Arab American University- Jenin

Faculty of Graduate Studies

Optimized Machine Learning Based Feature Selection Methods for Sentiment Classification

By

Mohammad Basheer

Supervisor: Dr. Labib Arafeh

This thesis was submitted in partial fulfillment of the requirements for the Master's degree in

Computer Science

December, 2017

Optimized Machine Learning Based Feature Selection Methods for Sentiment Classification

By

Mohammad Basheer

This thesis was defended successfully on 23/ 12 /2017 and approved by:

Committee Members	Signature
1. Supervisor: **Dr. Labib Arafeh**	**….....**
2. Internal Examiner: **Dr. Ahmad Ewais**	**.....**
3. External Examiner: **Dr. Radwan Tahboub**	**.....**

Dedication

This work is dedicated to my family who stood by my side all the way and without whom I would have never got the courage to embark on this rough journey.

Father, Mother, and dear Wife thank you for your support.

Acknowledgments

Praise be to Allah with whom all good deeds are fulfilled.

My thanks and gratitude to my thesis supervisor Dr. Labib Arafeh for his generous and kind advice. I would also like to extend my deepest thanks to the faculty of the computer science master department in the Arab American University for providing us with the basis of research that helped me conclude my research.

I also thank the Coordinator of the Computer Science Program Dr. Amjad Ratrout for his guidance during the study period, especially that he has experience in coordinating and managing the master program.

I would also like to thank my colleague Dr. Thabit Sabbah, whose thesis in the Master of Computer Science was approved by Dr. Labib Arafeh, thanking him for encouraging me to further continue my studies to get my master degree.

I would also like to express my thanks and gratitude to my colleagues who supported me during my studies, especially Eng. Ibrahim Amreya, the linguistic auditor of the thesis.

My deep thanks and gratitude to my father, my mother and my wife for encouraging me to complete my Master's degree.

Abstract

Analyzing the sentiment and opinions has become crucial, especially that institutions, governments, and private sector companies became very interested in knowing what people think about certain events or products. The data size on the web is enormous and growing rapidly. Processing and analyzing this size of data is hard and costly; therefore, existing solutions of sentiment analysis are suffering from deficiencies, such as high dimensionality and low accuracy. The process of selecting relevant features is a matter of research.

The selection of relevant features that can produce high accuracy in classification is not an easy task. Therefore, the goal of this thesis is to classify text based opinions into positive and negative sentiments effectively by selecting the relevant feature subset. To solve this problem, we present an approach that utilizes machine learning and optimization evolutionary algorithms in selecting an effective feature subset in four methods.

Firstly, choosing the feature subset is done based on machine learning algorithm. The support vector machine algorithm (SVM) is used to produce a weight vector after the learning process which contains values that represent the term importance for the classification process.

Secondly, evolutionary algorithm (G.A) is used to optimize the feature subset generated from the first method in order to enhance the sentiment classification.

The third method hybridizes the machine learning based feature subset that is generated by the first method with a statistical based feature subset produced using correlation feature selection method.

The fourth method is called optimized hybrid method, in which the optimization evolutionary (G.A) is applied on top of the feature subset that resulted from the third method.

Two well-known sentiment analysis datasets that are publicly available were used to test and validate the proposed approaches. The first is polarity dataset v2.0 (D1), the second is polarity dataset v1.0 (D2), and a third dataset (D3) which is a combination of D1 and D2.

Sentiment classification performance in this research is evaluated using accuracy, recall, precision, and f-measure.

The results achieved in this research outperform the results reported in existing studies. In our approach in the first method using machine learning for feature weighting we achieved high accuracy reaching 98.79, and we were able to improve those results in the second method to reach higher results of 99.21 using optimization evolutionary. And when we merged the features obtained using machine learning weighting using weight by SVM with those obtained using statistical method of weight by correlation, the classification accuracy reached 99.46 which is even better than both previous methods. After improving the feature subset using the hybrid method we were able to improve the accuracy to reach 99.71 and decrease the feature subset size.

Processing produces a large feature set, the results that we achieved were based on feature subsets extracted from this large set. These subsets contain the most relevant

features for classification, and when compared to existing works the subset sizes we used were smaller which reduces the computation time required for classification.

Table of Contents

Dedication ..... **II**
Acknowledgments ..... **III**
Abstract ..... **IV**
Table of Contents ..... **VII**
List of Tables ..... **X**
List of Figures ..... **XI**
List of Abbreviations ..... **XII**
1. Introduction **2**
1.1 Introduction 2
1.2 Motivation and Problem Studied 2
1.3 Sentiment Analysis 3
1.4 Objectives and Goal 4
1.5 Research Scope 4
1.6 Contribution 4
1.7 Research Methodology 6
1.8 Thesis Overview 9
1.9 Summary 10
2. Background **12**
2.1 Introduction 12
2.2 Data Preprocessing 12
2.2.1 Tokenization 12
2.2.2 Stop Words Removal 12
2.2.3 Case Normalization 13
2.3 Vector Space Model (VSM) 13
2.4 Feature Selection 14
2.4.1 Feature Types 14
2.4.1.1 Unigrams 14
2.4.1.2 Part of Speech 15
2.4.2 Feature Weighting 15
2.4.2.1 Weight by SVM 15
2.4.2.2 Weight by Correlation 15
2.4.2.3 Weight by Information Gain 15
2.4.2.4 Weight by Gini Index 16

2.4.2.5 Term frequency-inverse document frequency (TF-IDF) 16
2.5 Classification 17
2.5.1 Machine Learning Algorithms 17
2.5.2 Validation and Evaluation 19
2.5.2.1 Cross-validation 19
2.5.2.2 Evaluation Measures 19
2.5.2.2.1 Accuracy 20
2.5.2.2.2 Precision 20
2.5.2.2.3 Recall 21
2.5.2.2.4 F-Measure 21
2.6 Summary 21
3. Literature Review 23
3.1 Introduction 23
3.2 Feature Selection in Sentiment Analysis 23
3.2.1 Optimization Approaches 24
3.2.2 Statistical and Machine Learning Approaches 28
3.3 Hybrid Methods 33
3.3.1 Hybrid GA Approaches 33
3.3.2 Neuro Fuzzy Approaches 34
3.4 Summary 38
4. Methodology 40
4.1 Introduction 40
4.1.1 Feature Types 40
4.1.2 Preprocessing 41
4.1.3 Feature Selection 42
4.1.4 Classification 43
4.2 Genetic Algorithms (GA) 46
4.3 Method 1: Machine Learning Based Feature Selection Method 48
4.4 Method 2: Optimized Machine Learning Method 49
4.5 Method 3: Hybrid Statistical and Machine Learning Method 51
4.6 Method 4: Optimized Hybrid Method 53
4.7 Summary 54
5. Results and Evaluation 56
5.1 Introduction 56

5.2 Experimental Setup 56
5.2.1 Tools and Hardware Specifications 56
5.2.2 Datasets 57
5.3 Experiments and Results 65
5.3.1 Classification Using Machine Learning Based Feature Selection Method 65
5.3.2 Classification Using Optimized Machine Learning Method 66
5.3.3 Classification Using Hybrid Statistical and Machine Learning Method 68
5.3.4 Classification Using Optimized Hybrid Method 70
5.4 Discussion 72
5.5 Comparison 73
5.5.1 Comparison on D1 Polarity Dataset v2.0 74
5.5.2 Comparison on D2 Polarity Dataset v1.0 76
5.5.3 Comparison on D3 77
5.6 Limitations 78
5.7 Summary 79
6. Conclusion and Future Work 84
6.1 Conclusion 84
6.2 Future work 85
References 86
ملخص 94

List of Tables

Table 2.1 Dataset example..... 14
Table 2.2 Confusion Matrix..... 20
Table 3.1 Optimization Approaches..... 27
Table 3.2 Statistical and Machine Learning Approaches..... 31
Table 3.3 Hybrid GA Approaches..... 34
Table 3.4 Neuro Fuzzy Approaches..... 36
Table 5.1 Datasets Details..... 58
Table 5.2 Number of Features Obtained for Each Type from Each Dataset..... 59
Table 5.3 Accuracy for different weighing methods for unigrams and POS for D1..... 59
Table 5.4 Accuracy for different weighing methods for unigrams and POS for D2..... 60
Table 5.5 Accuracy for different weighing methods for unigrams and POS for D3..... 60
Table 5.6 Accuracy for different weighing methods at different values of K for Unigrams..... 63
Table 5.7 Accuracy, Precision And Recall For All Datasets at K=4000 for Method 1 . 66
Table 5.8 Accuracy of Method 1: Machine Learning Based Feature Selection Method and Method 2: Optimized Machine Learning Method..... 67
Table 5.9 Accuracy, Recall, and Precision for Optimization Methods at K = 4000..... 67
Table 5.10 Union of Features: 1: Top 4000 Features from Weight by SVM, 2: Top 4000 Features from Weight by Correlation..... 68
Table 5.11 Accuracy for all datasets using the Hybrid Method at different K values. .. 69
Table 5.12 Accuracy, recall and precision of the hybrid method at top k =1000..... 69
Table 5.13 Number of Unigram Features before and after union for each dataset. 70
Table 5.14 Accuracy and number of features for hybrid method and Optimized Hybrid method at K = 1000..... 70
Table 5.15 Accuracy, Precision, Recall, and F-Measure of Optimized Hybrid Method.71
Table 5.16 Comparison on Dataset 1= polarity dataset v2.0=2000..... 74
Table 5.17 Comparison on Dataset 2= polarity dataset v1.0=1400..... 76
Table 5.18 Comparison on Dataset 3= Dataset 1+ Dataset 2=D1+D2=3400..... 77
Table 5.19 Approach Summary on Dataset D1..... 80
Table 5.20 Approach Summary on Dataset D2..... 81
Table 5.21 Approach Summary on Dataset D3..... 82

List of Figures

Figure 1.1 Sentiment analysis process on product reviews 3
Figure 1.2 Research Methodology of the Thesis. 6
Figure 2.1 VSM representation of a collection of documents 13
Figure 2.2 VSM weights matrix based on TF-IDF 17
Figure 2.3 Machine learning in sentiment analysis 18
Figure 4.1 Method 1: Machine Learning Based Feature Selection Method 49
Figure 4.2 Method 2: GA on Subset feature 1 50
Figure 4.3 Weight by Correlation 51
Figure 4.4 Hybrid Method 52
Figure 4.5 Method 4: Optimized Hybrid Method 53
Figure 5.1 Accuracy of Unigrams and POS for D1 61
Figure 5.2 Accuracy of Unigrams and POS for D2 61
Figure 5.3 Accuracy of Unigrams and POS for D3 62
Figure 5.4 Accuracy of different weighing methods for D1 at different K values 64
Figure 5.5 Accuracy of different weighing methods for D2 at different K values 64
Figure 5.6 Accuracy of different weighing methods for D3 at different K values 64
Figure 5.7 Accuracy of Method 1 and Method 2. 68
Figure 5.8 Accuracy of Hybrid Method alone and Optimized Hybrid Method. 71

List of Abbreviations

ABC	Artificial Bee Colony Algorithm
ACO	Ant Colony Optimization Algorithm
BP	Back Propagation
BON	Bag Of Nouns
BOW	Bag Of Words
CAFS	Constructive Approach for Feature selection
CART	Classification and Regression Tree
CFRM	Classic Fuzzy Reasoning Method
CFS	Correlation-based Feature Selection
CharSCNN	Character to Sentence Convolutional Neural Network
CHI	CHI Square
CNN	Convolutional Neural Networks
CRF	Conditional Random Field
D1	Polarity Dataset V2.0
D2	Polarity Dataset V1.0
D3	a combination of D1 and D2.
DF	Document Frequency
F N	False-negative
F P	False-positive
FSGATC	Feature Selection Genetic Algorithm Text Clustering
FURIA	Fuzzy Unordered Rule Induction
GA	Genetic Algorithm
GFRM	General Fuzzy Reasoning Method
ID3	Interactive Dichotomizer version 3 (Decision tree classifier)
IG	Information Gain
IHAC	Improved High Adjective Count
K-NN	Nearest Neighbor
LDF	Linear Discriminant Function
LHNFCSF	Linguistic Hedges Neural-Fuzzy Classifier
LSI	Latent Semantic Indexing
LVQ	Learning Vector Quantization
ME	Maximum Entropy
MI	Mutual Information
mRMR	Minimum Redundancy and Maximum Relevance
MSPSO	Multi-Swarm Particle Swarm Optimization
NB	Naive Bayes
NN	Neural Network
NNs	Neural Networks
OM	Opinion Mining
PCA	Principal Components Analysis

POS	Parts-Of-Speech
PSO	Particle Swarm Optimization
RIDOR	Ripple down Rule Learner
RNN	Layer Recursive Neural Network
RNTN	Recursive Neural Tensor Net (RNTN).
RST	Rough Set Theory
S A	Sentiment Analysis
SGD	Stochastic Gradient Descent
SVD	Singular Value Decomposition
SVM	Support Vector Machine
T N	True-Negative
TF	Term Frequency
TF-IDF	Term Frequency-Inverse Document Frequency
TP	True-positive
VSM	Vector Space Model
WLLR	Weighted Log Likelihood Ratio

CHAPTER 1

INTRODUCTION

1. Introduction

1.1 Introduction

Thousands of comments on news, business, politics, and stock market flow every day through the web. This enormous size of data makes it impossible to follow and analyze such data manually. According to Liu, B. and Zhang, L. in [1], financial news interpretation is important for the stock market investors as it makes decision making less risky. However, humans tend to be biased in their analysis of information and interpret textual information in a way that fits their preferences, therefore, building a system capable to analyze the opinions and news helps in making accurate interpretations and simplifies the task of decision making.

1.2 Motivation and Problem Studied

Governments would like to know how the voters feel towards the states policy in running its departments, agreements, public services, etc. There is a necessity to know the public opinion in events that affect security, shakes the economy, or the state's ability to manage risks and provide safe living for the people [2].

Universities seek to know the public opinion and that of their own students about the electronic services they provide, such as registration, faculty, continuing education, etc.[2].

Health care institutions need to know patients' opinions to provide the satisfactory healthcare that people want and deserve [3].

Companies want to know what the consumers feel about a product or movie, what are their recommendations or satisfaction about this product, and marketing campaign analysis and the most affected geographic locations by them. They also need to know

what would improve and boost their investment return in that product, and peoples' opinion about the services they offer or the trademark they own. All this knowledge could be used to satisfy customer demand immediately and help improve the product quality [1].

1.3 Sentiment Analysis

Finding people's opinions towards an entity is called Sentiment Analysis (SA) or Opinion Mining (OM). An entity is a certain topic of interest which can be a person, a product, a service, or an event that is being reviewed. Both expressions (i.e. SA and OM) are interchangeable, although some of the works that we have reviewed tend to differentiate between the two [4], such that OM is used for extracting and analyzing opinions, while SA is used for the works of finding opinions, identifying the sentiment, and then classifying text's polarity as shown in Figure 1.1. The figure shows that SA is a classification process [5].

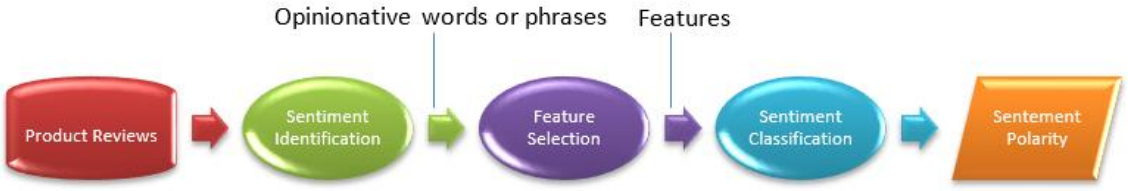

Figure 1.1 Sentiment analysis process on product reviews

Sentiment analysis comes in three levels [6]: document level which classifies an entire document as positive or negative, sentence level where a document is partitioned into sentences and each sentence is classified as positive or negative, and aspect level which is used when a document or sentence refers to different aspects or topics and we need

the sentiment about each aspect or topic. In this thesis we conduct document level sentiment analysis.

1.4 Objectives and Goal

High dimensionality and low accuracy are examples of the deficiencies that sentiment analysis suffer from.

The main goal of this thesis is to reduce the size of the feature subset that resulted from reviews processing which was very high. This is achieved by eliminating irrelevant and redundant features to get only the relevant and important features that enables the classifier to classify the samples datasets into two classes. The approach also aimed at reaching high accuracy better than what others achieved. This can be done by utilizing:

1. Feature weighting by machine learning in this case svm.
2. Combining statistical and machine learning weighting, correlation and svm hybrid.
3. Optimizing the feature subset with Genetic Algorithm.

1.5 Research Scope

The scope of this thesis is to analyze people's opinions and their sentiment towards movies, by referring to English movie reviews data sets and classify the reviews as positive or negative in order to determine their attitude.

1.6 Contribution

Statistical methods are the most common among researchers for feature selection, in our approach we used:

1. Machine learning (weight by SVM):

   In statistical methods the weight vector is not adjustable, while in weight by SVM the weight vector is tunable to fit the learning process in order to achieve higher accuracies.

2. Optimize Selection Evolutionary (GA):

   GA is a generic population-based metaheuristic optimization algorithm. This algorithm is used to enhance the subset of features produced by machine learning, to reduce its size while maintaining high classification accuracy.

1.7 Research Methodology

Figure 1.2 shows the research methodology of this thesis.

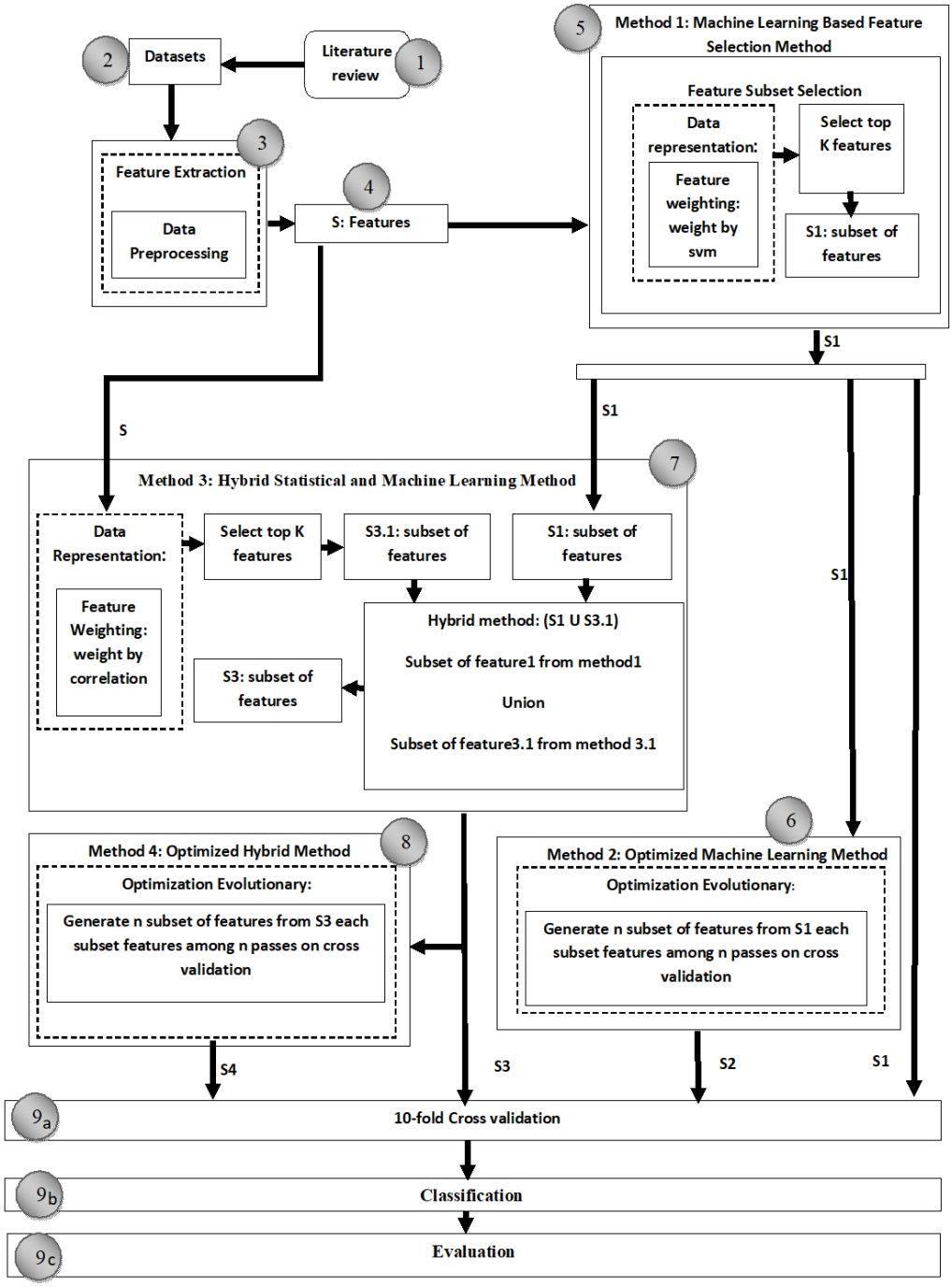

Figure 1.2 Research Methodology of the Thesis.

Where in the above figure:

S: the feature set resulting from the preprocessing of dataset reviews

S1: the feature subset resulting from Method 1

S2: the feature subset resulting from Method 2

S3.1: the feature subset resulting from Method 3 using weight by correlation

S3: the final feature subset resulting from Method 3

S4: the feature subset resulting from Method 4

The following lists more information about Figure 1.2:

1. Literature review shown in chapter 3.
2. For sentiment analysis classification in our approach two well-known datasets (D1, D2) along with a third (D3) resulting from their merger were used. This data was widely used by researchers in sentiment analysis and classification. Section 5.2.2 discusses the data set.
3. Feature extraction includes preprocessing done on the reviews in these datasets which included tokenization, case normalization, and stop word removal to produce the complete feature set S. Preprocessing is discussed in section 4.1.2.
4. The types of features that were experimented with included POS and unigrams. Results using unigrams were better than using POS that is why unigrams was used in our approach as feature type to produce the feature set S. See section 4.1.1.
5. In method 1: Machine learning based feature selection method the feature subset used resulted from weighting by SVM and then choosing the top K features to extract S1 Subset of features. See section 4.3.

6. In Method 2: Optimized machine learning method, we used S1 that resulted from method 1 and reduced the size of the feature subset to better improve the accuracy using optimization evolutionary (GA). See section 4.4.

7. In Method 3: Hybrid statistical and machine learning method, features were weighted using statistical method weight by correlation then top K features were selected to get S3.1 feature subset. Then S1 feature subset from method 1 was merged with S3.1 and then weighted again using weight by SVM to get S3 feature subset. See section 4.5.

8. In Method 4: Optimized Hybrid Method, we used S3 feature subset that resulted from method 3 and further enhanced it using optimization evolutionary (GA) decrease its size to produce S4 and increase the classification accuracy. See section 4.6.

9. In each of the four methods:

    a. Stratified sampling 10 fold validation was used on the datasets which is common. See section 2.5.2.1.

    b. SVM classifier used for classification. See section 4.1.4.

    c. Evaluation measures included: Accuracy, Recall, Precision, and F-measure. See section 2.5.2.2.

1.8 Thesis Overview

This thesis is organized as follows:

Chapter 2, introduces background topics. We begin by explaining the data preprocessing steps which include tokenization, stop words removal, and case normalization. Then we explain the concept of Vector Space Model (VSM). We also go through feature selection concepts including feature types, and feature weighting methods like weight by SVM, Correlation, IG, and TF-IDF. Then we present Classification algorithms where we explain machine learning algorithms, validation especially cross-validation, and evaluation measures like accuracy, precision, recall, and F-measure.

Chapter 3, literature review, goes through some of the latest proposed approaches in SA. Here we categorized the approaches reviewed into feature selection targeted approaches that separate into optimization and statistical approaches, machine learning approaches, and hybrid methods which are also subcategorized into hybridized and neuro fuzzy approaches.

In chapter 4, the methodology of the proposed approach is presented. Firstly, we discuss the basic concepts of SA. Then we explain GA and how it works. After that we present our first proposed method called Machine Learning Based Feature Selection Method where we use weight by SVM for feature weighting and SVM classifier for classification. Then the second method called Optimized Machine Learning Method is introduced. In method 2 we apply optimization evolutionary algorithm on the resulted feature subset of method 1 to further enhance the results. After so, Method 3 is presented. This method is called Hybrid Statistical and Machine Learning Method. Here

we combine by a union between the top 4000 features resulted from weight by SVM and the top 4000 features resulted from weight by correlation. Then we use the SVM classifier and cross-validation to train the model. Then we introduce method 4 called Optimized Hybrid Method, where we use optimization evolutionary on the resultant feature subset from method 3, and show that this method gives higher accuracy in classification.

Chapter 5, goes through the experiments conducted during the research. At first, we describe the setup. We explain why we chose RapidMiner as a tool for experimentation. We talk about the datasets which were used to conduct experiments. Then we go through each one of the four methods and its experiment. In each method we present all results obtained for each dataset including accuracy, precision, recall, and f-measure. Then we discuss the results showing that weight by SVM is the best feature weighting method, and using GA enhances those results. In the comparison section, we compare our approach to some of the latest published SA approaches. In our comparison we focused mainly on two factors, accuracy, and number of features used for classification.

Lastly, **Chapter 6** is the conclusion and future work.

1.9 Summary

The competition is high, the market is risky, and consumers are very hard to be satisfied. The edge goes to producers who know what their customer needs. This is why product reviews are very important and their analysis is crucial. Hence the attention of researchers in sentiment analysis increased. The main point is to analyze people's sentiment towards a certain product, service, or event to help decision makers make the right decisions.

CHAPTER 2

BACKGROUND

2. Background

2.1 Introduction

Sentiment is simply a negative or positive opinion [7][8][9]. The process of finding the opinion towards a particular entity computationally using natural language processing is called sentiment analysis.

In this chapter we will go through some of the main concepts related to sentiment analysis, for example: data preprocessing, feature weighting, feature selection, and machine learning.

2.2 Data Preprocessing

When using supervised machine learning techniques data preprocessing is a very important step. Standard preprocessing techniques [10] are used to reduce the feature space. These techniques include tokenization, stop word removal, and case normalization.

2.2.1 Tokenization

Tokenization is splitting the text into individual tokens, usually words. The simplest way is to split the document into separate words using the space character for example the sentence “Jenin is a city in the northern part of the west bank” consists of 12 different words or tokens: “Jenin, is, a, city, in, the, northern, part, of, the, west, bank”.

2.2.2 Stop Words Removal

Removing words that bear no importance in sentiment analysis like a, an, and, the, but, if, or, etc. For example, the sentence “Jenin is a city in the northern parts of the west bank” has 6 stop words “is, a, in, the, of, the”. That leaves only “Jenin, city, northern, parts, west, bank” that have importance.

2.2.3 Case Normalization

The English language letters have two cases either upper case or lower case. To normalize the text is to transfer all letters of the document into lower case.

2.3 Vector Space Model (VSM)

An effective and common statistical representation of a collection of documents proposed by Salton et al. in 1975 [11]. VSM considers the document as a vector of terms d = (t_1,t_2,…,t_n) with a corresponding vector of weights w = (w_1,w_2,…,w_n) with w_i being the weight of the term t_i. VSM model is visualized as a 2-dimetional matrix, where documents are represented as rows and features (words or tokens) are represented as columns as shown in Figure 2.1.

Document	still	text	black	detect	Sentence	in	is	short	evil	this	we
DOC A					1		1	1		1	
DOC B		1	1				1			1	
DOC C		1	1		1	1	1			2	
DOC D	1	2	1	1					1	2	1
***Document Frequency	1	3	3	1	2	1	3	1	1	4	1

*** Number of documents in which the word appears

Figure 2.1 VSM representation of a collection of documents

As an example on VSM creation [12], consider the dataset in Table 2.1, each document is comprised of a number of words separated by spaces. Each word in the example is considered a feature, so the set of words in all documents without repetition comprise the feature space. Figure 2.1 shows the words in bold that constitute the feature space of this dataset.

Table 2.1 Dataset example

Document	Content
Doc A	This Sentence is short.
Doc B	This text is black.
Doc C	This text in this Sentence is black.
Doc D	We still detect this black text, this evil text.

The VSM model in Figure 2.1 shows the representation of this dataset in VSM format. The word occurrence is used as weight w, where the empty cells indicate that this word does not occur in this document. The dataset frequency in Figure 2.1 represents the sum of word occurrence in all documents. The document frequency row represents the number of document that the word occurred in, and the document length column represents the unique word count in each document. The feature space is represented by bold words in row 1.

2.4 Feature Selection

Extracting features from unstructured text is the key to sentiment analysis. Mostly existing approaches of sentiment analysis develop machine learning models based on N-grams (i.e. unigrams, bigrams, trigrams, etc.) and some use Part-of-Speech (POS). A feature is relevant if removing it degrades the performance. Irrelevant features are those that are not necessary for classification [13]. In this research we experimented on two types of feature: unigrams, and POS.

2.4.1 Feature Types

2.4.1.1 Unigrams

An n-gram feature is a sequence of n contiguous terms of text. These terms can be letters, words, phonemes, syllables or base pairs. The shortest n-gram is of size 1, i.e. when each single word is taken separately, this is called a Unigram [14].

2.4.1.2 Part of Speech

The English language has 9 categories that each word can be assigned according to the syntactic meaning. These categories are called parts of speech and they are noun, pronoun, adjective, determiner, verb, adverb, preposition, conjunction, and interjection [15].

2.4.2 Feature Weighting

There are many weighting methods in which features could be weighed. In this section we focus on few of them.

2.4.2.1 Weight by SVM

This method calculates how relevant a feature is by calculating weights of the features with respect to the class. It uses the coefficients of the normal vector of a linear SVM as feature weights [16] [17].

2.4.2.2 Weight by Correlation

Weight by correlation calculates the relevance of the feature by computing the value of correlation for each feature of the input features with respect to the class attribute. This weighting scheme is based upon correlation and it returns the absolute or squared value of correlation as feature weight [18].

2.4.2.3 Weight by Information Gain

Weight by Information Gain calculates the weight of features with respect to the class attribute by using the information gain ratio. The higher the weight of a feature, the more relevant it is considered [19].

2.4.2.4 Weight by Gini Index

Weight by Gini Index calculates the weight of a feature with respect to the class attribute by computing the Gini index of the class distribution, if the given processed dataset would have been split according to the feature. Gini Index is a measure of impurity of a processed dataset. The higher the weight of a feature, the more relevant it is considered [20].

2.4.2.5 Term frequency-inverse document frequency (TF-IDF)

Term frequency (TF) is the weight of the term in the document, it assumes that more occurrences of a term in a short document gives it more significance. Document Frequency (DF) represents the number of documents in the dataset in which the term occurred, it assumes the more documents the term occurs in the more significant it is.

Term frequency-inverse document frequency (TF-IDF) assumes that less frequent terms in the dataset are more significant in the document and vice versa [12].

Equations 2.1, 2.2, and 2.3 [12] show the formulas of TF-IDF, where:

$fr_{t,d}$ is the word (term) *t* frequency in document *d*,

n is the number of unique words (terms) in a document,

N is the total of documents in the dataset,

F_t is the frequency of the word (term) *t* at the dataset level.

$$TF_{t,\mathrm{d}} = \frac{fr_{t,\mathrm{d}}}{\sqrt{\sum_{t=1}^{n} t \; f\,\mathrm{r}_{t,\mathrm{d}}{}^{2}}} \tag{2.1}$$

$$\mathrm{DF}_t = \sum_{d=1}^{N} \begin{cases} 1\ldots, \mathrm{t} \in d \\ {}_0\ldots, \mathrm{t} \notin d \end{cases} \tag{2.2}$$

$$TF - IDF_{t,d} = TF_{t,d}.IDF_t \qquad \text{where} \quad IDF_t = \log\left(N/DF_t + 1\right) \tag{2.3}$$

The VSM matrix of the data set in Table 2.1 based on the TF-IDF weighting formula is shown in Figure 2.2.

Document	still	text	black	detect	Sentence	in	is	short	evil	this	we
DOC A					0.8466		0.6438	1.1931		0.5000	
DOC B		0.6438	0.6438				0.6438			0.5000	
DOC C		0.4292	0.4292		0.5644	0.79540	0.4292			0.6667	
DOC D	0.6618	0.7143	0.3571	0.662					0.6618	0.5547	0.6618

Figure 2.2 VSM weights matrix based on TF-IDF

In the above figure the term “short” has the highest TF-IDF value since it appears only once in one document that is the shortest document in the dataset. Words like “detect” and “evil” also appear only once but in longer documents and that’s why “short” scored higher.

2.5 Classification

2.5.1 Machine Learning Algorithms

Machine learning is used in text classification in two ways: traditional topic-based text classification that uses topic-related words to categorize documents into different classes like “sports, news, etc.”, and sentiment detection that uses sentiment bearing words to determine the sentiment of a document. Machine learning-based sentiment analysis go through multiple phases that are:

1. Preprocessing.
2. Feature selection
3. Feature weighting.
4. Machine Learning Algorithm.

Three categories of learning algorithms are used in sentiment analysis:

1. Supervised Learning Algorithms
2. Unsupervised Learning Algorithms
3. Semi-supervised learning

These types are further divided into subcategories as shown in Figure 2.3 [21].

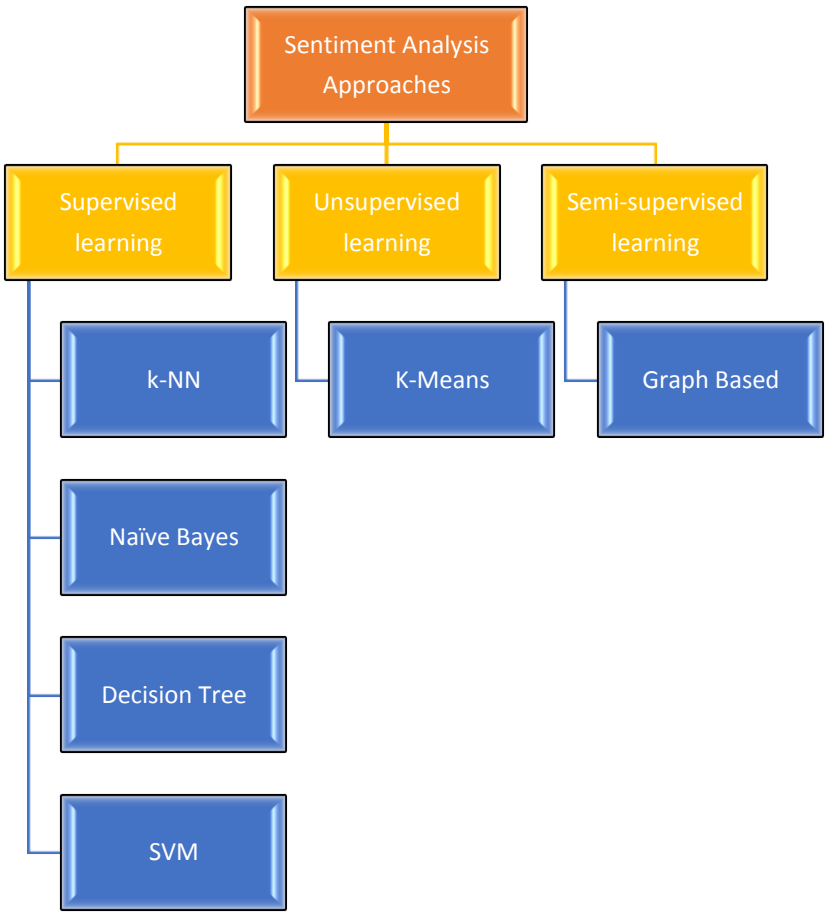

Figure 2.3 Machine learning in sentiment analysis

In this research we used a supervised machine learning algorithm called Support Vector Machine (SVM).

The SVM classifier has outperformed other classifiers and became dominant in text classification [22]. The SVM classifier is superior when compared to the Decision Tree

(DT) classifier [23, 24]. Besides that, SVM support limitless features. SVM classifier represents examples as points in a high dimensional space where different classes are separated by a distance as wide as possible. New examples belong to a class if their points are closer to its side of the hyper plane. Support vectors, which are small subsets of the training set, are used to determine the hyper planes.

2.5.2 Validation and Evaluation

2.5.2.1 Cross-validation

Cross-validation is a popular statistical and machine learning method for modeling and tuning parameter selection due to its simplicity and applicability [25]. Cross-validation aims to make the performance evaluation unbiased by fitting and evaluating each candidate model on separate data sets. One of the most popular methods to evaluate classification algorithms is k-fold cross-validation. In this method the data set is randomly divided into k subsets of approximately equal sizes and the model generated from k-1 folds is tested against the remaining fold in turn. The performance of the model is the average of the k accuracies that resulted from the k-fold cross validation. In the proposed approach we used 10-fold cross validation [26].

2.5.2.2 Evaluation Measures

To evaluate the classification results a few definitions need to be made and measured. True-positive (TP) is the number of positive reviews correctly classified as positive. True-Negative (TN) is the number of negative reviews correctly classified as negative. False-positive (FP) is the number of negative reviews that were misclassified as positive. False-negative (FN) is the number of positive reviews that were misclassified as negative.

Table 2.2 shows an example confusion matrix that has 500 reviews with 250 positive and 250 negative reviews. After applying the classification to the dataset 245 of the positive reviews were classified as positive and the remaining 5 wrongly classified as negative. 247 negative reviews were correctly classified as negative and the remaining 3 were wrongly classified as positive. That means that true-positive is 245, false-negative is 5, true-negative is 247, and false-positive is 3.

Table 2.2 Confusion Matrix

Reviews = 500	Predicted N	Predicted P	
Actual N	TN = 247	FP = 3	250
Actual P	FN = 5	TP = 245	250
	252	248	

2.5.2.2.1 Accuracy

One of the most important measures of classification is the accuracy (a) which is the ratio between reviews correctly classified and the total number of reviews (n) [27]:

$$a = \frac{TP+TN}{n} \tag{2.4}$$

If we apply equation 2.4 to Table 2.2 then the accuracy is as follows:

$$a = \frac{245 + 247}{500} = 0.984$$

That means the accuracy is 98.4%.

2.5.2.2.2 Precision

Precision (p) is the ratio between correctly classified reviews and all classified reviews [27]:

$$p = \frac{TP}{TP+FP} \tag{2.5}$$

If we apply equation 2.5 to Table 2.2 then the precision is as follows:

$$p = \frac{245}{245 + 3} = 0.987$$

2.5.2.2.3 Recall

Recall is the ratio between correctly classified positive reviews and all positive reviews [27]:

$$r = \frac{TP}{TP+FN} \quad (2.6)$$

If we apply equation 2.6 to Table 2.2 then the Recall is as follows:

$$r = \frac{245}{245 + 5} = 0.98$$

2.5.2.2.4 F-Measure

F-measure is the harmonic mean of precision and recall. It is used to optimize the classification system towards either recall or precision which will affect final results [28].

$$F - Measure = \frac{2*p*r}{p+r} \quad (2.7)$$

Where p is the precision and r is the recall. In our example, f-measure = (2*0.987*0.98)/(0.987+0.98) = 0.9835.

2.6 Summary

To further simplify the understanding of the thesis, and to make it more beneficial to others, we have shed the light on some of the concepts related to sentiment analysis. These concepts included for example types of features, feature weighting, feature selection, cross validation, SVM classifier, machine learning, and many other concepts which will aid the reader in understanding the details of our approach accurately and easily.

CHAPTER 3

LITERATURE REVIEW

3. Literature Review

3.1 Introduction

Researchers take great interest in opinion mining, especially that the web has provided a huge size of data. Since people tend to be biased when analyzing data according to their personal preferences, building a system that analyzes opinions accurately and in an unbiased manner became a necessity in order to aid decision makers take the right decisions. This process is called sentiment analysis.

Sentiment analysis approaches on product reviews mostly focus on classifying text entities as positive or negative. Existing approaches can be categorized as supervised methods that need training data, and lexicon-based methods that rely on dictionaries. Recently, heuristic search based algorithms are being used in many approaches for feature selection in sentiment analysis.

In this chapter we go through some of the latest approaches and published works related to text classification and sentiment analysis which we will divide into subcategories which include works that have focused on feature selection in both statistical and optimization approaches. We will also go through some studies aimed at machine learning in sentiment analysis, and others that hybridized multiple techniques especially in the field of genetic algorithms and neuro fuzzy.

3.2 Feature Selection in Sentiment Analysis

After the weighting process is done a part of the resulting features are selected for classification purposes. Some researchers select features that have a weight less than a certain weight threshold [29], others chose terms that have weights above a certain

threshold [30], some chose terms that have a weight equal to a certain weight, and some chose a number of the highest weighting features (top K) [31].

3.2.1 Optimization Approaches

Many optimization algorithms are Nature inspired algorithms that are very efficient and robust in various domains. One widely used algorithm due to its fast convergence and simple implementation is Particle Swarm Optimization (PSO) algorithm [32].

A great number of studies implemented PSO. Wahyudi et al. in [33] implemented PSO optimization with SVM classifier to analyze user sentiment on a smartphone dataset that they obtained from gsmarena website (www.gsmarena.com). They implement their system using RapidMiner tool. When comparing the results of SVM with and without optimization it was found that PSO improved the accuracy from 82% to 94.5%.

Liu, Zhi, et al. [34] used an improved version of PSO called Multi-Swarm Particle Swarm Optimization (MSPSO) to improve the micro f-measure of SVM classifier when they built their sentiment analysis model using Liblinear tool for a course reviews dataset from MOOC platform. The authors of [35] used CRF classifier with PSO on Restaurant Review dataset and found that using PSO increased the accuracy of the system.

Umamaheswari et al. [36] used a hybrid method combining PSO and SVM to do sentiment analysis on movie reviews in twitter, they used LIBSVM tool and found that PSO increased the accuracy of the SVM classifier by 13.93%. In [37] Stylios and his colleges used Decision Tree classifier along with hybrid PSO/ACO2 to build a system to analyze product reviews dataset that they obtained from Greek Fora and were able to increase the accuracy to 90.59%.

Using a dataset called movie review in twitter[1] from the Stanford repository, the authors of [22] integrated PSO with SVM to improve the classification accuracy of SVM. In [38] Aghdam and his co-researchers integrated k Nearest Neighbor (KNN) classifier with PSO on Reuters-21578 dataset, they found that PSO converges quickly, has a strong search capability, and efficiently finds minimal feature subsets.

Another nature inspired algorithm is Artificial Bee Colony (ABC). ABC was used by many researches to optimize their systems. Saravanan et al. in [39] implemented ABC with Interactive Dichotomizer version 3 (ID3) which is a decision tree classifier and introduced a weighting algorithm called improved high adjective count (IHAC) to build an effective system for opinion mining. They extracted a product review dataset on cameras from amazon and achieved a precision of 94.45%.

Dhurve et al. [40] integrated SVM with ABC. They used a product review data set from the University of Illinois at Chicago repository. They found that using ABC increased the accuracy of SVM by 15%. Sumathi et al. [41] used ABC to optimize the Naïve Bayes based system they built for the movie review IMDb dataset and showed that using ABC increased the accuracy of classification.

Yousefpour et al. [42] experimented with multiple classifiers including SVM, NB, ME, and LDF. For feature selection, they employed meta-heuristic algorithms (HS and GA) based on IG, CHS, DF, SD, WLLR on integrated POS and n-grams extracted from a book review, electronic review, and music review datasets. The accuracies they achieved were 94.98%, 93.25%, and 92.97% for the three datasets respectively.

[1] http://www.stanford.edu/~alecmgo/cs224n/trainingandtestdata.zip

Using genetic algorithms for feature selection and a Bayes network classifier built in Weka to classify documents in two datasets (Stanford Twitter Sentiment (STS) and Sanders), Keshavarz et al. [43] achieved an accuracy of 84.1%. Wahyudi and Putri [44] used the SVM classifier along with GA depending on term presence and frequency of n-grams and POS on a twitter dataset extracted from peoplebrowser.com and achieved an accuracy of 93.5%. Abualigah et al. [45] employed k-mean classifier along with TF-IDF, GA, and FSGATC for feature selection on terms of the Reuters-21578 and 20Newsgroups datasets which they evaluated using Accuracy, Precision, Recall, and F-measure where the top accuracy they achieved was 55.78%.

S. R. Ahmad et al. [46] compared different types of feature selection in sentiment analysis based on natural language processing and modern methods such as genetic algorithm and rough set theory. They emphasized that feature selection is an important step in sentiment analysis because a suitable feature selection can identify the actual product features criticized or discussed by consumers. And they suggested that metaheuristic based algorithms have the potential to be implemented in sentiment analysis research and can produce an optimal subset of features by eliminating features that are irrelevant and redundant.

Table 3.1 Optimization Approaches

Reference	Year	Approach	Feature	Dataset	Evaluation	Achievement
[22]	2013	PSO with SVM	Term	Stanford Movie Review in twitter	Precision, Recall, Accuracy	Accuracy: 77%
[33]	2016	PSO with SVM	Term	Gsmarena smartphone dataset	Accuracy Precision Recall	Accuracy: 94.5%
[34]	2016	MSPSO with SVM	Term	Course reviews from MOOC	Precision Recall F-Measure	F-measure 88%
[35]	2015	PSO with CRF	Lexical	Restaurant Review	Recall, precision F-Measure Accuracy	Accuracy 78.48%
[36]	2015	PSO with SVM	POS	Movie Review	Precision, Recall, Accuracy	Accuracy: 81.6%
[37]	2014	hybrid PSO/ ACO2 with Decision Tree	POS	Product Reviews	Accuracy	Accuracy: 90.59%
[38]	2015	PSO with K-NN	Term	Reuters dataset	Precision, Recall	Precision 77.4619% Recall 79.8411%
[39]	2016	ABC/ IHAC with ID3	POS	Product Review from amazon	Precision Recall F-measure Accuracy	Precision: 94.45%
[40]	2015	ABC with SVM	BOW, BON	Product Reviews	Precision Recall F-measure Accuracy	Accuracy: 70%
[41]	2014	ABC/ IDF with NB/FURIA/RIDOR	Term	IMDb	Precision, Recall, Accuracy	Accuracy: 93.75%
[42]	2016	HS/GA with SVM/NB/ME/LDF	POS, n-grams	Product Reviews	Accuracy	Accuracy: 94.98%
[43]	2016	GA with Bayes network	objective and subjective words	STS and sanders	Accuracy	Accuracy: 84.1%
[44]	2016	GA with SVM	N-gram and POS	Twitter dataset from peoplebrowser.com	Accuracy	Accuracy 93.50%
[45]	2016	TF-IDF/GA/ FSGATC with K-Mean	Terms	Reuters-21578 and 20Newsgroups	Accuracy F-measure	Accuracy: 55.78%
[46]	2015	Comparative study of natural language processing and modern methods for feature selection such as G.A and RST.				

In Table 3.1 the researchers used optimization methods like G.A, PSO, Ant colony, etc. to improve feature selection which gave high results. This drove me to conduct experiments on choosing feature subsets using these advanced methods. I found that G.A gave better results than the others, and that is why it was the method chosen to enhance feature subset selection in my experiments as shown in sections 4.4 and 4.6.

3.2.2 Statistical and Machine Learning Approaches

Deng et al. [47] used the importance of a term in a document (ITD) and the importance of a term for expressing sentiment (ITS) along with SVM Classifier. They experimented with multiple datasets: the polarity dataset, a product review dataset from amazon.com, and a large movie review dataset from the Stanford repository in which they achieved accuracies of 88.5%, 88.7%, and 88% respectively. Using the movie review dataset V1.0 and V2.0, Parmar et al. [48] with a random forest classifier along with information gain and gain ratio for feature selection used on unigrams achieved accuracies of 87.85% for dataset v1.0 and 91.0% for dataset v2.0.

Gautam (2014) et al. [49] studied an approach in which they extracted the adjective from a dataset (labeled tweets) that have some meaning which is called feature vector, then selected the feature vector list and thereafter applied machine learning based classification algorithms namely: Naïve Bayes, maximum entropy and SVM along with the semantic orientation based wordnet which extracts synonyms and similarity for the content feature. The results showed that the Naïve Bayes technique when subjected to unigram model gives a better result than the maximum entropy and SVM. Further the accuracy is again improved when the semantic analysis wordnet is followed up, taking it to 89.9% from 88.2%.

Po-Wei Liang et.al. [50] used twitter API to collect twitter data. Their training data falls in three different categories (camera, movie, mobile). The data is labeled as positive, negative and non-opinions. Tweets containing opinions were filtered. Unigram Naive Bayes model was implemented and the Naive Bayes simplifying independence assumption was employed. They also eliminated useless features by using the mutual information(MI) and chi square feature extraction method. Finally, the orientation of a tweet is predicted (i.e. positive or negative.) with a better accuracy of 70.39% in comparison with the traditional model 67.58%.

In [51] Cardoso et al. used General Fuzzy Reasoning Method (GFRM) and Correlation-Based Feature Selection (CFS) on movie reviews dataset from amazon and achieved an accuracy of 71.25%. Agarwal et al. [52] built their approach using the SVM and NB classifier based on Rough Set Theory (RST) and IG for feature selection on features of type unigram extracted from the movie review and product review datasets where they increased the f-measure from 84.2% to 87.7%. Agarwal et al. [53] proposed an approach based on SVM, dependency parsing-based semantic parser with common-sense knowledge and mRMR with multiple feature types containing unigrams, bigrams, and POS. Their system was tested on movie reviews, books, DVD, and electronics datasets achieving an accuracy of 90.1%.

Li et al. [54] proposed a clustering-based approach that utilizes k-means and TF-IDF on terms extracted from the movie reviews dataset resulted in an accuracy of 78%. Paltoglou et al. [55] used SVM and TF-IDF on unigrams to classify the documents of the movie reviews, multi-domain sentiment, and BLOGS06 corpora datasets. The system achieved an accuracy of 96.9%.

Sumathi et al. [56], in their research aimed at analyzing the performance of classification methods for opinion mining, used Naïve Bayes, Fuzzy Unordered Rule Induction Algorithm (FURIA), and the Ripple Down Rule Linear Algorithm (RIDOR). For feature selection IDF was used on the Internet Movie Database IMDb dataset. Their comparison was evaluated using accuracy, precision, recall, and root mean square error (RMSE) their accuracy was 92.25%. In [57] Jotheeswaran et al. proposed a technique for feature reduction using principal component analysis for opinion mining. The implemented Naïve Bayes, Linear Vector Quantization (LVQ), and Classification and Regression Tree (CART). For selecting terms, they used IDF and principal component analysis (PCA) on movie reviews from the Internet Movie Database (IMDb). They used root mean square error (RMSE), accuracy, precision, and recall for evaluation where LVQ achieved 75% accuracy.

Valarmathi and Palanisamy [58] proposed a classification method using key word summarization based on singular value decomposition. For the classifier CART and Bayes Net were used to classify words extracted from the movie review dataset. Where the accuracy was 78.667%, and 76% for Bayes Net and CART respectively.

Sahayak et al. [59] proposed an approach which automatically classifies the tweets as positive, negative or neutral with respect to the query term. It uses the POS-tagging and the tree kernel to prevent the need for feature engineering. But the difficulty increases with the complexity.

Pablo et al. [60] presented variations of Naive Bayes classifiers for detecting polarity of English tweets. Two different variants of Naive Bayes classifiers were built namely baseline (trained to classify tweets as positive, negative and neutral), and binary (makes

use of a polarity lexicon and classifies as positive and negative. neutral tweets neglected). The features considered by classifiers were lemmas (nouns, verbs, adjectives and adverbs), polarity lexicons, and multiword from different sources and valence shifters. The results show that there is an improvement in performance (f-score = 0.63) when the classifiers are implemented with the binary strategy, when they use a polarity lexicon, and when multiword are considered as features.

Agarwal et al. [61] used SVM and NB classifiers along with Minimum Redundancy Maximum Relevancy (mRMR) and Information Gain (IG) on terms of types unigram, bi-gram, and POS to study their effect on four standard datasets. They have observed that mRMR has better performance than IG due to its capability to select relevant features and eliminate redundancy unlike IG which only finds feature importance. Where BMNB scored 82.7% in F-measure and ComPmRMR scored 91.1%.

Table 3.2 Statistical and Machine Learning Approaches

Reference	**Year**	**Approach**	**Feature**	**Dataset**	**Evaluation**	**Achievement**
[47]	2014	ITD/ ITS with SVM	Term (word)	Polarity Dataset, Product Reviews from amazon, large movie review from Stanford	Accuracy	Accuracy: 88.5%, 88.7%, 88%
[48]	2014	IG and gain ration with Random forest	Unigrams	Movie Review v1.0 and v2.0	Accuracy	Accuracy: 87.85%, 91.0%
[49]	2014	NB, Maximum entropy(ME), and SVM	POS	Labeled Tweets	Accuracy Recall Precision	Accuracy: 89.9%
[50]	2013	NB	Unigrams	Twitter	Accuracy	Accuracy: 70.39%
[51]	2015	CFS, C4.5 with GFRM and CFRM	N-gram	Movie Review from Amazon	Precision, Recall, Accuracy, F-Measure	Accuracy: 71.25%

Reference	Year	Approach	Feature	Dataset	Evaluation	Achievement
[52]	2013	RST and IG(Hybrid IG-Rough) with SVM and NB	Unigrams	Movie review, product (book, DVD and electronics) review dataset.	F-measure	F- measure is increased from 84.2% to 87.7
[53]	2015	mRMR with SVM	Unigrams, Bigrams, and POS	Movie and product reviews	Accuracy	Accuracy: 90.1%
[54]	2010	TF-IDF with K-Means	Terms	Movie Reviews	Accuracy	Accuracy: 78%
[55]	2010	TF-IDF with SVM	Unigrams	Movie Reviews, multi-domain sentiment, and BLOGS06 corpora datasets	Accuracy	Accuracy: 96.9%
[56]	2013	IDF with NB, FURIA, RIDOR	words	IMDb	Accuracy Recall Precision RMSE	Accuracy: 92.25%
[57]	2012	IDF and PCA with NB, LVQ and CART	Terms	IMDb	RMSE Accuracy Recall Precision	LVQ accuracy: 75%
[58]	2011	CART/ SVD and Bayes Net	Words	IMDb.	Accuracy.	Accuracy: BayesNet: 78.667% CART: 76%
[59]	2015	POS-tagging And tree kernel SVM NB	POS	Tweets	-	-
[60]	2014	NB	unigrams of lemmas and valence shifters polarity lexicon multiwords	SemEval-2014 (tweeti-b.dist.tsv).	F-score	F-score: 0.63
[61]	2013	mRMR and IG with SVM and Boolean Multinomial Naïve Bayes (BMNB) algorithm	Unigram, Bigram, and POS	Cornell Movie Review Dataset , amazon product reviews of books, DVD and electronics	F- measure	F-Measure BMNB: 82.7% ComPmRMR 91.1%

It’s shown in the above table that researchers used statistical methods in the weighting of features like IG, Gini Index, MI, etc. and these methods are common among researchers for choosing features. This made me look for an unconventional method (none statistical) for feature weighting. I found that it is possible to weight features via machine learning which allows for tuning the weight vector during the learning process. This is shown in section 4.1.4.

3.3 Hybrid Methods

3.3.1 Hybrid GA Approaches

In [62] Govindarajan, hybridized Naïve Bayes with Genetic Algorithm using Arcing Classifier to classify documents of the movie review dataset v2.0. He used term frequency-inverse document frequency (TF-IDF), and latent semantic indexing (LSI) for feature selection. He depended on the grammatical categories and the use of bi-grams achieving an accuracy of 93.8%. Elawady et al. [63] proposed a hybrid feature selection method based on Rough Set Theory (RST) and IG. They considered RST algorithms better than decision tree algorithms, such as ID3 and C4.5 since it gives higher accuracies, finds the minimal feature subsets, and does not need membership functions and prior parameter settings due to its simplicity. They used SVM and NB for classifying product reviews and movie reviews datasets depending on features of type unigram achieving an f-measure of 87.7%.

Jotheeswaran et al. [64] classified the IMDB dataset a binary classification using a multi-layer perceptron neural network. They used decision tree for feature ranking and extraction. GA was used for weight training obtaining an accuracy of 83.25%.

Table 3.3 Hybrid GA Approaches

Reference	Year	Approach	Feature	Dataset	Evaluation	Achievement
[62]	2013	TF-IDF with GA, NB and arcing classifier hybrid NB-GA	Bi-grams	Movie reviews v2.0	Accuracy	Accuracy: 93.8%
[63]	2014	Hybrid IG and RST with SVM and NB	Unigrams	Movie Reviews	F- measure	F-measure 87.7%
[64]	2015	GA with multilayer perceptron NN and decision tree	Term	Movie review from IMDb	Precision, Recall, Accuracy and F-measure	Accuracy: 83.25%

3.3.2 Neuro Fuzzy Approaches

The biological neuron in the main part of the human brain, it's a very complex system that can handle and process huge amounts of information simultaneously. This system inspired the mathematical model called Neural Networks (NNs) [65].

Since its first introduction in 1943 by Warren McCulloch and Walter Pitts [65] NNs have been employed in many researches. In [66] Yuan et al. used Stochastic Gradient Descent (SGD) along with Recursive Neural Network (RNN) and Recursive Neural Tensor Net (RNTN). Their model was built for SemEval-2013 data set collected by York University where they achieved accuracies of 84.17% and 80.68% for one-hidden-layer RNN and two-hidden-layer RNN respectively. Kim and Yoon in [67] used Convolutional Neural Networks (CNN) classifier on multiple datasets and noticed good performance gain for various datasets.

Kabir et al. in [68] implemented Deep Convolutional Neural Networks which is called Character to Sentence Convolutional Neural Network (CharSCNN). They used the Stanford Sentiment Treebank (SSTb) and Stanford Twitter Sentiment corpus (STS) movie review datasets and achieved accuracy improvements on both datasets (SSTb 85.7% STS 86.4%). In [69] the authors implement neural networks with constructive approach for feature selection (CAFS) on various disease datasets from the UCI machine learning repository. They were able to achieve better accuracy in classification with lower number of features when comparing their work with previous researches using other algorithms.

Zhu et al. in [70] integrated Back Propagation (BP) neural networks with Singular Value Decomposition (SVD) to build a model for classifying the news data set obtained from SemEval-200 and SinaNews with accuracies of 40.1% and 59.57% respectively. Azar and Taher in [71] developed a Linguistic Hedges Neural-Fuzzy classifier (LHNFCSF) to classify a medical dataset extracted from the UCI machine learning repository. Their model reduced the dimension of the problem and improved the classification performance.

Akhtar et al. [72] proposed a novel hybrid deep learning architecture (A deep neural network (DNN) is an ANN with multiple hidden layers between the input and output layers. [73]) for sentiment analysis in resource-poor languages. They used CNN architecture to learn sentiment embedded vector and make a final prediction using SVM classifier at the output layer of CNN. A multi-objective GA based feature selection technique was used to optimize the feature set to assist the training of SVM. They observed performance enhancements across the domains and languages compared to the state-of-the-art methods. They used multiple Hindi and English datasets in their

experiments, were their best accuracy achieved using the English restaurant review dataset was 77.16%.

Table 3.4 Neuro Fuzzy Approaches

Reference	**Year**	**Approach**	**Feature**	**Dataset**	**Evaluation**	**Achievement**
[66]	2015	SGD with RNN and RNTN	BOW	SemEval-2013	Accuracy	Accuracy: One-hidden-layer RNN 84.17% Two-hidden-layer RNN 80.68%
[67]	2014	CNN	BOW	Movie reviews, Customer reviews, TREC question dataset, MPQA dataset	Performance	Performance gain for various datasets
[68]	2014	CharSCNN	BOW	SSTb and STS from stanford	Accuracy	Accuracy improved (SSTb 85.7% STS 86.4%)
[69]	2010	CAFS (based on IG) and NN	Term	Diabetes, breast cancer, glass, vehicle, hepatitis, horse colic, ionosphere, and splice junction. Source: UCI machine learning repository	Mean accuracy	NNs produced by CAFS achieved the best classification accuracy with smallest number of features compared to other algorithms
[70]	2016	BP with SVD	Word	SemEval-200, SinaNews	Accuracy	On *SemEval* 40.10% ON *SinaNews* Accuracy: 59.57%
[71]	2014	LHS with LHNFCF	Term	medical datasets: breast cancer, erythemato-squamous disease, thyroid disease UCI machine learning repository	Performance Accuracy error	reduces the dimensions of the problem, and improves classification performance
[72]	2016	CNN multi-objective GA	PoS, Word N-grams, Character Ngrams	Twitter and Reviews datasets	Accuracy	Restaurant Reviews accuracy 77.16%

Table 3.3 and 3.4 show that some researchers hybridized their approaches in many ways:

- One way was choosing the feature subset that was weighted using statistical methods like IG and then applied G.A to get a smaller feature subset with better classification.
- Some combined Neural Networks and Fuzzy logic what is known as neurofuzzy.

I used the hybridization in my methodology in "Method 3: Hybrid Statistical and Machine Learning Method" and I was able by that to minimize the feature subset size and increase the classification accuracy. All of that is shown in section 4.5.

Atmaca et al. [74] compared fuzzy inference systems and neural network approaches with ANFIS method for predicting fuel consumption in a study done using the auto-mpg dataset obtained from the UCI repository. They came to a conclusion that the learning duration of ANFIS is shorter than neural networks but has a constraint on the number of inputs; for large number of inputs the number of fuzzy if-then rules, which equals m^n where m is the number of member functions and n is the number of inputs, is enormous and leads to memory errors.

Relich [75] used ANFIS on Taiwanese manufacturing industry data for management support to build a system for decision support for alternative project choice. He concluded that ANFIS gave better results than neural networks and statistical techniques; but had difficulties in preprocessing, repeatability of experiments, and convergence. Shirvan [76] also used ANFIS and FIS in a comparative study using a concrete mix design dataset and deduced that ANFIS model has better results than Fuzzy expert system.

3.4 Summary

In choosing the features for sentiment analysis, researchers depended on different approaches, nature inspired methods called optimization methods, statistical methods that do not employ optimization techniques, hybrid methods, and machine learning methods.

Many factors have been noticed to have an effect on the results accuracy that the researchers obtained in S.A. these factors included, the weighting method, the type of classifier, the preprocessing of documents, and the type of features. Going through the different approaches that others used, the size of the feature subset and the accuracy obtained varied from one approach to another.

CHAPTER 4

METHODOLOGY

4. Methodology

4.1 Introduction

People use the web to express their opinions on various topics, like product quality, politics, war, services, education, and many other fields of interest. Hence, comes the necessity to analyze this large amount of text regarding a certain subject and figuring out what people think of it. Sentiment Analysis (SA) is a text-mining field that computationally treats opinions, sentiments and subjectivity of text. SA classifies expressions as positive or negative opinions towards the subject of interest after identifying the sentiment expressions, determining their polarity, and relationship to the subject.

Sentiment Analysis and opinion mining have various methods, which include machine learning, lexicon-based, and discourse analysis. After reviewing existing approaches for feature selection in sentiment analysis and analyzing their strengths and weaknesses, we developed our own approach that is based on optimization methods built on genetic algorithm (GA).

4.1.1 Feature Types

During the research two types of feature sets were extracted and experimented:

1. Unigrams: where each single word is considered a feature.
2. Part of Speech: selecting particular word types, here we chose to select only nouns.

Later we will show which feature type gave the best results and was chosen for the further development of our approach.

4.1.2 Preprocessing

To convert the data into a statistical data representation that can be treated by computer algorithms (understandable format) that will be usable in our approach. The preprocessing stage does three things to the raw data in the datasets:

1. Tokenization: braking up the stream of text of each review into words using none letters as the separators.
2. Stop-words removal: removing the English language most common words using a list of stop words in order to only keep the words that are important in specifying the sentiment.
3. Case Normalization: All words are converted into lowercase letters.

For more details on tokenization, stop words, and case normalization see sections 2.2.1, 2.2.2, and 2.2.3 in the background chapter.

After these steps a VSM matrix is created by weighting the resulting features from the preprocessing step using TF-IDF. Sections 2.5.2.5, and 2.3 of the background chapter discuss in details the concepts of TF-IDF and VSM matrix respectively. Algorithm 0 shows the pseudo code for the preprocessing step.

Algorithm 0 Pseudo Code of Preprocessing

```
Preprocessing
 For Each Review in Dataset Do
 {
       //Split each review into a Token list
       TokenList1 = Tokenize (Review) // using none letters as mode of split
         For each Token in TokenList1 Do
                  {
                           // convert all litters into small letter
                           TransformCase (Token)
                           Add Token to TokenList2
                  }
         //Remove English Stop words
        RemoveStopWords(TokenList2)
 }
/* VSM model is a 2-dimetional matrix, where reviews are represented as rows and features
(words or tokens) are represented as columns, containing TF-IDF weights for all features */
 Output Vector Space Model (VSM)
```

4.1.3 Feature Selection

For feature selection different methods are used and compared to obtain higher accuracy, such as statistical and machine Learning methods.

Weighting methods give each feature a weight that reflects its importance to the classification process of the data. Many weighting techniques have been used by

researchers such as Term Frequency (TF) [77] which assumes that more occurrences of a term in a document indicates that this term is significant, Document Frequency (DF) [78] which gives a term a greater weight if it occurs in more documents, Inverse Document Frequency (IDF) used in [79] is the opposite of DF where the term is more relevant if it occurs in less documents, Term Frequency-Inverse Document Frequency (TF-IDF) used in [80] which assumes that less frequent terms are more significant, Entropy used in [81] defines relevant terms as terms that have high occurrence frequency in less documents.

4.1.4 Classification

The main task in sentiment analysis is selecting the minimal feature subset that achieves high accuracy. Achieving satisfying results means selecting the right feature subset, the hard part is minimizing the number of features in such a subset.

For achieving this main task, a series of methods in the experimentation stage have been conducted, by using machine learning for feature weighting and evolutionary algorithm (G.A) for subset optimization. The dataset was split into training and testing sets using stratified sampling and 10-fold cross validation, and for the classification a support vector machine (SVM) classifier was used.

Stratified sampling builds subsets randomly ensuring equal class distribution between subsets i.e. all subsets are homogeneous containing equal proportions of the classes in order to better represent the population.

The SVM classifier constructs a hyperplane in an infinite-dimensional space that could be used for regression, classification, or other tasks. A good feature can reduce the

dimension vector, reduce the computational complexity, improve the computational efficiency, and prevent overfitting [82].

When talking about effectiveness and popularity in supervised learning algorithms SVM takes a place in the front seats. The hyper-plane that separates the classes is found by learning from the training set. This plane is located in hyper-space at a point that maximizes the distance between the closest positive and negative samples that are called support vectors. The weight vector W plays a major role in linear SVM. This vector is perpendicular to the hyper-plane. Also, the bias b that represents the offset of the hyper-plane from the origin has a great impact. To determine the class of unlabeled example x the value of f(x) is calculated by equation 4.1 [12]:

$$F(X) = W^T.\mathrm{X} + b \tag{4.1}$$

The example *X* is classified as positive if *F(X)* is greater than or equal zero and negative otherwise.

The weight vector w is calculated using equation 4.2 [83] which shows that $\overrightarrow{W}$ of the hyper-plane is constructed as a linear combination of $\overrightarrow{d_i}$. Let y_i equal +1(-1), if document

d_i is in class +(-). The solution can be written as:

$$\overrightarrow{W} = \sum_{i=1}^{n} \propto^*_i \; y_i \overrightarrow{d_i} \qquad \propto^*_i \geq 0 \tag{4.2}$$

$\overrightarrow{W}$: Weight vector

$\propto^*$: Lagrange multipliers,

y_i: Equal +1(-1),

$\overrightarrow{d_i}$: Documents vector

n: Number of documents

Where a dual optimization problem is solved to find $\propto^*_i$. Examples with α_i greater than zero are considered of high importance. Those vectors are called support vectors because they contribute in $\overrightarrow{W}$.

In the training phase w is changed if the value of f(x) gave a misclassification, this way the weight is always updated until the best weight is obtained so that it gives the correct value of f(x) with the least error percentage or the highest possible accuracy. When choosing the best values of w with f(x) giving the highest accuracy (i.e. selecting top k w) we are selecting the corresponding top k features from the VSM matrix.

One strong feature of SVM is its ability to handle huge feature spaces, as well as its ability to deal with redundant features. Furthermore, performance wise it's been proven that SVM in many domains including text classification is one of the best performing machine learning algorithms. Although SVM is a binary classifier it can be used in multi-label text classification, Y. Aphinyanaphongs et al. in [84] presented a comparative study between different SVM algorithms applied on various datasets for text classification that are publicly available.

Correlation considers both the usefulness of each feature for the classification process and the inter-correlation of features. If there is no relation between the two features then their correlation is zero. The idea of correlation is to select the features that have low correlation with each other and high correlation to the class label. Equation 4.3 defines heuristic evaluation function as listed by Onan in [85] where he cited it from [86]:

$$M_s = \frac{k\overline{r_{cf}}}{\sqrt{k + k(\mathrm{k}-1)\overline{r_{ff}}}} \tag{4.3}$$

If S is a feature subset that contains k features then M_s is its heuristic merit, r_{cf} is the mean feature-class correlation ($f \in S$) and r_{ff} is the mean feature–feature inter-correlation.

If n is the number of feature pairs, then the correlation r between two Features (X, Y) can be computed by equation 4.4 as follows [87]:

$$r = \frac{\sum xy - \frac{\sum x \sum y}{n}}{\sqrt{\left(\sum x^2 - \frac{(\sum x)^2}{n}\right)\left(\sum y^2 - \frac{(\sum y)^2}{n}\right)}} \tag{4.4}$$

4.2 Genetic Algorithms (GA)

Survival of the fittest is the main concept of these algorithms. Genetic algorithms were first introduced by John Holand in 1960s, he along with his colleges and students developed these algorithms based on Darwin's theory of evolution and natural selection[88].

To understand the basics of genetic algorithms lets tell a small story about rabbits. We have a group of fast and clever rabbits and another slow and dump. The first generation of rabbits lived safe and happy, until a group of foxes attacked them. Unfortunately, many of the rabbits didn't make it, only the fast ones survived and a limited number of the slow ones also did by mere luck. The following generation which is a mix of the genes of fast and slow fathers continued to live and reproduce. With the continuous attacks from foxes and with time the fast rabbit's percentage increased and the slow ones decreased. And this is the story of natural selection.

Let's assume that the group of solutions is the group of rabbits. The fitness function which is how good the solution is, represents the rabbit's speed. The mating process

represents the mixing of the solution parts which we'll call crossover. The survival of rabbits by luck from one generation to the next we'll call selection. Sometimes a small gene change gives better solutions which we'll call mutation [72].

So, we have an initial population of rabbits (solutions) P(t), where P is the first group in time t which we will divide into groups based on the fitness function (speed) which is called evaluation. And recursively go through next generations until a certain condition is met which depends on the problem to be solved.

So, to put it in steps:

1. Select the best from P(t) to go to the next step.
2. Crossover: Mating between the groups to produce better solutions (generations).
3. Mutation: sometimes a certain solution (rabbit) goes through a mutation to produce better breeds.
4. Go through 1 to 3 again until a certain stopping criterion is met.
5. The output will be the best solution (better rabbits).

How do we employ the GA concept in sentiment analysis? The features in the beginning represent the qualities and genes which any member of the population can carry. That can be represented in binary 1 if the feature exists and 0 if it doesn't.

The initial population is a subset of features that are chosen randomly, let's give it the symbol Pi: subset i of features. Such that if we had 10 subsets of features randomly chosen that means if we had 40000 features then P1,P2,…,P10 will be our 10 feature subsets. P1 represents a feature vector that contains a group of features taken from the existing features that were produced from the preprocessing process which represents the genetic features that can describe the members of the population. In this vector if a

feature exists is represented by a binary 1 and if not 0. The features in the vector are randomly chosen such that P1 can have 20000 features, while P2 could have only 7000, and so on.

Each subset P is used for classification and the results (accuracy) are recorded. That means we will have 10 different results. The results are sorted in decreasing order. Then the strongest half of the subsets are taken, these represent the stronger members of the population. By strongest we mean the ones with the highest accuracy.

We make a crossover between the strongest vectors (highest accuracy), from each two we get a new one. From this step we get 10 new feature subsets or vectors that represent the members of the population.

Let's assume that the feature subsets p1, p3, p4, p7, and p9 came out with the highest accuracy with p1 being the highest, p3 second highest, and so on. We mate p1 with p3, p1 with p4, p1 with p7, p1 with p9, p3 with p4, to get five new vectors which will be combined with the original subset (p1, p3, p4, p7, and p9) to get 10 vectors. We do the classification step again and go through all the previous steps until a certain condition is met or a certain accuracy is reached.

4.3 Method 1: Machine Learning Based Feature Selection Method

In choosing the feature subset multiple experiments have been conducted using the well-known methods among researchers like weight by SVM, correlation, GINI-Index, and information Gain. Weight by SVM was selected as the best one as it achieved the best results. Figure 4.1 shows the work flow of method 1 and Algorithm 1 shows the pseudo code.

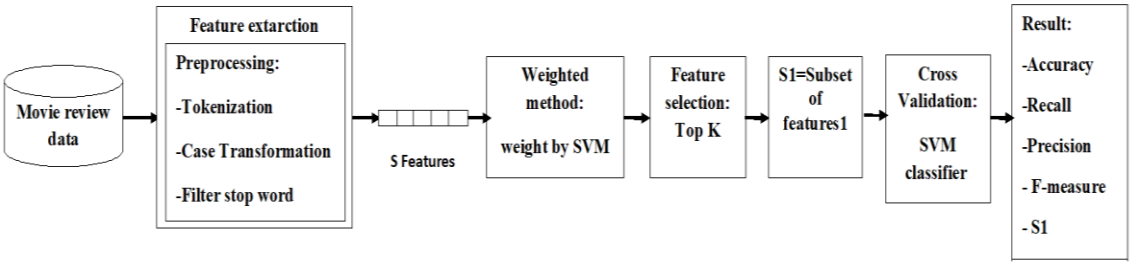

Figure 4.1 Method 1: Machine Learning Based Feature Selection Method

Algorithm 1, Pseudo Code: Method 1 Machine Learning Based Feature Selection Method

```
For Each Review in the Data Set

DO
    {

        Feature Extarction :  call preprocessing ();
        Add document vector to VSM;

    }

        Weight by method:  apply Weight by method (weight by SVM)
        Subset of features:  selects top k attributes.  When k=n, where n is constant.
        n: 500 - 4500      // Subset of features 1 is generated when k=4000
                           // top k: higher weight attributes for classification

For each sub-feature set

Do
    {

        Apply 10-fold cross validation classification.

    }
        Result: accuracy, recall, precision, F-measure, Subset of Featurs1
```

4.4 Method 2: Optimized Machine Learning Method

The highest accuracy feature subset that resulted from method 1 was taken and fed to optimization evolutionary algorithm to select a subset out of it with less number of

feature and yet give a higher accuracy than that of method 1. Figure 4.2 and Algorithm 2 show the workflow of method 2.

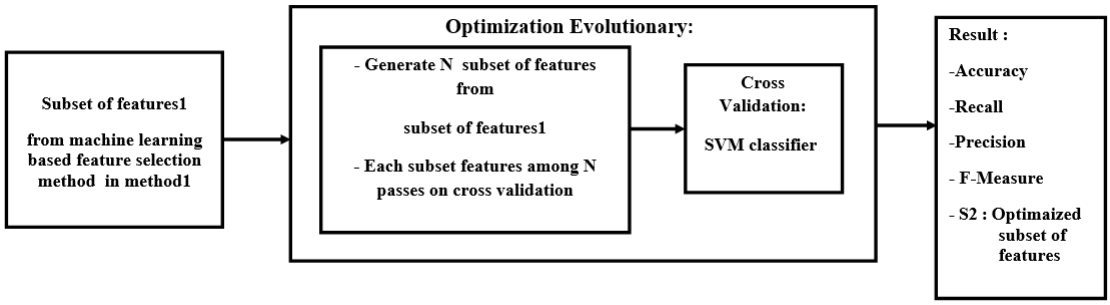

Figure 4.2 Method 2: GA on Subset feature 1

Algorithm 2, Pseudo Code: Method 2 Optimized Machine Learning Method

```
T = call (Subset of Features1)
        // Subset features1 is generate in algorithm 1 when k=4000 using weight by svm method
        // Weight by method: optimization evolutionary algorithms (Genetic Algorithm – GA)

Initialize population size = p
Initialize generations count = g
Generate p subsets from T

Do while not stopping criteria (number of iterations < g)

        {
          For each subset in P
                                Apply 10 fold cross validation classification
                                Record performance, feature subset
          End for each

Keep highest performance, feature subset

Do crossover to generate new P

        }

 Output: highest performance (accuracy), feature subset
 Subset of Features2 = subset features for classification to achieve more accuracy

 Result: accuracy, recall, precision, F-measure, Subset of Features 2
```

4.5 Method 3: Hybrid Statistical and Machine Learning Method

The weighting of features was done using both weight by SVM and weight by Correlation shown in Figure 4.3 and algorithm 3.1, after that the resultant feature subsets were merged using the union operator.

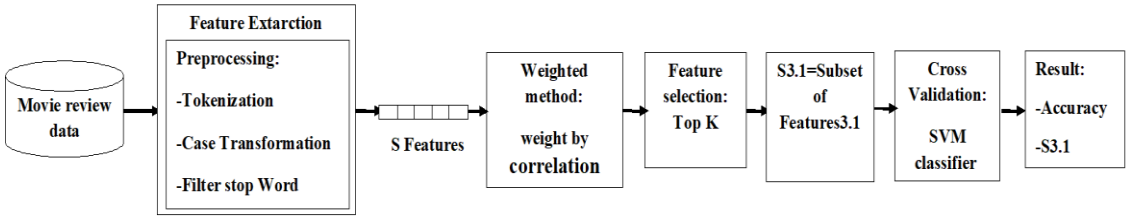

Figure 4.3 Weight by Correlation

Algorithm 3.1: Pseudo Code for Proposed First Stage Using Weight by Correlation

```
For Each Review in the Data Set
        {

                Feature Extarction :  call preprocessing ();
                Add document vector to VSM;
        }

Weight by method:  Apply weight by method (weight by correlation)

 Subset of features:  Selects top k attributes.  Where k=n, n is a constant whose value is
between 500 and 4500 with step size 500.

                        // Subset of features3.1 is generated when k=4000
                        //Top k: select higher weight attributes for classification

For each sub-feature set

        {
                        Apply 10 fold cross validation classification.
        }


Result: accuracy, subset of features 3.1
```

The union was made between top K features of each weighting method (weight by SVM and Weight by Correlation) at K =4000 from each one. Where the top 4000 features of weight by SVM are called feature subset 1 and top 4000 features of weight by correlation are called feature subset 3.1 as shown in Figure 4.4. The pseudo code of the hybrid method is shown in algorithm 3.2.

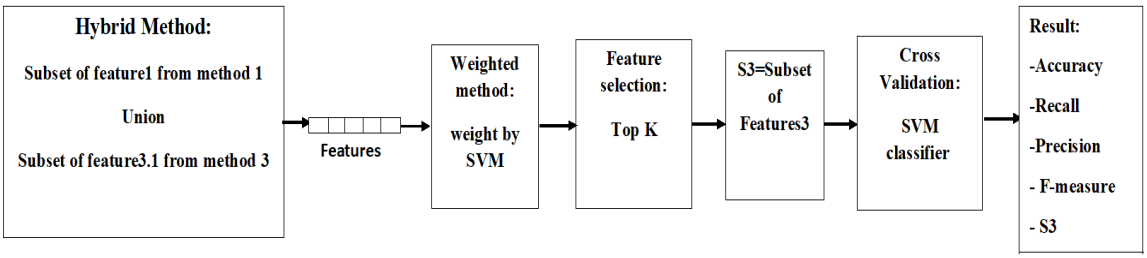

Figure 4.4 Hybrid Method

Algorithm 3.2, Pseudo Code: Method 3 Hybrid Statistical and Machine Learning Method

Call (Subset of Features1 and Subset of Features3.1)

Hybrid method : Union for subset of features1 and subset of features 3.1

Weight by method : Apply weight by method (weight by svm) based feature selection

Subset of features : Selects top k attributes. When k=n (n=500-2500), where n is constant.

// Subset of features 3 is generated when k=1000

For each sub-feature set

{

Apply 10-fold cross validation classification.

}

Result: accuracy, recall, precision, F-measure, Subset of Features 3

4.6 Method 4: Optimized Hybrid Method

The highest accuracy feature subset that resulted from method 3 was fed into the optimization evolutionary algorithm to derive a feature subset 4 with smaller size or higher accuracy or both than that of method3. Method4 is detailed in Figure 4.5 and algorithm 4.

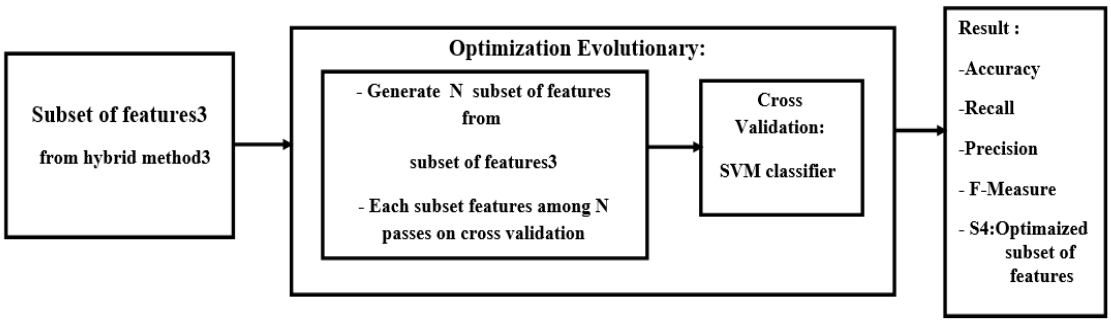

Figure 4.5 Method 4: Optimized Hybrid Method

Algorithm 4, Pseudo Code: Method 4 Optimized Hybrid Method

```
T = call (Subset of Features 3)
Initialize population size = p
Initialize generations count = g
Generate p subsets from T
Do while not stopping criteria
        {
            For each subset in P
                {
                    Apply 10-fold cross validation classification
                    Record performance, feature subset
                }
            End for each
Keep highest performance, feature subset

Do crossover to generate new P
        }
Output: highest performance (accuracy), feature subset

Subset of Features 4 =  Subset of Features for classification to achieve more accuracy

Result: accuracy, recall, precision, F-measure, subset of features 4
```

4.7 Summary

In our approach we used machine learning for weighting features so that the weight could be tuned during the learning process to achieve the desired results. The weighting method used is called weight by SVM. We also merged between two weighting methods, weight by SVM and weight by correlation, where features resulting from both methods were merged in order to reduce feature subset size used in classification while giving better results.

In order to evaluate classification accuracy, stratified sampling 10 fold cross validation was chosen.

For feature selection there are optimization methods based on nature inspired algorithms like, PSO, ANT Colony, etc. In our approach we used optimize evolutionary which is based on G.A. the results using this G.A method were better in regards of accuracy and feature subset size than machine learning methods.

CHAPTER 5

RESULTS AND EVALUATION

5. Results and Evaluation

5.1 Introduction

A sentiment analysis approach would be successful if it achieves a high classification accuracy using the least number of possible features. That is why selecting the feature subset that gives the required results is very important. Our approach increased the accuracy by using machine learning to weight the features and update the weight vector during the learning process. It also employed optimization to decrease the feature subset size used by the classifier. The improvements in classification in sentiment analysis is an ongoing task of great importance for researchers, because based on this classification process decision makers would be able to take the right decisions in the field of interest.

Experiments were conducted to choose the effective features in multiple methods. Some using machine learning methods, others employed optimization evolutionary algorithm, and some were hybrid. All of this will be discussed in detail in the following sections.

5.2 Experimental Setup

5.2.1 Tools and Hardware Specifications

The hardware used during the experiments is a laptop with Intel core i7 CPU 2.4 GHz, 8 GB of RAM, 256 GB hard drive, and 64 bit Microsoft windows 10 enterprise edition.

For applying the experiments RapidMiner Studio v7.4 was chosen. RapidMiner Studio is an environment for building predictive analytic workflows. This application uses a visual programing environment with more than 1500 machine learning and data preprocessing functions. It integrates R and python scripts and supports many correct model validation methods and supports access to all types of data.

Manek, Asha S., et al. in [89] obtained a remarkably high accuracy in sentiment analysis. They used RapidMiner in their work and this was the motive to check out this tool. RapidMiner included all the tools that I needed to complete my experiments, especially weight by SVM. In the literature that I reviewed, no one used weight by SVM in past studies in the domain of SA; most of them used statistical methods, that's why I used this weighting method in my research and the results were better.

5.2.2 Datasets

The Movie review datasets used in this thesis are listed as follows:

- D1: polarity dataset v2.0 (3.0Mb)[2] : 1000 positive and 1000 negative processed reviews. Introduced in Pang/Lee ACL 2004. Released June 2004.
- D2: polarity dataset v1.0 (2.8Mb)[3]: 700 positive and 700 negative processed reviews. Released July 2002
- D3: A combination of D1 and D2

Some researchers combined D1 and D2 in their researches [89], so I also combined the two datasets in order to compare my results with their results as shown in the comparison section. Also by merging the two datasets D1 and D2 a new dataset is resulted which give the chance to further test the approach and prove its validity and correctness by showing the high results it gave. Table 5.1 shows the details of the datasets.

[2] http://www.cs.cornell.edu/people/pabo/movie-review-data/review_polarity.tar.gz

[3] http://www.cs.cornell.edu/people/pabo/movie-review-data/mix20_rand700_tokens_cleaned.zip (links copied in 1/10/2016)

Table 5.1 Datasets Details

Dataset	Number of Documents (Reviews)		
	Positive	Negative	Total
Polarity Dataset V2.0 D1	1000	1000	2000
Polarity Dataset V1.0 D2	700	700	1400
Combination of D1 And D2 D3 = D1+D2	1700	1700	3400

Feature types that could be used in sentiment analysis include N-grams: unigrams, bigrams, POS, phrases. There are many weighing methods to weigh features in order to choose the feature subset for classification such as: weight by SVM, Weight by Correlation, Weight by Gini index, Weight by information gain, etc. In the classification of documents into positive and negative the results differed for a number of reasons:

1. The Type of feature used (POS, Unigram, and Bigrams).
2. The weighting method.
3. The used classifier.

Two types of features were extracted for experiments: Unigrams and POS (Part-of-Speech), for POS nouns were extracted. Table 5.2 shows the number of features that was obtained for each type of feature from each dataset.

Table 5.2 Number of Features Obtained for Each Type from Each Dataset.

Dataset	Type of feature	Number of Features
D1	Unigram	38557
	POS	26331
D2	Unigram	34675
	POS	23885
D3	Unigram	43491
	POS	30184

The features that result from preprocessing were weighed for classification at different values of top k, where top k means the k features that have the highest weights. The classification accuracy for features of type unigram were better than those of type POS no matter what the weighing method was, as shown in Table 5.3, 5.4, and 5.5. Also the preprocessing time was notably shorter when extracting unigrams than that of POS features, for example in processing D1 to extract unigrams the processing time was nearly 15 seconds while in extracting POS (nouns) the time was nearly 44 minutes.

Table 5.3 Accuracy for different weighing methods for unigrams and POS for D1.

Weight by Method	**Type of Feature**	**K 500**	**K 1000**	**K 1500**	**K 2000**	**K 2500**	**K 3000**	**K 3500**	**K 4000**	**K 4500**
Weight By SVM	Unigram	91.35	94.25	95.7	97.1	97.7	98.45	98.8	98.7	98.95
Weight By SVM	POS	89.1	92.75	95.25	95.55	96.3	96.95	97	97.2	97
Weight by Correlation	Unigram	89.95	92.85	93.4	94.1	95.05	95.5	96	96.7	96.8
Weight by Correlation	POS	86.95	90.35	92.65	93.25	93.7	94.95	95.3	95.8	96.1
Weight By IG	Unigram	87.25	91.95	92.6	92.6	93.7	93.8	94.45	94.7	95.2
Weight By IG	POS	85	87.8	89.4	89.9	91.2	91	91.35	92	93.1
Weight by Gini Index	Unigram	88.15	91.1	91.5	91.9	92.45	93.4	93.1	93.7	94.3
Gini Index	POS	83.55	87.3	87.75	89.3	89.85	91.15	90.6	90.65	91.85

Table 5.4 Accuracy for different weighing methods for unigrams and POS for D2.

Weight by Method	Type of Feature	K 500	K 1000	K 1500	K 2000	K 2500	K 3000	K 3500	K 4000	K 4500
Weight By SVM	Unigram	92.57	95.29	96.29	97.43	98.36	98.36	99	98.79	98.93
	POS	88.86	93.64	95.43	96.79	97.36	97.71	97.71	98.21	98.57
Weight by Correlation	Unigram	91.29	94.21	95.21	96.43	96.71	97.29	98.29	97.71	98.21
	POS	89.29	92.71	94.21	95.56	95.79	95.86	96.5	97.07	97
Weight By IG	Unigram	87	89.43	91.29	91.29	93.14	93.43	94.36	94.86	94.93
	POS	85.86	88.64	91.14	91.57	92.14	92.5	93.86	92.93	93.21
Weight by Gini Index	Unigram	87.43	90.29	90.29	92.36	91.36	93.29	94.21	94.5	94.79
	POS	84.5	88.14	89.57	90.71	90.07	92.14	92.64	93.64	93.21

Table 5.5 Accuracy for different weighing methods for unigrams and POS for D3.

Weight by Method	Type of Feature	K 500	K 1000	K 1500	K 2000	K 2500	K 3000	K 3500	K 4000	K 4500
Weight By SVM	Unigram	92.56	94.79	95.79	96.53	96.97	97.18	97.59	97.85	98.09
	POS	88.24	92.79	94.15	95.18	95.97	96.41	96.68	96.76	97.24
Weight by Correlation	Unigram	90.82	92.44	93.97	93.91	94.97	95.21	95.26	95.82	95.94
	POS	87.24	90.47	91.76	92.79	93.94	94.24	94.68	94.74	94.97
Weight By IG	Unigram	88.29	91.38	91.85	92.79	93.35	94.47	94	94.44	94.56
	POS	84.65	87.79	89.26	90.71	91.5	91.76	93.15	93.12	94
Weight by Gini Index	Unigram	88.44	90.71	92.09	92.71	92.68	93.47	93.85	93.97	94.44
	POS	84.18	88.15	89.47	89.79	91	91.44	92.26	92.97	93.44

For example, in D1 at k=500, 2500, and 4500 it is obvious in Figure 5.1 that unigrams are better than POS in every weighing method. Also Figure 5.2 and Figure 5.3 show the same for datasets D2 and D3 respectively.

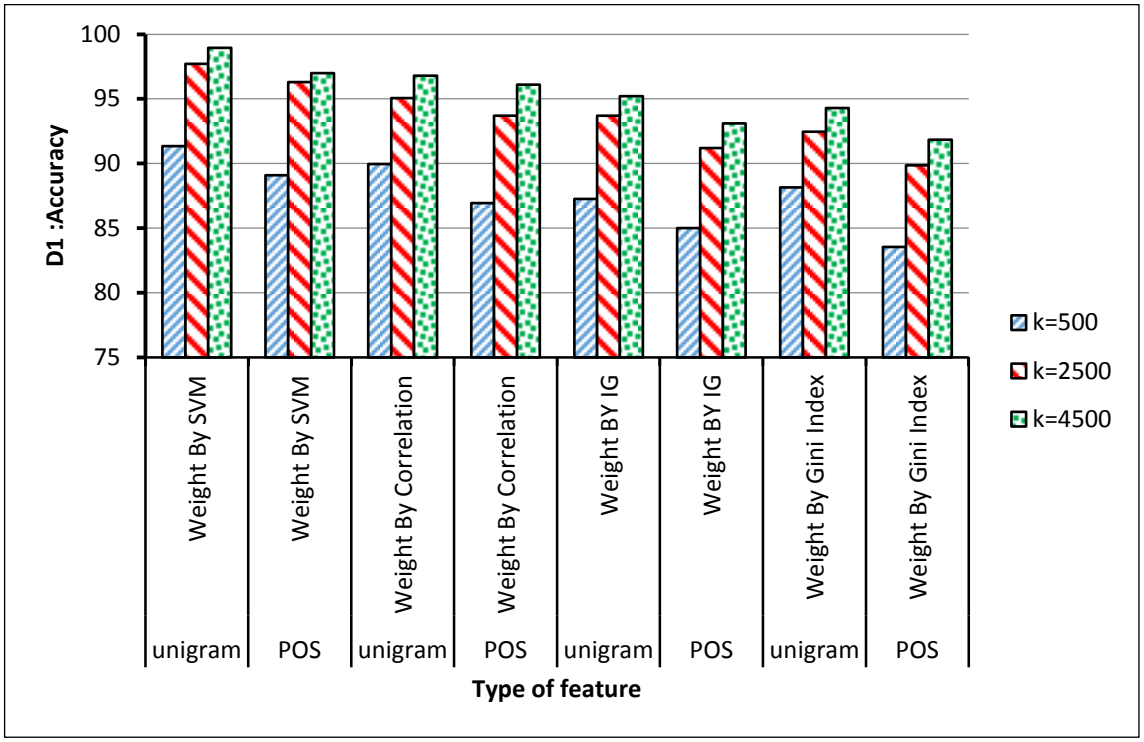

Figure 5.1 Accuracy of Unigrams and POS for D1

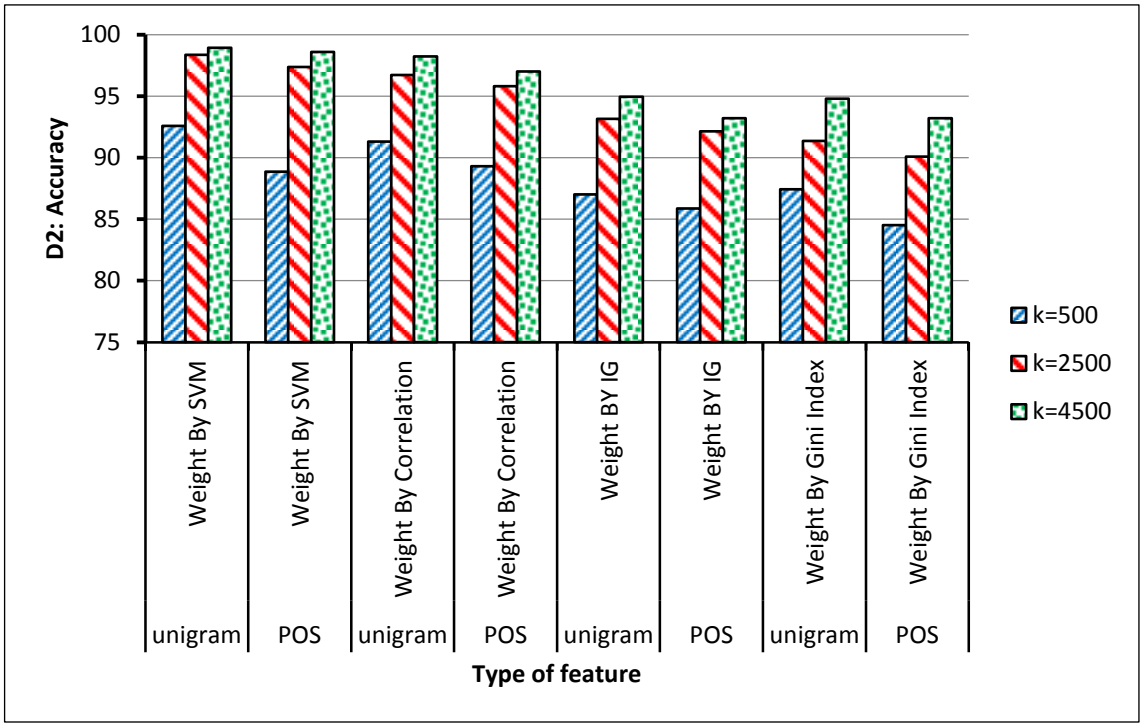

Figure 5.2 Accuracy of Unigrams and POS for D2

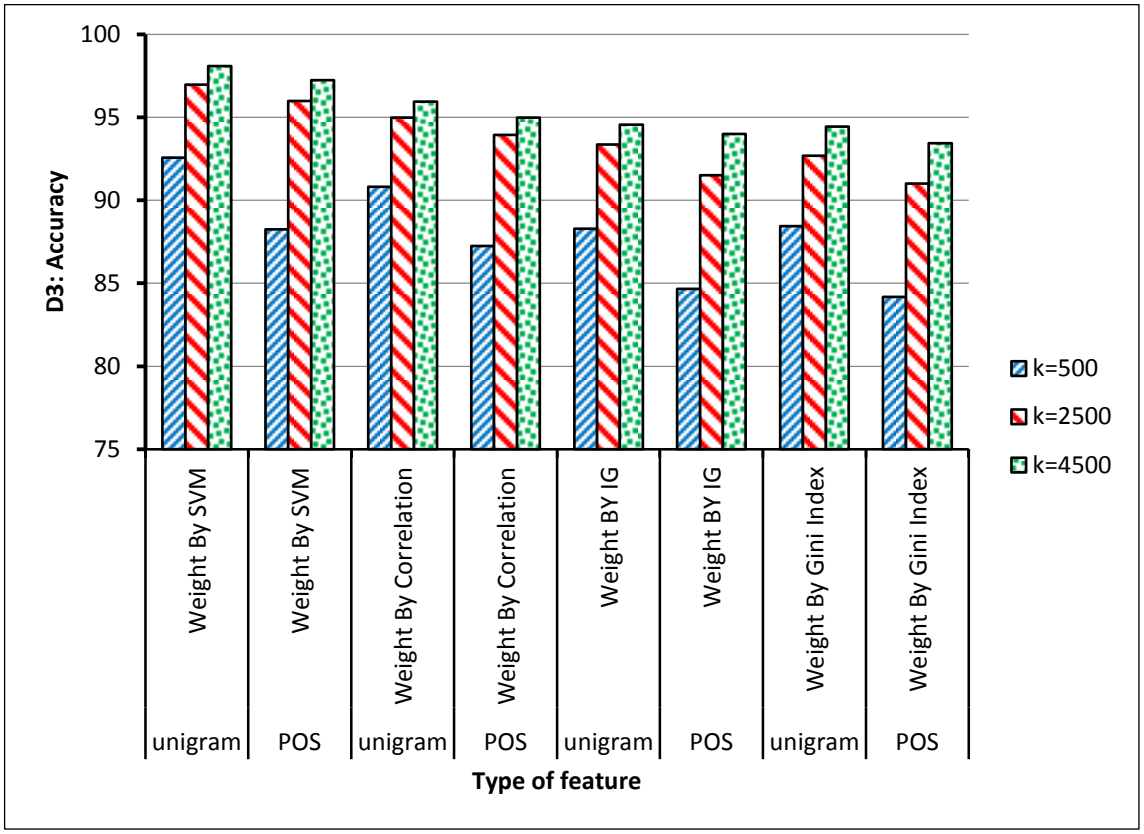

Figure 5.3 Accuracy of Unigrams and POS for D3

Also, the accuracy for the unigrams weighed by SVM was higher and better than the accuracy obtained in classifying the features that were weighed in any other weighing method like weight by correlation, weight by GINI Index, etc. Table 5.6 shows accuracy for different weighing methods for different K values for features of type Unigram.

Table 5.6 Accuracy for different weighing methods at different values of K for Unigrams

Dataset	Weight by Method	K 500	K 1000	K 1500	K 2000	K 2500	K 3000	K 3500	K 4000	K 4500	Note
D1	Weight By SVM	91.35	94.25	95.7	97.1	97.7	98.45	98.8	98.7	98.95	Highest Accuracy
	Weight by Correlation	89.95	92.85	93.4	94.1	95.05	95.5	96	96.7	96.8	Second Highest Accuracy
	Weight By IG	87.25	91.95	92.6	92.6	93.7	93.8	94.45	94.7	95.2	
	Weight by Gini Index	88.15	91.1	91.5	91.9	92.45	93.4	93.1	93.7	94.3	
D2	Weight By SVM	92.57	95.29	96.29	97.43	98.36	98.36	99	98.79	98.93	Highest Accuracy
	Weight by Correlation	91.29	94.21	95.21	96.43	96.71	97.29	98.29	97.71	98.21	Second Highest Accuracy
	Weight By IG	87	89.43	91.29	91.29	93.14	93.43	94.36	94.86	94.93	
	Weight by Gini Index	87.43	90.29	90.29	92.36	91.36	93.29	94.21	94.5	94.79	
D3	Weight By SVM	92.56	94.79	95.79	96.53	96.97	97.18	97.59	97.85	98.09	Highest Accuracy
	Weight by Correlation	90.82	92.44	93.97	93.91	94.97	95.21	95.26	95.82	95.94	Second Highest Accuracy
	Weight By IG	88.29	91.38	91.85	92.79	93.35	94.47	94	94.44	94.56	
	Weight by Gini Index	88.44	90.71	92.09	92.71	92.68	93.47	93.85	93.97	94.44	

The classification for the features weighed by correlation came in second place in accuracy when compared to other weighing methods as shown in Table 5.6. That is why our approach combines both weight by SVM and Correlation.

It is shown in Figures 5.4, 5.5, and 5.6 that weight by SVM has the highest accuracy followed by weight by Correlation.

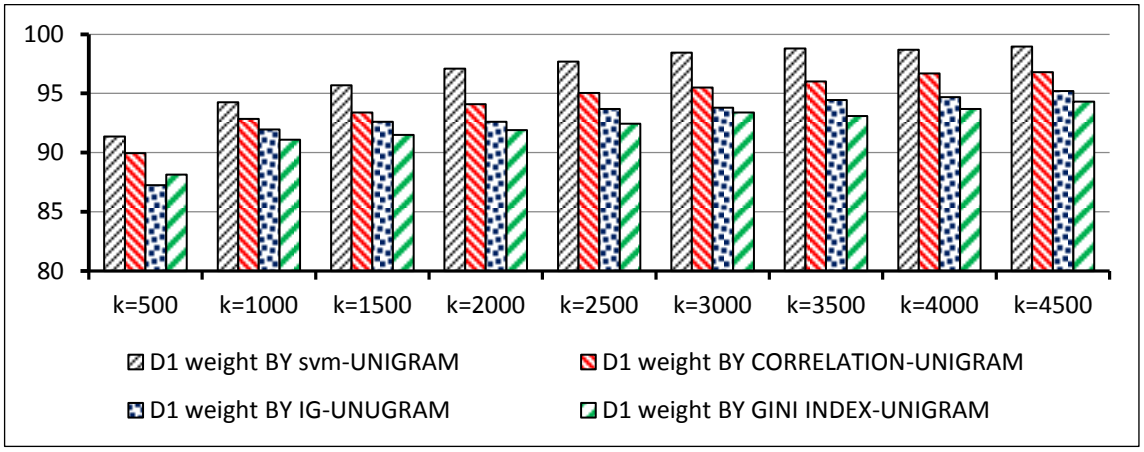

Figure 5.4 Accuracy of different weighing methods for D1 at different K values.

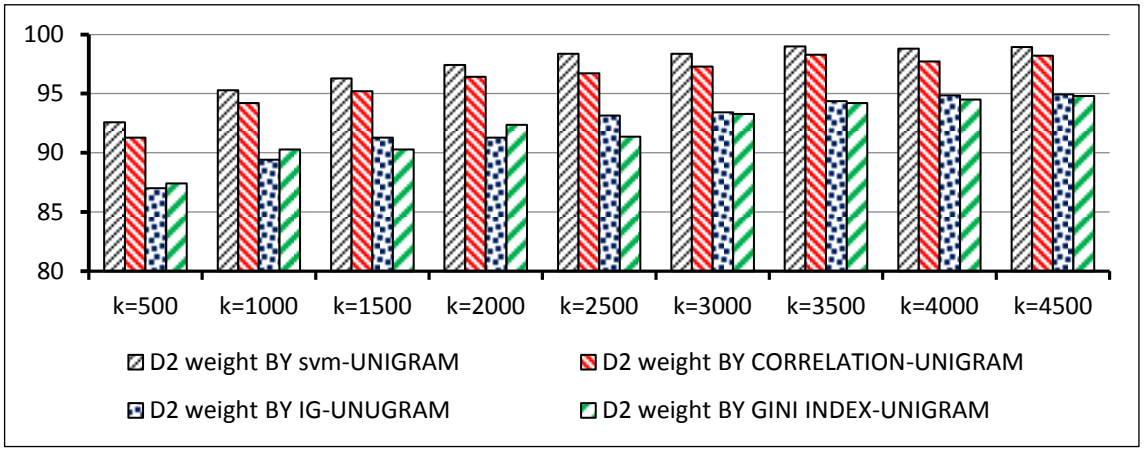

Figure 5.5 Accuracy of different weighing methods for D2 at different K values.

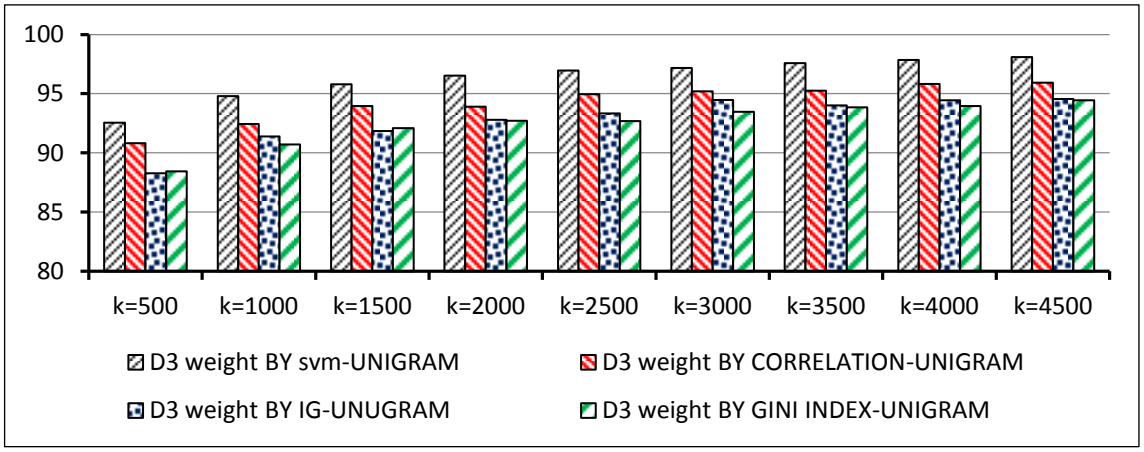

Figure 5.6 Accuracy of different weighing methods for D3 at different K values.

5.3 Experiments and Results

The following experiments were applied:

1. Classification Using Machine Learning Based Feature Selection Method.
2. Classification Using Optimized Machine Learning Method.
3. Classification Using Hybrid Statistical and Machine Learning Method.
4. Classification Using Optimized Hybrid Method.

In all experiments the features that were extracted from preprocessing of each dataset were weighed using weight by SVM, then a subset of top K features that have the highest weights were selected. For the training and testing process of the dataset cross validation of 10-fold stratified sampling method was used due to its popularity as shown in section 2.5.2.1 and the resulting output of features were passed to the classifier to be used in classifying the reviews into positive or negative using SVM classifier. The results that were obtained had very high accuracy with minimal feature subset size.

5.3.1 Classification Using Machine Learning Based Feature Selection Method

Doing classification using known methods like information gain, GINI-Index, etc.., is called statistical methods. In this experiment different K values were selected ranging from 500 to 4500 in steps of size 500. The results of classification using different K values show that as the number of input features increases the accuracy of the classifier increases as shown in Table 5.6.

The accuracy at K=4000 was high and better than what others have obtained in classifying the same datasets as shown in Table 5.6. For D3 using 4000 feature the

accuracy obtained was 97.85. Table 5.7 shows the accuracy, precision and recall at k=4000 for all datasets.

Table 5.7 Accuracy, Precision And Recall For All Datasets at K=4000 for Method 1

Dataset	Weight by Method	Results			
		Accuracy	Precision	Recall	F-Measure
D1	Weight By SVM	98.7%	98.8%	98.6%	98.69%
D2	Weight By SVM	98.79%	98.58%	99%	98.78%
D3	Weight By SVM	97.85%	97.27%	98.47%	97.86%

5.3.2 Classification Using Optimized Machine Learning Method

One of the main data mining tasks is selecting the most relevant features for classification problems. A genetic algorithm (GA) computes the weights of features and uses them in a way that mimics natural evolution in order to reduce classification error [90]. GA Mutates features by switching them on and off and crosses them over by interchanging used features.

When GA was applied to the Machine Learning method the accuracy got higher with less number of features used in classification as shown in Table 5.8.

Table 5.8 Accuracy of Method 1: Machine Learning Based Feature Selection Method and Method 2: Optimized Machine Learning Method.

Dataset	Method		K 500	K 1000	K 1500	K 2000	K 2500	K 3000	K 3500	K 4000	K 4500
D1	Machine Learning Based Feature Selection Method	accuracy	91.35	94.25	95.7	97.1	97.7	98.45	98.8	98.7	98.95
		Features	500	1000	1500	2000	2500	3000	3500	4000	4500
	Optimized Machine Learning Method	accuracy	91.95	94.9	96.1	97.35	97.9	98.65	98.85	99.15	99.15
		Features	473	980	1458	1950	2458	2950	3450	3950	4450
D2	Machine Learning Based Feature Selection Method	accuracy	92.57	95.29	96.29	97.43	98.36	98.36	99	98.79	98.93
		Features	500	1000	1500	2000	2500	3000	3500	4000	4500
	Optimized Machine Learning Method	accuracy	92.86	95.36	96.79	97.79	98.64	98.64	99.36	99.21	99.36
		Features	471	954	1451	1950	2462	2957	3458	3950	4450
D3	Machine Learning Based Feature Selection Method	accuracy	92.56	94.79	95.79	96.53	96.97	97.18	97.59	97.85	98.09
		Features	500	1000	1500	2000	2500	3000	3500	4000	4500
	Optimized Machine Learning Method	accuracy	92.76	94.88	96.21	96.88	97.15	97.53	97.65	98.12	98.15
		Features	472	570	1450	1957	2450	2950	3451	3950	4450

Table 5.9 shows the accuracy, recall, and precision at k=4000. Figure 5.7 shows the accuracy of method 1 and method 2.

Table 5.9 Accuracy, Recall, and Precision for Optimization Methods at K = 4000

Dataset	Results			
	Accuracy	Precision	Recall	F-Measure
D1	99.15%	99.1%	99.2%	99.14%
D2	99.21%	99.14%	99.29%	99.21%
D3	98.12%	97.67%	98.59%	98.12%

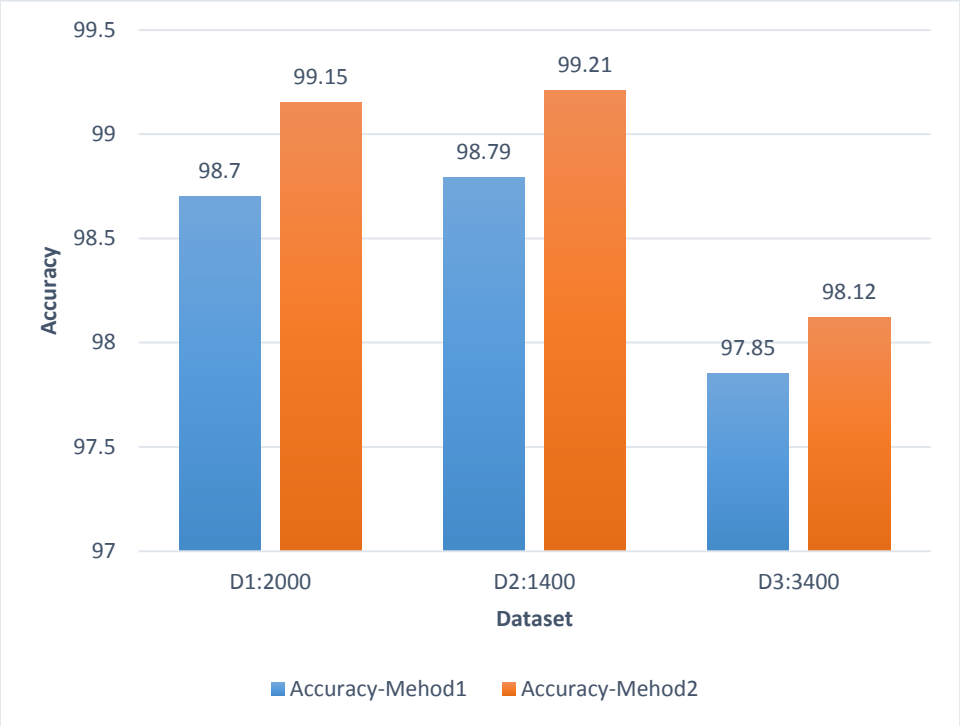

Figure 5.7 Accuracy of Method 1 and Method 2.

5.3.3 Classification Using Hybrid Statistical and Machine Learning Method

In order to increase accuracy using the least number of features the hybrid method was developed. In this method two feature weighing methods were used: weight by SVM and weight by Correlation. The top 4000 features of weight by SVM and top 4000 features of weight by correlation that are shown in Table 5.6 were taken and a union was made between them to generate the features that will be used to obtain the feature subset to be used in classification as shown in Table 5.10.

Table 5.10 Union of Features: 1: Top 4000 Features from Weight by SVM, 2: Top 4000 Features from Weight by Correlation

Dataset	Union	Number of Features After Union
D1	1 U 2	5238
D2	1 U 2	5020
D3	1 U 2	5651

The features resulting from the hybrid method were used in classification at different top k values. Table 5.11 shows that the accuracy obtained by the hybrid method is higher using a smaller number of features in the classification process.

Table 5.11 Accuracy for all datasets using the Hybrid Method at different K values.

Dataset	K 500	K 1000	K 1500	K 2000
D1	95.7	99.45	99.75	99.7
D2	95.64	99.46	99.86	99.82
D3	95.5	98.66	99.03	99.12

At k=1000 the accuracy obtained in this method for all datasets was higher than that of the previous method. Table 5.12 shows the accuracy, recall and precision of the hybrid method.

Table 5.12 Accuracy, recall and precision of the hybrid method at top k =1000.

Dataset	Accuracy	Precision	Recall	F-Measure
D1	99.45%	99.3%	99.6%	99.44%
D2	99.46%	99.22%	99.71%	99.46%
D3	98.66%	98.88%	98.44%	98.65%

The original number of features was truncated after the union such that then number of features that the feature subset will be selected from became very small when compared to the resulting features from preprocessing (Unigrams) as shown in Table 5.13 for each dataset. That means that we were able to pinpoint the relative features that will give high accuracy in classification.

Table 5.13 Number of Unigram Features before and after union for each dataset.

Dataset	Original Features	Number of features After union
D1	38557	5238
D2	34675	5020
D3	43491	5651

5.3.4 Classification Using Optimized Hybrid Method

In this method reviews were classified using optimization evolutionary algorithm for the features resulting from the hybrid method at k=1000 to obtain a less number of features than that of the hybrid method while obtaining higher accuracy as shown in Table 5.14.

Table 5.14 Accuracy and number of features for hybrid method and Optimized Hybrid method at K = 1000.

Dataset	Method	Result	
D1	Hybrid Method	Accuracy	99.45
		Features	1000
	Optimized Hybrid Method	Accuracy	99.53
		Features	970
D2	Hybrid Method	Accuracy	99.46
		Features	1000
	Optimized Hybrid Method	Accuracy	99.71
		Features	950
D3	Hybrid Method	Accuracy	98.66
		Features	1000
	Optimized Hybrid Method	Accuracy	98.69
		Features	958

For each dataset the accuracy was better when using optimization evolutionary algorithm on the hybrid method than that using the hybrid without optimization as shown in Figure 5.8.

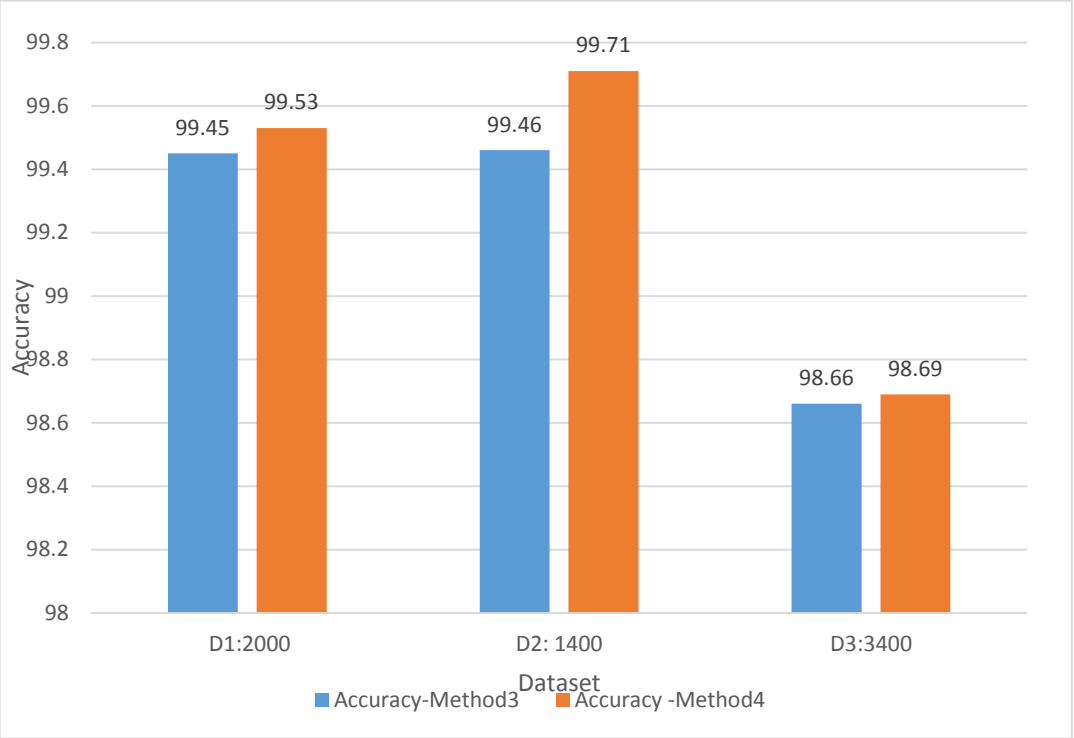

Figure 5.8 Accuracy of Hybrid Method alone and Optimized Hybrid Method.

The accuracy, precision, recall, and F-measure of this approach are shown in Table 5.15.

Table 5.15 Accuracy, Precision, Recall, and F-Measure of Optimized Hybrid Method.

Dataset	**Results**			
	Accuracy	**Precision**	**Recall**	**F-Measure**
D1	99.53%	99.4%	99.65%	99.52%
D2	99.71%	99.43%	100%	99.8%
D3	98.69%	99.08%	98.29%	98.68%

5.4 Discussion

To achieve high results in classification of opinions into positive and negative, one must take into consideration the following:

1. The ability to choose the relevant features to classification.
2. The classifier strength in using a strong and successful learning process using the important and relevant features.

There are many methods in choosing the features, such as IG, Correlation, MI, and GINI index. These are traditional methods (statistical methods) that depend on a law that defines a constant value called weight for every feature. There are ways to choose the qualities that specify the weight of a feature that could be tuned to accommodate the learning process in order to achieve high results like weight by SVM.

Weight by SVM depends in feature weighting on machine learning. And the experiments we did have shown that when weighting the features by weight by SVM achieved high and even better results than the results obtained when weighting with statistical methods as shown in table 5.6. The results also show that using features of type unigram gives higher accuracy than features of type POS as shown in Figures 5.1, 5.2, and 5.3 for datasets D1, D2, and D3 respectively.

The results were very high and came in first place regarding accuracy when the features were weighted using weight by SVM, the second highest accuracy was obtained using weight by correlation as shown in table 5.6 and Figures 5.4, 5.5, and 5.6.

A good and effective feature subset is the one that has the least number of features while achieving high accuracies in classification. In order to reduce the size of the feature

subset while sustaining high accuracy we used optimization methods as in method 2 of the experiments. Optimization Evolutionary methods based on Genetic algorithms were using in feature selection.

By using G.A. we obtained an effective feature subset with minimum number of features and higher accuracy in classification, the results using G.A. were better than the ones obtained in method 1 regarding the size of feature subset used and accuracy achieved. All of that is shown in table 5.8.

Insisting on achieving higher accuracy using relevant features of a small size we turned to Hybrid Methods as in method 3. A union was made between the features weighted by SVM and the features weighted by correlation at top k = 4000 of each weighting method. The resulting feature subset sizes for each dataset are shown in table 5.13.

The features were weighted using weight by SVM in preparation to use them in classification. After classification using 10-fold cross validation by the SVM classifier the accuracy was very high at k=1000.

The results gained in method 3 were better than of the previous methods regarding the feature subset size and also the accuracy level as shown in tables 5.9 and 5.12.

In method 4 optimization methods were used on the features resulting from method 3. Using optimization methods achieved higher accuracy in classification of opinions than method 3 with even less number of features as shown in table 5.14.

5.5 Comparison

When comparing the work that we have done and its results with others we should go through research papers that used the same datasets.

Here we focus on two factors, accuracy and number of features used in the classification of reviews. The papers that we discuss are as recent as possible published in 2016 and 2017.

We compare each dataset separately with others that have used it using their results in their own experiments without conducting any experiments of our own using their approaches. Sections 5.5.1, 5.5.2, and 5.5.3 compare our approach when applied to each dataset with what others have achieved using these datasets.

5.5.1 Comparison on D1 Polarity Dataset v2.0

Table 5.16 Comparison on Dataset 1= polarity dataset v2.0=2000

Reference	Year	Approach	Features	Accuracy
[91]	2016	Composite unigrams and bi-tagged, Bi-gram, Dependency features with mRMR feature selection (Minimum redundancy maximum relevance) Information gain, Naive Bayes, SVM	1130+1114	91.8
[92]	2017	Unigram, CHI, IG POS, SVM ,NB	2311	91.33
			16669	94.13
[93]	2016	Naive Bayes classifier POS Feature lexicon-based	not available	83.7
[62]	2013	Proposed Hybrid NB-GA Method TF-IDF Feature Selection: Best First Search	not available	93.8
[48]	2014	Unigrams Random Forest Classifier	1942	91
[53]	2015	*TF-IDF* Bigrams, Bi-tagged Unigrams SVM Dependency parsing-based semantic parser with common-sense knowledge; mRMR	not available	90.1
Our		Method 1: Unigram, Weight by SVM, SVM Classifier	1000	94.25
			4000	98.7
		Method 2: Unigram, Weight by SVM Optimize Selection (Evolutionary): Genetic Algorithm, SVM Classifier	980	94.9
			3950	99.15
		Method 3: Unigram, Weight by SVM & Correlation, SVM Classifier	1000	99.45
		Method 4: Unigram, Weight by SVM & Correlation Optimize Selection (Evolutionary): Genetic Algorithm, SVM Classifier	970	99.53

Table 5.16 compares our approach when applied to dataset D1 with others who used it. Basant Agarwal and Namita Mittal in [91] used 2244 features in the classification process and obtained an accuracy of 91.8, while Alireza Yousefpour et al. achieved 91.33 in [92] using 2311 features through SVM classifier, and achieved 94.13 through naïve bias classifier using 16669 features.

Also, Govindarajan in his paper titled " Sentiment analysis of movie reviews using hybrid method of Naive Bayes and genetic algorithm" [62] reached an accuracy of 93.8 using Hybrid NB-GA Method without specifying the number of features used.

Agarwal, Basant et al in [53] scored an accuracy of 90.1 without mentioning the number of features used in the classification process. Others listed in Table 5.16 used the same dataset but did not achieve high accuracy.

In our approach using the same dataset we achieved high accuracy that reached 94.25 using 1000 features, and an accuracy of 98.7 using 4000 features using Traditional methods. And when using optimization methods, the accuracy was higher and better as shown in table 5.16. The accuracy was boosted from 98.7 to 99.15 and number of features lowered from 4000 to 3950 using optimization. And when using the Hybrid Method, the accuracy was 99.45 using only 1000 features without optimization, after implementing optimization in the hybrid method the accuracy increased to 99.53 using 970 features.

The results that we obtained for D1 was better in accuracy and number of features than others.

5.5.2 Comparison on D2 Polarity Dataset v1.0

Table 5.17 Comparison on Dataset 2= polarity dataset v1.0=1400

Reference	Year	Approach	Features	Accuracy
[93]	2016	Naive Bayes classifier POS feature lexicon-based	not available	82.4
[48]	2014	Unigrams Random Forest Classifier	2275	87.85
Our		Method 1: Unigram, Weight by SVM , SVM Classifier	1000	95.29
			4000	98.79
		Method 2: Unigram, Weight by SVM Optimize Selection (Evolutionary): Genetic Algorithm, SVM Classifier	954	95.36
			3950	99.21
		Method 3: Unigram, Weight by SVM & Correlation, SVM Classifier	1000	99.46
		Method 4: Unigram, Weight by SVM & Correlation Optimize Selection (Evolutionary): Genetic Algorithm, SVM Classifier	950	99.71

Table 5.17 compares our approach when applied to dataset D2 with others who used it. Madhavi, Piryani, and Singh in [93] achieved an accuracy of 82.4 without mentioning the number of features used in the classification process. Parmer et al in [48] achieved an accuracy of 87.85 using 2275 features. All of which is shown in Table 5.17.

In our approach the accuracy was 95.29 using 1000 features, and was increased to 98.79 using 4000 features using traditional methods on the same dataset. And when using optimization, it was even better than that of the traditional alone, the accuracy was boosted from 95.29 to 95.36 using only 954 features and from 98.79 to 99.21 using 3950 features instead of 4000 on the same dataset as shown in table 5.17.

In the Hybrid method we achieved better and higher results with an accuracy of 99.46 using 1000 features, and when applied with optimization it was even higher with accuracy of 99.71 using 950 features as shown in table 5.17.

5.5.3 Comparison on D3

Table 5.18 Comparison on Dataset 3= Dataset 1+ Dataset 2=D1+D2=3400

Reference	Year	Approach	Features	Accuracy
[89]	2017	SVM(Linear), Weight by Correlation, Weight by Gini Index, Word	20	97.25
Our		Method 1: Unigram, Weight by SVM , SVM Classifier	4000	97.85
		Method 2: Unigram, Weight by SVM Optimize Selection (Evolutionary): Genetic Algorithm, SVM Classifier	3950	98.12
		Method 3: Unigram, Weight by SVM & Correlation, SVM Classifier	1000	98.66
		Method 4: Unigram, Weight by SVM & Correlation Optimize Selection (Evolutionary): Genetic Algorithm, SVM Classifier	958	98.69

Table 5.18 compares our approach when applied to dataset D3 with others who used it. Manek, Asha S., et al in [89] achieved an accuracy of 97.25 using 20 features only as stated in their research titled “Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier." i.e. he used 20 features in the classification process to achieve an accuracy of 97.25. I wrote the author (Asha) to ask about how a high accuracy of 97.25 could be achieved using such a low number of features. He replied “it all depends on the preprocessing stage”. I wrote him another email asking for the details of the preprocessing steps but no reply was received.

Our approach achieved 97.85 using 4000 features in traditional method, and we were able to improve the results with optimization to get an accuracy of 98.12 using 3950

features. And in the Hybrid method accuracy was increased to 98.66 using 1000 features, and with optimization the accuracy was increased further to 98.69 and the number of features decreased to 958.

5.6 Limitations

There are two main limitations that were obvious in the research. The first, fixed none time related data was used in the analysis. Using dynamic data that changes over time due to multiple reasons such as user mood, sequence of events, event outcome, and so on could lead to more effective results and shows the change of public opinions towards the entity being analyzed which could help take better decisions.

The second, choosing the feature subset using metaheuristic based algorithms such as G.A is very time consuming. In method 2: Optimized Machine Learning Method and Method 4: Hybrid Method which both used optimization using G.A the time was relatively long when compared to Method 1: Machine Learning Based Feature Selection Method and Method 3: Hybrid Statistical and Machine Learning Method where optimization was not used and time was not a factor due to its noticeable shortness.

5.7 Summary

The experiments were conducted on all datasets in each of the four methods that were outlined in the methodology. These methods are:

1. Method 1: Machine Learning Based Feature Selection Method.
2. Method 2: Optimized Machine Learning Method.
3. Method 3: Hybrid Statistical and Machine Learning Method.
4. Method 4: Optimized Hybrid Method.

The features were weighted using weight by SVM in method 1 and the results were better than others regarding feature subset size and accuracy as shown in the comparison section.

In method 2 we were able to increase the accuracy and decrease the feature subset size using optimization evolutionary algorithm based on G.A to select the effective feature subset.

To further focus on the important features that are effective for classification to increase the accuracy we developed the hybrid method in method 3. A union was made between the features weighted by two different weighting methods to extract a feature subset that is less in size than the previous methods and produces higher accuracy.

The feature subset that resulted from method 3 was used in method 4 as an initial population for G.A optimization, which generates multiple subsets and selects the one

with the best results of classification accuracy as the feature subset to be used in classification. So method 4 further enhanced the accuracy and lowered the feature subset size even more than previous methods.

Tables 5.19, 5.20, and 5.21 summarize our approach on D1, D2, and D3 respectively.

Table 5.19 Approach Summary on Dataset D1

Dataset	Method Name	Feature Extraction	Feature Selection		Metaheuristic Algorithms For Feature Selection	Validation	Classifier	Tool	Evaluation Measures	Achievement	
			Feature Type	Feature Weighting						# Features	Accuracy
D1: Polarity Dataset V2.0 1000 Positive And 1000 Negative	Method 1: Machine Learning Based Feature Selection Method	Preprocessing: Tokenization Stop-Words Removal Case Normalization	Unigram	Weight By SVM	---	10 Fold Cross Validation	SVM	RapidMiner Studio V7.4	Precision, Recall, Accuracy And F-Measure	4000	98.7%
	Method 2: Optimized Machine Learning Method				Optimization Evolutionary Algorithms (G.A)					3950	99.15%
	Method 3: Hybrid Statistical And Machine Learning Method			Weight By SVM	---					1000	99.45%
	Method 4: Optimized Hybrid Method			Weight By Correlation	Optimization Evolutionary Algorithms (G.A)					970	99.53%

Table 5.20 Approach Summary on Dataset D2

Dataset	Method Name	Feature Extraction	Feature Selection			Validation	Classifier	Tool	Evaluation Measures	Achievement	
			Feature Type	Feature Weighting	Metaheuristic Algorithms For Feature Selection					# Features	Accuracy
D2: Polarity Dataset V1.0 700 Positive And 700 Negative	Method 1: Machine Learning Based Feature Selection Method	Preprocessing: Tokenization Stop-Words Removal Case Normalization	Unigram	Weight By SVM	---	10 Fold Cross Validation	SVM	RapidMiner Studio V7.4	Precision, Recall, Accuracy And F-Measure	4000	98.79%
	Method 2: Optimized Machine Learning Method				Optimization Evolutionary Algorithms (G.A)					3950	99.21%
	Method 3: Hybrid Statistical And Machine Learning Method			Weight By SVM						1000	99.46%
	Method 4: Optimized Hybrid Method			Weight By Correlation	Optimization Evolutionary Algorithms (G.A)					950	99.71%

Table 5.21 Approach Summary on Dataset D3

Dataset	Method Name	Feature Extraction	Feature Selection		Metaheuristic Algorithms For Feature Selection	Validation	Classifier	Tool	Evaluation Measures	Achievement	
			Feature Type	Feature Weighting						# Features	Accuracy
D3: A Combination Of D1 And D2	Method 1: Machine Learning Based Feature Selection Method	Preprocessing: Tokenization Stop-Words Removal Case Normalization	Unigram	Weight By Svm	---	10 Fold Cross Validation	Svm	RapidMiner Studio V7.4	Precision, Recall, Accuracy And F-Measure	4000	97.85%
	Method 2: Optimized Machine Learning Method				Optimization Evolutionary Algorithms (G.A)					3950	98.12%
	Method 3: Hybrid Statistical And Machine Learning Method			Weight By Svm	---					1000	98.66%
	Method 4: Optimized Hybrid Method			Weight By Correlation	Optimization Evolutionary Algorithms (G.A)					958	98.69%

CHAPTER 6

CONCLUSION AND FUTURE WORK

6. Conclusion and Future Work

6.1 Conclusion

Choosing the effective subset of features on which depends the process of classifying the reviews into either positive or negative is the essence to solving the problem of S.A. Unlike what is dominant among researchers in weighting features using statistical methods, like Gini Index, IG, etc. features were weighted in our approach in machine learning using SVM then a feature subset was chosen and enhanced using optimization evolutionary algorithm (GA).

Using weight by SVM for feature selection meant that the weight vector produced was tunable during the learning process to achieve the desired results. GA enhanced the feature subset produced by machine learning by reducing its size while preserving high classification accuracy.

Feature types that were experimented with are POS, and Unigrams. Results when using unigrams were better than those using POS, that's why we focused on using features of type unigrams in our experiments.

Stratified sampling 10 fold cross validation was used to evaluate the classification accuracy.

The experiments were conducted in four methods:

1. Method 1: Machine Learning Based Feature Selection Method
2. Method 2: Optimized Machine Learning Method
3. Method 3: Hybrid Statistical and Machine Learning Method
4. Method 4: Optimized Hybrid Method

For evaluation measures we used Accuracy, Recall, Precision, and F-Measure. In each of the methods above the results for each one were better than the previous one respectively in regards of feature subset size and accuracy obtained.

In the comparison section 5.5 it showed that the results we obtained were better than the results of others in both feature subset size and accuracy achieved.

6.2 Future work

Some of the suggested future work fields of study include areas where exact results must be found due to their criticality, such as security where we need high accuracy in sentiments, towards a particular event to help in decision making for security measures to be taken with little risk; here simple positive and negative results are not enough we need to know how positive and negative sentiments are with precise number such as 20% positive emotion i.e. each sentiment along with its magnitude.

Another area of interest for future work is revealing fake opinions i.e. an analytic study that shows honesty in an opinion through checking the language, the way it’s written, and the date and time of the writing.

One other area is time oriented. Sentiments can change with time according to the mood, world events, and other factors; so it’s important to look at data from the point of view of time. Building a system that takes time into consideration is another field of interest, such that it computes sentiment at the present and how it changed from the past.

References

[1]Liu, Bing, and Lei Zhang. "A survey of opinion mining and sentiment analysis." Mining text data. Springer US, 2012. 415-463.

[2] Binali, Haji, Vidyasagar Potdar, and Chen Wu. "A state of the art opinion mining and its application domains." Industrial Technology, 2009. ICIT 2009. IEEE International Conference on. IEEE, 2009.

[3] Greaves, Felix, et al. "Use of sentiment analysis for capturing patient experience from free-text comments posted online." Journal of medical Internet research 15.11 (2013).

[4]Tsytsarau, Mikalai, and Themis Palpanas. "Survey on mining subjective data on the web." Data Mining and Knowledge Discovery 24.3 (2012): 478-514.

[5]Medhat, Walaa, Ahmed Hassan, and Hoda Korashy. "Sentiment analysis algorithms and applications: A survey." Ain Shams Engineering Journal 5.4 (2014): 1093-1113.

[6] Liu, Bing. "Sentiment analysis and opinion mining." Synthesis lectures on human language technologies 5.1 (2012): 1-167.

[7] Hu, Minqing, and Bing Liu. "Mining and summarizing customer reviews." Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2004.

[8] Melville, Prem, Wojciech Gryc, and Richard D. Lawrence. "Sentiment analysis of blogs by combining lexical knowledge with text classification." Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009.

[9] Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. "Thumbs up?: sentiment classification using machine learning techniques." Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. Association for Computational Linguistics, 2002.

[10] Feldman, Ronen, and James Sanger. The text mining handbook: advanced approaches in analyzing unstructured data. Cambridge university press, 2007.

[11] Salton, Gerard, Anita Wong, and Chung-Shu Yang. "A vector space model for automatic indexing." Communications of the ACM 18.11 (1975): 613-620.

[12] Sabbah, Thabit, et al. "Hybridized term-weighting method for dark web classification." Neurocomputing 173 (2016): 1908-1926.

[13] Yu, Lei, and Huan Liu. "Efficient feature selection via analysis of relevance and redundancy." Journal of machine learning research 5.Oct (2004): 1205-1224.

[14] Saif, Hassan, Yulan He, and Harith Alani. "Semantic sentiment analysis of twitter." The Semantic Web–ISWC 2012 (2012): 508-524.

[15] Hu, Minqing, and Bing Liu. "Mining opinion features in customer reviews." AAAI. Vol. 4. No. 4. 2004.

[16] Bron, Esther E., et al. "Feature selection based on the SVM weight vector for classification of dementia." IEEE journal of biomedical and health informatics 19.5 (2015): 1617-1626.

[17] Maldonado, Sebastián, Richard Weber, and Fazel Famili. "Feature selection for high-dimensional class-imbalanced data sets using Support Vector Machines." Information Sciences 286 (2014): 228-246.

[18] Koprinska, Irena, Mashud Rana, and Vassilios G. Agelidis. "Correlation and instance based feature selection for electricity load forecasting." Knowledge-Based Systems 82 (2015): 29-40.

[19] Tang, Jiliang, Salem Alelyani, and Huan Liu. "Feature selection for classification: A review." Data Classification: Algorithms and Applications (2014): 37.

[20] Li, Jundong, et al. "Feature selection: A data perspective." ACM Computing Surveys (CSUR) 50.6 (2017): 94.

[21] Joshi, Neha S., and Suhasini A. Itkat. "A survey on feature level sentiment analysis." International Journal of Computer Science and Information Technologies 5 (2014): 5422-5425.

[22] Abd. Samad Hasan Basaria, Burairah Hussina, I. Gede Pramudya Anantaa, Junta Zeniarjab," Opinion Mining of Movie Review using Hybrid Method of Support Vector Machine and Particle Swarm Optimization", Procedia Engineering 53 (2013) 453 – 462

[23] Chen, Hsinchun. "Exploring extremism and terrorism on the web: the dark web project." Intelligence and security informatics (2007): 1-20.

[24] Zheng, Rong, et al. "A framework for authorship identification of online messages: Writing- style features and classification techniques." Journal of the Association for Information Science and Technology 57.3 (2006): 378-393.

[25]Lei, Jing. "Cross-Validation with Confidence." arXiv preprint arXiv:1703.07904 (2017).

[26] Charte, Francisco, et al. "On the impact of dataset complexity and sampling strategy in multilabel classifiers performance." International Conference on Hybrid Artificial Intelligence Systems. Springer, Cham, 2016.

[27] Tayel, Salma. "Rule-based complaint detection." (2013).

[28] Tripathy, Abinash, Ankit Agrawal, and Santanu Kumar Rath. "Classification of sentiment reviews using n-gram machine learning approach." Expert Systems with Applications 57 (2016): 117-126.

[29] Agre, Gennady, and Anton Dzhondzhorov. "A Weighted Feature Selection Method for Instance-Based Classification." International Conference on Artificial Intelligence: Methodology, Systems, and Applications. Springer International Publishing, 2016.

[30] Kalaivani, P., and K. L. Shunmuganathan. "Feature reduction based on genetic algorithm and hybrid model for opinion mining." Scientific Programming 2015 (2015): 12.

[31] Poria, Soujanya, Erik Cambria, and Alexander F. Gelbukh. "Deep Convolutional Neural Network Textual Features and Multiple Kernel Learning for Utterance-level Multimodal Sentiment Analysis." EMNLP. 2015.

[32] Huang, Lan, et al. "Particle Swarm Optimization for Traveling Salesman Problems [J]." Acta Scientiarium Naturalium Universitatis Jilinensis 4 (2003): 012.

[33] Wahyudi, Mochamad, And Dinar Ajeng Kristiyanti. "Sentiment Analysis of Smartphone Product Review Using Support Vector Machine Algorithm-Based Particle Swarm Optimization." Journal of Theoretical & Applied Information Technology 91.1 (2016).

[34] Liu, Zhi, et al. "Sentiment recognition of online course reviews using multi-swarm optimization-based selected features." Neurocomputing 185 (2016): 11-20.

[35] Gupta, Deepak Kumar, Kandula Srikanth Reddy, and Asif Ekbal. "PSO-ASent: Feature Selection Using Particle Swarm Optimization for Aspect Based Sentiment Analysis." International Conference on Applications of Natural Language to Information Systems. Springer International Publishing, 2015.

[36] Umamaheswari, K., S. P. Rajamohana, and G. Aishwaryalakshmi. "Opinion Mining using Hybrid Methods." International Journal of Computer Application, ISSN (2015): 0975-8887.

[37] Stylios, George, Christos D. Katsis, and Dimitris Christodoulakis. "Using Bio-inspired intelligence for Web Opinion Mining." International Journal of Computer Applications 87.5 (2014).

[38] Aghdam, Mehdi Hosseinzadeh, and Setareh Heidari. "Feature selection using particle swarm optimization in text categorization." Journal of Artificial Intelligence and Soft Computing Research 5.4 (2015): 231-238.

[39] Saravanan, T. M., and A. Tamilarasi. "Effective Sentiment Analysis for Opinion Mining Using Artificial Bee Colony Optimization." Research Journal of Applied Sciences, Engineering and Technology 12.8 (2016): 828-840.

[40] Dhurve, Ruby, and Megha Seth. "Weighted Sentiment Analysis Using Artificial Bee Colony Algorithm." International Journal of Science and Research (IJSR), ISSN (Online) (2015): 2319-7064.

[41] Sumathi, T., S. Karthik, And M. Marikkannan. "Artificial Bee Colony Optimization for Feature Selection in Opinion Mining." Journal of Theoretical & Applied Information Technology 66.1 (2014).

[42] Yousefpour, Alireza, et al. "Integrated Feature Selection Methods Using Metaheuristic Algorithms for Sentiment Analysis." Asian Conference on Intelligent Information and Database Systems. Springer Berlin Heidelberg, 2016.

[43] Keshavarz, Hamidreza, and Mohammad Saniee Abadeh. "SubLex: Generating subjectivity lexicons using genetic algorithm for subjectivity classification of big social data." Swarm Intelligence and Evolutionary Computation (CSIEC), 2016 1st Conference on. IEEE, 2016.

[44] Wahyudi, Mochamad, and Dwi Andini Putri. "Algorithm Application Support Vector Machine with Genetic Algorithm Optimization Technique for Selection Features for The Analysis of Sentiment on Twitter." Journal of Theoretical and Applied Information Technology 84.3 (2016): 321.

[45] Abualigah, Laith Mohammad, Ahamad Tajudin Khader, and Mohammed Azmi Al-Betar. "Unsupervised feature selection technique based on genetic algorithm for improving the Text Clustering." Computer Science and Information Technology (CSIT), 2016 7th International Conference on. IEEE, 2016.

[46] Siti Rohaidah Ahmad, Azuraliza Abu Bakar, and Mohd Ridzwan Yaakub, 2015 July, ”metaheuristic algorithms for feature selection in sentiment analysis”” , science and information conference 2015, London UK, p. 222-226.

[47] Deng, Zhi-Hong, Kun-Hu Luo, and Hong-Liang Yu. "A study of supervised term weighting scheme for sentiment analysis." Expert Systems with Applications41.7 (2014): 3506-3513.

[48]Parmar, Hitesh, Sanjay Bhanderi, and Glory Shah. "Sentiment Mining of Movie Reviews using Random Forest with Tuned Hyperparameters." (2014).

[49] Gautam, Geetika, and Divakar Yadav. "Sentiment analysis of twitter data using machine learning approaches and semantic analysis." Contemporary computing (IC3), 2014 seventh international conference on. IEEE, 2014.

[50] Liang, Po-Wei, and Bi-Ru Dai. "Opinion mining on social media data." Mobile Data Management (MDM), 2013 IEEE 14th International Conference on. Vol. 2. IEEE, 2013.

[51] Cardoso, Matheus, Angelo Loula, and Matheus Giovanni Pires. "Building a fuzzy system for opinion classification across different domains." Proceedings of the 2015 International Conference on Fuzzy Logic in Aartificial Intelligence-Volume 1424. CEUR-WS. org, 2015.

[52] Agarwal, Basant, and Namita Mittal. "Sentiment Classification using Rough Set based Hybrid Feature Selection." WASSA@ NAACL-HLT. 2013.

[53] Agarwal, Basant, et al. "Concept-level sentiment analysis with dependency-based semantic parsing: a novel approach." Cognitive Computation 7.4 (2015): 487-499.

[54]Li, Gang, and Fei Liu. "A clustering-based approach on sentiment analysis." Intelligent Systems and Knowledge Engineering (ISKE), 2010 International Conference on. IEEE, 2010.

[55] Paltoglou, Georgios, and Mike Thelwall. "A study of information retrieval weighting schemes for sentiment analysis." Proceedings of the 48th annual meeting of the association for computational linguistics. Association for Computational Linguistics, 2010.

[56] Sumathi, T., S. Karthik, and M. Marikannan. "Performance Analysis of Classification Methods for Opinion Mining." International Journal of Innovations in Engineering and Technology (IJIET) 2.4 (2013): 171-177.

[57] Jotheeswaran, Jeevanandam, R. Loganathan, and B. Madhu Sudhanan. "Feature reduction using principal component analysis for opinion mining." International Journal of Computer Science and Telecommunications 3.5 (2012): 118-121.

[58] Valarmathi, B., and V. Palanisamy. "Opinion mining classification using key word summarization based on singular value decomposition." International Journal on computer science and Engineering 3.1 (2011): 212-215.

[59] Sahayak, Varsha, Vijaya Shete, and Apashabi Pathan. "Sentiment Analysis on Twitter Data." International Journal of Innovative Research in Advanced Engineering (IJIRAE) 2.1 (2015): 178-183.

[60] Gamallo, Pablo, and Marcos Garcia. "Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets." SemEval@ COLING. 2014.

[61] Agarwal, Basant, and Namita Mittal. "Optimal feature selection for sentiment analysis." International Conference on Intelligent Text Processing and Computational Linguistics. Springer, Berlin, Heidelberg, 2013.

[62] Govindarajan, M. "Sentiment analysis of movie reviews using hybrid method of naive bayes and genetic algorithm." International Journal of Advanced Computer Research 3.4 (2013): 139.

[63] Elawady, Rasheed M., Sherif Barakat, and Nora M. Elrashidy. "Different feature selection for sentiment classification." International Journal of Information Science and Intelligent System 3.1 (2014): 137-150.

[64] Jotheeswaran, Jeevanandam, and S. Koteeswaran. "Decision tree based feature selection and multilayer perceptron for sentiment analysis." ARPN Journal of Engineering and Applied Sciences 10.14 (2015): 5883-94.

[65] MCCULLOCH, Warren S.; PITTS, Walter. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 1943, 5.4: 115-133.

[66] Yuan, Ye, and You Zhou. "Twitter Sentiment Analysis with Recursive Neural Networks." CS224D Course Projects (2015).

[67] Kim, Yoon. "Convolutional neural networks for sentence classification." arXiv preprint arXiv:1408.5882 (2014).

[68] dos Santos, Cícero Nogueira, and Maira Gatti. "Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts." COLING. 2014.

[69] Kabir, Md Monirul, Md Monirul Islam, and Kazuyuki Murase. "A new wrapper feature selection approach using neural network." Neurocomputing 73.16 (2010): 3273-3283.

[70] Zhu, Endong, et al. "Features Extraction Based on Neural Network for Cross-Domain Sentiment Classification." International Conference on Database Systems for Advanced Applications. Springer International Publishing, 2016

[71] Azar, Ahmad Taher. "Neuro-fuzzy feature selection approach based on linguistic hedges for medical diagnosis." International Journal of Modelling, Identification and Control 22.3 (2014): 195-206.

[72] Akhtar, Md Shad, et al. "A Hybrid Deep Learning Architecture for Sentiment Analysis." COLING. 2016.

[73] Bengio, Yoshua. "Learning deep architectures for AI." Foundations and trends® in Machine Learning 2.1 (2009): 1-127.

[74] Atmaca, Hamdi, B. Cetisli, and H. Serhan Yavuz. "The comparison of fuzzy inference systems and neural network approaches with ANFIS method for fuel consumption data." Second International Conference on Electrical and Electronics Engineering Papers ELECO. 2001.

[75]Relich, Marcin. "A decision support system for alternative project choice based on fuzzy neural networks." Management and Production Engineering Review 1.4 (2010): 46-54.

[76] Shirvan, Iran. "A comparative study on ANFIS and fuzzy expert system models for concrete mix design." (2011).

[77] Salton, Gerard, and Christopher Buckley. "Term-weighting approaches in automatic text retrieval." Information processing & management 24.5 (1988): 513-523.

[78] Alghamdi, Hanan M., and Ali Selamat. "Topic detections in Arabic dark websites using improved vector space model." Data Mining and Optimization (DMO), 2012 4th Conference on. IEEE, 2012.

[79] Li, Ran, and Xianjiu Guo. "An improved algorithm to term weighting in text classification." Multimedia Technology (ICMT), 2010 International Conference on. IEEE, 2010.

[80] Selamat, Ali, and Sigeru Omatu. "Web page feature selection and classification using neural networks." Information Sciences 158 (2004): 69-88.

[81] Crestani, Fabio, et al. "Short queries, natural language and spoken documents retrieval: Experiments at Glasgow University." NIST Special Publication 500-240. NIST, 1997.

[82] Quan, Changqin, Xiquan Wei, and Fuji Ren. "Comparison of SVM classification method and semantic similarity method for sentiment classification." Cloud Computing and Intelligence Systems (CCIS), 2014 IEEE 3rd International Conference on. IEEE, 2014.

[83] Ye, Qiang, Ziqiong Zhang, and Rob Law. "Sentiment classification of online reviews to travel destinations by supervised machine learning approaches." Expert systems with applications 36.3 (2009): 6527-6535.

[84] Y. Aphinyanaphongs, L. D. Fu, Z. Li, E. R. Peskin, E. Efstathiadis, C. F. Aliferis, et al., “A comprehensive empirical comparison of modern supervised classification and feature selection methods for text categorization,” Journal of the Association for Information Science and Technology, vol. 65, pp. 1964-1987, 2014.

[85] Onan, Aytuğ. "Classifier and feature set ensembles for web page classification." Journal of Information Science 42.2 (2016): 150-165.

[86] Hall MA. Correlation-based feature selection for machine learning. PhD Thesis, University of Waikato, New Zealand, 1999.

[87] Higgins, Jim. "The radical statistician: A beginners guide to unleashing the power of applied Statistics in the Real World." Jim Higgins Publishing (2006).

[88] Holland, John H. "Outline for a logical theory of adaptive systems." Journal of the ACM (JACM) 9.3 (1962): 297-314.

[89] Manek, Asha S., et al. "Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier." World wide web 20.2 (2017): 135-154.

[90] Paramesha, K., and K. C. Ravishankar. "Optimization of cross domain sentiment analysis using SentiWordNet." arXiv preprint arXiv:1401.3230(2013).

[91] Agarwal, Basant, and Namita Mittal. "Prominent feature extraction for review analysis: an empirical study." Journal of Experimental & Theoretical Artificial Intelligence 28.3 (2016): 485-498.

[92] Yousefpour, Alireza, Roliana Ibrahim, and Haza Nuzly Abdel Hamed. "Ordinal-based and frequency-based integration of feature selection methods for sentiment analysis." Expert Systems with Applications 75 (2017): 80-93.

[93] Devaraj, Madhavi, Rajesh Piryani, and Vivek Kumar Singh. "Lexicon Ensemble and Lexicon Pooling for Sentiment Polarity Detection." IETE Technical Review 33.3 (2016): 332-340.

ملخص

تحليل المشاعر والآراء أصبح امرا ملحا وضروريا، خاصة أن المؤسسات والحكومات وشركات القطاع الخاص مهتمون جدا بمعرفة ما يفكر به الناس تجاه أحداث أو منتجات معينة. حجم البيانات على شبكة الإنترنت هائل وينمو بسرعة. معالجة وتحليل هذا الحجم من البيانات أمر صعب ومكلف؛ ولهذا السبب فان الحلول القائمة لتحليل المشاعر تعاني من أوجه القصور، مثل الأبعاد العالية والدقة المنخفضة. ولذلك فان عملية اختيار الميزات ذات الصلة بالتحليل والتصنيف هي مسألة بحث.

ان اختيار السمات ذات الصلة التي تعطي دقة عالية في التصنيف ليست مهمة سهلة. ولذلك، فإن الهدف من هذه الأطروحة هو تصنيف الآراء على أساس النص إلى مشاعر إيجابية وسلبية بشكل فعال عن طريق اختيار مجموعة ميزات فرعية ذات الصلة. لحل هذه المشكلة، نقدم نهجا يستخدم التعلم الآلي وخوارزميات التحسين التطورية في اختيار مجموعة ميزات فرعية فعالة في أربع طرق.

أولا، يتم اختيار مجموعة ميزات فرعية على أساس خوارزمية التعلم الآلي. وتستخدم خوارزمية متجهات الدعم الآلي لإنتاج متجه الوزن بعد عملية التعلم التي تحتوي على القيم التي تمثل أهمية المصطلح لعملية التصنيف.

ثانيا، يتم استخدام خوارزمية التطور لتحسين عملية اختيار مجموعة ميزات فرعية من المجموعة التي نتجت من الطريقة الأولى من أجل تحسين عملية التصنيف.

الطريقة الثالثة تهجن ما بين التعلم الآلي القائم على مجموعة الميزات الفرعية التي تم إنشاؤها بواسطة الطريقة الأولى مع الطرق الإحصائية القائمة على مجموعة الميزات الفرعية التي نتجت باستخدام طريقة اختيار ارتباط الميزات.

الطريقة الرابعة تسمى الطريقة الهجينة المحسنة، حيث يتم تطبيق التطور التحسيني على المجموعة الفرعية للمميزات التي نتجت عن الطريقة الثالثة.

واستخدمت ثلاث مجموعات بيانات معروفة لتحليل المشاعر متاحة للجمهور لاختبار النهج المقترح والتحقق منه. الأول هو مجموعة البيانات القطبية (D1) V2.0، والثاني هو مجموعة البيانات القطبية (D2) V1.0، والثالث هو D3 الذي هو مزيج من D1 وD2.

يتم تقييم أداء تصنيف المشاعر في هذا البحث باستخدام الدقة، والارجاع، و الضبط، و مقياس متوسط الدقة و الارجاع.

النتائج التي تحققت في هذا البحث تفوق النتائج الواردة في الدراسات الحالية. في نهجنا حققنا في الطريقة الأولى باستخدام التعلم الآلي كطريقة لنوزين الميزات نتيجة عالية وصلت الى 98.79 ، واستطعنا ان نحسن من نتائج الطريقة الأولى لتصل الى نتيجة أعلى فقد وصلت نتائج الطريقة الثانية الى 99.21 باستخدام التطور التحسيني ، وعندما دمجت الميزات التي تم توزينها عن طريق التعلم الآلي باستخدام التوزين بخوارزمية الدعم الآلي مع الميزات التي تم توزينها بالطريق الاحصائي باستخدام التوزين بارتباط الميزات وصلت نتيجة التصنيف للميزات الناتجة عن الميزات المدموجةالى 99.46 وهذه نتيجة افضل من النتائج السابقة ، وعندما تم التحسين على مجموعة الميزات الفرعية الناتجة عن الطرق الهجينة استطعنا ان نزيد الدقة لتصل الى 99.71 و ان نقلل حجم المجموعة الفرعية. الميزات التي نتجت عن عملية المعالجة المشاعر كانت كبيرة جدا ، وقد تحققت هذه النتائج باستخدام أفضل الميزات ذات العدد القليل لمأخوذة من الميزات التي استخرجناها من عملية المعالجة للمشاعر بالمقارنة مع الأعمال القائمة مما يقلل من الوقت اللازم في عملية التصنيف.