem of Morphological variations and Negations which may have resulted in low accuracy needs to be handled. An appropriate solution to word sense ambiguity can be presented if a word with more than one sense is present in HSWN.

3.5 Summary

Performing Sentiment Analysis in Hindi is a very challenging task as is evident from previous discussion. One of the approaches being used for performing Sentiment analysis in Hindi is by making use of the lexical resource Hindi SentiWordNet. The terms present in the sentences are matched with the lexicons in the HSWN and the polarity is determined whether it is positive or negative.

The Simple Lexicon HSWN approach is tested on single domain and multidomain datasets for experimentation and the results are obtained. The results are also compared with various machine learning classifiers so as to get an idea about the efficiency of a Lexicon HSWN approach. The accuracy obtained for Simple Lexicon HSWN approach is substantially less as compared to individual Machine Learning classifiers and needs a lot of improvement. The Hindi SentiWordNet has different scores for different senses of the same word. Hence the word sense ambiguity problem needs to be solved to get the correct sense of the word from the Hindi SentiWordNet.

From above discussion it can be concluded that Lexicon based methods in case of Hindi Language have low performance and should be made efficient. Research in this direction could lead to lexicon-based efficient solutions for Sentiment Analysis. Having seen how lexicon-based methods work and what are its limitations, the next chapter proposes an Enhanced Lexicon algorithm which performs Word Sense Disambiguation, Negation Handling and Rule-based Intensifier and Diminisher handling and tries to find an efficient solution for Lexicon-based Hindi Sentiment analysis.

Chapter 4

Enhancing the Lexicon-based Sentiment Analysis Method

4.1 Introduction

The critical analysis of the Simple lexicon-based Sentiment Analysis method using HSWN in case of Hindi Language show that they have low accuracy and are not very effective in performing SA. There could be several reasons:

- Morphological Variations may be a problem. Morphological handling has to be done so as to bring the word in its root form or appropriate base form before performing the search in HSWN.
- Word Sense Disambiguation is a known problem of NLP and if different senses of the same word are available with different scores in the HSWN, then this issue needs to be effectively handled in case of Lexicon based SA method. Obtaining the appropriate score from HSWN is required to get an accurate output.
- Negations, Intensifiers and Diminishers when not handled may result in low accuracy. These need to be handled to perform SA effectively

This work proposes an Enhanced Lexicon algorithm which performs Morphological Handling, Word Sense Disambiguation, Negation Handling and Rule-based Intensifier and Diminisher handling to find an efficient solution for Lexicon-based Hindi Sentiment analysis. The Enhanced Lexicon model was tested on the 4000 sentences multidomain review dataset and the results were found to be promising.

4.2 Morphological Handling

Morphological Variations form a significant challenge in case of Hindi language. The meaning of Morphological Variations is that for the same root word, there can be different variations of the word in Hindi language. E.g.: आएगा, आएगी, आयेंगे (He will come, she will come, they will come)

In presence of variations, there is a need to find out which variation is present in the HSWN and appropriate score of that word must be obtained from the Hindi SentiWordNet. For

example, the word सड़ is present in the HSWN and its negative score is 0.875 which clearly tells that it is a negative word. A sentence having the word सड़ will not be able to get the score from the HSWN. Hence variations of the word need to be considered and the input word should be converted to the specific word whose score is present in the HSWN so as to get an accurate output. The HSWN specific morphological handling algorithm is shown in Figure 4.1.

ALGORITHM 2: HSWN SPECIFIC MORPHOLOGICAL HANDLING

INPUT: Word which can be noun, verb, adverb and adjective.

OUTPUT: positive and Negative Score

```

1  Search the word in HSWN
2  If the word is present in HSWN
3      pos= Obtain the positive score of the word from HSWN
4      neg = Obtain the negative score of the word from HSWN
5      return (word, pos, neg)
6  root_word = root - suffix
7  If the root_word is present in HSWN
8      pos= Obtain the positive score of the word from HSWN
9      neg = Obtain the negative score of the word from HSWN
10     return (root_word, pos, neg)
11 while (Matras list is not equal to NULL)
12     var=root_word+Matra
13     if var is in HSWN
14         pos= Obtain the positive score of the word from HSWN
15         neg = Obtain the negative score of the word from HSWN
16         return (var, pos, neg)

```

Figure 4.1. HSWN specific morphological Handling

The algorithm first checks whether the input word is present in the HSWN and if present the relevant score is obtained. If the word is not present, the algorithm checks whether its root is present by removing the suffix. If the root is present, its corresponding score is obtained otherwise it tries to find which variation of the root is present and gets its corresponding score. Thus, the morphological handling algorithm along with the HSWN

approach was used on the multidomain dataset and the graph results of comparison of HSWN specific Morphological Handling with simple Lexicon HSWN approach were obtained as shown in Figure 4.2.

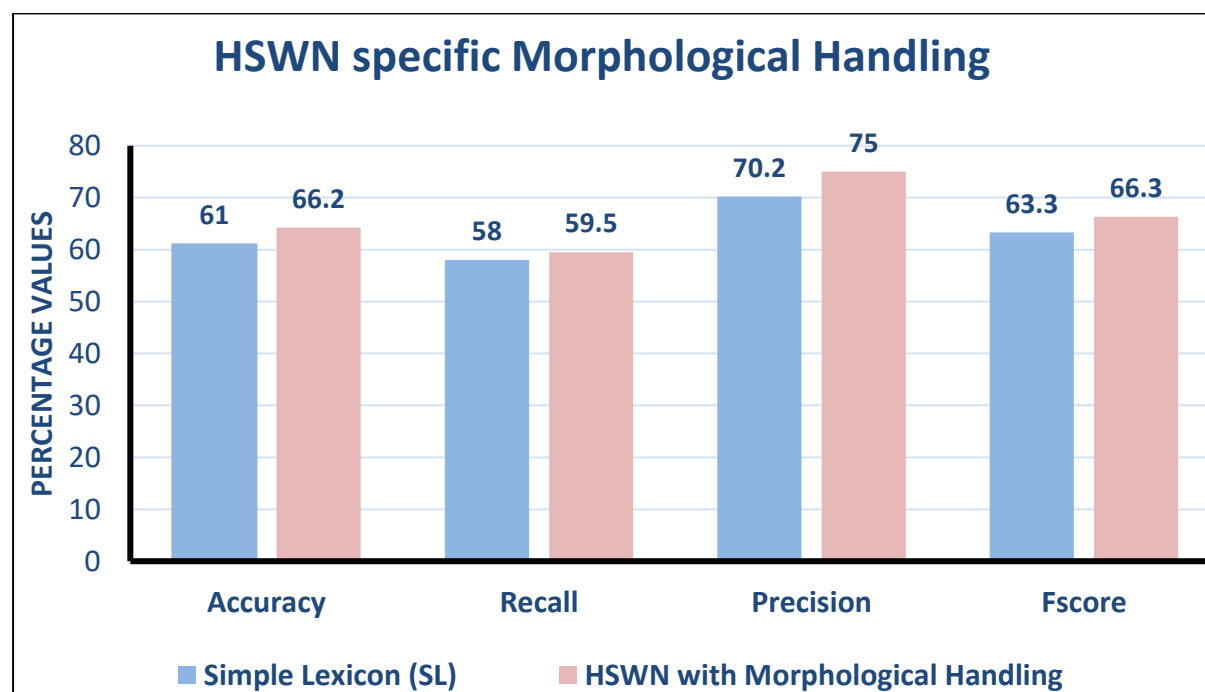


Figure 4.2. Comparison of HSWN specific morphological Handling with Simple HSWN

4.3 Word Sense Disambiguation

A sub-field of NLP is Sentiment analysis and one of the challenging problems of NLP is word sense disambiguation. Finding out correct sense of the word depending on the context is required if an efficient sentiment analysis system needs to be built. Word sense disambiguation problem is related to other applications also such as Machine Translation, speech synthesis, information retrieval, text processing, grammatical analysis etc.

Hindi language is said to be a very ambiguous language and the meaning of the word may be different considering the context in which it is being used. Therefore, an exact sense for the given word should be selected from the Hindi SentiWordNet. Majority times, words from the lexicon are selected directly without handling the word ambiguity issue. The lexicon does include polysemous words i.e., words having more than one meaning and in order to get the exact sentiment score, it is required that the correct sense of the word is extracted from the lexicon.

There are two approaches that have been commonly used for solving the problem of word sense disambiguation namely first sense of the word [5] or taking the average score of the words [6]. In cases where obtaining score of the word that comes first in the lexicon is used, the performance of the opinion mining system may decrease in some cases. When averaging the scores of all senses is used, the results may yield better performance than first sense but the solution does not focus on getting the correct sense of the word and in a way may not consider the effects that it has on the domain knowledge [7]. The two approaches were tried as a part of experimentation on the multidomain dataset. The graph results of the same are shown in Figure 4.3. The results clearly show that Averaging gives better results than first sense and hence may be used as one the approaches. But an appropriate word sense disambiguation method is required which will try to find the correct sense of the word and obtain that respective score from the HSWN.

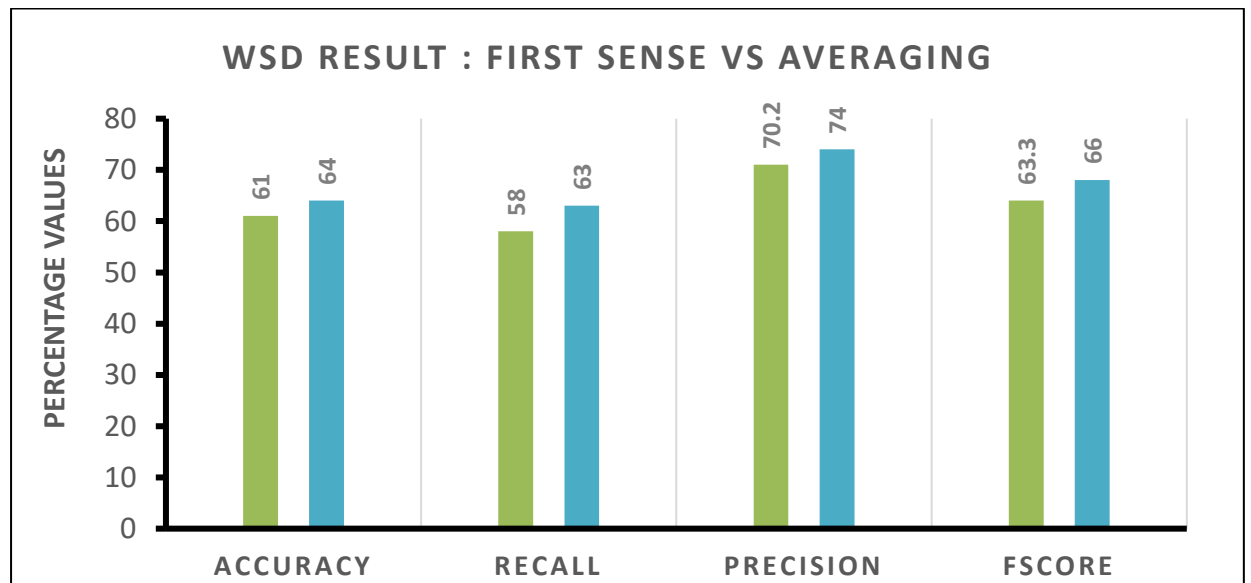


Figure 4.3. Comparison of WSD Results: First Sense and Averaging

Lesk approach for WSD was introduced by Michael Lesk [80] and has been used for English language to solve the problem of word sense disambiguation. In case of the Lesk Approach if there is an ambiguous word present say W and CONTEXT is the collection of the surrounding words in the sentence then the Wordnet is searched and using the overlapping concept for all the senses, the sense which is having the maximum overlapping is selected. The pseudocode of Lesk algorithm is shown in Figure 4.4.

ALGORITHM 3: LESK ALGORITHM FOR WSD

INPUT: *ambiguous word and Sentence S*

OUTPUT: *Sense*

```
1  Initialize variable best-sense to value 0
2  Initialize variable maximum-overlap to value 0
3  CONTEXT = collection of words in a sentence
4  for every sense in senses of word do
5      A = set of words in the gloss of senses from Wordnet
6      overlap = COMPUTEOVERLAP (A, CONTEXT)
7      if overlap > maximum-overlap then
8          maximum-overlap = overlap
9          best-sense = sense
10 return (best-sense)
```

Figure 4.4. Comparison of WSD Results: First Sense and Averaging

The proposed methodology makes modifications to the Lesk approach so as to use it along with the Hindi SentiWordNet to improve the accuracy of the lexicon approach. A graph-based technique along with the LESK approach is implemented to solve the WSD problem. Figure 4.5 shows an example of the graph-based method. Consider that three words W1, W2, W3 are taken as input. The different Senses of every word are extracted from the Hindi wordnet. Lesk approach is used to calculate the similarity index and then the indegree is calculated. S1² has a maximum indegree of 6 and hence, it will be selected as the sense for W1. Likewise, depending on maximum indegree the senses for word W2 and word W3 will be carefully chosen. For all the word senses that have been selected, corresponding scores are got from the HSWN by matching the corresponding synset ids.

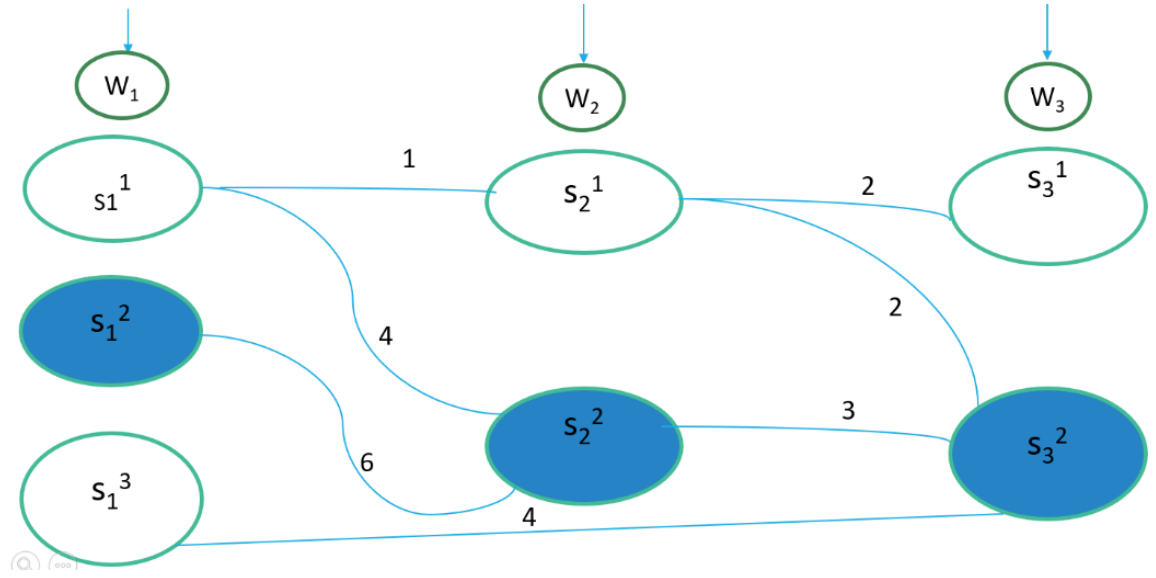


Figure 4.5. Example of Graph based WSD

The Algorithm for Graph-based Modified Lesk Approach (GBMLA) is presented in Figure 4.6. As depicted in the algorithm, the words of a sentence will be taken as candidate vertices of the graph. Senses are extracted from the Hindi Wordnet and are represented as nodes of the graph. Once it is done, using the LESK approach, similarity index between various word senses is calculated using the overlapping concept. An edge is formed amid the two-word senses of the two words. A suitable weight is allocated depending on the similarity index after which indegree of all nodes is calculated and the node having maximum indegree for every word is finally selected. The algorithm uses proximity concept incorporated along with the LESK approach to make it an optimized one. In the proximity-based method instead of finding out the senses of all words in the sentence and then using the LESK approach, only $n/2$ words in the left of the target word and $n/2$ words on the right of the target word are taken. This is based on the fact that neighboring words of the target word contribute more towards the context than those words which are far away from the target word.

ALGORITHM 4: GRAPH-BASED MODIFIED LESK APPROACH (GBMLA)

Input: Sentence S

Output: positive and Negative Score

```
1  Let  $w_1, w_2...w_n$  be words of the sentence
2  for every word  $w$ 
3      find the senses  $s_1, s_2...s_t$  from the Hindi Wordnet and add as nodes of the
4  for every word  $w$  and its respective senses
5      for every  $n/2$  words on left of  $w$  and for every  $n/2$  words on right of  $w$  do
6          sim = calculate Similarity index between word-senses using Lesk
              approach
7          Assign sim as weight of the edge between the two senses
8  Calculate indegree of all the nodes in the graph
9  Select the node(word-sense) for every word which has the maximum indegree
10 for every chosen word-sense
11     match the synset_id with the one in HSWN and return corresponding positive
        score and negative score
```

Figure. 4.6. Algorithm for Graph based Modified LESK approach

The results of GBMLA were compared with first sense and averaging methods and the corresponding results obtained for the multidomain dataset are as shown in Figure 4.7. When HSWN was used with graph based modified Lesk it showed a significant increase in accuracy when compared with other two approaches. This was mainly due to the reason that the scores obtained from the HSWN were extracted depending on the correct sense of the word. The results show that GBMLA approach is significantly better than the other two approaches with respect to all parameters.

Experimentation was done on the Lesk approach to include not only glosses but examples, hypernym, hyponym, meronym and antonym from the Hindi Wordnet [84]. The results show that there was as such no significant change in accuracy, but Recall and F-measure reduced. After trying out different combinations it was seen that glosses along with examples could produce a much better output. Hence, in the final Graph based Modified Lesk method only glosses and examples were considered. Hence the algorithm was designed to produce an efficient and optimized output by considering only glosses and examples for finding overlapping and reducing the overhead of processing of hypernyms, hyponyms, meronyms and antonyms

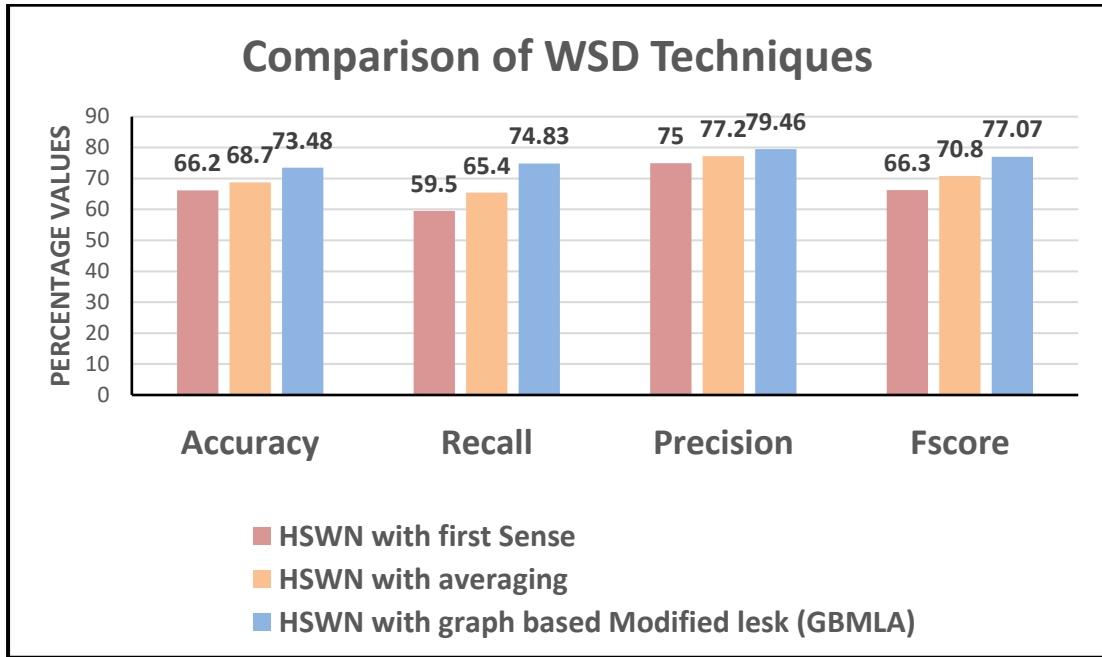


Figure. 4.7. Comparison of GBMLA with first sense and averaging techniques

. When the morphological handling technique and the GBMLA word sense disambiguation technique was used as a part of enhanced lexicon model and experimented on the simple lexicon HSWN approach, the results were substantially better when compared with Simple Lexicon approach. With morphological handling and GBMLA technique the accuracy of simple lexicon was increased from 61% to 73.48% as shown in Figure 4.8.

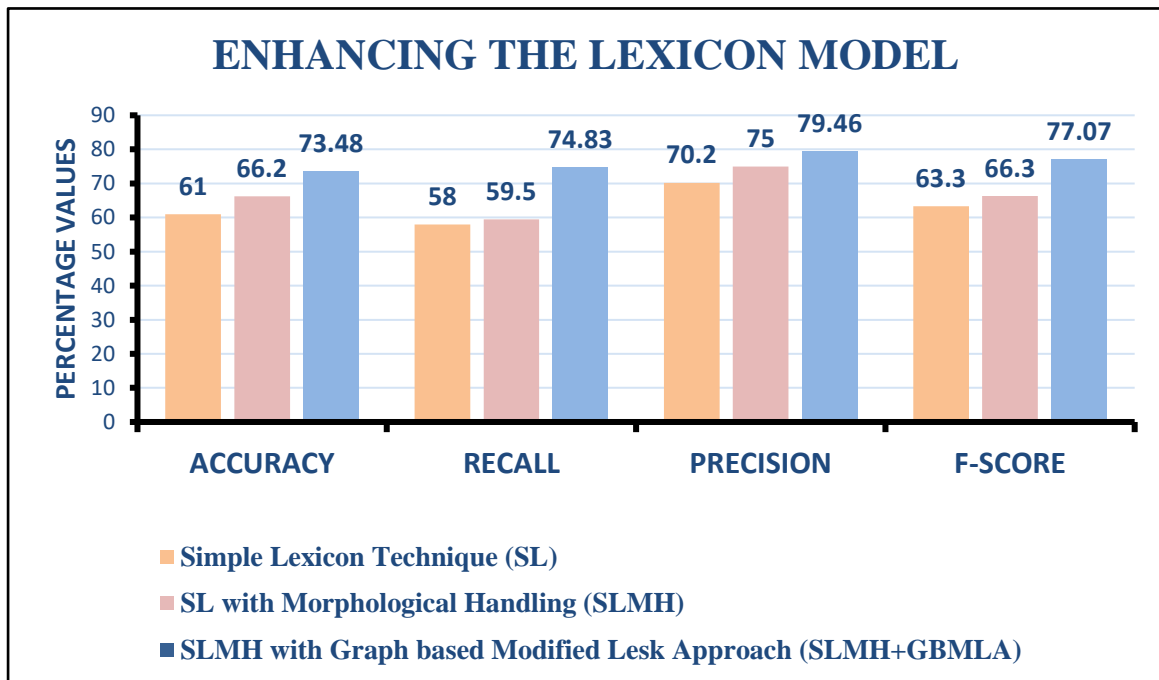


Figure. 4.8. Output of Enhanced Lexicon Model

4.4 Conjunction Handling

Conjunctions such as लेकिन, मगर, और (but, and) are used in Hindi language to link two sentences and they also contribute in determining the polarity of the statement. Conjunctions such as लेकिन, मगर (but) are known to change the orientation to a opposite one whereas conjunction such as और (and) adds strength to the orientation. Handling of conjunctions if done properly can successfully contribute in finding out the exact polarity.

The algorithm for conjunction handling is listed in Figure 4.9. The list of Conjunctions is maintained as shown in the algorithm. The list can be updated by adding new conjunctions if any. Some of the conjunctions which are predominantly used in Hindi language are लेकिन, मगर, किन्तु, परंतु, फिर भी, मात्र, तो, और, तथा. (but, but also, and) The Hindi linguistics study shows that since conjunctions join two statements, in case of SA, the statement that comes after the conjunction generally contains the correct sentiment. This approach can be used for solving the problem of conjunctions. As shown in the algorithm, the first step is to find out whether a conjunction is present in the sentence. If conjunction is present only then those words that appear after the conjunction are considered and included in list T. This means polarity of the sentence depends on the words which are appearing after the conjunction word. Hence, only those words will be used to find out the final score of the sentence.

ALGORITHM 5: ALGORITHM FOR HANDLING CONJUNCTIONS

Input: sentence S

Output: sentence T

- 1 **CONJUNCTIONS:** { लेकिन, मगर, किन्तु, परंतु, फिर भी, मात्र, तो, और, तथा }
 - 2 **T = {}**
 - 3 **for** all words in Sentence S **do**
 - 4 **if** C is a word at position k and C belongs to CONJUNCTIONS **then**
 - 5 add all the words after the conjunction word C from position k+1 to N in T
 - 6 **Return (T)**
-

Figure. 4.9. Conjunction handling

4.5 Modifier Handling

Modifiers are the ones who can modify the intensity and meaning of the linguistic variables. Modifiers are generally divided into three types namely Intensifiers, Diminishers and Negators. A rule-based approach can be devised to handle intensifiers, diminishers and negators efficiently. The rule-based approach alters the sentiment scoring of the word based on some linguistic rules and thus tries to enhance the performance of the sentiment analysis system. The words considered for score alteration are mainly adjectives as according to Hindi linguistic rules it is seen that an intensifier or diminisher may come before an adjective and in that case the score of the adjective word needs to be increased or decreased respectively.

Sentences wherein a negator word comes after an adjective also needs to be handled as it totally reverses the sentence. Such linguistic properties of the Hindi Language are considered for defining the rules so as to handle the modifiers and increase the performance of the sentiment analysis system. Figure 4.10 shows the rule-based algorithm for handling modifiers. Intensifier, Diminishers and Negators are three lists which maintain the respective modifiers. All these together form the modifier lexicon. The commonly used intensifiers and diminishers in Hindi language have been added in the lists.

According to the algorithm, if the word w is adjective and previous word of w is an adverb then it may be an intensifier or diminisher. If it is an intensifier, then we increase the score of w . This may not change the polarity of the sentiment but may be very useful rule in terms of fine-grained sentiment analysis. For example, if a user is interested in differentiating between very positive review and just positive review this rule can turn out be very useful. The second rule considers the diminishers. If the word w is preceded by a diminisher it may change the polarity of the sentiment. If a diminisher is present, then the rule applied is to decrease the score of the adjective word w . The final rule applied is, if the word w is succeeded by a negator, it reverses the polarity. E.g.: अच्छा नहीं, बुरा नहीं (not good, not bad) etc. Thus, the rule-based algorithm tries to handle various types of modifiers which may have an effect on the polarity of the statements. The algorithms for modifier and conjunction handling were implemented and tested on the multidomain dataset.

ALGORITHM 6: ALGORITHM FOR RULE BASED SCORING

Input: Sentence S

Output: Final Score of word

INTENSIFIERS = { 'बहुत', 'बेहद', 'बिल्कुल', 'अत्यधिक', 'अविश्वसनीय', 'सकारात्मक',
'महत्वपूर्ण', 'सचमुच' }

DIMINISHERS = { 'कम', 'थोड़ा', 'मुश्किल से', 'शायद ही कभी', 'कुछ-कुछ' }

NEGATORS = { 'नहीं', 'कभी नहीं', 'ऐसा नहीं', 'कुछ नहीं' }

*if word w at position j in Sentence S is an adjective && is a sentiment word
present in HSWN then*

Extract Pos_Score and Neg_Score of the word w from HSWN

if word x at position $j - 1$ in the Sentence S is an adverb then

if word x is specified in Intensifiers list, then

$$\text{Pos_Score of } w = 2 * \text{Pos_score} - \text{pos_score}^2$$

$$\text{Neg_Score of } w = 2 * \text{Neg_Score} - \text{neg_score}^2$$

else if word x is specified in diminishers then

$$\text{Pos_Score of } w = 1 - \sqrt{1 - \text{pos_score}}$$

$$\text{Neg_Score of } w = 1 - \sqrt{1 - \text{neg_score}}$$

if the next word of w is a negator then

$$\text{Pos_Score of } w = 1 - \text{pos_score}$$

$$\text{Neg_Score of } w = 1 - \text{neg_score}$$

return Pos_Score, Neg_Score

Figure. 4.10. Rule based Algorithm for handling Modifiers

The final enhanced lexicon algorithm is presented in Figure 4.11. The input to the algorithm is a sentence or Review S . Preprocessing is performed which includes tokenization, stopword removal, and POS tagging. If the sentence contains conjunction, the conjunction handling algorithm is called which will in turn return the words after the conjunction as the sentence S . For every word in the sentence S , morphological handling is done by calling the HSWN specific Morphological Handling algorithm. If the number of senses for a particular word in the HSWN is greater than 1 then finding correct sense and getting the score from the HSWN is done by calling the GBMLA algorithm. Once the positive and negative score is obtained, the alteration to the score is done depending on presence of intensifiers, diminishers and negators by using the rule-based scoring algorithm. The total

of positive score and negative score is compared and the maximum of the two is selected as the final polarity of the sentence.

ALGORITHM 7: ALGORITHM FOR ENHANCED LEXICON MODEL

INPUT: Sentence S

OUTPUT: polarity

```

1  Perform preprocessing by applying stop word removal, stemming
2  Perform POS tagging of each word
3  Initialization of List: VotingList = {}
4  Initialization of variables: assign zero to variable pos and neg
5   $S = \text{Conjunction handling}()$ 
6  For every word in sentence  $S$ 
7       $W, \text{Pos\_Score}, \text{Neg\_Score} = \text{HSWN specific Morphological Handling}()$ 
8      Senses = count of  $W$  in HSWN for specific tag  $t$ 
9      if number of senses for  $W$  is greater than 1
10         Pos_Score, Neg_Score = GBMLA ( $S$ )
11     Pos_Score, Neg_Score = Rule_based_scoring (Pos_Score, Neg_Score)
12     If (Pos_Score > Neg_Score) then
13         VotingList.append (1)
14         pos = pos + Pos_score // add positive score from HSWN to pos
16     else if (Neg_Score > Pos_Score) then
17         VotingList.append (0)
18         neg = neg + Neg_Score // add negative score from HSWN to neg
19      $a = \text{number of ones in the VotingList}$ 
20      $b = \text{number of zeroes in the VotingList}$ 
21     if ( $a > b$ ) then
22         polarity = 1 (positive)
23     else if  $b > a$  then
24         Polarity = 0 (negative)
25     else if (pos > neg) then
26         polarity = 1 (positive)
27     else if (neg > pos) then
28         Polarity = 0 (negative)

```

Figure. 4.11 Final Enhanced Lexicon Algorithm

The results of the final enhanced lexicon after performing morphological handling, graph based Modified Lesk WSD and Rule based scoring for Modifier and conjunction handling were significantly better when compared to the Simple Lexicon model. The results of Simple Lexicon HSWN model and Enhanced Lexicon model is shown in Table 4.1 The results show almost 23% increase in accuracy of enhanced lexicon model as compared to the simple Lexicon approach. There was a substantial change in percentage values of Recall, precision and F-score also. This improved performance was a result of extracting the appropriate score from HSWN by handling morphological variations and WSD and also due to handling Negations, conjunctions and diminishers. The graph results of comparison are shown in Figure 4.12.

Table 4.1 Comparison of Simple Lexicon and Enhanced Lexicon model

Algorithm	Simple Lexicon	Enhanced Lexicon
Accuracy	61	83.72
Recall	58	90.4
Precision	70.2	83.5
F-score	63.3	86.8

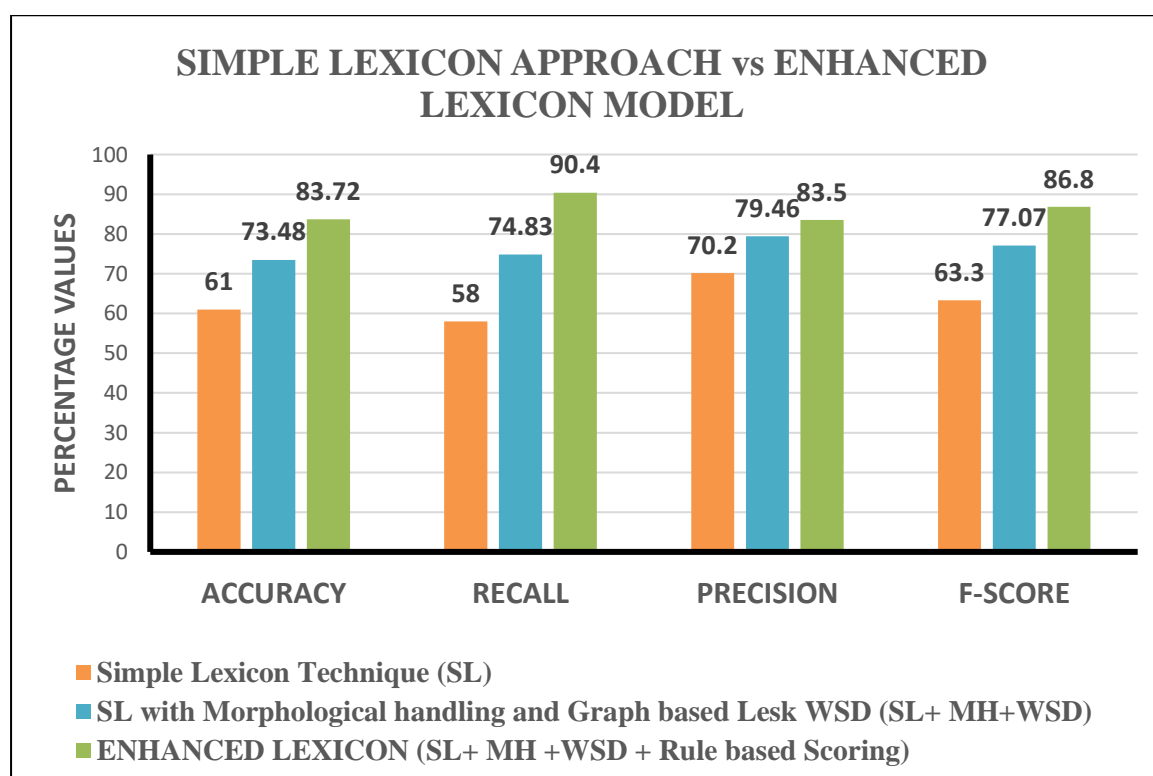


Figure. 4.12. Simple Lexicon vs Enhanced Lexicon Model

4.6 Summary

The Simple Lexicon HSWN approach faces various challenges like Morphological variations, Word sense ambiguity, presence of negators, intensifiers, diminishers, conjunctions which needs to be handled if the sentiment analysis system has to be made efficient.

An algorithm is proposed to handle morphological variations which resulted in 5% increase in Accuracy. A graph based Modified Lesk approached was proposed to solve the problem of word sense disambiguation which further increased the accuracy of the system. The presence of modifiers such as intensifiers, diminishers and negators along with conjunctions were handled using a Rule based scoring method which altered the final score and helped in determining the correct polarity of the sentence.

The enhanced Lexicon model performed considerably better than the simple Lexicon approach and also surpassed the accuracies obtained by individual machine learning classifiers that were experimented as a part of investigation of the Lexicon approach. The next chapter presents a hybrid model by combining the enhanced Lexicon model with the machine learning classifiers and shows its effectiveness in comparison with Simple Lexicon approach.

Chapter 5

Building a Hybrid Model and evaluating its efficiency and effectiveness

5.1 Introduction

. As already discussed in chapter 1, the two main approaches used for performing sentiment analysis in Hindi are lexicon-based approach and Machine learning approach. Lexicon approach is very simple but is reliant on a dictionary. The quality of the lexicon decides the output and the problem of absence of lexicons can affect the accuracy of the classification. Machine learning approaches are considerably more accurate as compared to Lexicon based techniques and they do not require a well-built lexicon dictionary. But for performing classification, they do require as input a properly labelled training dataset.

A combinational method may be proposed which makes use of knowledge-based approach as one phase that aims to provide stability and statistical approach as another phase which aims to provide more accuracy. The amount of training data that is required by the statistical approach is considerably high. Further, the models built by the statistical approaches may be domain specific which means the same model may not work well across different domains. On the other hand, the lexicon-based technique which is a knowledge-based method may show a steady and reliable performance, even if different domains are considered.

Considering the pros and cons of both the techniques it turns out to be beneficial to combine the advantages of both the techniques and build a hybrid model which could perform SA on the Hindi text. This chapter explores the efficiency and effectiveness of the hybrid model built for performing sentiment analysis on Hindi text. The output of the hybrid model was compared with simple lexicon method and empirical evaluation was done on three different datasets.

5.2 Proposed Hybrid Model

5.2.1 Building the Hybrid Model

As discussed in Chapter 4, the lexicon approach is enhanced by proposing methods for morphological analysis, word sense disambiguation, negation handling and Intensifier/Diminisher handling. The enhanced Lexicon model performed better than the simple Lexicon approach and gave considerable accuracy. When the results of the enhanced Lexicon model were evaluated, it was seen that some of the sentences were not classified into positive and negative and remained as unclassified (neutral class) mainly due to absence of lexicons. To solve the problem of these unclassified text reviews, a hybrid model was built as shown in Figure. 5.1. The proposed hybrid model combines the enhanced Lexicon model and the Classifier model built by the Learning algorithm to produce more accurate results.

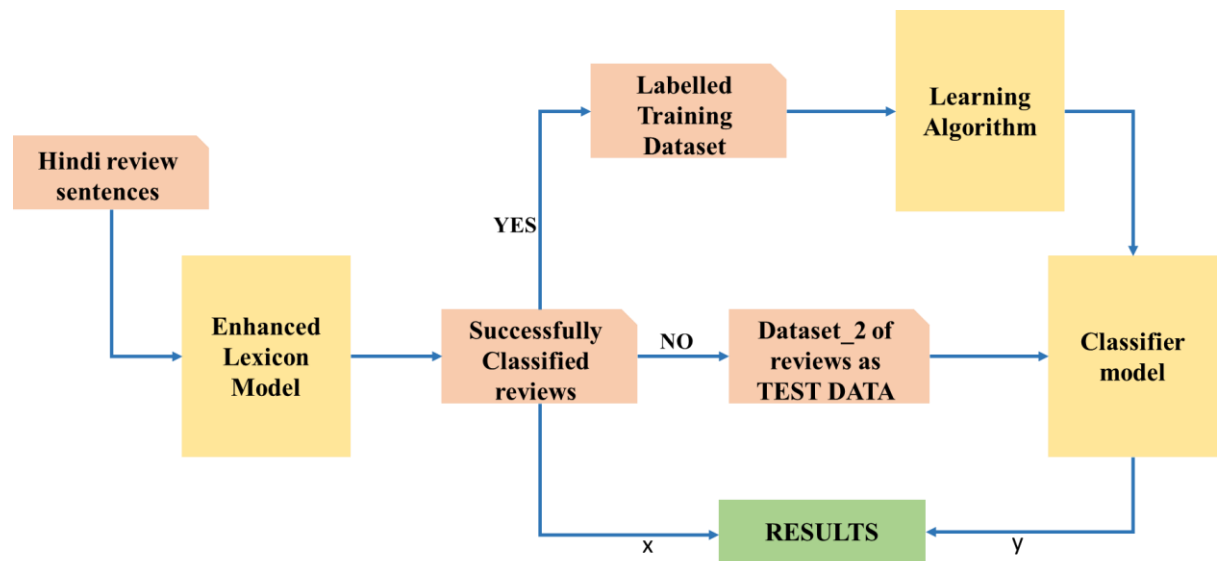


Figure. 5.1 Proposed Hybrid Model

The Hindi text reviews which need to be classified is given as input to the Enhanced Lexicon model which performs the classification. All the sentences which are appropriately classified (result x) and hence labelled, are given as a training dataset to the Learning algorithm which produced a classifier model by getting trained from the successfully classified reviews. The sentences which were not properly classified or the unclassified review sentences were then given as test data to the classifier model and the results of the model (y) were obtained. The output of the hybrid model was thus taken as accuracy

obtained from the enhanced lexicon model(x) along with the accuracy obtained from the classifier model (y).

In terms of calculation, the final Accuracy (FA) of the hybrid model was given by:

$$FA = x + (100 - x) * (y / 100) \quad (5.1)$$

where x = accuracy obtained from the enhanced lexicon approach

y = accuracy obtained from the classifier model.

5.2.2 Checking Effectiveness of the hybrid model

To check the effectiveness of the proposed hybrid model, the following was done:

- Different Classifiers were used to test which works better and fits best as a part of the hybrid model
- The model was tested on different datasets such as:
 - a. Single Domain: Movie Review Dataset Of 1200 sentences
 - b. Multi-Domain: A dataset of 4000 sentences which includes combination of Movie reviews, Product reviews and tourism reviews.
 - c. BHAAV Emotion Dataset: A dataset of around 20,304 sentences containing 4 classes Joy, Sad, Anger, Suspense and Neutral.
- The computational complexity of the hybrid model was evaluated by calculating the time taken for the hybrid model.

5.2.3 Results of Single domain dataset

The movie review dataset when used on the simple Lexicon approach gave an accuracy of around 53.52%. The Enhanced Lexicon model increased the accuracy from 53.52 % to 64.75 %. The successfully classified reviews were given as input to different learning algorithms and the accuracy was obtained for unclassified reviews as shown in Figure 5.2. Random Forest Classifier proved to be better in categorizing the unclassified sentences by achieving an accuracy of 78.3%. Various classifiers were evaluated as a part of the hybrid model and the final accuracy for the various classifiers were obtained using the Equation 5.1. The accuracies and the F-measure for the hybrid model considering different classifiers is shown in Figure 5.3. The best classifier which could be used as a part of the hybrid model was Random Forest Classifier and it obtained a maximum accuracy of 92.5%.

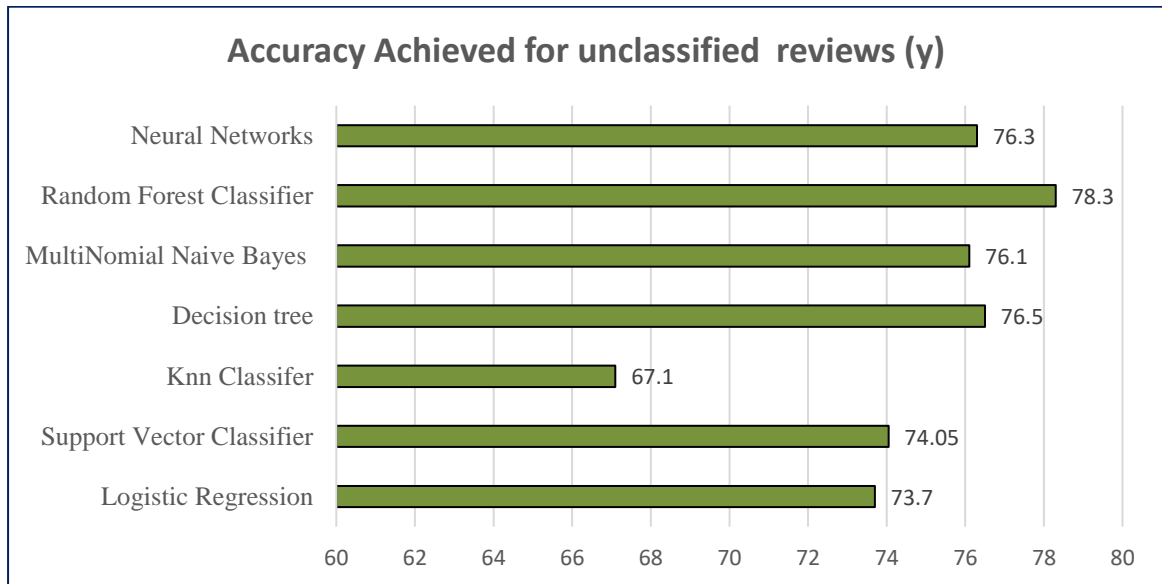


Figure.5.2 Accuracy obtained for unclassified reviews for Single domain dataset

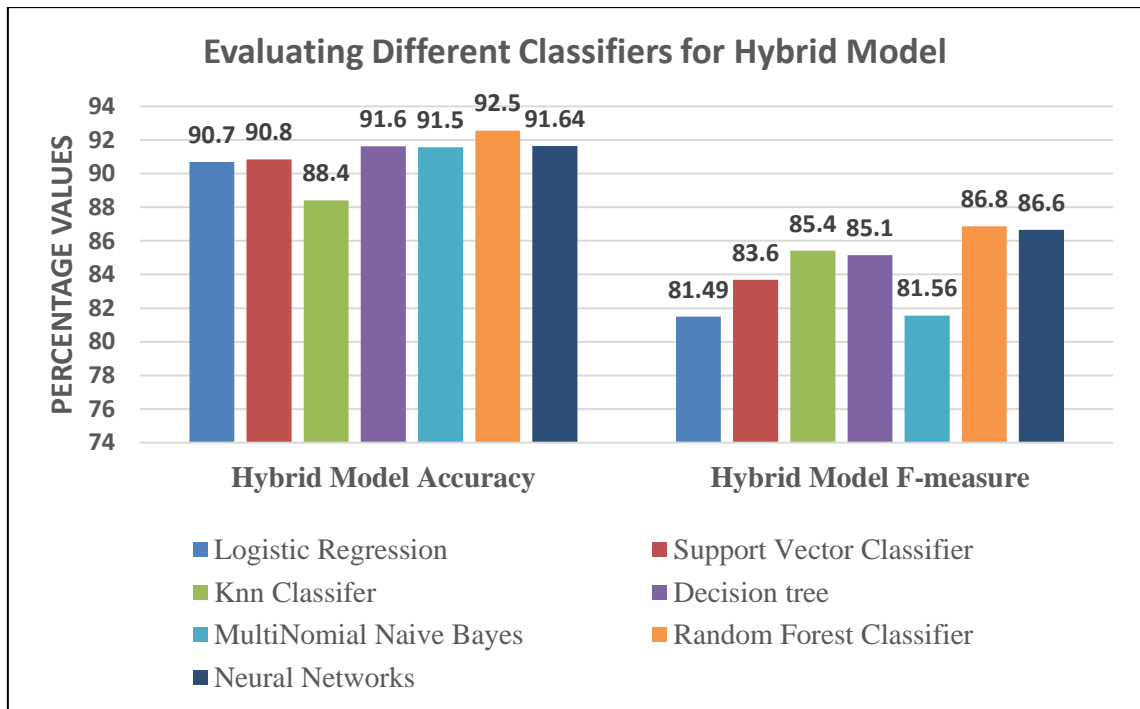


Figure.5.3 Evaluating classifiers as a part of Hybrid model for Single domain dataset

5.2.4 Results of Multi domain dataset

The multidomain dataset consisting of reviews from movies domain, product domain and tourism domain when used on the simple Lexicon approach gave an accuracy of around 61.2%. The Enhanced Lexicon approach which included Morphological Handling, WSD handling and Negation Handling increased the accuracy from 61.2 % to 83.4%. The reviews which were successfully classified are then given as input to various machine

learning algorithms and the accuracy is obtained for unclassified reviews as shown in Figure 5.4. Random Forest Classifier and Multinomial Naïve Bayes proved to be better in categorizing the unclassified sentences by achieving an accuracy of almost 61%. Various classifiers were evaluated as a part of the hybrid model and the final accuracy for the various classifiers were obtained using the Equation 5.1. The accuracies and the F-measure for the hybrid model considering different classifiers is shown in Figure 5.5.

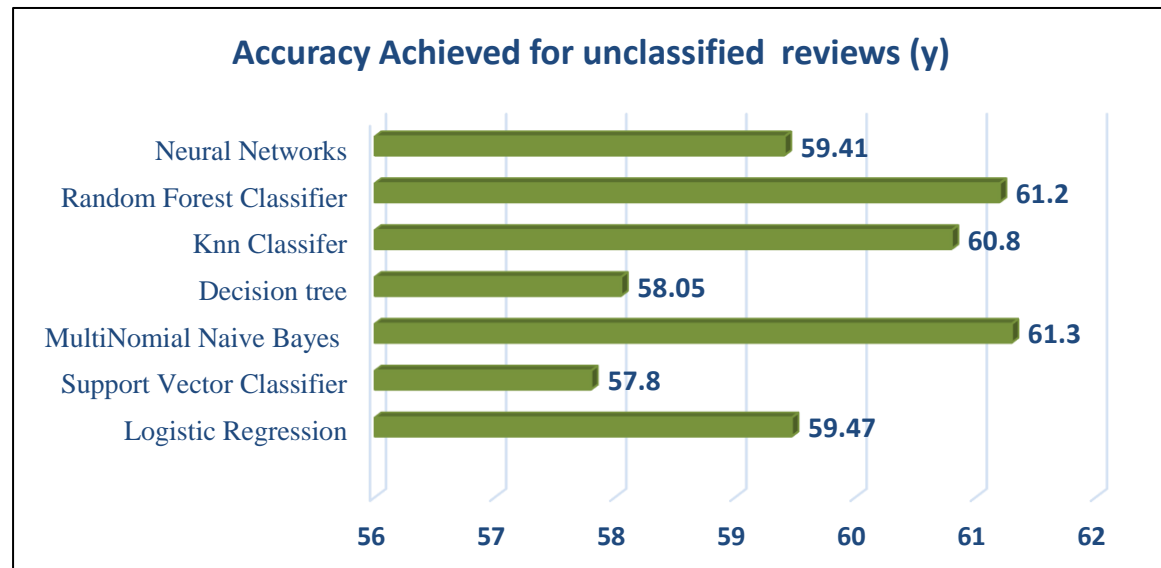


Figure.5.4 Accuracy obtained for unclassified reviews for Multidomain dataset

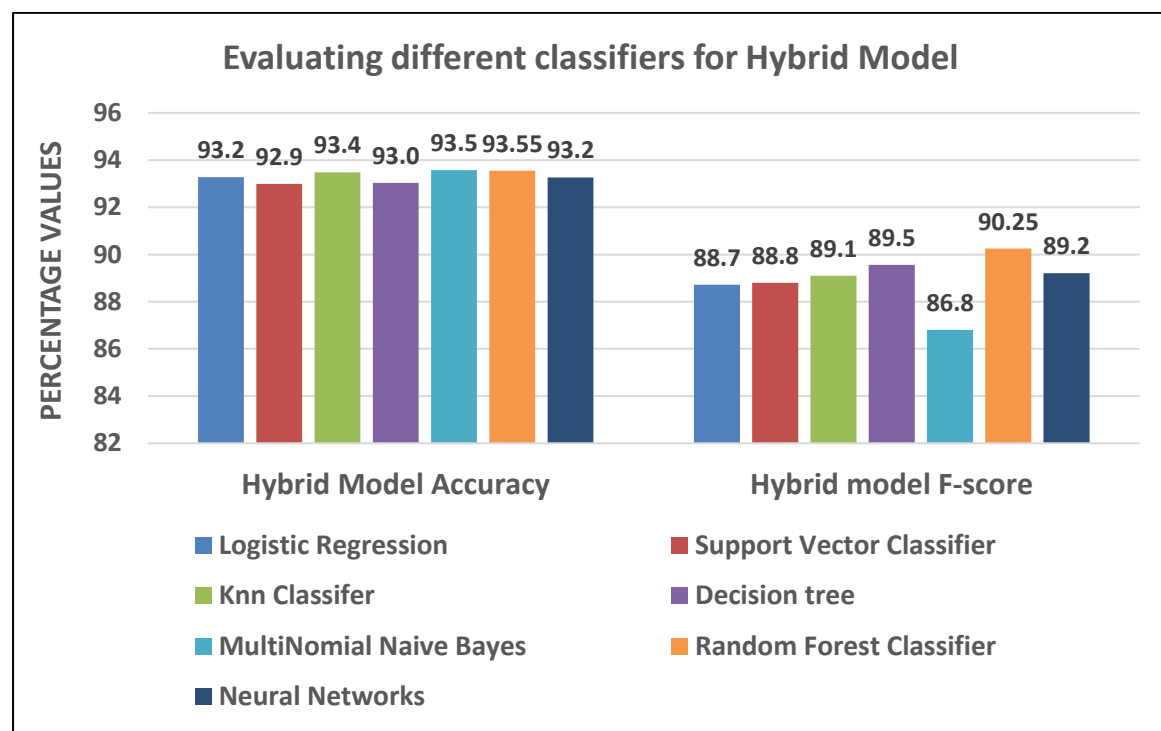


Figure.5.5 Evaluating classifiers as a part of Hybrid model for multidomain dataset

5.3 Hybrid Approach for Emotion Classification

Emotion is a very important part of human expression and experience. There are different kinds of emotions such as happy, sad, angry, fear, disgust, surprise etc. Text based emotion recognition is a very challenging task and finding out the human emotion in textual data is turning out to be very significant from an application viewpoint.

Text based Emotion mining is divided into two types namely Emotion detection which deals with detecting whether the sentence contains an emotion and Emotion classification which deals with categorizing the sentence based on emotions such as happy, sad, fear etc. Emotions can be explicit wherein the emotion word is present in the sentence or can be implicit wherein no emotion words are present in the sentence. Example: A sentence like “ मुझे बहुत खुशी हुई”

[I am very happy]

gives an explicit emotion happy but a sentence like

“क्या तुम ठीक हो?”

[Are you ok?"]

shows a caring emotion but is implicit.

Very little work has been done in Hindi language emotion classification. There are very few built resources as such which are available for emotion mining. This work mainly aims at creating a highly covered emotion lexicon resource which can be used for emotion mining and then building the hybrid approach and evaluating for multiclass classification.

5.3.1 BHAAV dataset

BHAAV dataset is a well build corpus containing around 20,304 sentences and it was developed by Yaman Kumar et al [78]. This was constructed using 230 various short stories that included around 18 different genres. Stories are a unique way of expressing emotions because they include different characters and various plots which have been used by the author in his writing. In case of BHAAV dataset, all the text that was extracted from stories were split into sentences using an automated method. There were five emotions considered and annotation of sentences was done by well-educated Hindi speakers. BHAAV dataset is a valuable resource that is developed for Hindi language and hence has been used for experimentation.

There were five different classes that were considered in BHAAV dataset which form a good dataset for multiclass classification. The five classes and their respective count of sentences are as shown in Table 5.1

Table 5.1 Sentences in BHAAV dataset

Type of Emotion	Total number of sentences
Sad	3168
Suspense	1512
Neutral	11,697
Joy	2463
Anger	1,464

BHAAV dataset has been tested previously for performance using various supervised machine learning classifiers like logistic regression, SVM, Random-Forest, CNN etc. [78]. The results on this dataset obtained by the authors who built the dataset are mentioned in the Table 5.2

Table 5.2 Performance of BHAAV dataset in [78]

Name of Classifier	Accuracy
Logistic Regression	62%
SVM	52%
CNN	55%
Random forest	59%
BLSTM	60%

According to the results obtained in [78] Logistic Regression performed well among all the classifiers obtaining an accuracy of 62%. The results will be compared with the results of the built hybrid model.

5.3.2 Creation of Emotion Lexicon

In case of opinion-based mining, HSWN can be used as the lexicon for obtaining the polarity of the words and getting the exact sentiment. Such built-in resources are very less in case of emotion sentiment analysis. Taking this into consideration, an emotion lexicon was built using EmoSenticNet [82][83] and NRC-EmotionNet [84]. A well-known lexical resource of English language is EmoSenticNet that is known to allocate six different WordNet labels of emotion namely joy, anger, disgust, fear, surprise and sad. Using Google Translate the words were translated into Hindi. To add more lexicon words in the lexicon resource NRC EmotionNet also was used for including emotion lexicon words. The NRC

EmotionNet consists of eight different emotions namely joy, surprise, anger, disgust, fear, anticipation, trust and sad. The NRC Emotion has already converted its English version of the lexicon in other languages by translating using Google Translate. So, the Hindi conversion was included in building the emotion Lexicon resource by eliminating the duplicate words that were already added by EmoSentNet. Using the EmoSentNet and NRC Emotion Net, a new Emotion Lexicon resource was built which contained Emotions Sad, Joy and Anger. Since Google translation was used it was most important to incorporate human validation. A validator who has a command over the language evaluated the built lexicon.

Once the validation is over, to further enhance the emotion Lexicon, synonyms of the Hindi words that were present in the built EmotionNet are added if not already added in the Emotion Lexicon. The group of words along with the respective synonyms is referred to as a seed. Each word which is validated in the seed lexicon is further extended using the Hindi Wordnet. It is verified before adding the synonyms that they have been included from synsets that refer to emotions.

To test on the BHAAV dataset, another emotion that was required was Suspense. The suspense emotion was created by a thorough study and with the help of two well-known Hindi annotators who helped in finding and annotating the words to the proper emotion. Once this was done, the final Emotion Lexical Resource was built with the count of words as mentioned in Table 5.3. The total number of Lexicons in the built Emotion lexicon is 7318.

Table 5.3 Total number of Lexicons in the built Emotion Lexicon

Type of Emotion	Total number of lexicons
Sad	1495
Suspense	653
Joy	3515
Anger	1655
TOTAL	7,318

Once the emotion lexical resource was built, following procedure was followed for sentences in the BHAAV dataset:

- Preprocessing which includes tokenization and stop-word removal is done on the extracted Hindi sentences of the BHAAV dataset.
- POS tagging is done as the next step in Preprocessing. Adjectives, adverbs and nouns are extracted as the target words of the sentence.

- The target words are then compared against the built Emotion Lexical resource to find the precise emotion of the sentence.

The results are obtained after using the built-in Emotion lexical resource. Out of all the sentences of the BHAAV dataset, the ones which are appropriately classified (result x) by using the Emotion lexical resource and hence are labelled are then given as a training dataset to the different Machine Learning algorithms. These classifiers yield a model by getting trained from the successfully classified reviews. The sentences which are not correctly classified i.e., the unclassified review sentences are then given as test data to the classifier model and the results of the model (y) are obtained as shown in Figure 5.6. As shown Random Forest and Multinomial Naïve Bayes could classify the unclassified reviews better compared to other classifiers. The output of the hybrid model is taken as accuracy obtained from the enhanced lexicon model (x) along with the accuracy obtained from the classifier model (y).

The results of the hybrid model for emotion multiclass classification are shown in Figure 5.7. In terms of accuracy, Random Forest and Multinomial Naïve Bayes showed good results and when F-score results were investigated Support Vector Classifier turned out to be better than other classifiers. Decision Tree and KNN classifiers did not perform well. Though the results of Decision trees were not up to the mark, Random Forest does perform well as it considers the power of multiple decision trees. Decision trees have a tendency to give higher importance to a particular group of features whereas Random Forest is known to select the features in a random fashion when training process is being done. This means, Random Forest classifier does not rely majorly on some particular feature which is why its results are way better than Decision trees. Due to the randomized feature selection and the advantage of generalizing over the data in a much better way, results of random forest have turned out to be more accurate.

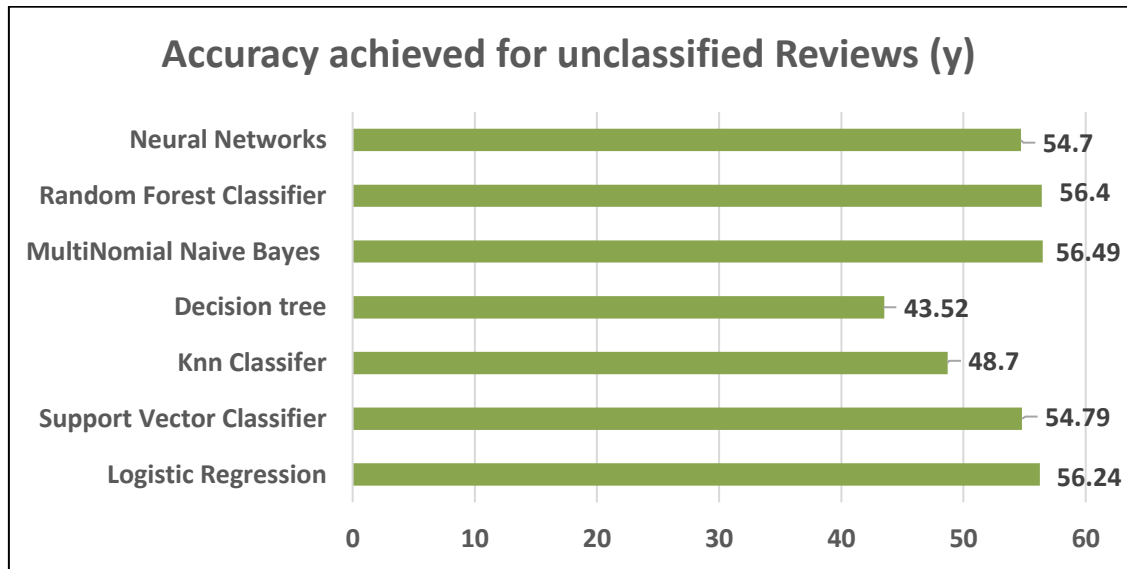


Figure.5.6 Accuracy obtained for unclassified reviews for BHAAV dataset

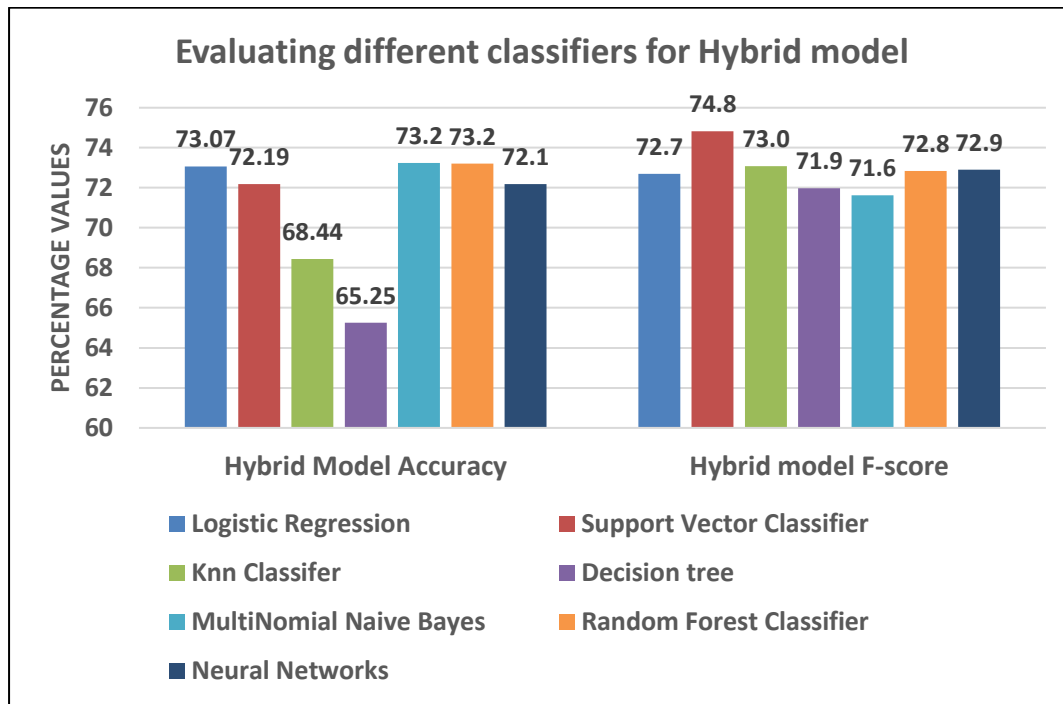


Figure.5.7 Evaluation of different classifiers for Hybrid model for BHAAV dataset

The accuracy of individual classifiers obtained on BHAAV dataset in [63] are compared with the hybrid model wherein the classifier was used as a part of hybrid model trained by the output resulting from the emotion lexicon. The comparison is as shown in Table 5.4.

Table 5.4 Comparison of results of BHAAV dataset in [78] with Hybrid model

Name of Classifier	Accuracy obtained in [78]	Accuracy when classifier was used as a part of Hybrid model
Logistic Regression	62%	73.07 %
SVM	52%	72.19 %
Random forest	59%	73.2 %

The important findings of the experimentation of the hybrid model are:

- The Hybrid model shows substantial increase in accuracy since emotion lexicon was built and when the correctly classified sentences were given as input to the ML classifiers, they significantly decreased the number of unclassified reviews.
- Random Forest Classifier when used as a part of the hybrid model achieved better results across various datasets due to its ensemble approach and its ability to work on unbalanced dataset.

5.3.3 Time complexity of Hybrid model

As mentioned earlier, when two approaches are combined along with the advantages, its limitations also have to be considered. The major disadvantage of any hybrid model is its runtime complexity since time taken by both the approaches is considered. In case of the Machine Learning models the training time taken for various models is different and should be considered as it can have an overall effect on the hybrid model time complexity. Further as the size of the training dataset goes on increasing the time taken may increase. Taking this into consideration a graph was plotted with the training dataset size along with time taken for training as shown in Figure 5.8.

The results in Figure 5.8 clearly show that as the dataset size increases the training time taken by the classifiers also increases. Multinomial Naïve Bayes is the only classifier taking very less time for training. Support Vector Classifier, Random Forest classifier and Neural Networks have a higher training time when compared to other classifiers

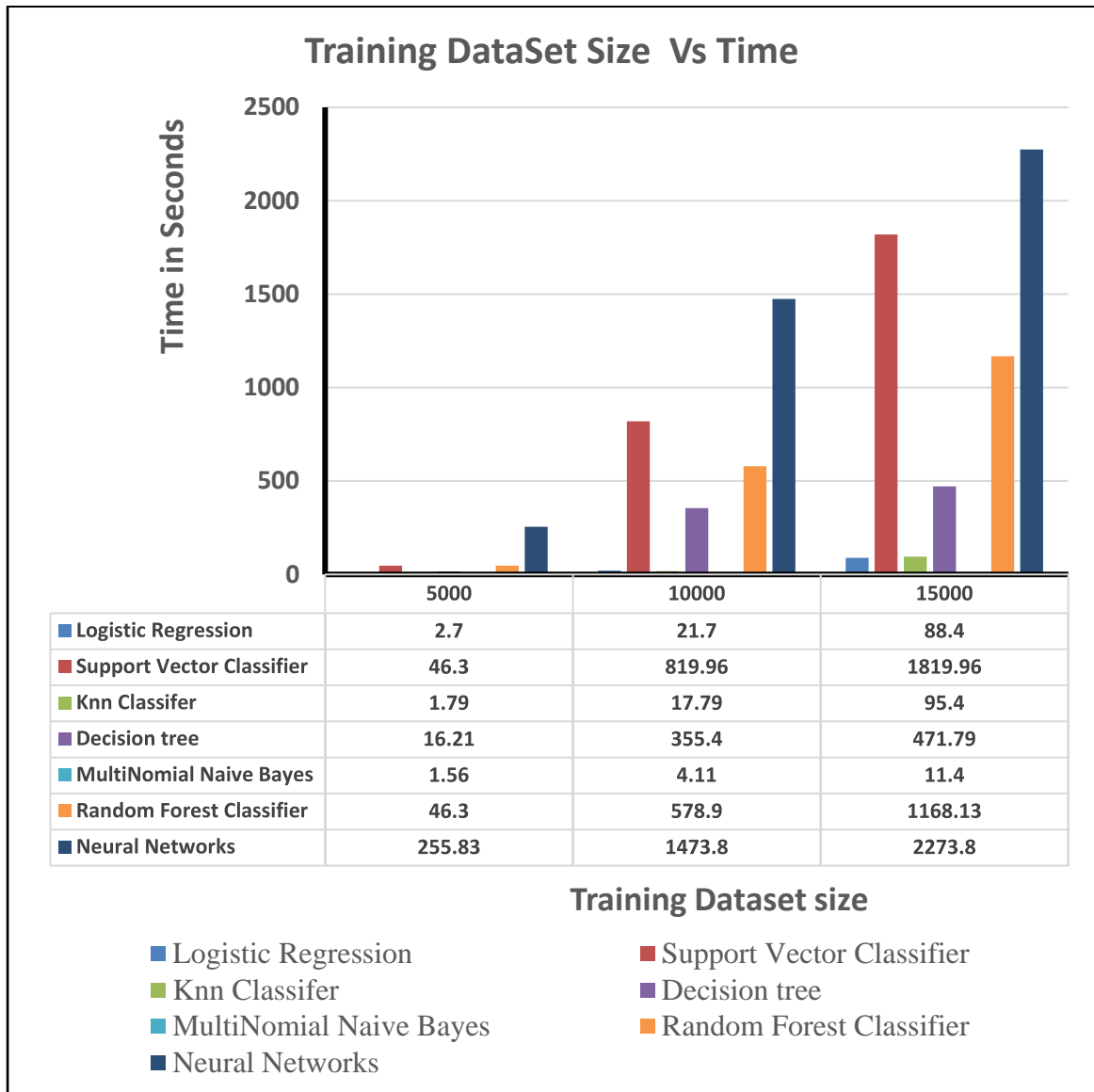


Figure.5.8 Training Dataset Size Vs Time taken for training

The accuracy of hybrid model considering Random Forest gave very good results for all the three datasets used for experimentation. But it should also be taken into consideration that as the number of trees in case of a random forest increases, the time taken to train each of the trees also increases. This can turn out to be very crucial in time constraint applications.

The selection of classifier for the hybrid model can be based on the tradeoff between accuracy and training time constraint. Table 5.5 shows the accuracy and Training Time comparison of two models, Multinomial Naïve Bayes and Random Forest classifier.

Table 5.5 Comparison of Multinomial Naïve Bayes and Random Forest classifier for all three datasets

	Training dataset size	Multinomial Naïve Bayes		Random Forest	
		Accuracy	Time taken	Accuracy	Time taken
Single Domain Dataset (1200 sentences)	777	91.5	0.96	92.5	8.4
Multidomain Dataset (4000 sentences)	3336	93.7	2.05	93.16	187
BHAAV Dataset (20,304 sentences)	6497	73.2	10.4	73.2	668.13

Considering the results of comparison, it is evident that Multinomial Naïve Bayes can be a perfect fit in the hybrid model which has almost similar accuracy results as of Random Forest and takes very less training time.

5.4 Summary

A hybrid method is proposed that combines a knowledge-based approach that aims to provide stability and statistical approach which aims to provide more accuracy. The amount of training data required by the ML approach is noticeably high. Further, the same model may not work well across different domains whereas the lexicon-based technique may show a steady performance, across different domains. Considering these pros and cons of both the techniques a hybrid model is built which could perform SA on the Hindi text. The output of the hybrid model was compared with simple lexicon method and empirical evaluation was done on three different datasets namely Single domain dataset, Multidomain dataset and BHAAV emotion dataset.

After analyzing the accuracy results it is evident that, the hybrid model works better compared to individual classifiers and the simple lexicon approach. A comparison was made with the dataset size used for training and the time taken by classifiers so as to build an efficient hybrid model. In this case, Multinomial naïve Bayes, when used as a part of Hybrid model gives good results. Overall, the hybrid model is a worthy approach to be considered for Hindi Sentiment Analysis. The next chapter investigates the effect of word count on the accuracy parameter and presents some interesting findings.

Chapter 6

Investigating the impact of length of messages on sentiment classification accuracy

Different datasets have different properties and it is interesting to know what impact these properties may have on the results. The performance of sentiment analysis is affected by different properties like dataset size or training corpus size, feature extraction, length of target documents, preprocessing constraints, length of the review/word count, readability, purity of data, subjectivity of data etc. Efforts need to be taken to not only increase the performance of the algorithms but to investigate the impact of some of the linguistic properties of the dataset on the performance. One such data property is Word count.

Majority of existing research aims at just providing comparison of performance in terms of the accuracy of different algorithms but does not consider the role of various data properties and settings which may result in alterations in the accuracy. The results and the performances may differ depending on the nature of the data used and the way experimentation was done.

6.1 Impact of Wordcount

It has been seen that the length of words in reviews varies widely depending on the domains. Now-a-days well known review websites or social networking platforms enforce its users to write comments or reviews by providing them with a defined length. Twitter is a well-known online micro-blogging and social-networking platform that allows its users to write a message of maximum length 280 characters. However, the exact nature of the relationship between the length of textual data and classifier performance has been unclear. The main objective of this study was to analyze the effect of length of words of a review/sentence on accuracy of sentiment analysis.

The steps and the approaches used in the investigation are as shown in Figure 6.1.

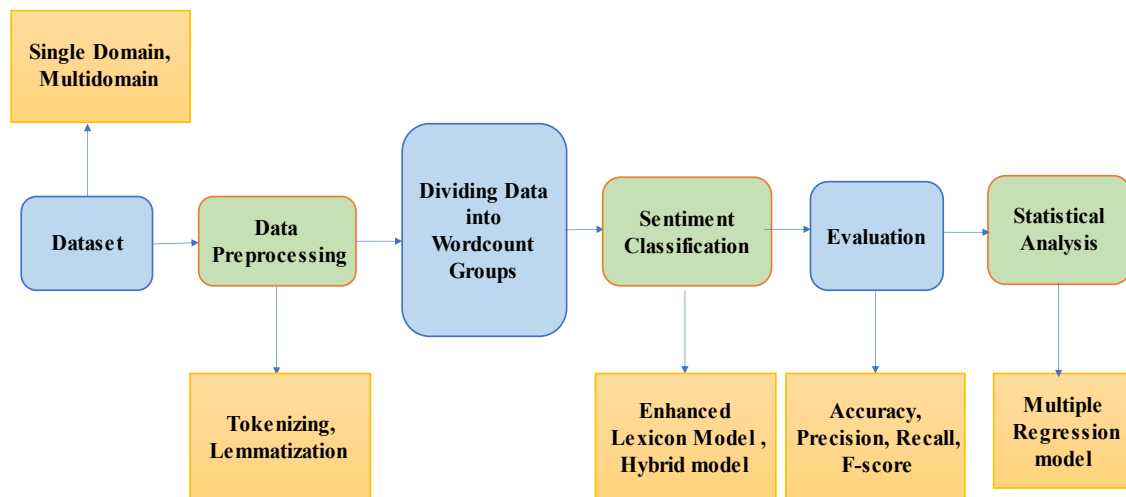


Figure.6.1 Steps and Approaches used for investigation of impact of wordcount

As shown in figure 6.1, in order to investigate the sensitivity or the impact of wordcount on the sentiment classification accuracy, the sentences in the dataset are preprocessed and then are split into several categories and each category or group has a different word-count. The four groups of wordcount that are formed for the investigation are:

- Sentences having wordcount 1-50
- Sentences having wordcount 51-80
- Sentences having wordcount 81-120
- Sentences having wordcount > 120

The next step is to perform the sentiment classification using the two proposed models enhanced lexicon model and the hybrid model and then evaluate the results considering the main parameter accuracy as well as other parameters Recall, precision and F-score. To verify whether the investigation being done is statistically significant, a statistical analysis is done using the Multiple Regression model.

The first approach experimented for effect of wordcount was the Enhanced Lexicon model. The enhanced Lexicon model was tested on both the single and multidomain datasets to find the effect of wordcount on Accuracy and other parameters by considering the four wordcount categories. Figure 6.2 shows the impact of wordcount on Accuracy for single domain dataset of 1200 sentences. The graph clearly shows that for the group 1-50 the

accuracy is 66.78 %. As the number of words increases for the next wordcount group the accuracy slightly decreases. For the sentences having words greater than 120, the accuracy is decreased by almost 4% which means as the number of words go on increasing the accuracy goes on decreasing. This may happen due to the fact that when less words are used in reviews, they tend to be more specific about the sentiment. As the number of words goes on increasing, due to noise, analysis of the reviews is affected and accuracy is decreased significantly.

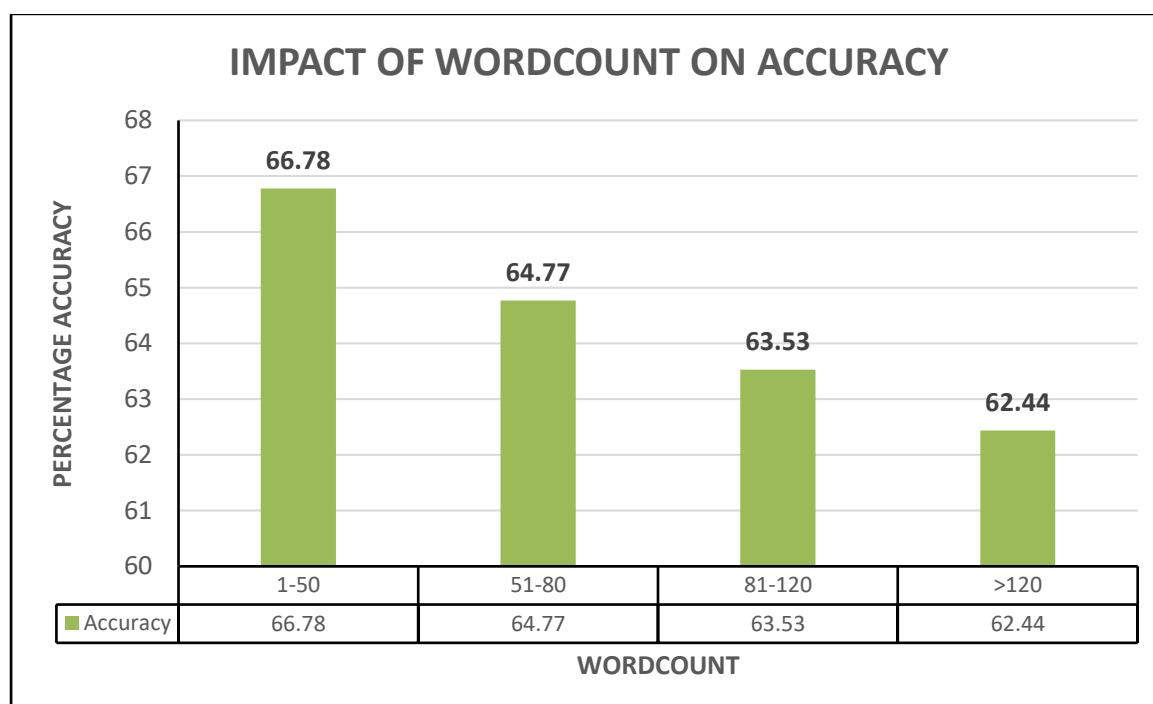


Figure.6.2 Effect of wordcount on Accuracy of enhanced Lexicon model for Single Domain dataset

The effect of wordcount needs to be checked for other parameters also. Figure 6.3 shows the impact of word count on Recall, Precision and F-score. Effect on Recall is same as Accuracy. As the wordcount increases Recall decreases. There is a slight increase in Precision as word count increases. But since Recall decreased significantly, the impact was seen on F-score also. There was a significant decrease in F-score results as the wordcount increased.

The effect of wordcount on accuracy of enhanced Lexicon model was also tested for multidomain dataset containing 4000 sentences for various wordcount groups. The results of the effect of wordcount on Accuracy for multidomain dataset is shown in Figure 6.4. A drop in accuracy by 4% was seen for wordcount group 81-120. For the words greater than 120 the accuracy was almost 10% less as compared to first wordcount group of 1-50 words.

The impact of wordcount was also checked for other three parameters Precision, Recall and F-score and the results of the same have been shown in the Figure 6.5. As the wordcount increased, Recall decreased. There was as such no effect on Precision as the wordcount increased. But since there was a decrease in Recall, the impact was seen on F-score also.

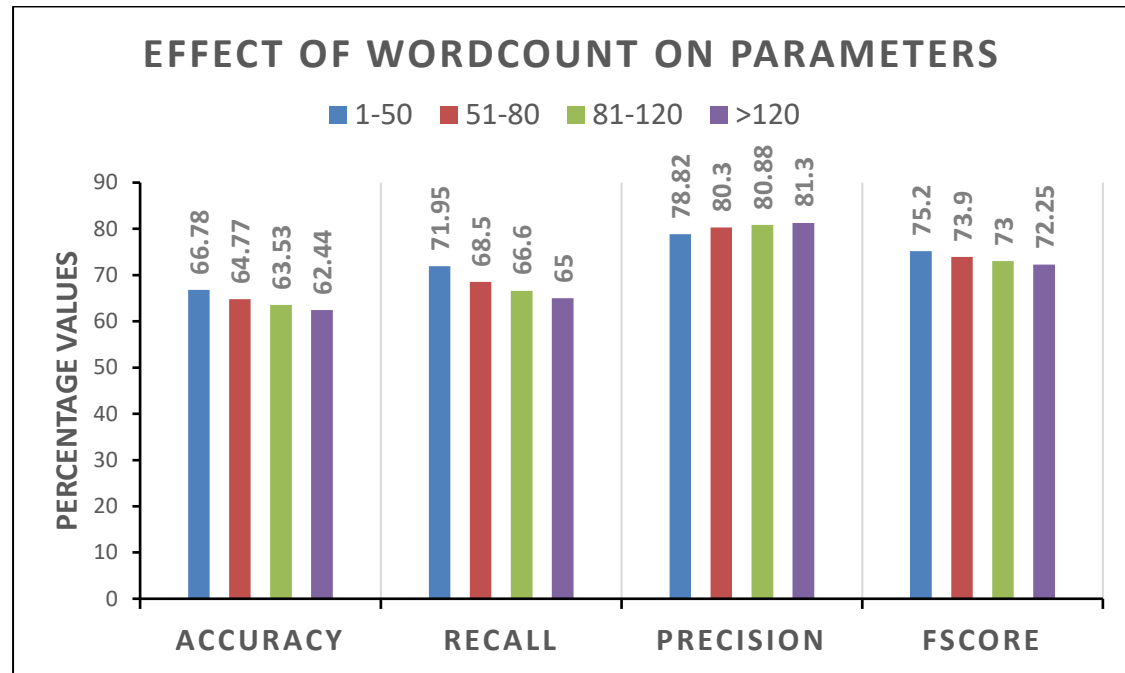


Figure.6.3 Effect of wordcount on Recall, Precision and F-score of enhanced Lexicon model for Single domain dataset

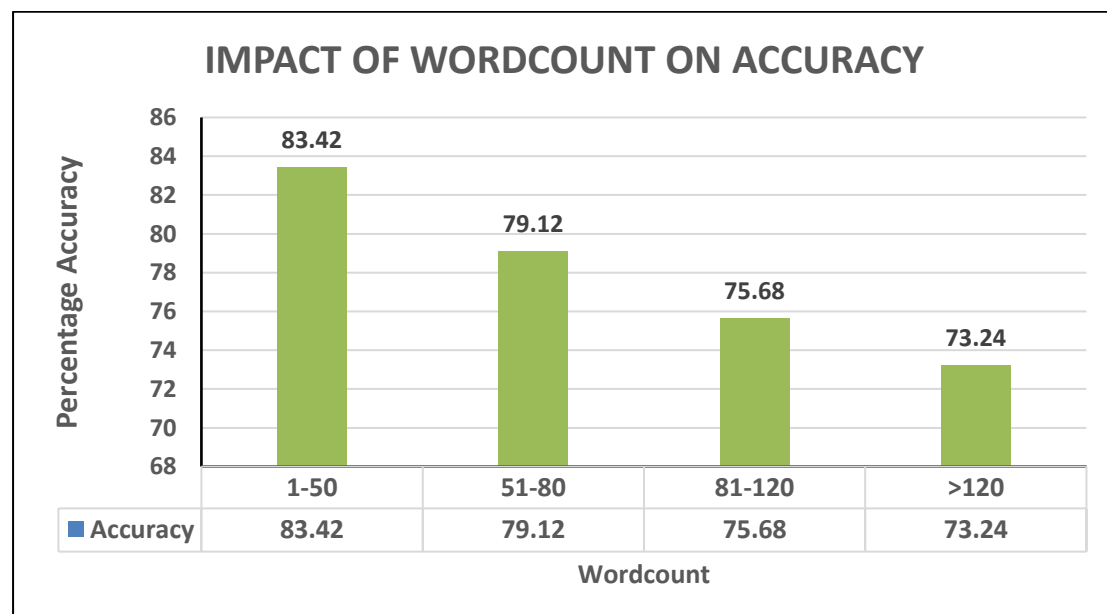


Figure.6.4 Effect of wordcount on Accuracy of enhanced Lexicon model for Multidomain dataset.

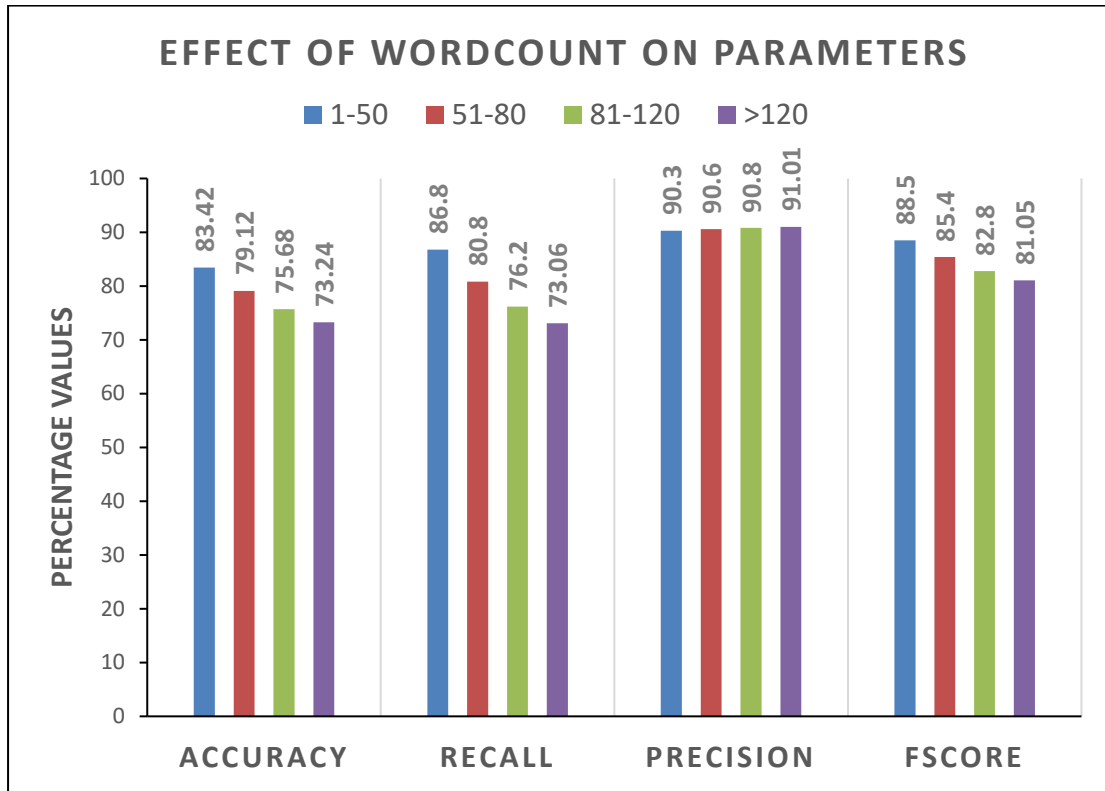


Figure.6.5 Effect of wordcount on Recall, Precision and F-score of enhanced Lexicon model for Multidomain dataset

In case of hybrid model, along with the enhanced lexicon model the effect of wordcount on accuracy of machine learning classifiers also needs to be seen. Considering the Multinomial Naïve Bayes as a part of the hybrid model, the results for single domain dataset were obtained as shown in Figure 6.6 and Figure 6.7. The accuracy results in Figure 6.7 makes it clear that, for hybrid model too as the wordcount increases the accuracy decreases. Accuracy in first two wordcount groups was almost same. As the wordcount increased further than 80 words, significant drop in accuracy was seen. Precision increased slightly but there was a decrease in Recall and F-score values.

The effect of wordcount on accuracy of hybrid model was evaluated on multidomain dataset of 4000 sentences and the corresponding results were obtained as shown in Figure 6.8 and 6.9. Values of accuracy in the wordcount group 51-80 is substantially high. When the wordcount goes on increasing further higher, eventually the accuracy decreases as seen in Figure 6.8. Even parameters Recall and F-score decrease as Accuracy increases. Only a slight increase in value of Precision is seen in the results shown in Figure 6.9.

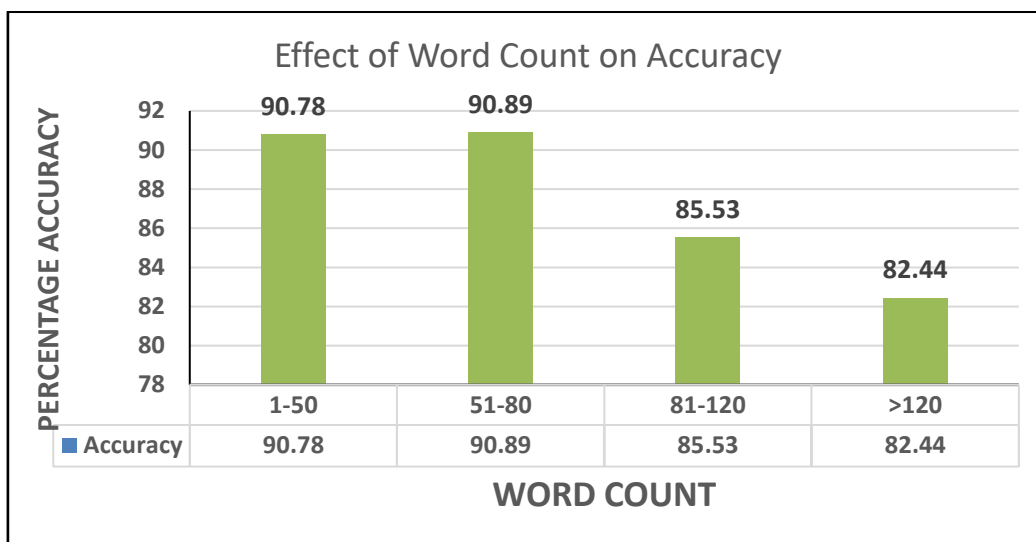


Figure.6.6 Effect of wordcount on Accuracy of Hybrid Model for Single domain dataset.

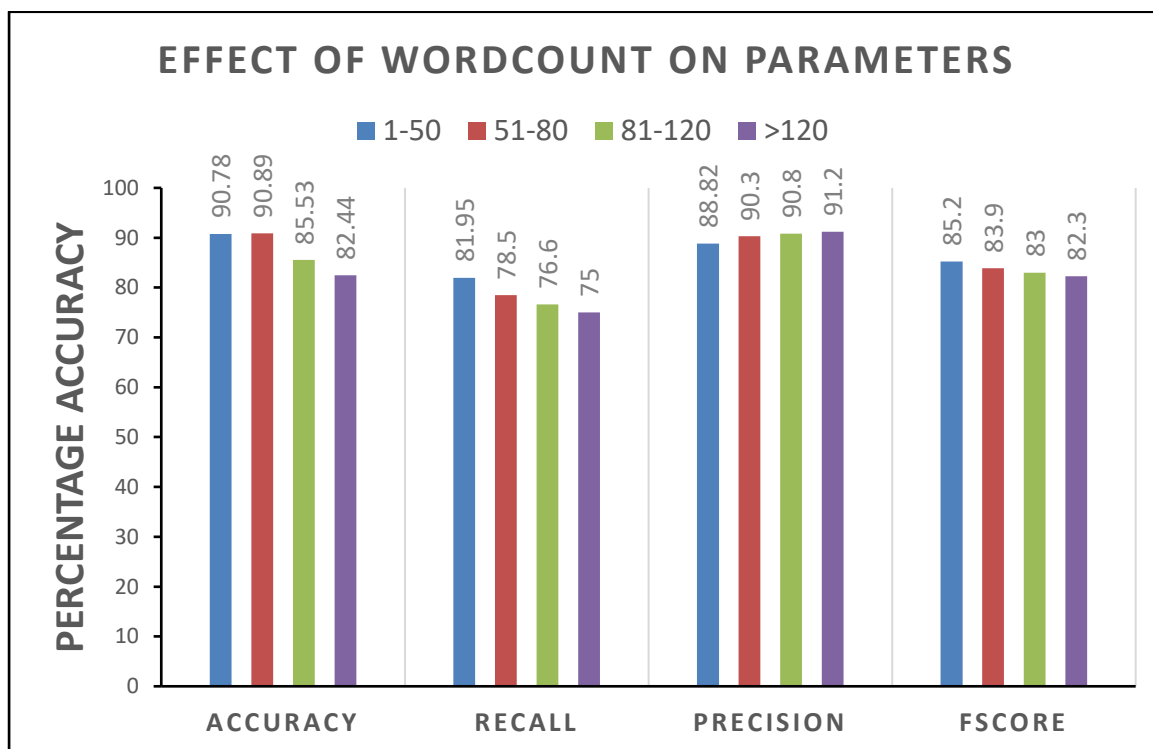


Figure.6.7 Effect of wordcount on Recall, Precision and F-score of Hybrid Model for Single domain dataset.

All the results obtained for both Enhanced Lexicon model and hybrid model for Single domain dataset and multidomain dataset clearly show the effect of wordcount on Accuracy and other parameters Precision, Recall and F-score. The conclusion here can be done from the results that for sentences with shorter words the accuracy is significantly high. After a

particular threshold the accuracy starts decreasing as the number of words in the sentence goes on increasing.

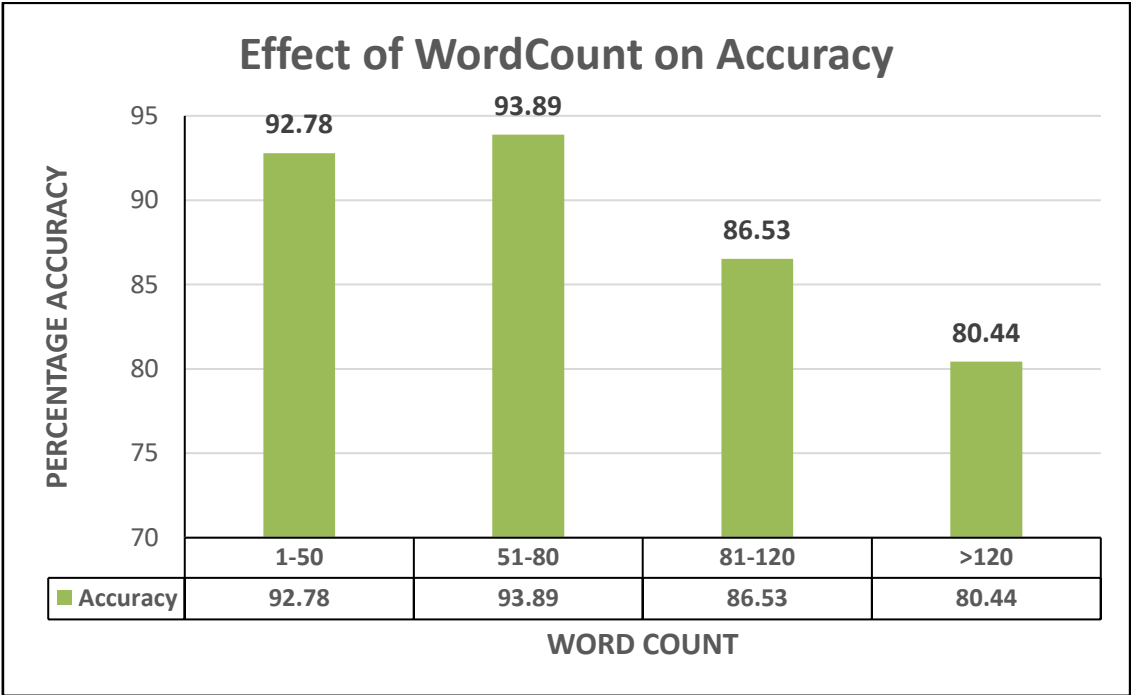


Figure.6.8 Effect of wordcount on Accuracy of Hybrid Model for Multidomain dataset.

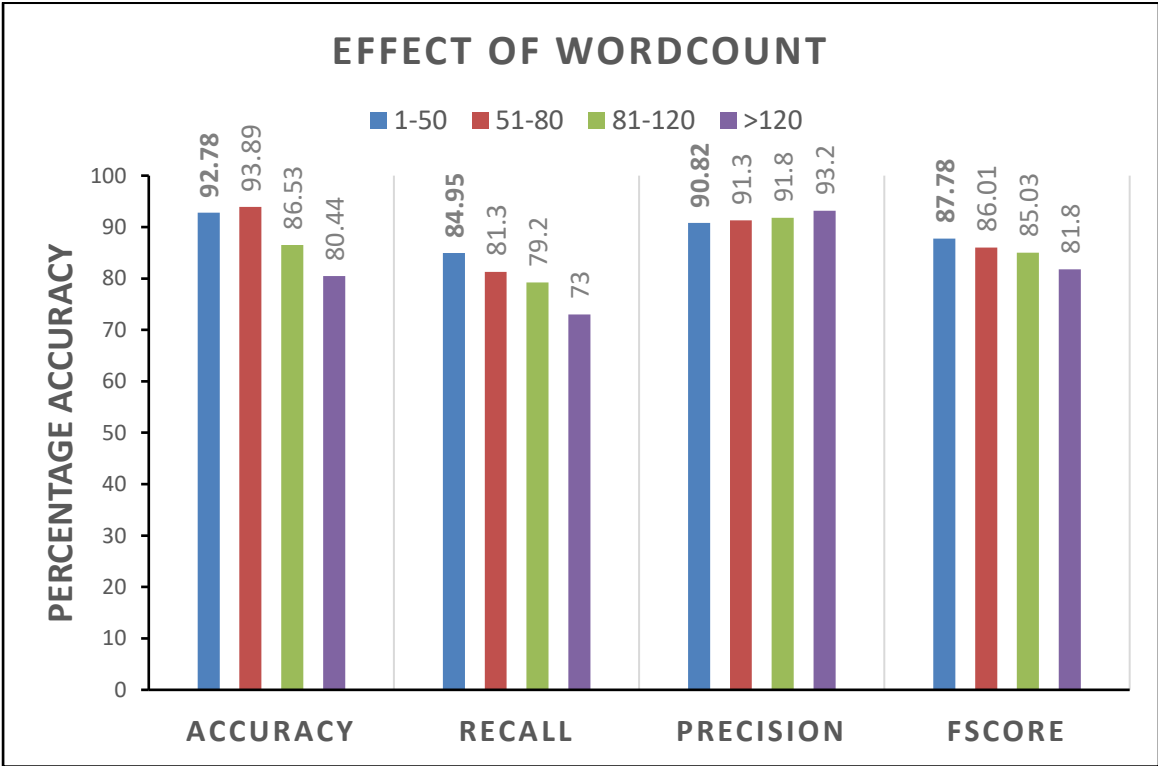


Figure.6.9 Effect of wordcount on Recall, Precision and F-score of Hybrid Model for Multidomain dataset.

6.2 Statistical Significance

Statistical significance, as the name signifies denotes the claim that an output generated from data by experimentation is not expected to happen by chance but is considered likely to be for a specific cause. In order to verify the statistical significance of the impact of word-count on the classification accuracy, a multiple regression model is built. In case of regression model, the default hypothesis is always that every independent variable is having absolutely no effect. The size of the coefficient for every independent variable gives the size of the effect that it is having on the dependent variable, and the sign on the coefficient whether it is positive or negative gives the direction of the effect. In regression with multiple independent variables, the coefficient tells you how much the dependent variable is expected to increase when that independent variable increases by one, holding all the other independent variables constant.

. The regression model is described as

$$Accuracy_i = \beta_0 + \beta_1 Word_count_i + \beta_2 LexModel_i + \beta_3 Data_size_i$$

where Accuracy is the dependent variable, denoting the accuracy of the sentiment classification. For the independent variables, a variable Word_count is introduced that denotes the level of the words count. Considering the experimentation, the short-review level is defined as having a threshold count of 80 words in each review and the long-review level having beyond 80 words. With reference to the short review level, the relative impact of the word count on the accuracy is examined. LexModel is used to control the effect of Lexicon based model. Similarly, Data_size is used to control the effect of the dataset size and are taken as 1 when the dataset size is 4000. The regression analysis is made by using Stata software.

Table 6.1 shows the results of the statistical significance. From the results, it is clear that the P value for the estimate of Word_count is smaller than 0.0001, which implies that the impact of the word count on the sentiment classification accuracy is statistically significant at a 99% confidence level. Also, the estimate for Word_count is negative, which means that if the word count has long level, then it could lower the accuracy of the sentiment classification. This explains the importance of the fact that people can express their sentiments clearly with a limited number of words and if too long reviews are there then it is due to the noise. For Hindi sentiment analysis considering lexicon-based methods, less than 80 words is an ideal level for words count and it will guarantee that this much

information is sufficient for expressing the reviewers' sentiment and will involve very less noise. Results clearly indicate that the impact of the review words count on the classification accuracy is statistically significant at a 99% confidence level. Less than 80 words is an ideal level for the review words count

Table 6.1 Results of the statistical significance

Parameter	Estimate	Standard Error	T-statistic	P
W-count	-2.46	0.319	-8.44	<0.0001
LexModel	12.67	0.825	15.35	<0.0001
Dsize	3.44	0.45	7.52	<0.0001

6.3 Summary

The performance of sentiment analysis is affected by different properties like dataset size or training corpus size, feature extraction, length of target documents, preprocessing constraints, length of the review/word count, readability, purity of data, subjectivity of data etc. Efforts need to be taken to not only increase the performance of the algorithms but to investigate the impact of some of the linguistic properties of the dataset on the performance. One such data property is Word count. The first approach experimented for effect of wordcount was the Enhanced Lexicon model. The enhanced Lexicon model was tested on both the single and multidomain datasets to find the effect of wordcount on Accuracy and other parameters by considering the four wordcount categories. In case of hybrid model, along with the enhanced lexicon model the effect of wordcount on accuracy of machine learning classifiers also needs to be seen. Considering the Multinomial Naïve Bayes as a part of the hybrid model, the results for single domain dataset were obtained. All the results obtained for both Enhanced Lexicon model and hybrid model for Single domain dataset and multidomain dataset clearly show the effect of wordcount on Accuracy and other parameters Precision, Recall and F-score. The conclusion here can be drawn from the results that for sentences with shorter words the accuracy is significantly high. After a particular threshold the accuracy starts decreasing as the number of words in the sentence goes on increasing. In order to verify the statistical significance of the impact of word-count on the classification accuracy, a multiple regression model is built. Results clearly indicate that the impact of the review words count on the classification accuracy is statistically significant at a 99% confidence level. Less than 80 words is an ideal level for the review

words count. In the next chapter some of the important findings of the research are highlighted with some concluding remarks and a few important directions are provided for taking the research work in this field forward.

Chapter 7

Conclusion and Scope for Future Work

7.1 Conclusion

The work proposes an enhanced Lexicon model and a hybrid model to perform sentiment analysis in Hindi Language. The simple lexicon-based approach for Hindi language makes use of HSWN but suffers from various limitations like absence of lexicons, word sense ambiguity, morphological variations, low accuracy due to presence of diminishers and negators etc. An enhanced lexicon-based algorithm is proposed which performs morphological handling, provides an effective solution for word sense disambiguation and enhances the performance of the sentiment analysis system by handling intensifiers, diminishers, conjunctions and negators. A hybrid model is also built by combining the enhanced lexicon model along with machine learning approach and its performance is evaluated for efficiency and effectiveness.

Experimental results demonstrate that the enhanced lexicon model produces highly accurate results compared to other machine learning algorithms. The enhanced lexicon model makes efficient use of HSWN and tries to solve the problems related to NLP. The hybrid method is tested on various datasets and also for multiclass classification to find out which machine learning algorithm can fit better in the hybrid model. The hybrid method proves its effectiveness in terms of performance evaluation measures such as Accuracy, Precision, Recall and F1-score. The hybrid model provides stability and consistency due to use of enhanced lexicon model and provides improved accuracy when combined with machine learning classifier. The hybrid model was also tested for time complexity so as to get an efficient sentiment classification system.

This research work has following noteworthy contributions:

1. The work presented in the thesis is a step towards an effective lexicon-based sentiment classification system for Hindi language.
2. The work demonstrated an enhanced Lexicon HSWN model that was used to solve NLP related issues like word sense ambiguity, morphological variations, negation handling that may affect the performance of a lexicon-based system.

3. The work implemented a graph based Modified Lesk approach for providing a solution to word sense ambiguity of HSWN lexicon and proved to be better than other two methods of averaging and first sense. The graph based Modified Lesk Approach obtained an accuracy of 73.48% which was higher as compared to Averaging (68.7%) and First sense (66.2%).
4. A rule-based scoring approach was implemented as a part of the Enhanced Lexicon model algorithm to handle various intensifiers, diminishers, negators and conjunctions in Hindi Language. The accuracy increased from 73.48% to 83.72% by using the rule-based approach.
5. The work explored a hybrid method to perform sentiment analysis in Hindi language by combining Enhanced Lexicon based HSWN model with machine learning classifiers to enhance the accuracy of the system. The experimental results clearly express the effectiveness of the hybrid method when Multinomial Naïve bayes or Random Forest is used along with the enhanced Lexicon model.
6. The hybrid approach was tested on various datasets such as single domain, multidomain as well as multiclass classification emotion dataset and was found to be effective on all the datasets compared to simple lexicon approach or individual machine learning classifiers.
7. The work attempted at finding out the effect of word count or length of reviews on sentiment classification accuracy and experimentation confirmed that the ideal wordcount for Lexicon model is 80 words

7.2 Scope for future work

The proposed algorithm is intended for performing sentiment analysis in Hindi language by enhancing the Lexicon based approach and proposing a hybrid approach. It can further be extended in following ways:

1. Rule-based method to handle intensifiers, diminishers, and conjunctions as a part of enhanced Lexicon algorithm can be extended to define rules for handling Sarcasm and idioms as a part of enhanced lexicon model
2. The work only handles explicit negations. Handling implicit negations can be explored.
3. Automated approaches for enhancing the HSWN lexicon coverage can be devised.
4. Research in Indian languages is always limited generally due to unavailability of relevant resources. New large datasets should be created and previously existing

ones should be extended. Researchers should also aim at introducing new optimized lexical and linguistic resources which can help in developing an efficient and robust sentiment analysis system.

5. Sentiment analysis of mixed language employing Hindi-English combination can be addressed. Work in English-Hindi mixed code is turning out to be the new area of research since people use some English words while writing reviews in Hindi. This has been observed a lot on the social media like Twitter and Facebook.
6. The work stresses on sentence level sentiment analysis. There is also scope for fine grained Sentiment Analysis which can be done mainly at the aspect level. Aspect level analysis amounts to only 10% of the research work which means researchers have to focus more on the aspect level [86]. Considering feature level or the aspect level is certainly going to be more valuable and useful with respect to sentiment analysis
7. Appropriate solution for issues such as emotion detection, humor detection, data sparsity, discourse connectors issue which are common sentiment analysis issues can be devised.
8. Multiclass classification is undoubtedly an exploring and rising area of research and investigators may research and evaluate different techniques that can be used for the same.

Appendix A

Basics of Python

Python is a very well-known object-oriented programming language which has the competences of any high-level programming language. It has become popular because it is quite easy to learn syntax and its ability to be portable has made it even more useful.

Guido van Rossum was the one who developed Python and the very first version was said to be released in the year 1991. Python was named after a TV show called Monty Python's Flying Circus.

Python is an open source programming language and it has features which are taken from The elegance of C language is combined with the object oriented programming features of Java. Python is an interpreted language.

Features of Python

as well as the most popular language for machine learning and data science. It is because of the following strengths that Python has –

- Python is easy to learn and understand and is simpler in terms of syntax.
- Python is very robust and adaptive.
- Python is an open source programming language and gets full support from the python community. So the errors if present are easily fixed by the Python community
- Since python supports structured programming, object-oriented programming as well as functional programming it can be called as a multi-purpose programming language
- Python is a scalable programming language because it gives a better-quality structure in order to support larger programs in comparison with shell-scripts.
- Python is an extensible language that is having very large number of modules that is said to cover almost each programming aspect. It is known to have a wide and

powerful set of packages like scipy, pandas, scikitlearn, numpy etc. that can be utilized in different arenas.

The only weakness of Python is its slow execution speed because it is an interpreted language.

Appendix B

Python Libraries

Numpy

NumPy is a very useful component and fundamentally stands for Numerical Python. NumPy includes multidimensional array objects and is known to execute essential operations such as Mathematical operations on arrays, logical operations on arrays, Operations involving linear algebra, operations including Fourier transformation etc.

To use NumPy, we just need to import the package into the Python script using the following statement:

- `import numpy as np`

In case of standard Python distribution, NumPy can be installed using the python package installer, pip by giving the command:

- `pip install NumPy`

Pandas

Pandas is another beneficial Python library developed by Wes McKinney that is generally utilized for data manipulation and analysis. With the help of Pandas, data processing can be achieved by following steps Load, Prepare, Manipulate, Model and Analyze

Data representation in Pandas is done by using three data structures namely series, data-frame, panel. Series is similar to an array containing homogeneous data. Data Frame is utilized for nearly every type of data representation and manipulation required in case of pandas and is a two-dimensional data structure holding heterogeneous data. Usually, Data Frames is used for representation of tabular data. Panel is a 3-dimensional data structure which can contain heterogeneous data and is usually showed as a container of Data Frame.

Pandas can be imported as a package into the Python script using the statement:

- `import pandas as pd`

In case of standard Python distribution, Pandas can be installed using the python package installer, pip.

- `pip install Pandas`

Scikit-learn

Scikit-Learn is another very useful and important python library which is predominantly used for Data Science and machine learning in Python. Some of the well-known features of Scikit-learn which has made it a useful resource are given below:

- ♦ Scikit-Learn is built on NumPy, SciPy, and Matplotlib.
- ♦ It is accessible to everyone and is known for its benefit of reusing it in different contexts.
- ♦ It is an open source which can be reused under BSD license
- ♦ It is known to provide a huge range of machine learning algorithms which covers all aspects of Machine learning like classification, regression, model selection, clustering, etc. Scikit-Learn package can be used into the Python script as shown below. For example, with following line of script we are importing accuracy score metric from Scikit-learn:

```
from sklearn.metrics import accuracy_score.
```

The confusion matrix can be found out by using the `confusion_matrix ()` function of sklearn as shown below:

- `from sklearn.metrics import confusion_matrix`

Various classifiers can be imported using Scikit Learn using the import statement as shown below:

- `from sklearn.tree import DecisionTreeClassifier`
- `from sklearn.naive_bayes import GaussianNB`
- `from sklearn.ensemble import RandomForestClassifier`

If standard Python distribution is being used and is having NumPy and SciPy then by using python package installer, pip Scikit-learn can be installed.

- `pip install -U scikit-learn`

References

- [1] Pang B, Lee L. 2008. Opinion mining and sentiment analysis. *Foundations and Trends R in Information Retrieval*, vol 2, nos 1–2, 2008, 1–135,
- [2] Dr. Vivek Kumar Singh.2015. A Survey of Opinion Mining Research in Hindi Language, *International Journal of Advanced Scientific Research & Development*, Vol. 02, Issue. 03, Ver. II, 2015, pp. 135 – 138.
- [3] D. Jain, A. Kumar and G. Garg. 2020. Sarcasm detection in mash-up language using soft-attention based bi-directional LSTM and feature-rich CNN, *Applied Soft Computing Journal*, Volume 91, Elsevier 2020.
- [4] Narang, S.R., Jindal, M.K., Ahuja, S. et al. 2020. On the recognition of Devanagari ancient handwritten characters using SIFT and Gabor features. *Soft Comput* 24, 17279–17289 2020. <https://doi.org/10.1007/s00500-020-05018-z>
- [5] Dargan, S., Kumar, M., Garg, A. et al. 2020. Writer identification system for pre-segmented offline handwritten Devanagari characters using k-NN and SVM. *Soft Comput* 24, 10111–10122, 2020. <https://doi.org/10.1007/s00500-019-04525-y>
- [6] Richa Sharma, Shweta Nigam and Rekha Jain. 2014. Opinion Mining in Hindi Language: A Survey, *International Journal in Foundations of Computer Science & Technology (IJFCST)*, Vol.4, No.2, 2014.
- [7] Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., and Manandhar, S. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, Dublin, 2014.
- [8] Rani, S, Kumar. P. 2019. A journey of Indian languages over sentiment analysis: a systematic review. *Artificial Intelligence Review* 52, 2019, 1415– 1462
- [9] Sneha Mulatkar. 2014. Sentiment Classification in Hindi, *International Journal of Scientific & Technology Research* Volume 3, Issue 5, 2014.
- [10] Akshat Bakliwal, Piyush Arora, Vasudeva Varma. 2012. Hindi subjective lexicon: A lexical resource for Hindi polarity classification, In: *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC)*, 2012.
- [11] A. Das, S. Bandyopadhyay. 2010. SentiWordNet for Indian Languages. In: *Proceedings of the 8th Workshop on Asian Language Resources*,2010, pp. 56–63.

- [12] Soonh Taj, Baby Bakhtawer Shaikh, Areej Fatemah Meghji. 2019. Sentiment Analysis of News Articles: A Lexicon based Approach, International Conference on Computing, Mathematics and Engineering Technologies – iCoMET IEEE ,2019.
- [13] Rahul Rajput, Arun Kumar Solanki. 2016. Review of Sentimental Analysis Methods using Lexicon Based Approach International Journal of Computer Science and Mobile Computing, Vol.5 Issue.2, 2016, pg. 159-16
- [14] A. Joshi, A. R. Balamurali and P. Bhattacharyya.2010. A Fallback Strategy for Sentiment Analysis in Hindi: A Case Study. In: Proceedings of the 8th International Conference on Natural Language Processing (ICON), 2010.
- [15] Arora P. 2013. Sentiment analysis for Hindi language. MS by Research in Computer Science,2013.
- [16] Jasleen Kaur and Jatinderkumar R. Saini. 2014. A Study and Analysis of Opinion Mining Research in Indo-Aryan, Dravidian and Tibeto-Burman Language Families, International Journal of Data Mining and Emerging Technologies, vol 4, issue 2,2014, pp 53-60.
- [17] Mukesh Yadav, Varunakshi Bhojane. 2015. Sentiment Analysis on Hindi Content: A Survey, International Journal of Innovations & Advancement in Computer Science IJIACS, Volume 4, Issue 12, 2015.
- [18] Garg, Komal, and Preetpal Kaur Buttar.2017. "Survey on Sentiment Analysis in Hindi Language." International Journal of Advanced Research in Computer Science 8.5 ,2017.
- [19] Mittal N, Agarwal B, Chouhan G, Bania N, Pareek P. 2013. Sentiment analysis of Hindi review based on negation and discourse relation. In: Proceedings of International joint conference on natural language processing,2013, pp 45–50.
- [20] Piyush Arora, Akshat Bakliwal, And Vasudeva Varma. 2012 Hindi Subjective Lexicon Generation using WordNet Graph Traversal, International Journal of Computational Linguistics and Applications (IJCLA) VOL. 3, NO. 1, 2012, PP. 25–39.
- [21] Sharma Raksha, and Pushpak Bhattacharyya. 2014. A sentiment analyzer for Hindi using Hindi Senti lexicon, Proceedings of the 11th International Conference on Natural Language Processing (ICON), 2014, pp 150–155.

- [22] Vandana Jha, Manjunath N, P Deepa Shenoy and Venugopal K R. 2016. Sentiment Analysis in a Resource Scarce Language: Hindi, International Journal of Scientific & Engineering Research, Volume 7, Issue 9, 2016, pp no 968-990.
- [23] Mishra Deepali, Manju Venugopalan, and Deepa Gupta. 2016. Context specific Lexicon for Hindi reviews. In: Procedia Computer Science 93, 2016, pp 554-563.
- [24] Deepa Modi and Neeta Nain. 2016. Part-of-Speech Tagging of Hindi Corpus Using Rule-Based Method, Proceedings of the International Conference on Recent Cognizance in Wireless Communication & Image Processing Springer, 2016, pp 241-247.
- [25] Vandana Jha, Savitha R, P Deepa Shenoy. 2018. Venugopal K R , Arun Kumar Sangaiah A novel sentiment aware dictionary for multi-domain sentiment classification , Computers & Electrical Engineering , Volume 69, Elsevier 2018, Pages 585-597.
- [26] Firdous Hussaini, S. Padmaja and S. Sameen Fatima. 2018. Score-Based Sentiment Analysis of Book Reviews in Hindi Language, International Journal on Natural Language Computing (IJNLC) Vol.7, No.5, 2018, pp 115-127.
- [27] Yakshi Sharma, Veenu Mangat and Mandeep Kaur. 2015. A Practical Approach to Sentiment Analysis of Hindi Tweets 1st International Conference on Next Generation Computing Technologies (NGCT-2015), 2015, pg. no. 677-680.
- [28] Vandana Jha, Manjunath N, P Deepa Shenoy and Venugopal K R. 2015. HSAS: Hindi Subjectivity Analysis System, Annual IEEE India Conference (INDICON), IEEE, 2015, pp 1-6.
- [29] Shapiro, Adam Hale, Moritz Sudhof, Daniel Wilson. 2020. Measuring News Sentiment, Federal Reserve Bank of San Francisco Working Paper 2017-01, 2020.
- [30] Kanika Garg. 2019. Sentiment analysis of Indian PM's "Mann Ki Baat", International journal of information technology, 08 Jul 2019, Vol. 12, Issue 1, pp 37 – 48.
- [31] Sharma Richa, Shweta Nigam, and Rekha Jain. 2014. Polarity detection movie reviews in Hindi language. arXiv preprint arXiv:1409.3942, 2014.
- [32] Gilliar Meng, Heba Saddeh. 2020. Applications of Machine Learning and Soft Computing Techniques in Real World, International Journal of Computer Applications & Information Technology Vol. 12, Issue No. 1, 2020.
- [33] Ansari Fatima Anees, Arsalaan Shaikh, Arbaz Shaikh and Sufiyan Shaikh. 2020. Survey Paper on Sentiment Analysis: Techniques and Challenges, Easy Chair Preprint № 2389, 2020.

- [34] Sudha Shanker Prasad, Jitendra Kumar, Dinesh Kumar Prabhakar, Sukomal Pal. 2015. Sentiment classification: an approach for Indian language tweets using decision tree, International Conference on Mining Intelligence and Knowledge Exploration. Springer, 2015, pp 656-663.
- [35] Sarkar K, Chakraborty S. 2015. A sentiment analysis system for Indian language tweets. International conference on mining intelligence and knowledge exploration. Springer, 2015, pp 694–702.
- [36] Sachin Kumar S., B. Premjith, Anand Kumar M., and Dr. Soman K. P. 2015. AMRITA_CEN-NLP@SAIL2015: Sentiment analysis in Indian language using regularized least square approach with randomized feature learning, Lecture Notes in Computer Science, vol. 9468, 2015, pp. 671-683.
- [37] Kumar Ayush, Kohail S, Ekbal A, Biemann C. 2015. Iit-tuda: system for sentiment analysis in Indian language using lexical acquisition, International conference on mining intelligence and knowledge exploration, Springer, 2015, pp 684–693.
- [38] Shriya Se, Vinayakumar R, Kumar MA, Soman K. 2015. Amrita-cen@ sail2015: Sentiment analysis in Indian languages, international conference on mining intelligence and knowledge exploration. Springer, 2015, pp 703–710.
- [39] Jha V, Manjunath N, Shenoy PD, Venugopal K, Patnaik LM. 2015. Homs: Hindi opinion mining system, 2nd International conference on recent trends in information systems. IEEE, 2015, pp 366–37.
- [40] Sharma P, Moh TS. 2015. Prediction of Indian election using sentiment analysis on Hindi twitter. In: International conference on big data. IEEE, 2016, pp 1966–1971.
- [41] Akhtar MS, Ekbal A, Bhattacharyya P. 2016. Aspect based sentiment analysis in Hindi: resource creation and evaluation, Proceedings of the 10th international conference on language resources and evaluation, 2016, pp 1–7.
- [42] Akhtar MS, Ekbal A, Bhattacharyya P. 2016. Aspect based sentiment analysis: category detection and sentiment classification for Hindi, 17th International conference on intelligent text processing and computational linguistics, 2016, pp 1–12.
- [43] Prof. Nikita Desai, Anandkumar D. 2016. Sarcasm Detection in Hindi sentences using Support Vector machine, International Journal of Advance Research in Computer Science and Management Studies, Volume 4, Issue 7, 2016.
- [44] Prafulla B. Bafna, Jatinderkumar R. Saini. 2020. On Exhaustive Evaluation of Eager Machine Learning Algorithms for Classification of Hindi Verses, International Journal of Advanced Computer Science and Applications, (IJACSA), Vol. 11, No. 2, 2020.

- [45] Ankush Khandelwal, Sahil Swami, Syed Sarfaraz Akhtar, Manish Shrivastava. 2018. Gender Prediction in English-Hindi Code-Mixed Social Media Content: Corpus and Baseline System, *Computación y Sistemas*, Vol. 22, No. 4, 2018, pp. 1241–1247.
- [46] Deepanshu Vijay, Aditya Bohra, Vinay Singh, Syed S. Akhtar, Manish Shrivastava. 2018 A Dataset for Detecting Irony in Hindi-English Code-Mixed social media Text, 15th Extended Semantic Web Conference (ESWC-2018), 2018.
- [47] Kumar Ravi, Vadlamani Ravi. 2016. Sentiment classification of Hinglish text, 3rd International Conference on Recent Advances in Information Technology (RAIT), IEEE, 2016.
- [48] Charu Nanda, Mohit Dua and Garima Nanda. 2018. Sentiment Analysis of Movie Reviews in Hindi Language using Machine Learning, International Conference on Communication and Signal Processing, 2018.
- [49] Mukesh Yadav, Varunakshi Bhojane. 2019. semi supervised mix Hindi sentiment analysis using neural network, 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2019.
- [50] Kaushika Pal and Biraj V. Patel. 2020. Model for Classification of Poems in Hindi Language Based on Ras, Smart Systems and IoT, *Innovations in Computing*, 2020 - Springer pp 655-661.
- [51] Basant Agarwal & Namita Mittal. 2016. Prominent feature extraction for review analysis: an empirical study, *Journal of Experimental & Theoretical Artificial Intelligence*, 28:3, 485-498, 2016.
- [52] Deepanshu Vijay, Aditya Bohra, Vinay Singh, Syed S. Akhtar, Manish Shrivastava. 2018. Corpus Creation and Emotion Prediction for Hindi-English Code-Mixed social media Text, *Proceedings of NAACL-HLT 2018: Student Research Workshop*, 2018, pp 128–135.
- [53] Swami S., Khandelwal, A., Singh, V., Akhtar, S.S., Shrivastava, M. 2018. A Corpus of English-Hindi Code-Mixed Tweets for Sarcasm Detection, 19th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing), 2018.
- [54] Ankush Khandelwal, Sahil Swami, Syed S Akhtar, and Manish Shrivastava. 2018. Humor detection in English Hindi code-mixed social media content: Corpus and baseline system. *arXiv preprint arXiv:1806.05513*, 2018.
- [55] Otter, D. W., Medina, J. R., & Kalita, J. K. 2020. A Survey of the Usages of Deep Learning for Natural Language Processing, *IEEE Transactions on Neural Networks and Learning Systems*, pp 1–21, 2020.

- [56] Dargan, S., Kumar, M., Ayyagari, M.R. 2020. A Survey of Deep Learning and Its Applications: A New Paradigm to Machine Learning. Arch Computat Methods Eng 27, 1071–1092 (2020). <https://doi.org/10.1007/s11831-019-09344-w>
- [57] Seshadri S, Madasamy AK, Padannayil SK. 2016. Analyzing sentiment in Indian languages micro text using recurrent neural network, Institute of Integrative Omics and Applied Biotechnology (IIOAB Journal) Volume 7, 2016, p.313-318.
- [58] Wenling Li, Bo Jin, Yu Quan. 2020. Review of Research on Text Sentiment Analysis Based on Deep Learning, Open Access Library Journal, Volume 7, 2020.
- [59] Akhtar Ms, Sawant P, Sen S, Ekbal, A, Bhattacharyya P. 2018. Solving data sparsity for aspect-based sentiment analysis using cross-linguality and multi-linguality, Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018, pp 572-582.
- [60] Rani S, Kumar P. 2018. Deep learning-based sentiment analysis using convolution neural network. Arabian Journal for Science and Engineering 44 (4), 2018, pp 3305-3314.
- [61] Puneet Mathur, Rajiv Ratn Shah, Ramit Sawhney, Debanjan Mahata. 2018. Detecting Offensive Tweets in Hindi-English Code-Switched Language, Proceedings of the Sixth International Workshop on Natural Language Processing for social media, 2018, pp 18–26.
- [62] Sushmitha Reddy Sane, Suraj Tripathi, Koushik Reddy Sane, Radhika Mamidi, 2019. Deep Learning Techniques for Humor Detection in Hindi-English Code-Mixed Tweets, Proceedings of the 10th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, 2019, pp 57–61.
- [63] Pranaydeep Singh and Els Lefever. 2020. Sentiment Analysis for Hinglish Code-mixed Tweets by means of Cross-lingual Word Embeddings, Proceedings of the LREC- 4th Workshop on Computational Approaches to Code Switching, 2020, pp 45–51.
- [64] T.Y.S.S. Santosh, K.V.S. Aravind. 2019. Hate Speech Detection in Hindi-English Code-Mixed social media Text, Proceedings of the ACM India Joint International Conference on Data Science and Management of Data, Cods-COMAD 2019, pp 310-313.
- [65] Singhal P, Bhattacharyya P. 2016. Borrow a little from your rich cousin: using embeddings and polarities of English words for multilingual sentiment classification.

In Proceedings of the International Conference on Computational Linguistics (COLING), 2016.

- [66] Bhargava R., Arora S., & Sharma Y. 2018. Neural Network-Based Architecture for Sentiment Analysis in Indian Languages. *Journal of Intelligent Systems*, 2018, 28 (3), pp 361-375.
- [67] Akhtar MS, Kumar A, Ekbal A, Bhattacharyya P. 2016. A hybrid deep learning architecture for sentiment analysis. In: *Proceedings of the 26th international conference on computational linguistics*, 2016, pp 482–493.
- [68] Madan Gopal Jhanwar, Arpita Das. 2018. An Ensemble Model for Sentiment Analysis of Hindi-English Code-Mixed Data, and Language, *arXiv:1806.04450* , 2018.
- [69] Kanika Garg and D. K. Lobiyal. 2018. Multi-class Classification of Sentiments in Hindi Sentences Based on Intensities, Towards Extensible and Adaptable Methods in Computing, Springer Nature Singapore Pte Ltd. 2018.
- [70] Shrikant Tarwani, Manan Jethanandani, and Vibhor Kant. 2019. Cyberbullying Detection in Hindi-English Code-Mixed Language Using Sentiment Classification, *Advances in Computing and Data Sciences Springer Singapore*, ICACDS 2019
- [71] Namita Mittal, Basant Agarwal, Garvit Chouhan, Prateek Pareek, and Nitin Bania. 2013. Discourse Based Sentiment Analysis for Hindi Reviews, *International Conference on Pattern Recognition and Machine Intelligence PReMI 2013: Pattern Recognition and Machine Intelligence*, pp 720-725.
- [72] Mr. Mohammed Arshad Ansari and Prof. Sharvari Govilkar. 2016., Sentiment Analysis of Transliterated Hindi and Marathi Script, *Sixth International Conference on Computational Intelligence and Information Technology (CIIT) 2016*.
- [73] Garg, Komal, and Preetpal Kaur Buttar. 2017. Aspect Based Sentiment Analysis of Hindi Text Review, *International Journal of Advanced Research in Computer Science*, Vol. 8, No. 7, 2017, pp. 831-836.
- [74] C. Dalal, S. Tandon, A. Mukerjee. 2014. Insult Detection in Hindi. *Technical report on Artificial Intelligence*, 18, 2014.
- [75] Pandey P, Govilkar S. 2015. A framework for sentiment analysis in Hindi using HSWN. *International journal of Computer Applications*, Vol 119 No 19, 2015, pp 23–26.
- [76] Archana Kumari and D. K. Lobiyal. 2020. Word2vec’s Distributed Word Representation for Hindi Word Sense Disambiguation, *Social Networking and Computational Intelligence. Lecture Notes in Networks and Systems*, 202

- [77] Jain A, Yadav D, Tayal DK. 2014. NER for Hindi language using association rules, International conference on data mining and intelligent computing, 2014.
- [78] Yaman Kumar, Debanjan Mahata. 2019. Sagar Aggarwal, Anmol Chugh, Rajat Maheshwari, Rajiv Ratn Shah BHAAV (भाव) - A Text Corpus for Emotion Analysis from Hindi Stories, Published in ArXiv 2019
- [79] Vikas Kumar Jhaa, Hrudya Pa, Vinu P Na, Vishnu Vijayana, Prabakaran Pa. 2020. DHOT-Repository and Classification of Offensive Tweets in the Hindi Language Third International Conference on Computing and Network Communications, Elsevier,2020.
- [80] C. Monica, N. Nagarathna. 2020., Detection of Fake Tweets Using Sentiment Analysis, SN Computer Science, Springer Nature,2020.
- [81] Singh, Satyendr, and Tanveer J. Siddiqui. "Role of semantic relations in Hindi word sense disambiguation." *Procedia Computer Science* 46 (2015): 240-248.
- [82] Soujanya Poria, Alexander Gelbukh, Erik Cambria, Amir Hussain, Guang-Bin Huang. EmoSentSpace: A Novel Framework for Affective Common-Sense Reasoning. *Knowledge-Based Systems*, vol. 69, 2014, ISSN 0950-7051, pp. 108–123, Doi: 10.1016/j.knosys.2014.06.011.
- [83] Soujanya Poria, Alexander Gelbukh, Amir Hussain, Dipankar Das, Sivaji Bandyopadhyay. Enhanced SenticNet with Affective Labels for Concept-based Opinion Mining. *IEEE Intelligent Systems*, vol. 28, issue 2, 2013, pp. 31–38, doi:10.1109/MIS.2013.4.
- [84] Crowdsourcing a Word-Emotion Association Lexicon, Saif Mohammad and Peter Turney, *Computational Intelligence*, 29 (3), 436-465, 2013.

PAPER PUBLICATIONS OUT OF RESEARCH WORK

SCOPUS INDEXED JOURNALS

- [1] Kulkarni, D. S., & Rodd, S. F. (2021). Sentiment Analysis in Hindi—A Survey on the State-of-the-art Techniques. *Transactions on Asian and Low-Resource Language Information Processing*, ACM, 21(1), 1-46.
- [2] Dhanashree. S. Kulkarni and Dr. Sunil. F. Rodd, (2022) Word Sense Disambiguation for Lexicon-based Sentiment Analysis in Hindi, *Webology* 19 (1), 592-600.
- [3] Kulkarni, D. S., & Rodd, S. F. (2022). Towards Enhancement of the Lexicon Approach for Hindi Sentiment Analysis. In *IOT with Smart Systems* (pp. 445-451). Springer, Singapore

Submitted

- [4] Paper Titled “Hybrid Approach to Multiclass classification of Hindi text” submitted to *International Journal of Information Technology*, Springer
- [5] Paper Titled “Impact on data properties on Hindi sentiment classification performance” submitted to Scopus Indexed Journal “*Indian Journal of Computer Science and Engineering*”

Conferences

- [6] Kulkarni, D. S., and S. F. Rodd. "Extensive study of text-based methods for opinion mining." *2018 2nd International Conference on Inventive Systems and Control (ICISC)*. IEEE, 2018.