Survey and improvisation for Sentiment Analysis of textual content

Submitted By

Krishna A. Upadhyay

18MCEC16

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2020

Survey and improvisation for Sentiment Analysis of textual content

Major Project

Submitted in fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering

Submitted By

Krishna A. Upadhyay

(18MCEC16)

Guided By

Dr. Ankit Thakkar

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2020

Certificate

This is to certify that the major project entitled **"Survey and improvisation for Sentiment Analysis of textual content"** submitted by **Krishna A. Upadhyay (18MCEC16)**, towards the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering of Nirma University, Ahmedabad, is the record of work carried out by her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project part-II, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Dr. Ankit Thakkar
Guide & Associate Professor,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Priyanka Sharma
Professor,
Coordinator M.Tech - CSE (CSE)
Institute of Technology,
Nirma University, Ahmedabad

Dr. Madhuri Bhavsar
Professor and Head,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr Rajesh N Patel
Director,
Institute of Technology,
Nirma University, Ahmedabad

Statement of Originality

I, **Krishna A Upadhyay**, **18MCEC16**, give undertaking that the Major Project entitled "**Survey and improvisation for Sentiment Analysis of textual content**" submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in **Computer Science & Engineering** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made.It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

_____

Signature of Student

Date:

Place:

Endorsed by

Dr. Ankit Thakkar

(Signature of Guide)

Acknowledgements

It gives me immense pleasure in expressing thanks and profound gratitude to **Prof. Ankit Thakkar**, Associate Professor, Computer Engineering Department, Institute of Technology, Nirma University, Ahmedabad for his valuable guidance and continual encouragement throughout this work. The appreciation and continual support he has imparted has been a great motivation to me in reaching a higher goal. His guidance has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come.

It gives me an immense pleasure to thank **Dr. Madhuri Bhavsar**, Hon'ble Head of Computer Science And Engineering Department, Institute of Technology, Nirma University, Ahmedabad for her kind support and providing basic infrastructure and healthy research environment.

A special thank you is expressed wholeheartedly to **Dr. Rajesh N Patel**, Hon'ble Director, Institute of Technology, Nirma University, Ahmedabad for the unmentionable motivation she has extended throughout course of this work.

I would also thank the Institution, all faculty members of Computer Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

- Krishna A. Upadhyay

18MCEC16

Abstract

Sentiment Analysis (SA) is useful to extract important information from textual content such as blogs, tweets, social media, websites, to name a few. Reviews derived from SA are useful for both the user and an organization. For example, if someone likes a product and want to purchase it, they will look at its reviews before making a purchase. Similarly, if an organization has launched a product, they'd need reviews from its users to improvise their product. SA extracts useful information from surveys, tweets, or social media content. This survey aims to give the readers an overall idea of how SA works. Figure. 1.1 describes the generalized flow of SA. This report presents SA procedure in detail including different levels, pre-processing techniques, types of features and feature selection techniques, approaches, challenges in SA, recent work in SA and underline results. Report also presents overview and improvement in recent approaches such as Fuzzy sentiment phrase approach, aspect based SA with auxiliary sentence construction and enhanced naiv̈e bayes approach with simulated and improved accuracy result.

Abbreviations

SA	Sentiment Analysis.
POS	Part Of Speech Tagging.
NLP	Natural Language Processing.
NN	Neural Network.
SVM	Support Vector Machine.
NB	Naïve Bayes.
ME	Maximum Entropy.
RF	Random Forest.
DT	Decision Tree.
ABSA	Aspect Based Sentiment Analysis.

Contents

List of Figures

Chapter 1

Introduction

Sentiment Analysis (SA) automates the process of classifying sentiments as per the polarity. Earlier, surveys were conducted using paper forms and were evaluated manually [57]. For manual analysis of reviews, surveys were designed and were shared with the target individuals. After collecting all the information, they'd be sorted in categories such as positive, negative, or neutral. The manual extraction of needed information from the survey is a time-consuming and tedious job. Thus, SA is used for analysis and classification process. It is also called opinion mining, which is a task of Natural Language Processing (NLP) that finds automatically subjective information conveyed by a text such as opinions, sentiments, evaluations, appraisals, and attitudes, among others [98]. SA automatically finds polarized opinions from the text. It is a combination of the NLP and the information extraction system [53]. SA will classify the text into different polarity classes. Polarity is an orientation of sentiment such as positive, negative, or neutral. There are three categories of polarity: i) Fine-grained which has five classes, that include very positive, positive, neutral, negative, and very negative. ii) A coarse-grained or binary classification has only two classes, positive and negative. iii) The three-way classification has three classes, positive, neutral, and negative [98]. The reviews or opinions will map to these classes using SA. The generalized flow of SA is as shown in figure 1.1.

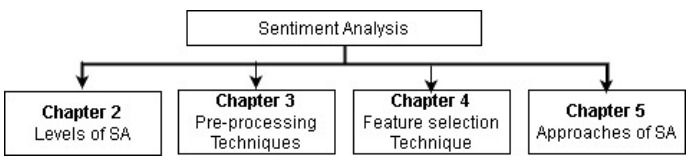

Figure 1.1: Generalized flow of SA

First step is to select the granularity. Based on the needed granularity, level of SA is selected. There are multiple levels of SA. i) Document-level, in which the entire document is considered to assess its polarity. ii) Sentence level, where the entire sentence is considered to assess the polarity. iii) Aspect level/or Feature level, which maps the user's likes and/or dislikes concerning any particular aspect. iv) Word level, in which the contribution of each word and their grammatical relation is taken into consideration [101]. v) Concept level, which works with ontologies(semantic network) at the entity level. Second step is to select the pre-processing techniques, that is used to reduce the noise and help in extract useful features from the data. After extracting features using pre-processing technique, next step is to select the useful features for SA, feature selection technique is used for that purpose. At this point, we have the selected features, and the last step is to analyse the feature and classify it to the polarity classes. To perform this step, different approaches are used, such as, i)Lexicon based approach, it works with dictionary ii)Machine learning-based approach, there are number of algorithms in ML that are used for analysis such as Naive Bayes(NB), Support Vector Machine(SVM), Decision tree, Maximum Entropy(ME), Neural Networks(NN) to name a few. iii)Hybrid approach and iv)Rule-based approach. Figure 1.2 describes a brief classification of SA levels and approaches used in SA.

There are multiple issues in SA that can result in mis-classification of a document. For example, i)Negation, "No one thinks it is good" [98]. here, 'good' is a positive word, however, 'no' makes it negative. ii)Comparison, iii)Sarcasm, iv)domain dependence, to name a few. These issues are rectified using pre-processing techniques before applying classification. The survey paper will describe the entire flow of the SA in detail, that will help the reader to understand the overall process of SA. Table 1.1 shows the comparison of different survey papers

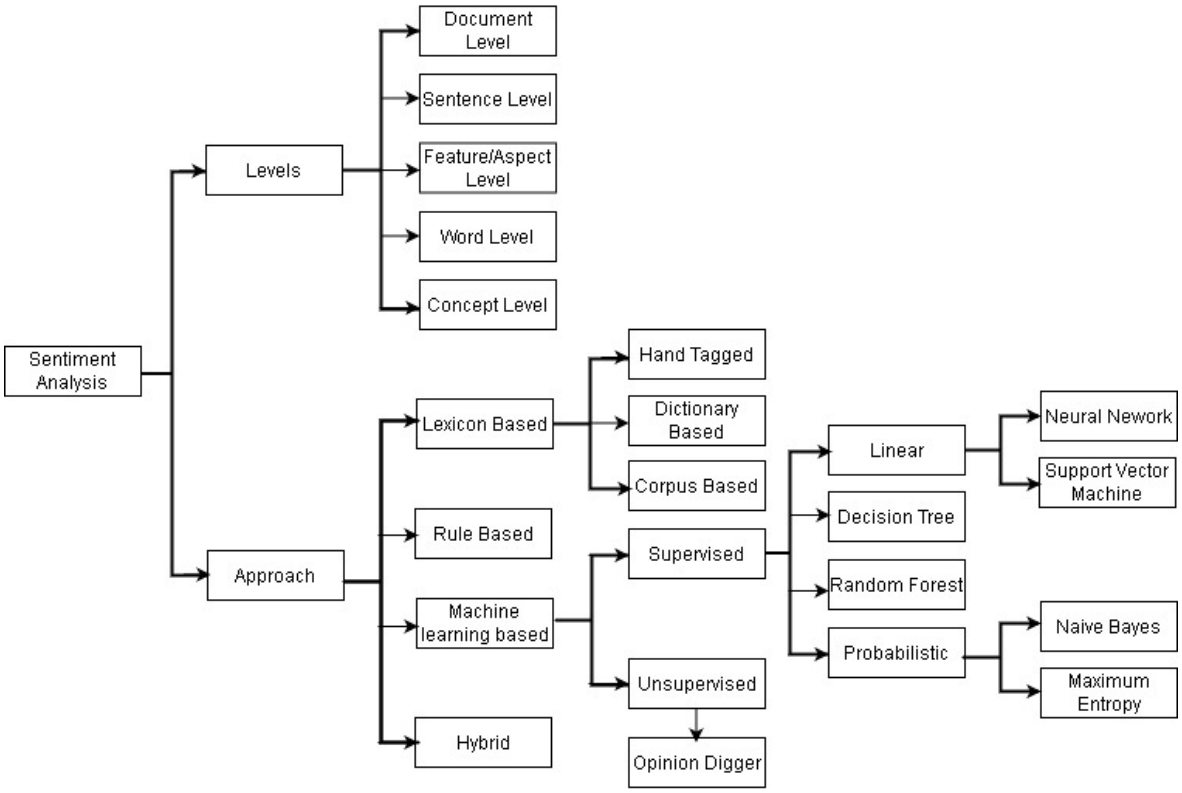

Figure 1.2: Overview of levels and approaches in SA

Reference	Levels of SA	Approaches of SA	Issues in SA	Pre-processing Techniques	Feature selection Techniques	Results of approaches
Our paper	✓	✓	✓	✓	✓	✓
Abiram et al.[1]	✓	✓	✓			✓
Hussein et al.[43]			✓			✓
Medhat et al.[65]		✓			✓	✓
Yue et al.[133]	✓	✓				
Kharde et al.[48]		✓	✓	✓	✓	✓
Hailong et al.[37]		✓			✓	✓
Varghese et al.[119]		✓				✓
Liu et al.[58]	✓	✓	✓			
Vinodhini et al.[121]		✓	✓			✓
Giachanou et al.[31]		✓	✓		✓	

Table 1.1: Comparative analysis of survey papers

The remaining article is organized as follows, chapter 2 explains different levels in SA such as document level, sentence level, feature level, word level and concept level. chapter 3 describes various types of pre-processing techniques, chapter 4 explains types of features and feature selection techniques. Chapter 5 will discuss the approaches in SA such as lexicon based, rule based, ML based and hybrid. Chapter 6 explains challenges in SA, chapter 8 describes the various applications of SA, chapter 7 describes the recent approaches in SA. Chapter 9 describes the overview and improvisation of the recent

approaches and chapter 10 will conclude the report.

Chapter 2

Levels of sentiment analysis

Based on the required granularity, level is selected to perform SA. If required granularity is high, then Document-level SA is performed, if there are multiple features or opinions in the sentence, then feature/aspect-based SA would be preferable.

2.1 Document level SA

Here the entire document is considered and classified in respective polarity classes [11]. The general approach is to check the polarity of each opinionated/subjective(sentence containing opinion) sentence present in the document, and combine them to find the polarity of the overall document [48]. In document-level SA, the assumption is that the entire document expresses opinions on a single entity such as, a specific product, restaurant or a place. If multiple entities are present in the document then it is not applicable [12].

2.2 Sentence level SA

The sentence level SA performs analyses sentences and further classifies it into different polarity classes. Document-level SA is mostly performed with binary classification because multiple sentences are present in the document that express either neutral, positive or negative polarity and based on the majority document will be classified. But in sentence-level SA, neutral class has to be considered, because such sentences may not express any opinion. There are two steps in sentence-level SA, i)Subjectivity detection - determines whether the sentence is opinionated or unopinionated, that is, subjective or objective ii)Classifying it into different polarity classes(positive, negative or neutral) [91].

Subjectivity detection is an important task of SA; it filters the opinion from factual(un opinionated/objective) information [19]. Opinionated sentences contain judgment, feelings or opinion of a person for example, "Phone's camera quality is very impressive". It varies from person to person because different people can have different opinion about the same object or device, that is useful for SA. On the other hand, facts will remain same for all, for example, "The phone has 16 MP camera". After extracting opinionated reviews, it is classified into different polarity classes.

2.3 Aspect/feature level SA

Sometimes, a sentence may have more than one aspect, for example, "Camera quality is very nice, but the phone is very heavy and difficult to hold". Here, two aspects/features, namely camera and weight are present in the sentence. The polarity towards the camera is positive, whereas for weight, it is negative. The aim of aspect/feature level SA is to find out the product features from the data and then check the polarity towards that aspect/feature [48]. This method maps user's opinion (likes and dislikes) with different features. There are three steps involved in the aspect level SA. i)identification, ii)classification and iii)aggregation [105]. Not all methods involve all three steps, mostly the first two steps are performed. In the first step, sentiment - feature/entity pairs are identified, for example, camera - very nice, weight - heavy. The second step will classify these pairs into polarity classes, for example, camera - positive, weight - negative. The third step will aggregate sentiment values for features/entities to check the overall score or class for different features/aspects, for example, if there are total 3 reviews of camera having sentiment scores of 5/5, 4/5, and 3.5/5 then the aggregated score for feature/aspect camera will be 12.5/3 that is 4.16

2.4 Word level SA

Word level SA is at the lowest level of granularity. The aim is to identify the sentiment of each word and label it with its polarity. Consider a label class L, which contains n labels l_1, l_2,.., l_n. For three-class classification, label value will be positive, neutral, or negative. A single sentence contains n number of words, say n_1, n_2,...,n_n. Here n_i corresponds to the word of natural language and l_i belongs to the restricted label class set [25]. Word level SA is supposed to tag word n_i with label l_i.

2.5 Concept level SA

It works with semantic analysis of the text using semantic networks or web ontologies [18]. An ontology represents a set of concepts as knowledge, together with the relationship between them. Here, concepts mean classes or entities that can be a living or non-living, such as, place, food, individual. This method is domain-dependent, thus, in order to create an ontology for a particular domain, knowledge of that domain is required. It contains a hierarchy with the classification of concept [130]. Consider Fig. 2.1. ,it contains three main classes, cultural attraction, sports attraction and natural attraction; natural attraction class contains sub-classes, fort, beach, wadhi, and desert, likewise,

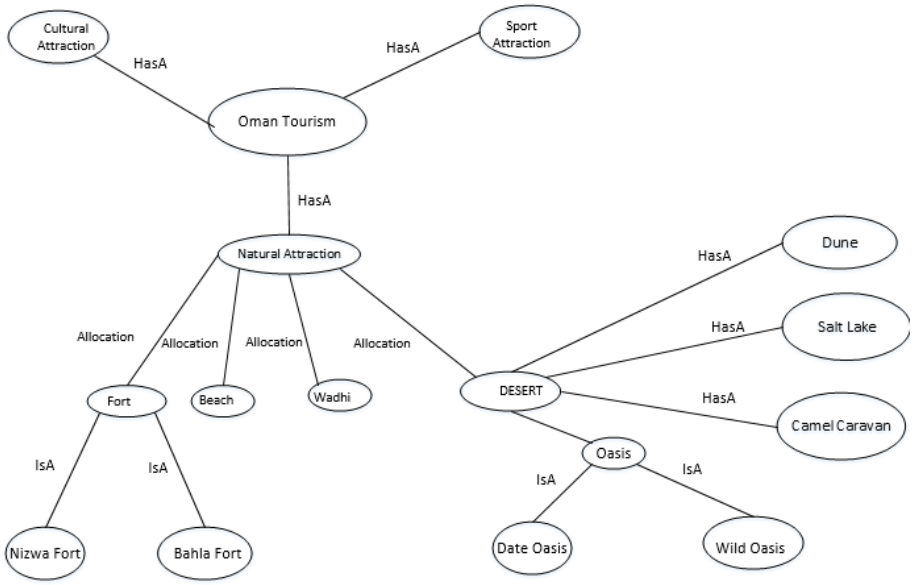

Figure 2.1: Domain Specific ontology for Oman Tourism [93]

the hierarchy is maintained [93]. After creating semantic networks for a domain, there are two more steps. i)training - detect the named entity of a sentence, map it with the ontology concepts/class and then calculate the polarity ii)testing - there are two scenarios for testing which depends on extracted entity being present in the concept net. If it is present, then retrieve the polarity of sentiment, otherwise, find it's polarity from the conceptual semantic concept.

Chapter 3

Pre-Processing Techniques

It is the technique of cleaning a text that classifies it to make it more optimized [111]. It is a mandatory part of SA, because it helps in reducing the noise present in the data. Here, the noise refers to the data that does not have any useful information for analysis. It extracts the needed relevant content from the text [33]. Users use slang, abbreviation or other unnecessary punctuation marks to emphasize their emotions [27]. Such extra characters do not help in classifying sentiments to their proper classes, however they increase the noise and in turn reduce the classification accuracy. Hence it is necessary to remove noise before proceeding further.The noise can reach up to 40 percent in the original data that creates ambiguity [27]. There are number of steps for pre-processing. Each step removes some amount of noise from the given text or opinion.

Tokenization is the first step of pre-processing [70], [79], [120], all the other steps are explained in this section. Tokenization creates the words list given a full text string [9]. For example, consider a sentence:"the movie was so interesting!" after applying tokenization on this sentence it will be 'the movie', 'was', 'so', 'interesting','!'. After all the words are separated from the original text/opinion, pre-processing can be applied. There are multiple pre-processing techniques using which unnecessary tokens from the text can be removed.

3.1 Remove Unicode strings and noise

The dataset is not necessarily clean, hence, first of all, non-English words and Unicode strings such as 'x06' and 'u002c' should be removed. One of the possible ways to perform this step is to use Regular Expression(RE). RE works on the patterns, if the patterns

provided to RE are found in a sentence, they will be removed. For example, consider the string: "remove Unicode strings like 'u002', and 'x96' ". After applying pre-processing on this string output string is as follows, "remove Unicode strings like, and". Here 'u002' and 'x96' is removed from the sentence.

3.2 Replace URLs and user mentions

In twitter text, many sentences contain URL, hashtags or user mentions, but this all does not contain any opinion or sentiment, so the approach is to replace them, with their predefined tag as performed by [4]. Symeonidis et al. [111] proposed that url present in the text should be replaced with URL tag, #(hashtag) should be removed and @, that is user mentions should be replaced with AT_USER tag. For example, consider a sentence from SS-Twitter dataset: "@BarCough it's enough to make you sick, eh? there's nothing sacred anymore. http://twittascope.com/twittascope #bored." After removing url, user mentions and hashtags, the sentence will be, "AT_USER it's enough to make you sick, eh? there's nothing sacred anymore. URL bored." Here, @BarCough is replaced with AT_USER, http://twittascope.com/twittascope is replaced with the URL tag, and is removed.

3.3 Replacing slang and abbreviations

In social media all the posts or tweets are written in an informal way. There are two concepts, slang words and abbreviations that are used repetitively. Slang is a type of language that has some informal words and used particularly by a small group [111]. People use these words, instead of formal language, to quickly communicate with each other. For example, frenemy, it is a slang word that is a combination of two words friends and enemy, 2day that means today, 4ward that means forward, to name a few.

Abbreviation: these are short form of words or phrases, but these words are not understandable hence, it needs to be replaced with its meaning. For example, LOL is an abbreviation of lots of laugh. For these type of words(slang and abbreviation), look-up table is maintained, that can be constructed manually, contains words and their meanings as shown in Table 3.1.

Hence, whenever these words will occur, it will be replaced with its meaning or full form [54], [131]. For example, "btw, this phone is amz." after pre-processing the sentence will

Slang or abbreviation	meaning	Slang or abbreviation	meaning
aight	alright	airhead	stupid
aka	as known as	lol	laughing out loud
amigo	friend	amz	amazing
app	application	armpit	undesirable
asap	as soon as possible	atm	at the moment
atw	all the way	b/c	because
b-day	birthday	b4	before
b4n	bye for now		

Table 3.1: Look-up table for slang or abbreviation and their meaning [23].

be, "by the way, this phone is amazing"

3.4 Replacing contractions

These are words like don't, can't, won't, to name a few. These words are replaced by their full forms [14], [100]. For Example, "This laptop is very costly, I can't buy it." after pre-processing the sentence will be, "This laptop is very costly, I can not buy it.". This technique will help in enriching the classifier model. If we don't use this technique, then all words may be neglected due to low frequency of occurrence.

3.5 Removing numbers

Numbers present in the sentence must be removed because they do not contain any sentiment [39]. Here, they don't have to be replaced with any other word or tag, but removed instead [45]. Example from SS-Twitter, "It took 2 hours rummaging through my receipt drawer only to learn I'd lost the receipt, isn't there a better way?" after pre-processing the sentence will be, "It took hours rummaging through my receipt drawer only to learn I'd lost the receipt, isn't there a better way?". Numbers should be removed after replacement of all the slang words because if there is any slang word like 2day(today) or 2c u(to see you) which contains number, removing them can entirely change the meaning of the word and the sentence. For example, "I will 4ward this document to you". after removing number this sentence will be, "I will ward this document to you". Here removing number prior to replacing slang word will make the sentence meaningless. However, Lin et al. [56] argues that the numbers can improve the effectiveness of classification. Consider the example, "this was a 4/5 star movie". In this sentence, it adds value to the sentiment and helps categorize it into a positive review.

3.6 Replacing repetitions of punctuation

There are three punctuation marks that are used frequently, full stop mark('.'), exclamation mark('!'), and question mark('?'). These punctuation marks are sometimes used consecutively to express the intensity of the statement. For example, "what an amazing movie it was!!!!!!!!!". Here, repetitive exclamation mark shows that the writer of this post or tweet extremely likes the movie. Multiple occurrences of punctuation marks should be replaced with a representative token [8]. For that, multiple occurrences of exclamation mark will be replaced with 'multiExclamation' tag, multiple stop marks will be replaced with 'multiStop' tag, and multiple question mark will be replaced with 'multiQuestion' tag with a blank space before and after the tag.

Consider a sentence, "what an amazing movie it was!!!!!!!!!!" will be "what an amazing movie it was multiExclamation" after applying this pre-processing technique. "This technique is used to normalize the language of tweets and generalize the vocabulary employed to represent sentiment"[111].

3.7 Replacing negations with antonyms

This technique is presented by perkins et al.[84]. It involves three steps. i)Search the word 'not' in the sentence. ii)Select the word next to the word 'not' and find the antonym of the selected word. iii)If the antonym of the selected word is present then replace both 'not' and selected word with the found antonym. For example, 'I am feeling not good.' Now as we reach to the word 'not', select the next word that is 'good', then find the antonym of a word good that is bad and replace 'not good' with 'bad'. After pre-processing the sentence will be, 'I am feeling bad.' [69], [80], [66].

3.8 Removing punctuation

Removing punctuation is a classical technique of pre-processing for data mining and information retrieval. This technique will remove all the punctuation mark from the given text. For example, "Urgh I'm so bored and tired. text. call, anything. hmm I must find things to multi-task on now!" [111], after removing punctuation marks, the sentence will be, "Urgh Im so bored and tired text call anything hmm I must find things to multi-task on now".

3.9 Handling capitalized words

In microblogs, users often use all capitalized words to express their emotions [33]. For example, REALYYY, it is known as 'e-shouting'. These type of words can help to classify that sentence correctly. To handle capitalized words, they will be given a pre-fix 'ALL_CAPS_' [33], [88] Consider a sentence, "HAHAHAHHAAA, it was a fun", after pre-processing the sentence will be "ALL_CAPS_HAHAHAHHAAA, it was a fun".

3.10 Lowercasing

In this pre-processing technique, the entire word is converted into lowercase. This technique helps mapping its corresponding feature, irrespective of their case [33]. For example, "I Love This Apple" will be converted to "I love this apple". Although it is very effective in reducing vocabulary size and sparsity, it can reduce performance by increasing ambiguity [17]. For example, "apple is asking its manufacturers to move mac-book air production to the united states". Here, apple(fruit) and Apple(technical firm) will be considered as the same entity in the above sentence.

3.11 Removing stopwords

Stopwords are words which don't affect the classification of text, such as, articles, pronouns. Different techniques can be used to remove stop words. Gokulakrishnan et al. [33] have used Term frequency - Inverted Document Frequency(TF-IDF) to remove the stop words. Here, TF is the number of times a word occurs in the given document. For example, "this is an apple and this one is an orange". In the sentence TF of words 'this', 'is' and 'an' is 2 and TF of all the other words is 1. IDF - If there are a list of documents, then IDF will find words that occur most frequently in every document and remove those words. Stop words can be removed using these technique.

There are a number of lists that contain stop words. If the given text contains any of these words in the list, it is removed. Krouska et al. [55] have used "Rainbow" list, Srividhya et al. [108] have used "SMART" list and Symeonidis et al. [111] have used standard stop word list provided by Natural Language Tool Kit(NLTK). Consider a sentence, "Its time for you to changed direction, This one is the answer, It will blow your socks", after pre-processing the sentence will be, "time changed direction, one answer, blow socks".

3.12 Replacing elongated words

Elongated are words that contain multiple consecutive occurrences of any character, such as cooooooool, happyyyyyy, to name a few. These words are used by people to impose emotion in their post or tweet. It is essential to replace these kinds of words with their original or source word, otherwise it will be considered as a different word and because of low frequency of occurrence they will be ignored.

This method was examined in [72] and [6]. If any character appears more than twice in any word then it will be considered as an elongated word [6]. For example consider the sentence, "I am soooo happpyyyyy with this laptop" after pre-processing the sentence will be, "I am so happy with this laptop".

3.13 Spelling correction

The spelling mistake is the most common mistake that users make on social media. These mistakes can make classification difficult or sometimes incorrect. This problem compounds when there is a domain such as politics, because jargon, names and all other non-dictionary words are standard [75]. Aspell is a program that is freely available for spell checking. To find out the mistakes made by users all the words in the sentence are fed to aspell, all the words that are flagged as misspelled, are replaced with the first word suggested by aspell. Mullen et al. [75] have observed that using this technique, accuracy can be improved by 3.27%. Even though it is not a big number, it can still yield some improvement. Consider a sentence, "I was wanderring, how beatiful this world is", after pre-processing the sentence will be, "I was wondering, how beautiful this world is".

3.14 Part-of-Speech(POS) tagging

All the words from a post or tweet are not important. Many of them do not help in classification. Hence, to select a few words from the sentence among many, this step of pre-processing is very important. In this technique all the words of the sentence are tagged or labeled such as, noun(NN), verb(VB), or adverb(RB). This label is known as POS label. More specialized POS like a superlative adverb(RBS), proper plural noun(NNPS), third person verb(VBZ) are also assigned to the word. But these specialized POS words are not much useful for classification, hence, only noun, verb and adverb are kept and rest

all words are removed [54], [111] Consider a sentence, “About to spend the next couple of hours of my life working for the man” [111]. After pre-processing the sentence will be, “about spend next couple hour life work man”.

3.15 Lemmatization

Lemma or morphology is a dictionary form/canonical form/citation form with respect to a set of words. For example, ‘produces’, ‘produced’, and ‘producing’ are a form of the same lexeme ‘produce’. It is a process of replacing token with its corresponding lemma or root word [17]. The main reason for using lemmatization is to reduce sparsity. This method can be used to reduce the number of extracted feature [36]. For example, consider a sentence, “This laptop can be used for running heavier programs very effectively”, after pre-processing the sentence will be, “ This laptop can be used for run heavy program very effect”.

3.16 Stemming

Stemming and lemmatization are similar, except that stemming cuts the end or beginning, that is, prefix or suffix from the word while lemmatization replaces the word with its root word [6]. For example, consider the word ‘studies’. Stemming will remove the suffix that is ‘ies’ and after stemming the word will be ‘studi’ while lemmatization converts the word to its root word that is ‘study’. Both these techniques are mutually exclusive hence cannot, be used together. Mejova et al. [68] have explained the effect of stemming in classification. For example consider a sentence, “Everyday he wishes Good morning to everyone”, after stemming the sentence will be, “Everyday he wish Good morn to everyone”.

3.17 Handling negations

Negation is a crucial part of the SA because it can entirely change the sentiment of a statement. Consider the sentence, “I don’t think you should buy this phone”. Here the word ‘don’t’ changes the polarity of the statement from positive to negative. To handle the negation, ‘NEG_ ’ prefix tag can be added to every word after negative word [111]. For example, after applying the tag, the above sentence will be, “I don’t NEG_think NEG_you NEG_should NEG_buy NEG_this NEG_phone”. Angiani et al. [6] have proposed to

replace all the negative words such as, can't, won't, don't with a word not. Consider a sentence, "I don't think you should buy this phone" will be "I not think you should buy this phone" after applying negation. Table. 3.2 explains comparative analysis of applying different pre-processing technique on ss-twitter dataset. In table 3.2, 0 refers to the tokenization technique and other numbers are the suffix of the different subsection. Such as 3 refers to the section 3.3.

References	**Steps followed**	**Tokens after pre-processing**
Symeonidis et. al[111]	0,3,4,5,14	18476
Kouloumpis et. al[54]	0,2,3,9,10,12,14	18429
Nayak et. al[79]	0,11,13,16	18491
Vijayarani et.al[120]	0,11,16	18491
Balahur et.al[8]	0,3,6,8,13	18849
Medhat et. al[66]	0,7,11,14	21132
Collados et.al[17]	0,9,14	31156
Krouska et. al[55]	0,11,16	18491
Angiani et. al[6]	0,2,8,10,12,16,17	18132
Bontcheva et. al[15]	0,3,14	21077

Table 3.2: Comparative analysis of number of tokens remaining after applying different pre-processing techniques used by different authors on SS-Twitter dataset.

Chapter 4

Feature Selection

Feature selection has a significant role in sentiment classification. Without feature selection, features can't be classified correctly. Feature searching process can be divided into two main categories, i) feature extraction, it is a process of extracting features from the given text. There are multiple weighing schemes available for this task to produce new non-correlated features and to reduce the available features [44]. ii) feature selection, its aim is to select features from a large group of feature set such that the accuracy of the classification can be increased [44]. This section describes different categories of features and feature selection techniques.

4.1 Feature categorization

There are different types of features that can be categorized based on the nature of that feature, such as semantic feature, frequent feature, and implicit feature as explained below.

1. Semantic feature: It works on semantic orientation (SO) and contextual information. For example, consider two features, "traveled to" and "lived to" relates to the LOCATION [97].

2. Frequent feature: These are the features of an entity that acquires more attention of people [42]. These features are also called as hot features. Association rule mining is applied to find out these features. For example, consider a data set that contains multiple transactions with multiple items bought by the customer. Such as transaction 1 with items bread, butter, cheese, transaction 2 with items bread, jam,

butter, and transaction 3 with items bread, butter, sugar. From these transactions, it can be said that people are more interested in Buying bread and butter.

3. Implicit feature: These features are not explicitly mentioned in the sentence, as explained in section 6.11. Adverbs and adjectives are the most common indicators of these type of features. For example, "This car is too expensive", here, expensive is an adjective which indicates the price [63].

4.2 Feature Selection Techniques

After applying various pre-processing and feature extraction technique, a large group of token will be available. All of these tokens can't be used as a feature, because they are big in number. If all of these tokens are used as feature then it may increase complexity. Feature selection is used to select features in order to increase the accuracy of classification. There are multiple feature selection techniques that can be applied on extracted features. This section describes different feature selection techniques.

4.2.1 Information Gain (IG)

It extracts informative features from the whole outset of corpus [113]. It evaluates information within the feature that can be used to classify documents, such as emails. The polarity class of the feature is to be decided from both the scenarios, whether the word is present or absent in the document. IG can be calculated by measuring a deduction in overall entropy because of the inclusion of any feature. If X and Y are discrete random features then the entropy of feature Y before and after inclusion of X can be calculated using equation 4.1 and 4.2, [113].

$$H(Y) = -\sum_{y \in Y} p(y) log_2 p(y) \tag{4.1}$$

$$H(Y/X) = -\sum_{x \in X} (x) \sum_{y \in Y} p(y/x) log_2 p(y/x) \tag{4.2}$$

IG is the value of additional information of feature Y, that is provided by feature X, using which, the entropy of Y can be decreased [113]. Formula for the same is given in

equation 4.3, 4.4, and 4.5.

$$IG(x) = H(Y) - H(Y/X) \tag{4.3}$$

$$IG(y) = H(X) - H(X/Y) \tag{4.4}$$

$$IG = H(Y) + H(X) - H(Y, X) \tag{4.5}$$

4.2.2 Gain Ratio

This method is an extension of the previous method IG. IG is biased toward the selection of features, and to compensate the bias, Gain Ratio(GR) is used, which is a non-symmetrical measurement [38]. Formula for GR is stated by equation 4.6.

$$GR = \frac{IG}{H(X)} \tag{4.6}$$

4.2.3 Chi-Squared test

It is a well-known and commonly used technique to select informative features from the documents. Chi-Squared $\chi\hat{2}$ method provides valuable features from the feature space with respect to the class by analyzing the value of $\chi\hat{2}$ statistics [59]. Chi-Squared $\chi\hat{2}$ method has an initial hypothesis H_0, which makes an assumption, "Two features are dissimilar" [59]. Formula for chi-Squared method is given by equation 4.7.

$$\chi^2 = \frac{1}{d}\sum_{k=1}^{n}\frac{(O_k - E_k)^2}{E_k} \tag{4.7}$$

Here, O_k is observed frequency and E_k is expected frequency. If the value of $\chi\hat{2}$ is higher than the given threshold, the null hypothesis is considered to be false, that is there are some correlation between features. On the other hand, if it is less than the threshold, the null hypothesis is correct, that is the features are dissimilar.

4.2.4 1 Rule (1R)

This is an algorithm proposed by Holte et al. [41]. It takes a set of instances that have many features and different classes. It iteratively selects the single best feature. The

algorithm is as follows:

For each feature f^x

1. For each value v^x from the domain f^x
2. Choose the set of example where feature f^x has value v^x
3. Consider c^x = most frequent class within set
4. Add the condition, for feature f^x, for value v^x, the class will be c^x (Rule for feature f^x)

4.2.5 NLP based Feature selection

NLP based feature selection can be performed using the following techniques.

1. Based on POS tagging. In this technique, every word in the sentence is tagged as described in section 3.14. It is found that noun phrases, noun, adverbs, and adjectives express features within a sentence. For example, "I saw a phone that has a very large screen" Here phone and screen are Noun phrase, very is adverb and large is adjective which is a feature within the sentence.
2. If any term is occurring near the subjective word then it is a feature [50].
3. If 'P has F' or 'F of P' scenario occurs, these words can be selected as features, where P is a product and F is a feature [86]. For example, "Phone has camera" is 'P has F' type sentence where camera(f) is a feature and "camera of phone" is 'F of P' type sentence with the same features.

4.2.6 Statistical techniques

This technique has two sub types, i) Univariate, this method is also known as feature filtering. As the name suggests, it ignores attribute interaction. IG, log likely-hood, occurrence frequency, chi-square, minimum frequency threshold are the techniques used to implement this method. ii) Multivariate, this method considers group of attributes. Wrapper model is used for attribute selection. This method is computationally expensive compared to the univariate method because it considers attribute interaction. Recursive feature elimination, DT and genetic algorithms are used to implement this technique.

4.2.7 Symmetrical Uncertainty (SU)

This technique evaluates a single attribute. It is a symmetrical measure of correlation between features. It calculates IG from the features. Formula for SU is given in equation 4.8, and 4.9, [117].

$$SU(X,Y) = 2 * [\frac{IG(X,Y)}{H(X)} + H(Y)] \tag{4.8}$$

$$Here,\ IG(X,Y) = H(X) - H(X|Y) \tag{4.9}$$

IG(X, Y) is the information gain of X, given Y, and H(X) is the entropy of X that gives randomness in X. It can be calculated using equation 4.10.

$$H(X) = -\sum_{x \in X} p(x) log_2 p(x) \tag{4.10}$$

4.2.8 Wrapper and Naïve Bayes(WNB)

It is a wrapper technique which makes use of the NB probabilistic model. It looks for an optimal subset in the search space of features [117]. If there are features F_1, F_2... F_n of any review and there is a class called C then the conditional probability is given by equation 4.11, [117].

$$p(C|F_1, F_2...F_n) = p(C) * \frac{p(F_1, F_2...F_n|C)}{p(F_1, F_2...F_n)} \tag{4.11}$$

Where p(C|F_1, F_2... F_n) is the probability of class C given all the features.

4.2.9 Hybrid

Hybrid is a technique where more than one methods/techniques are used. Hybrid method in SA is used to select features with multiple techniques, for example, Hu et al. [42] has applied this technique by using POS tagging with WordNet dictionary.

Chapter 5

Approaches of SA

There are multiple techniques/approaches to perform SA, such as, Lexicon based approach, Rule based approach, Machine Learning based approach or Hybrid approach. Lexicon based approach uses dictionary to classify the text, Rule based approach uses "if then else" rule for classification, Machine learning approach uses different models such as NN, SVM, NB, ME, decision tree, random forest to name a few. Hybrid approach uses more than one approach for example rule based approach with machine learning. This section will describe the approaches of SA in depth.

5.1 Lexicon based approach

The general approach for lexicon based approach is to find the polarity of extracted words from the given document. If it contains more positive lexicons(word) than negative lexicons(word), then, it is classified as a positive document. If the reverse is true then it is classifiedas a negative document. Vohra et al. [123] has described four steps. i)preprocess: preprocessing of document is mandatory to remove words that do not contain any sentiment value, for example, stop words, slang words, to name a few. ii)initialization: initialize the total sentiment score of the document as zero iii)tokenize and polarity check: tokenize lexicons from the document and check it's presence in the sentiment dictionary. If it is present and contains positive sentiment, then increase the sentiment score, and if it contains negative sentiment, then decrease the score. iv) check the final sentiment score: define the threshold first, for example On a scale of 1 to 5, threshold for negative will be <=2, for neutral it will be 3 and for positive it will be >=4. If the score is greater than the threshold score, classify the text as positive, otherwise, negative. There are three

methods for lexicon based SA

5.1.1 Hand tagged Method

This method is straight forward but tedious and time-consuming. In this method, manually create lexicons and then tag them with positive or negative polarity. For manually tagging the lexicons, thousands of messages must be read, then the sentiment contained in the words/lexicon is identified, and lastly, they are tagged. [52]. Examples of the manually created dataset is i)publicly available dataset named Multi-Perspective Question Answering(MPQA), created by [127], and it contains 4850 words. Manually tagged 8000 words list contains subjectivity clues, which is created by [126] and expanded by [129].

5.1.2 Dictionary based Method

In this method, the dictionary is used to find the polarity of the lexicons. There are three steps, i)initial seed words with known polarity are selected, ii)synonyms of that word are added to the dictionary with the same polarity to expand the dictionary iii)updated dictionary is used and step ii is repeated until no other synonyms are left. [42] has used 30 initially-hand tagged words as a seed list, expanded this list by searching in the WordNet dictionary. Park et al. [81] has used three different dictionaries, Oxford Dictionary, Collins Dictionary, and Thesaurus Web Service, to improve the reliability. Because if irrelevant synonyms are collected, the text may be classified incorrectly. 3090 hand tagged word were used by [81] as a seed list. The motive of this method is to establish the fact that a word and its synonym have the same polarity. After creating the dictionary, the approach explained in section 5.1 is followed to find the document polarity. The limitation of dictionary based method is its inability find the word with domain-specific orientation [123].

5.1.3 Corpus based method

Corpus-based approach has an advantage over the dictionary-based approach i.e., corpus carries domain specificity, and it can be used to find domain-specific sentiment word and their orientation. There are two methods for corpus-based SA

1. Statistical Approach: Here, the frequency of words with positive text and negative text is calculated. If a word appears more frequently with positive text, then it's polarity is considered to be positive, if it appears more with negative text then it's polarity is

considered to be negative, or if it has an equal frequency of appearance with positive and negative text then it is said to be neutral. If two words appear frequently in the same context, they can contain the same polarity, therefore polarity of an unknown word can be determined based on the relative frequency of co-occurrence [92]. Pointwise Mutual Information and Information Retrieval(PMI-IR) is an algorithm that is used to estimate the semantic orientation of a phrase [116].

It first extracts adverbs or adjectives from the text, as they are good indicators of probability. But context can not be determined with the single word. For example, the word "Unpredictable" has a positive polarity in the context of a movie review but has a negative polarity in the automotive review context, therefore, two words are extracted, one to indicate word and the other to indicate context. PMI between two words can be determined using the following equation 5.1 [116].

$$PMI(word1, word2) = log_2 \left[\frac{p(word_1 \& word_2)}{p(word_1)p(word_2)} \right] \tag{5.1}$$

Here, the ratio between p(word$_1$& word$_2$), p(word$_1$)p(word$_2$) states the measure of the degree of statistical dependence between the words.
Semantic orientation(SO) of a phrase is determined as given in equation 9.1 [116].

$$SO(Phrase) = PMI(Phrase, \text{“}Excellent\text{”}) - PMI(Phrase, \text{“}Poor\text{”}) \tag{5.2}$$

The word excellent and poor is used because it is observed that in five star rating, poor indicates one star and excellent indicates five stars.

2. Semantic approach: In this approach, similar sentiment values are assigned to semantically close words. Initially, a list of few words is taken as a seed list, then the list is expanded with the synonyms and antonyms of the words present in the seed list. Further, the unknown word's polarity is measured by the relative count of the positive and negative synonym of the underlying unknown word [92].

5.2 Rule based approach

It is a classification scheme that used the IF-THEN rule to classify data. It is an expression of the form LHS ⇒ RHS [115], where LHS is a list of rules called antecedent and RHS is called consequent of the result that is, conclusion. If all the conditions of antecedent are met then the conclusion will be derived. Tung et al. [115] has explained rule based approach, and stated three main components for the approach.

- Rule Induction algorithm: It is a process of extracting If-then rules from the data, that can be performed with the following three methods

    1. Sequential covering algorithm

       It is the generic algorithm, in which, the iterator learns one rule at a time from the given database. Then all the data covered by the rule is deleted from the database, before learning the next rule. This process will continue until the learning conditions are met, such as no more interesting rules can be generated from the remaining dataset [115].

    2. Data mining methods, such as decision making

       DT is used to perform this step. Three types of nodes are present in the DT: root node, intermediate node, and leaf node. If the data/text fulfills the conditions of root node, condition in the intermediate nodes are checked. Here in the DT, leaf node represents class labels. Data/text reaching the leaf node indicates that it has fulfilled all the conditions till the leaf node hence, class label is assigned to the given data/text [115].

    3. Association rule mining

       Association refers to the patterns [115]. Consider buying patterns of customers, buying items such as bread butter, bread jam, to name a few. If a customer buys bread then there is a high probability that (s)he will buy butter. These types of pattern or association will be found from the dataset using different algorithms such as Apriori.

- Rule ranking measures: These measures have values that are used to check the use-

fulness of the rule in providing an accurate prediction. It is used in the induction algorithm to prune the unnecessary rules that do not aid in improving accuracy. It is also used in class prediction algorithms by ranking the rules, that is then used to predict the class of the given text [115].

- Class prediction algorithm: It predicts the class of given 'text with unknown class'. It uses the IF-THEN rule, which is an output of the rule induction algorithm, to classify the given text. If all the antecedent match, the output will be a class label given at consequent. If multiple rules are matched with the given text to classify, then there are further two approaches, i) top rule approach, it uses rule ranking measure to select the best IF-THEN rule. ii)Aggregation approach, it aggregates the result generated by multiple IF-THEN rules for classification [115].

5.3 Machine learning-based approach

SA automatically classifies the text towards the subject. This section explains different methods of Machine Learning(ML) based algorithms to perform this task. As given in figure. 1.2 ML-based algorithms can be classified into two categories, supervised and unsupervised. i)Supervised algorithms: this algorithm trains the model with well-labeled data that can be further classified as linear classification, probabilistic classification, decision trees, and random forest. Linear classifier classifies the data by creating a decision boundary. Consider a case of two-class classifications, positive and negative. The linear classifier will create a decision boundary such that if the data is present at one side of the boundary, it will be classified as positive and if it is present on the other side it will be classified as negative. Neural Network(NN) and Support Vector Machine(SVM) are the types of linear classification models. A probabilistic classifier is a classifier that can predict. Given a set of input text or sentences, it will distribute the probability over a set of input classes [128]. Consider 3 classes (positive, negative and neutral) and a sentence is given to the classifier, it will predict the probability of a sentence to be in each class, say 60% - positive, 30% - negative and 10% neutral. Naïve Bayes(NB) and Maximum Entropy(ME) are the types of a probabilistic classification model. The Decision Tree(DT) is used to represent choices and their subsequent results in the form of a graph. Random Forest(RF) is an ensemble of decision trees. ii)Unsupervised algorithms:

this algorithm trains the data with information that is not labeled. It groups the data based on the similarity, pattern and differences [30]. Opinion Digger is an unsupervised approach introduced by Moghad et al. [71]

5.3.1 Neural Network (NN)

NN tries to mimic the human brain [102]. It has multiple nodes and edges connected in a layered format. For SA, first layer of the network contains initial feature vector.All the edges have a pre-defined weight. The aim of NN is to optimize the weights to increase the model's accuracy. Each node is multiplied to its connected edge and the summation of the multiplication for each layer is given to the activation function, that classifies the data into a particular class. The objective is to train the internal weights using gradient descent and Back-propagation [99]. After computing the result, the error function is used to check the difference between actual and expected result, then it propagates back using back propagation to update the neuron's weights. The process recurrs down to neglegible/zero error. After training, the trained model is used to make the classification.

5.3.2 Support Vector Machine (SVM)

SVM is a linear classifier, it maps input feature vectors into a higher-dimensional feature space through nonlinear mapping. SVM assumes that each feature lives inside a hyper-space, then it uses hyperplane to divide or separate the data fig. 5.1. Consider there are

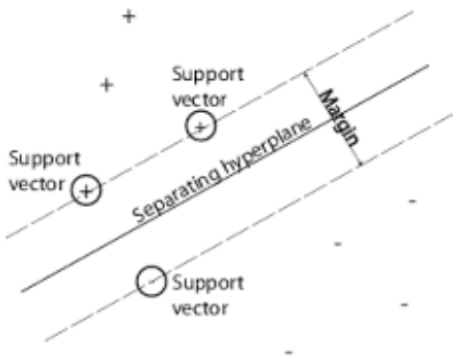

Figure 5.1: Support vector machine [99]

two points in the hyperspace, multiple hyperplanes are possible to separate these points. Aim of the SVM approach is to maximize the margin or distance between these points. SVM can classify the data into more than two classes that overcomes the limitation of linear classification. For example, if it is required to classify the data into three classes

negative, positive and neutral, then one vs all model is used that will create n model for n classes. For three class classification, three models will be created. For example, neutral vs positive, negative, positive vs negative, neutral and negative vs neutral, positive. This model is difficult to implement, for that, SVM allows some data to classify incorrectly as an outlier, called as soft margin [99].

5.3.3 Decision Tree (DT)

DT is a supervised algorithm that is used to represent choices and their subsequent results in the form of graphs [95], where the node represents an entity and edges of the graph represent a decision to be made [95]. DT is built in the top down approach using the training set given to the classifier. The top most node, called the root node will be the best predictor that is built, based on the entropy. Entropy is the measure of randomness that checks the homogeneity or similarity of the samples, if all the samples are homogeneous then the entropy is zero and if the samples are divided equally then the entropy is unity/one [106]. After selecting the root node, all the subsequent node are created the same way. The process of creating classification node is stopped if either all the samples belong to the same class or there are no attributes left for further partition [60]. Here, the leaf node represents the decision. A decision tree to predict whether a cricket match will be played is given in the fig. 5.2.

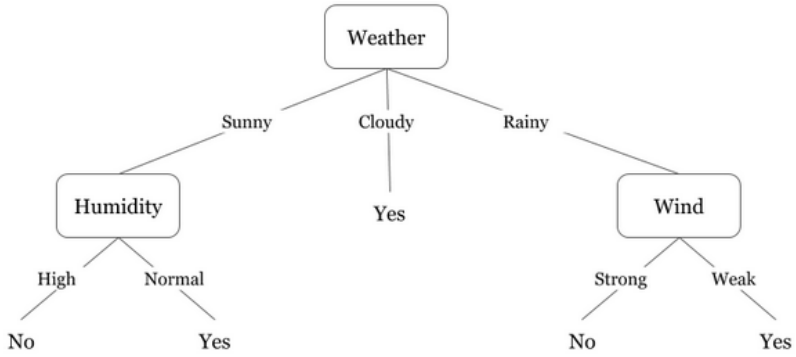

Figure 5.2: Simple DT to predict whether the cricket match will be played or not [107]

From the DT classifier it can be said that, if the weather is rainy and the wind is strong, the cricket match won't be played, or if the weather is sunny and the humidity is normal, it will be played. Similarly, the decision tree can be created according to the given dataset. To summarize, the first step is to create a decision tree with a given training data. While creating it, the entropy concept is taken into consideration to minimize the

randomness. The second step is to use the tree to classify the data. The advantage of a decision tree is that, it is easy to understand given its simplicity of representation. It can be applied to any data type, such as nominal, ordinal, numerical, etc [95], and it classifies the test data very fast [82]. Masrur et al. [3] have used DT for sentiment analysis of restaurant review.

5.3.4 Random Forest (RF)

RF is an ensemble of a DT that combines multiple decision trees to predict the output [82]. For DT, outliers or noise present in the dataset may affect the overall classification. RF classifier is resilient to noise and outliers because of the randomness it provides [82]. It provides two types of randomness, one with features and the other with the data. Parmar et al. [82] has explained the algorithm of RF as given below

Inputs: B = Number of Trees, N = Training Data, F = Total- Features, f = Subset of Features

Output: The class label for the input data.

For each tree in Forest B,

1. Select a bootstrap sample S of size N from training data.
2. Create the tree Tb by recursively repeating the following steps for each internal node of the tree.
   (a) Choose f at random from the F.
   (b) Select the best among f.
   (c) Split the node.

After B number of trees are created, test instances are passed on these trees and the class label is assigned based on the majority of the vote, which is known as a voting algorithm. Multiple hyper parameters affect the accuracy of the random forest model, such as i)Number of trees in the forest: more the trees, more the accuracy but only to an extent, after which accuracy does not increase. ii)Depth of the tree: depth of the tree also matters, if it is small then it may suffer from under fitting. and iii) the number of features to select.

5.3.5 Naïve Bayes (NB)

NB is a probability classifier, which gives the probability of document d to belong to class C. In mathematical terms, it is required to find more probable class given a document [51]. This algorithm is used because it is very simple to train and classify [29]. According to Bayes theorem [62],

$$P(s|d) = \frac{P(s)P(d|s)}{P(d)} \tag{5.3}$$

Where, P(s| d) can be pronounced as the probability of document class s, given document d. p(d) can be omitted as only computing of sentiment polarity is required. According to NB assumption, every feature's presence is independent of other features [51]. For example, consider three features for mobile phone, display, weight, and price. Presence of a feature price is independent of other features that is, weight and display.

$$Therefore, P(d|s) = \prod_{i=1}^{m} P(f_i|s) \tag{5.4}$$

where, f_1, f_2,.., f_m are the features of document that are assumed to be independent of each other.

Multinomial NB is slightly different, where different occurrences of the same word in a document are treated as separate events [62], e.g.,

$$P(d|s) = \prod_{i=1}^{m} P(f_i|s)^{n_i(d)} \tag{5.5}$$

where, P(f_i|s) is calculated by counting frequency of feature within a document.

5.3.6 Maximum Entropy (ME)

ME is one of the effective techniques of classification used in a number of NLP problems. Advantage of ME over NB is, no assumptions are made about the relationship between the features. In NB technique, features are assumed to be conditionally independent of each other. Hence, it performs better when the feature's conditional independence assumptions are not met [110]. For sentiment classification it can be equated using the

equation 5.6 [62].

$$P^*(d,s) = \pi \prod_{j-1}^{n} \alpha_j^{F_j(s,d)} \tag{5.6}$$

Where, $p^*(s,d)$ is the probability of document d, having sentiment s given a maximum entropy probability distribution [62]. Where, $\alpha_j > 0$ is the weight for the feature f_j, that is used to decide the significance of the feature in classification [29]. $F_j(s,d)$ is the feature function defines as follow [62].

$$F_j(s,d) = \begin{cases} 1, & if\ f_j\ appears\ in\ d\ with\ sentiment\ s \\ 0, & otherwise \end{cases}$$

The guiding principle here is to choose the model that makes less assumptions about the data while remaining consistent with it [62].

5.3.7 Opinion Digger

Opinion digger is an unsupervised approach introduced in [71]. It uses a set of known aspects of product and some initial guidelines such as correspondence between ratings and adjectives, say "Good" means 4 or "Excellent" means 5. Based on the rating guidelines and given aspects, aspect present in the review and their ratings are determined. It works in two phrases. In the first phrase, set of aspects are determined: it uses POS tagging for the same. It considers nouns as an aspect, hence if any noun appears frequently in the text, it will be considered as a "Potential aspect" [21]. If sentences match with a known aspect, opinion patterns are determined, then the sequence of POS tags are checked, that will express an opinion on an aspect. The frequent patterns used with known aspects are considered as opinion patterns [21]. If a review has a potential aspect noun, that matches at least two opinion aspects, it is considered an aspect.

Second phase rates the aspect. If any sentence contains an aspect, then the closest adjective is associated with that opinion. Two synonyms from the guidelines are searched in the WordNet synonym graph. The estimated rating of the aspect within opinion is the weighted average of the corresponding rating in the given guideline. "Weight is calculated by the inverse of the minimum path distance between the opinion adjective and the guideline's adjective in the WordNet hierarchy" [21]. Advantage of the unsupervised learning algorithms is this approach does not require training data.

5.4 Hybrid Approach

The hybrid approach uses multiple approaches. There are two types of a hybrid approaches, i) serial hybridization and ii)parallel hybridization. In the serial hybridization method, one phase depends upon the output of the other phase. For example, trained classifier depends upon the output of the pre processing task and in parallel hybridization, two or more methods are used to find complementary sets of aspects [105].

Popescu et al.[86] have used serial hybridization, in which Pointwise Mutual Information (PMI) is used to find the possible aspects from the given data in the first phase and then in the second phase, it feeds features into the NB algorithm to find the list of all explicit aspects. Parallel hybridization is used by Blair et al. [13]. They have used two approaches, machine learning-based and rule-based. ML based ME approach is used to find frequent features/aspects from the data, and a rule-based approach is used to find the less frequent features using syntactic patterns and frequency information [13].

Khan et al. [47] proposed a hybrid approach for the twitter dataset in which three classifiers are used, i)Enhanced Emoticon Classifier (EEC) ii) Improved Polarity Classifier (IPC) and iii) SentiWordNet Classifier (SWNC). EEC classification works on emoticons, IPC approach uses a list of positive and negative words and SWNC is based on the SentiWordNet dictionary. Three steps are involved in classifying a tweet in three class classification, that is positive, negative and neutral. First, the tweets are given to the EEC classifier. It has a set of 145 emoticons in which 70 are tagged as positive emoticons and 75 are tagged as negative. If a positive emoticon is found, it is classified in positive class,any negative emoticons, if found, are classified in negative class, otherwise they are classified in the neutral class. The tweets are then given to the IPC classifier. It has a list of 9493 positive and negative words. If it finds a positive word from the tweet then it will classify it as positive, if it finds a negative word from the tweet, it will classify it as a negative tweet or else it will classify it as neutral. After that tweet will be passed to the SWNC classifier. Word's sentiment will be calculated by the same approach used for IPC classifier.

Classification of tweets: to classify the tweet, it is passed to the EEC classifier first, then IPC classifier and lastly the tweet will be passed to the SWNC classifier. Classification formula given by [47] is as described in equation 5.7, where S_e is sentiment classified by

EEC classifier, S_w is sentiment classified by IPC classifier and S_s is sentiment classified by SWNC classifier. Consider, if all the classifiers that is, EEC classifier, IPC classifier and SWNC classifier classify the tweet as neutral then the final result will be neutral otherwise positive or negative as per the equation. This approach helps in reducing neutral tweets. Table 5.1 shows approach, method, dataset and accuracy of different classifiers.

$$Class = \begin{cases} Positive, & (S_e > 0) \vee (S_e = 0 \wedge S_w > 0) \vee (S_e = 0 \wedge S_w = 0 \wedge S_s > 0) \\ Negative, & (S_e < 0) \vee (S_e = 0 \wedge S_w < 0) \vee (S_e = 0 \wedge S_w = 0 \wedge S_s < 0) \\ Neutral, & (S_e = 0 \wedge S_w = 0 \wedge S_s = 0) \end{cases}$$

(5.7)

Approach	References	Method	Dataset	Accuracy
Lexicon Based	Abirami et al.[1]	Corpus based	Movie Review	77.6%
Machine Learning Based	Yadav et al.[132]	Naive Bayes	Cantonese-written restaurant review	85%
	Putri et al.[89]	Support Vector Machine	Grab Review dataset	89.01%
	Putri et al.[89]	Maximum Entropy	Grab Review dataset	90.46%
	Wazery et al.[125]	Neural Network	IMDB	87%
	Collomb et al.[21]	Unsupervised	Product Review	0.49 Ranking loss
	Wazery et al.[125]	Decision Tree	Amazon	81%
	Karthika et al.[46]	Random Forest	Flipkart review dataset	97%
Hybrid	Khan et al.[47]	EEC, IPC, EWNC	Tweeter Data	85.90%
Rule Based	Tung et al.[115]	-	SemEval 2013	72.3%

Table 5.1: Results of approaches in sentiment analysis

Chapter 6

Challenges of Sentiment analysis

There are multiple issues present in SA that affects or changes the polarity of the text, such as negation, sarcasm, domain dependence, to name a few. This section describes such challenges in detail.

6.1 Negation

Presence of negation in a sentence affects the polarity of other words present in the sentence. Here, one word can change the polarity of an entire sentence. For example "Bangalore's weather is very good", "Bangalore's weather is not very good". The word "not" has changed the polarity of a sentence from positive to negative. Negation may appears in two forms at higher structural level, morphological negation and syntactic negation [26]. In morphological negation, root words get modified with negative suffix or prefix known as function-words negator, such as "-less", "de-", "ir-", "il-", "dis-", "im-", "in", "miss-", "un-", "non-"to name a few [22]. Syntactic negation is most common and well known negation type where explicit words are used to invert the polarity of single word or multiple words in the sentence. Example of Syntactic negation are "no", "not", "rather", "shouldn't", "weren't", "wasn't", "didn't", "couldn't", "never", "none", "neither", "nor", "nobody", "nothing", "nowhere", "can't", "without" etc [26]. This issue can be handled using negation handling techniques. Authors have multiple techniques to handle negation. Some of them are explained in section 3.7.

6.2 Co-reference resolution

Co-reference resolution is a case in which a sentence contains multiple aspects(pronouns or subject) and a single reference. It is the problem of identifying what a pronoun or a noun phrase refers to [122]. For example, "yesterday I met Shyam and Raj, he was very shy". Here the word 'he' may refer to shyam or raj however, the reference is unclear. Co-reference resolution is useful for aspect/feature-based SA because in aspect/feature based SA there can be multiple features present in the sentence and can have single reference. For example, "I was comparing iPhone 11 and iPhone XR. it was a nice phone". In this sentence, 'it' may refers to iPhone 11 or iPhone XR.

6.3 Temporal Relations

Here, reviews or opinion changes over the period of time. For example, customer bought HP OmniBook 500 notebook in 2001 that has PIII(pentium III) processor, 256 Megabyte RAM, 30 Gigabyte hard disk drive and other features and later s/he rated it 4.5/5. Technology evolved and advanced with time. In 2019 nobody would buy HP omnibook 500 which was the best business laptop in 2001 because of outdated technology. Consider a phone review in 2013 "This phone has a very nice 5 megapixel camera" but now in 2019, the reviewer will not have the same review about it. This type of sentence may be useful in a case where we need to check the improvement in a product over a period of time [122].

6.4 Sarcasm

It is a situation where people use irony to mock other people to make a point or to be funny [64]. Sarcasm is very difficult to analyse not only for computer but for humans as well. There is a case in sarcasm where the meaning and sentiment are opposite to the literal interpretation. For example, "That movie was awesome". Normally this sentence represents positive sentiment, although sarcastically it is a negative statement. One way to calculate sentiment polarity of sarcastic statement is to calculate polarity for the statement and then reverse it, such as the above statement "That movie was great" literal polarity is positive and the reverse polarity is negative. Although, reversing the polarity does not always works. Consider a statement, "I am not happy that I woke up at 5:15

this morning.. #greatstart " [64]. Reversing the polarity of "not happy", it will become a positive statement which is not correct. Tsur et al [114] has used Amazon book review dataset and proposed a semi-supervised method to find the sarcastic statements from reviews and got 82.7% F-Score. Maynard et al [64] used hashtags to find the scope of sarcasm in tweets and got 97.25% F1-score.

6.5 Order dependence

Order can completely change the associated sentiment. Consider a sentence, "Restaurant A is preferable over restaurant B" is not similar to "Restaurant B is preferable than restaurant A". It will change the polarity attached to entities.

6.6 Subjectivity Identification

Subjectivity identification is used to identify sentences that contain an opinion or sentiment attached towards entities or event. Objective sentence represents truth or facts [57]. Subjective and objective sentences are also known as opinionated(contains opinions) and unopinionated(facts) sentence respectively. For example, "I love this laptop's Full HD display" is a subjective sentence as it has an opinion attached of the reviewer while "This laptop has a Full HD display" is an objective sentence as it is a fact or truth about the product laptop. Un-opinionated text does not contain sentiment hence, it is necessary to perform subjectivity identification that classify the sentence into opinionated(subjective) and unopinionated(objective) sentence then sentiment classification classifies the subjective sentence into polarity classes.

6.7 Comparative sentence

Many times, reviewers write reviews with the comparison of one entity or product to another. In this case, first it is necessary to identify the sentence containing comparison and then extract the feature or entity in each opinion [28]. Adjective or adverbs used in comparisons are, "less". "more","most", "higher", "lighter", "least" to name a few. For example, "phone A is lighter than phone B that makes it easier to use". With an aspect of weight, phone A has positive sentiment attached, whereas B has negative sentiment.

6.8 Domain Dependence

The same words can have a different meaning in a different domain, for example, the word "snake" refers to two domains, food and animal. Word "positive" is mostly associated with the positive polarity but in HIV or cancer reports context, it is associated with the negative polarity. Annotated dataset is limited and can't be used for all the domains [74] and for that cross domain SA can be used. Cross domain SA or domain adaption aims to train the model/classifier for the source domain, which has plenty of labelled data to a target domain, which has limited labelled data. Different methods can be used to implement cross domain SA. [40] has used supervised cross-domain method, in that classifier will be trained with the source domain samples and small number of target domain samples. [83] has proposed a novel method based on Central Moment Discrepancy (CMD) metric which is used to find the discrepancy between probability distribution of any two variables.

6.9 Thwarted Expressions

In this type of sentence, in which one part of the text can entirely change the polarity of the statement [48]. It can be defined as the phenomenon where the sentiment towards one of the important attribute/entity contradicts the sentiment towards all other attributes/entities [94]. For example, "The service was not good and the food presentation was also ok, place was not much good but I love the food it was awesome". There are four attributes/entities present in the sentence, service, presentation, place and food. Here the sentiment towards food which is an important attribute/entity for any restaurants contradict the sentiment towards all three other attributes/entities. Ramteke et al. [94] has proposed two approaches to detect thwarting, ontology based approach and SVM based machine learning approach.

6.10 World Knowledge Requirement

Sometimes knowledge about the world's events, facts, people are required to classify the text correctly. For example, "Casablanca and a lunch comprising of rice and fish: a good Sunday" [122]. Without prior knowledge that the Casablanca is a very famous movie, the sentence would be classified as positive.

6.11 Implicit feature

These are features that are not explicitly mentioned in the sentence. Consider these two sentence, 1."Phone's size is very large" and 2."Phone is not fitting into my pocket" [103]. Feature size is clearly mentioned in the first sentence but the second sentence refers to "not fitting in my pocket" and from that we can infer that the comment must be about the size. Inference is depend upon the other words of the sentence [104]. Wang et al. [124] has used hybrid association rule mining to find the implicit features from the sentence. Qiu et al. [90] has used set of implicit opinion words and explicit features. Using that they have created a co-occurrence matrix between explicit features and all the words in sentence. Using this technique they have found correct implicit feature.

Chapter 7

Recent work in SA

There are many approaches to perform the SA apart from the approaches explained in chapter 5. This chapter will describe the recent approaches that are currently used for SA, such as SA with emoticons, multiple dimension, partial textual entailment to name a few. These approaches will be explained in detail in this chapter.

7.1 Polarity Estimation of Emoticons by Polarity Scoring of Character Components

Utsu et al. [118] has proposed this approach for SA with emoticons. Emoticons are emotional icons that can be drawn with the keyboard, for example ':)' is an emoticon for smiley face, ':(' is an emoticon for sad face. People generally use the emoticons to express the feelings and they are good indicator of sentiment. Hence, using emoticons, polarized opinion can be acquired more accurately. First, extract one character immediately after '(' and immediately before ')', these characters are known as α and β respectively. They are most likely to express eyes, eyebrows or cheeks [118]. For example, consider the emoticon ':)', here the immediate character before ')'expresses eyes. Next step is to extract the character that is at the center of the emoticon. If there are odd characters in the emoticon then extract the center character and name it as both γ and δ, or if the number of characters are even then extract two character and name it as γ and δ respectively. These character are most likely to express the mouth.

Next step is to find the sentiment polarity score of the extracted characters. First, obtain the sentiment polarity of side characters(S_{side}), that is α and β. These scores are

obtained by the pre-determined value of characters. $S_{side} = (S_\alpha + S_\beta)/2$. Then, obtain the sentiment polarity of center characters, that is γ and δ. $S_{center} = (S_\gamma + S_\delta)/2$. After this step we have the polarity of both side and center characters. Now, calculate the overall polarity. Overall polarity is the average of the side polarity and the center polarity, $S_{overall} = (S_{side} + S_{center})/2$. After calculating the overall polarity, classify it using the threshold, that is if $S_overall >$ the threshold of positive, classify it as positive, same way classify the emoticons using the threshold for all the polarity classes. Two settings are used, i)maximize the positive and negative polarity and ii)maximize the overall polarity. In first setting accuracy of the neutral polarity class is very low and in the second setting overall polarity is considered hence the accuracy of the neutral polarity class is higher than in setting 1.

7.2 Joint Embedding of Emoticons and Labels

In microblogs, text and emoticons are present. Tao et al. [112] has proposed a joint embedding method for labels and emoticons using the Convolutional Neural Network(CNN) model.TO implement the model Public chinese microblog benchmark corpora NLPCC2013 & NLPCC2014 dataset is used. As shown in the figure 7.1, two encoders are used in the technique, input encoder and label encoder.

Input encoder encodes word and emoticons of each sentence and label encoder encodes the label sequence. Emoticons are transformed into the corresponding emotional word. According to the DUTIR, emotions can be divided into seven categories, happiness, like, sadness, anger, disgust, fear, surprise [112]. Two layers are there in each encoder, lookup layer and learning layer(CNN). Each sentence contains the series of words, lookup layer of input encoder converts word of each sentence into Distributed Representative Vector(DRV). Emoticons of each sentence are converted into the DRV of the corresponding emoticon type. CNN is used in learning layer for converting DRV to fix length semantic vector. Thereafter word vector and emoticon vector of each sentence is concatenated to form the final semantic vector of fix length. Label encoder does the same job as input encoder for a label sequence. Matcher will compare both the vectors obtained from input encoder and label encoder and find the matching score between them. Advantage of the approach is, original SA task is converted into vector comparison task. Acquired accuracy for different section is as given, subjectivity classification - 85%, polarity classification -

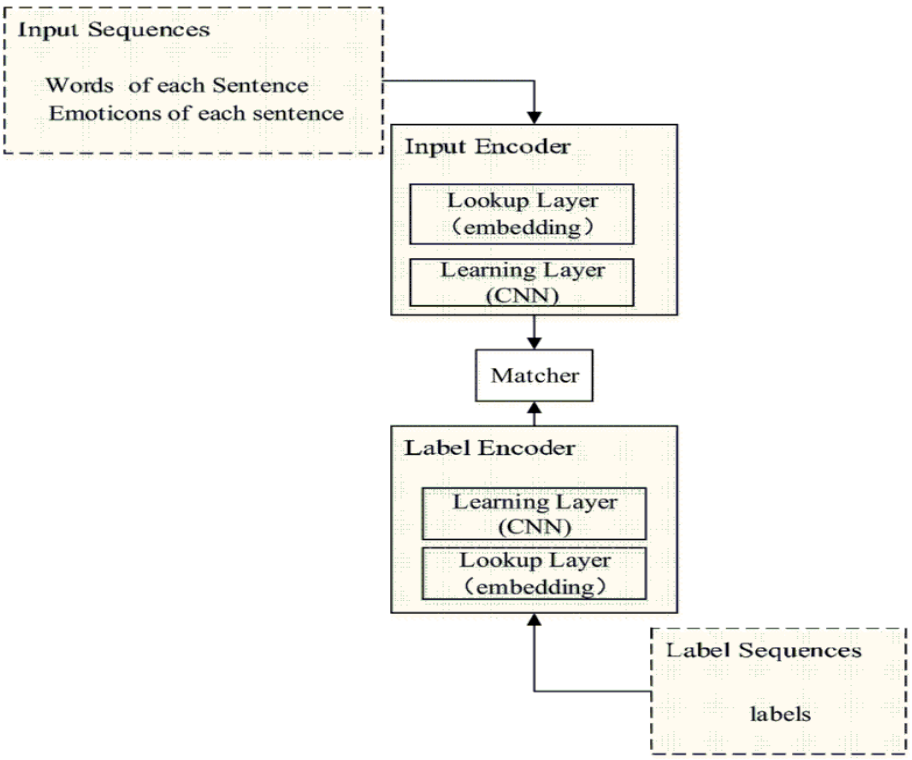

Figure 7.1: Flow of the Joint embedding approach [112]

88%, emoticon classification - 72%.

7.3 Sentiment analysis in multiple dimensions using sentiment compensation

Porntrakoon et al. [87] has proposed the method to analyze the consumer's reviews on multiple dimensions that are product, price and shipping. Thai sentiments are used for the analysis purpose. Dataset used is 'Thai consumers' reviews from Lazada Thailand website'. Longest word matching algorithm is used with the dictionary based approach. The general flow of the approach is as shown in the figure 7.2.

First step is to extract the Thai reviews. Then the text is getting segmented in sentence using the blank spaces. Third step is the text cleaning, where special characters and English symbols are getting deleted to ensure that only Thai characters and numbers are present. After cleaning the text, tokenization is performed with the longest matching algorithm using Lexitron dictionary. Thereafter each tokenized words are analyzed to examine whether the word belongs to the product dimension, shipping dimension or price dimension, then the next step is to check the polarity of each word for their respective dimension. Finally -1, 0, or 1 score is assigned to them. There are some scenarios

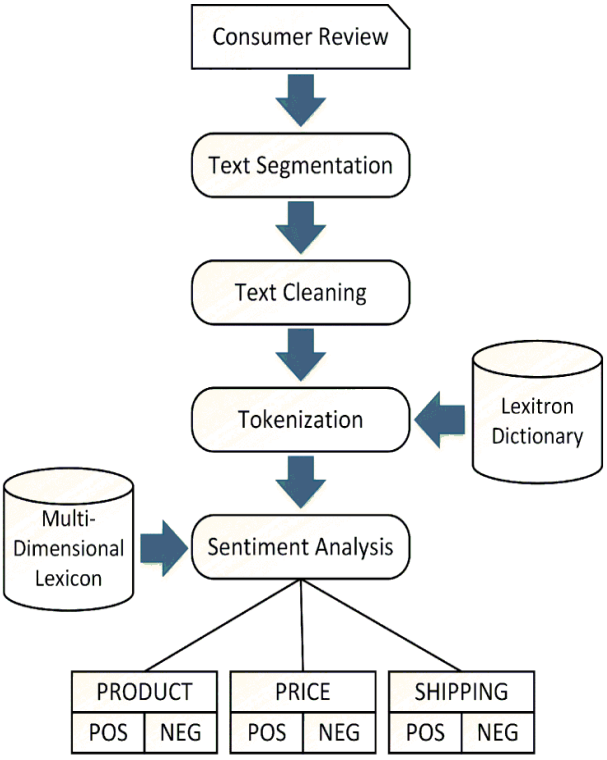

Figure 7.2: Flow of the sentiment compensation technique [87]

where the consumer review the entity without specify the entity. For example, 'very fast' is a review where the entity is not mentioned. Sentiment compensation(SensComp) is a technique that will automatically assign the dimension to the review if it is not explicitly mentioned. For example, 'very fast' refers to the speed, that is related to the shipping dimension, hence it will be assigned the review to the shipping dimension. Using the SensComp technique with the pre-defined dimensions, total accuracy of 93.60% is acquired [87].

7.4 SA using Partial Textual Entailment

Textual Entailment(TE) is a relationship between two text snippets, named text(T) and hypothesis(H). it is used to check where or not H can be concluded from the T. Gupta et al. [35] has proposed the approach of sentiment analysis using Partial Textual Entailment(PTE). TE is not much efficient because of the some reasons such as, complexity of the sentences and limitation of the linguistic resources. Hence, PTE is used to improve the task of SA. Here, partial entailment means, if a part of hypothesis is entailed by the text or not. Facets are used to perform the PTE for SA. First step is to decompose the hypothesis into the facets, that is an ordered pair of words. Second step s to decompose

the text into facets and check whether the facets of hypothesis is entailed by the facets of the text or not. Then, aggregate the matching score to verify the complete entailment.

Further, three methods are there to match the entailment. i)Exact match method: All words or lemma of hypothesis is entailed by the text. ii)Lexical inference method: Here, resnik similarity measure is used to estimate/measure the similarity present between the ordered pair of hypothesis and text. iii)Semantic inference method: This method is used to determine whether the meaning of one sentence is been entailed by the other or not. Using any of the three method PTE can be calculated. Table 7.1 shows the summary of all the explained approaches.

Approach	Reference	Method	Dataset	Accuracy	Advantage	Disadvantage
SA with emoticons	Utsu et al. [118]	Front faced emoticon-based approach having two settings. 1. Maximize accuracy of negative and positive polarity. 2. Maximize the overall polarity	Google japanese Input Emoticons	Setting 1: 79.6% Setting 2: 81.6%	Easy, Using emoticons, sentiments can be strongly expressed	Only front faced emoticons is used, Neutral emoticons polarity is very low
	Tao et al. [112]	Emoticons and Labels based on CNN (EL-CNN)	Public chinese microblog benchmark corpora NLPCC2013 & NLPCC2014	Subjectivity Classification - 85%, Polarity classification - 88%, Emoticon classification - 72%,	It converts original sentiment analysis task into vector comparison task	Emoticon classification accuracy is low
SA with multiple dimension	Porntrakoon et al. [87]	Multiple Dimensions Using Sentiment Compensation Technique (SenseComp)	Thai consumers' reviews from Lazada Thailand website	93.60% accuracy	Automatically assign dimension to sentiment if not mentioned in the sentence	It classifies based on three dimensions(Product, price, shipping) only
SA with partial texual entailment	Gupta et al.[35]	Partial Textual Entailment	None	None	Computational overhead of TE will be reduced	Creation of hypothesis is critical, It will not classify related text if PTE does not match with hypothesis

Table 7.1: Summary of the recent approaches in SA

Chapter 8

Applications of SA

SA can be useful in multiple applications to analyze the user's sentiment. This section describes the different applications such as recommendation system, summarization of reviews, medical data analysis, political SA, business applications of SA, stock market prediction, and music streaming using SA.

1. Recommendation system

   Recommendation system is used to recommend products that is similar to the user's interest [76]. SA can be useful to extract sentiment of user for any particular product, article, etc. Using the sentiment information, user statistic can be built that can be used to recommend any item to a particular user. Nabil et al. [76] have used SA in recommendation system using Apache Spark to process a large volume of data in distributed manner. They classify the data into 2 classes, negative and positive to check the user's likes and dislikes.

2. Summarization of reviews

   Text summarization is a process of extracting and interpreting useful data from a very large document [132]. There are multiple websites that provide summarization of reviews, such as Google product search. SA can be used for the same. Automated summarization of reviews helps in identifying the characteristics of a service or product without reading the complete document [78]. Alsaqer et al. [5] have used SA in movie review summarization with RapidMinor. There are two modules named SA module and summarize module, SA module finds the polarity of review

and summarize module summarizes the review in limited number of lines.

3. Medical data analysis

   Large volume of medical data is available on the internet today. Doctors and patients share health related opinions online [32]. SA can be used in this field for a number of applications such as feedback on a drug/physician, adverse drug effect, to name a few [67]. Gohil et al. [32] have reviewed tools and softwares used for health care data. Meena et al. [67] have used SA for a particular disease, Cancer, on three social media platforms namely twitter, google trends, and online forums.

4. Political SA

   SA can be used to mine political views online to predict the election outcome or to check the person's review or sentiment regarding a particular party/candidate [7]. Elghazaly et al. [24] have proposed SA of Arabic political data using SVM and NB, Caetano et al. [16] have used SA to find the homophily of user's classes during 2016 American presidential election.

5. Business Applications of SA

   Businesses have to understand their customer's needs and feedback for their products [49]. Online product reviews can help with the same. Manual approach is tedious, time consuming and inefficient, and SA can be used to automate the process. There are some fundamental questions to which organizations seek an answer, as stated by Seth Grimes [34].

   - Whether the customers are satisfied with the product, service and support
   - Customer's review about similar products in the market, their service and support
   - Customer's likes, dislikes and the improvements they expect in a product.
   - Additional features required by the customers
   - Customers that contribute to their profits the most

- Best selling product and their reviews

SA can analyze the reviews, comments and can help to find the answers to the organization's questions.

6. Stock market prediction

Stock market prediction is a field of finance, maths and engineering and it is a subject/topic of interest for many investors and financial analyst [10]. Stock market forecasting can be used to predict the future value of the company stock. Public's positive or negative outlook for any organization can have a ripple effect on the stock price of that organization [61]. SA can be used to predict the buy/sell signal for the investor [10]. Mankar et al. [61] have used SVM to predict the future stock price. Mohan et al. [73] have used the correlation between SA and published news articles to predict the stock price. They have used five years of SP500 company's dataset along with more than 265000 news articles of these companies.

7. Music Streaming

Multiple streaming platforms provide music steaming such as Spotify, Apple Music, Google Play Music, Amazon Music, Gaana, to name a few. Songs can be classified into different categories by genres, albums, or singers. These applications can recommend the music that the use might be interested in using SA. Choi et al [20] have used lyrics to classify the music using SA. Music lyrics are in textual format and hence, contain human emotion. Music can be classified using the sensibility of the text [20]. They have proposed the approach to recommend next song using the k-nearest neighbor with the most similar lyrics.

Chapter 9

Proposed approach

This chapter explains the improvisation of the existing approach using the limitation present in the approach to increase the efficiency or the scope of the existing approach.

9.1 Fuzzy Sentiment Phrases(FSP) approach

FPS approach focuses on each type of sentence separately, for example consider a statement "The book is not too bad". Here, the sentiment phrase 'too' is not clear. In the most cases this sentence will be considered as a positive sentence with words 'not bad' and the effect of too will be ignored. FSP considers these type of word to accurately express the sentiment present in the sentence.

9.1.1 Overview of FSP approach

Phan et al. [85] has proposed this approach that will focus on the each type of sentence. The approach is that each sentence contains three types of words, i)fundamental sentiment word, ii)negation word and iii)fuzzy semantic word. Consider a sentence "The movie was relatively good". Fundamental sentiment word includes negative and positive polarity words, such as 'good' is a fundamental word in the sentence. Negation words can be present or absent in the sentence, in the above sentence the it is absent. Fuzzy semantic word can be of two types, i)intensifier or ii)diminisher. Intensifier intensify the sentiment polarity and diminisher decreases/reduces the polarity. For example in fine-gained polarity class, five classes are there namely very negative, negative, neutral, positive and very positive. Suppose if the classifier is supposed to classify the tweet as positive, it will be classified as very positive after considering the intensifier or it can

be classified as neutral if the diminisher is present. Consider the above sentence "The movie was relatively good", here, if we ignore the word 'relatively', the sentence will be classified as positive but if we consider the effect of the word 'relatively' then it will be classified as neutral.

There are two categories of the FSP. i)Type 1: one fundamental word and one fuzzy semantic word. For example, 'relatively good'. ii)Type 2: One fundamental word, one negation word, and one fuzzy semantic word, for example 'not so bad'. Score of the FSP can be calculated as shown in the equation 9.1 [85]

$$w_f = (-1)^i((-1)^k w_{p1} + (-1)^j w_{p2}) \tag{9.1}$$

here, p_1 is the fundamental sentiment word, p_2 is the fuzzy semantic phrase, w_{p1} and w_{p2} is the sentiment score of the fundamental sentiment word and fuzzy semantic word respectively. k is the indicator of the FSP type, k=1 if FSP type is 2 and k=2 if FSP type is 1. i is the indicator of whether the word is positive or negative, i=1 if p_1 is negative, i=2 if p_1 is positive. j is the indicator of whether the fuzzy semantic word is intensifier or diminisher, j=1 if p_2 is diminisher, j=2 if p_2 is intensifier. w_f is the total sentiment score of the sentence containing FSP. Figure 9.1 shows the overall flow of the FSP approach.

9.1.2 Improvements in the existing FSP approach

The FSP approach will first detect if the tweet contains FSP or not. For that Window approach is used. Here, the window size is three, that is in the entire sentence if there is any word sequence that matches any type of FSP. Here, the limitation is the window size. If any sentence containes FSP but it doesn't fit into the window size of three, it will be classified as tweets not containing FSP which is incorrect. Consider a tweet "The book is not like extremely good", here the intensifier 'extremely' is present in the sentence but it doesn't fit into the defined window size hence, it will be ignored which should not be the case. Here I've increase the window size to 5 to not miss any sentence that containes FSP.

Another improvisation that can be addressed is sarcasm detection. Sometimes user tweets with sarcasm. For example, "The movie was so awesome that i slept watching it #sarcasm". Here classifier will classify the tweet as very positive because of the sentiment "so awesome" but it is a sarcastic tweet, hence it should be classified as a negative.

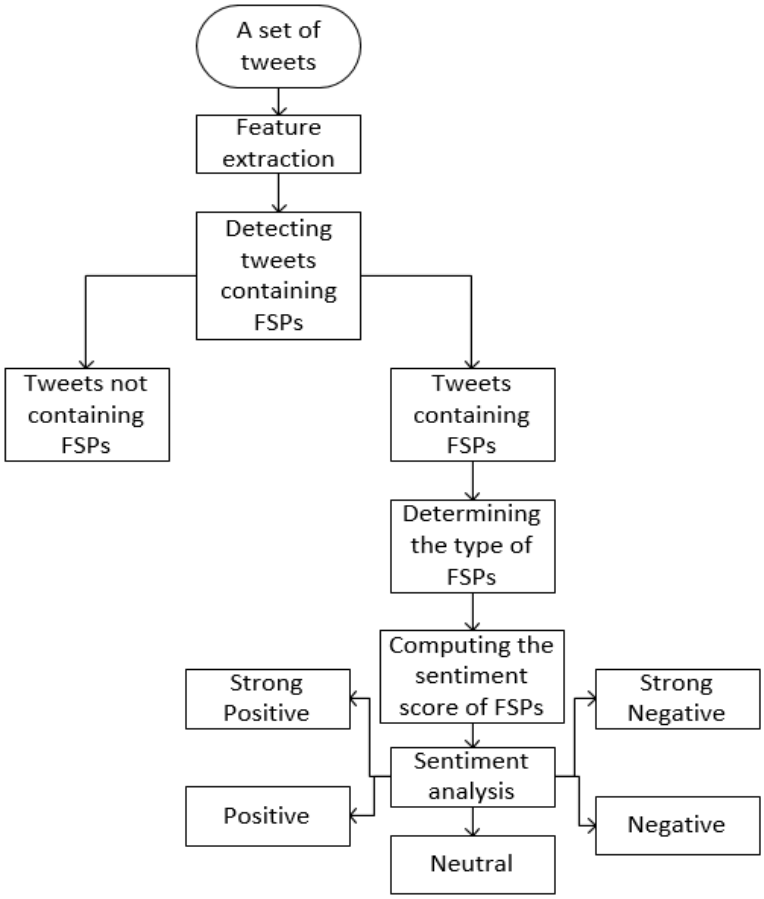

Figure 9.1: Fuzzy sentiment approach flowchart [85]

Maynard et al. [64] has proposed a techniques where tweets will be classified as sarcastic tweet if there is a token '#sarcasm' or '#sarcastic', but this case is not always possible that the sarcastic tweets contains '#sarcasm' or '#sarcastic' tag. Hence it is necessary to have an algorithm to detect the sarcastic tweet. Three steps are there to detect the sarcastic tweets.

1. Detect sarcasm with '#sarcasm' or '#sarcastic' tag

   If these tags are present in the tweet then the tweet must be a sarcastic tweet.

2. Detect sarcasm with polarity contradiction between words and emoticons

   This method divides the tweet in text and emoticon parts. First step is to extract the text part and then find out the emotion conveyed using the word based emotion detection, then extract the emoticons present in the tweet and find out the emotion conveyed by emoticons using emoticon based emoticon detection. If both

the emotion contradict to each other then the tweet can be classified as sarcastic tweet [96]. For example the emotion conveyed by text is negative and the emotion conveyed by emoticon is positive then the tweet is sarcastic tweet

3. Detect sarcasm with two halves of sentence

   It is noticed that if the sentence starts with one intent and ends with another intent then the sarcasm can be present in the sentence [2]. To find the sarcasm with this method, first divide the sentence in two halves, then find the positive and negative words in each half. If the negative word in the second half is greater than the positive word in the first half or the negative word in the first half is greater than the positive word in the second half then the tweet will be classified as sarcastic tweet.

If the sarcasm is present in the tweet, the polarity will be reversed for such sentences, for example, the above sentence will be classified as very positive and the sarcasm is present in the sentence then the polarity will be reversed to very negative.

9.1.3 Experiment setup and result

Python package Tweepy is used to collect the 3200 English tweets. These tweets contains many different topics. After extracting tweets, pre-processing techniques will be applied such as, removing punctuation mark, retweet symbol and URLs, spelling correction. Then the Multilayer Perceptron(MLP) model will be used with window size and scaled exponential linear units (SELU) activation function to detect whether the tweets contains FSP or not. After screening the tweets, if it contains FSP then sarcasm detection will be performed and tweets classified with sarcasm will be tagged. Then the type of the sentiment will be determined, after that the sentiment score of the tweet will be determined as shown in the equation 9.1. Here, the scale of -1 to 1 is used to classify the tweets, it will be classified as shown in the equation 9.2 [85]. While classifying the tweets, if the tweet contains sarcasm then the polarity of those tweets will be reversed. for example, if the polarity score is 0.7 and it contains sarcasm then the score will be -0.7. Overall accuracy of 81.9% is acquired with the improvement in the approach that

was 78.6% before.

$$Polarity = \begin{cases} VeryPositive, & if(0.6 < w_f < 1) \\ Positive, & if(0.2 < w_f < 0.6) \\ Neutral, & if(-0.2 < w_f < 0.2) \\ Negative, & if(-0.6 < w_f < -0.2) \\ Verynegative, & if(-1 < w_f < -0.6) \end{cases} \tag{9.2}$$

9.2 Aspect-based sentiment analysis via constructing auxiliary sentence

Aim of aspect based SA is to handle the sentence with multiple targets. For example, 'This phone has a very good camera but the display is not good'. Here, there are two aspects present in the sentence, i)camera and ii)display. To handle this type of sentence aspect based SA is used.

9.2.1 Overview of the approach

Sun et al. [109] has proposed the approach of Aspect Based Sentiment Analysis(ABSA) using auxiliary sentence. To implement the approach, Targeted ABSA that is TABSA is used which is an approach with known targets. In any sentence there are number of words, in that some words are pre-defined targets. Three things will be taken as an input that is, sentence s, set of target entities and a fixed aspect set general, safety, price, transit-location. As an output, sentiment polarity will be predicted for all the pair of target-aspect pair. Here for classification, 3 class classification is used. Using the auxiliary sentence approach TABSA task can be converted to sentence-pair classification task. Here, the sentence pair will be target aspect pair. For the sentence "In LOCATION1 price is less but it is not safe", auxiliary sentences will be LOCATION1-price and LOCATION1-safety, that is a target aspect pair, where the target is Location 1 and the aspects are price and safety.

9.2.2 Improvements in the existing TABSA approach

Here, the targeted aspect set is fixed, there are only four aspects that will be detected from any sentence that are general, safety, price, transit-location. If any sentence contains

any other aspect, then it will not be detected by this approach. For example, consider a sentence 'This car is awesome, it is under budget, cabin is comfortable and most importantly the engine is excellent'. Here there are three aspects, price, cabin and engine, but with the current approach it will consider only one that is price, because the other features/aspects are not there in the predefined aspect list. As the improvement the approach should work on any aspects present in the sentence, the same way it worked for the other pre-defined aspects.

9.2.3 Experimental setup and result

SemEval-2014 dataset is used. The following steps are performed to implement the approach. i)Given a sentence s and the target t in the sentence, detect the mention of an aspect. For example consider a sentence, "In LOCATION1 price is less but it is not safe", here the target is LOCATION1 and the aspects are price and safety, the result of this step will be 'LOCATION1-price', 'LOCATION1-safety'. ii)Determine the polarity word for detected target-aspect pair. Polarity word for aspect 'price' and 'safety' is 'less' and 'not safe' respectively. The output of this step will be LOCATION1-price-less, LOCATION1-safety-not safe. iii)Determine positive or negative polarity from the auxiliary sentence using pre-trained BERT model. The output will be LOCATION1-price-less: Positive and LOCATION1-safety-not safe: Negative. Here, the polarity of each aspect will be determined by the softmax function. The accuracy result of the initial approach is 85.2%. Improvement of the approach was suggested to increase the scope of the paper. After applying the improvements, 83.7% accuracy is obtained.

9.3 Enhanced Naïve Bayes classification approach

Narayanan et al.[77] has proposed an enhanced naive Bayes classification approach, in which they have used Bernoulli NB, laplacian Smoothing, negation handling, n-gram features and feature selection with mutual information. Bernoulli Naïve Bayes is used, classify the data into two classes. Naïve Bayes classifier has zero probability problem. It assumes that all the features are independent of each other, it multiplies the probability of each feature. Hence, if a feature has probability zero(not seen in the training data set) then the product of the probability of all the features would be zero. To overcome this problem, laplacian smoothing is used by [77]. Negation handling was one of the factors

that contributed significantly to the accuracy of the classifier. It transforms the word followed by n't or not into "not_" + word until the punctuation mark or double negation is found. an n-gram is used to increase the probability of document to be negatively or positively classified. For example, the word " definitely" doesn't provide the sentiment on its own, but " definitely try" will increase the probability of a text to be negatively or positively biased. In feature selection, redundant features are being removed and those features will be retained which has high disambiguation capabilities. Mutual information was used to select the feature.

9.3.1 Improvements in the existing approach

The existing approach can be further improved by using other pre-processing and feature selection techniques, as Naïve Bayes is fast and can be more accurate if used with proper techniques. I have used stop word removal, punctuation removal, and replace elongated words apart from negation handling. Stop word removal will remove unnecessary words from the document. Some users put so many punctuation marks, which will not help to classify a document. Elongated words will be replaced by its root word to reduce the number of features. For example, "happy" and "happyyyyyyyy" are the same word which will be considered as a two different feature without this pre-processing step.

Symmetrical Uncertainty is used as a feature selection technique. It will check the information gain of the correlation between feature, Higher the gain, more useful the feature. Then we can select top features among the list of features. Flowchart of the approach is in figure 9.2.

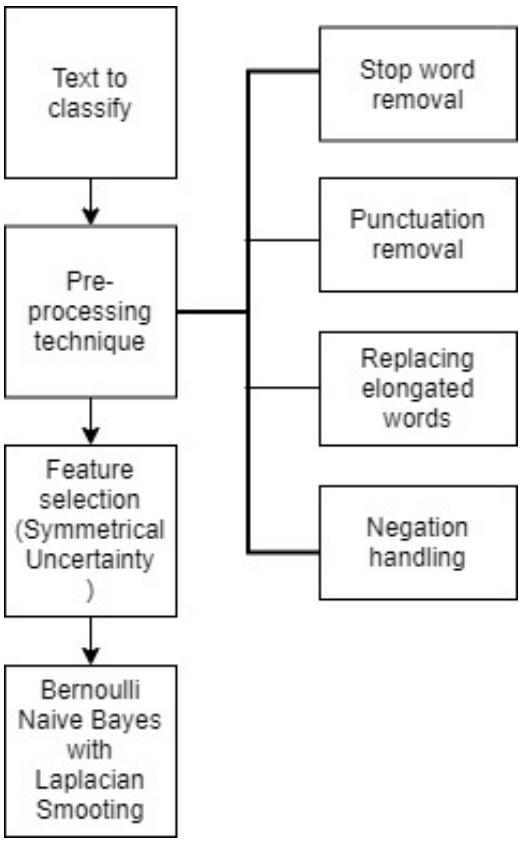

Figure 9.2: Flowchart of the approach

9.3.2 Experimental setup and result

IMDB online movie review dataset is used. Initially pre-processing techniques will be applied to the text to remove noise from the dataset. IMDB dataset has a list of positive and negative words. Each word in the document will be compared with the list and initial count of positive and negative word will be noted. Now all the word will passed to feature selection technique(Symmetrical Uncertainty), which will sort the text based on the higher to lower information gain. Then classification of text is performed using Bernoulli Naïve Bayes approach with the top k features, then data will be plotted with different values of K

The plot of accuracy vs number of features, for existing approach and proposed approach is in figure 9.3 and figure 9.4.

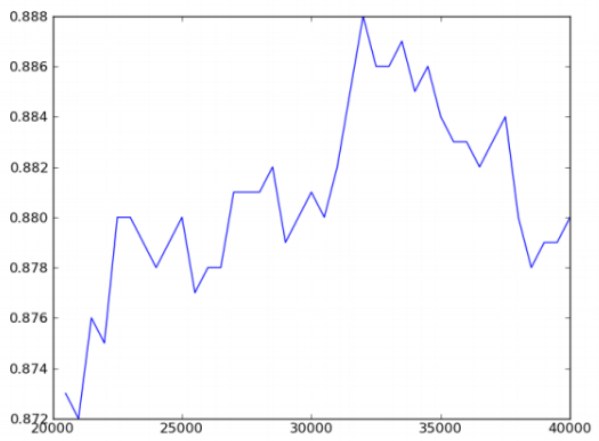

Figure 9.3: Accuracy vs number of features - existing

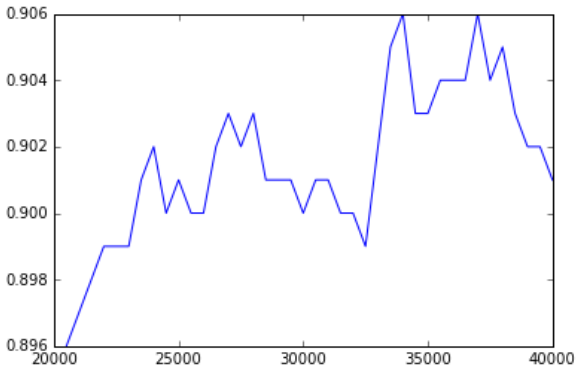

Figure 9.4: Accuracy vs number of features - proposed

Using these pre-processing and feature selection techniques, we can increase the accuracy of the classifier to 2% (88.8% to 90.6%).

Chapter 10

Result and Conclusion

There are multiple ways through which SA can be performed by using levels, pre-processing techniques, feature selection techniques and approaches. We have seen different levels such as, document level, sentence level, aspect/feature level, word level, and concept level SA. Thereafter, various pre-processing techniques are discussed and the generated token is compared on applying different pre-processing techniques used by different authors on SS-Twitter dataset. It is found that stemming, stop word removal, POS tagging, lower casing and replacing the elongated word are the most commonly used pre-processing techniques. Later, types of features, importance of feature selection and different techniques for the same are discussed. Approaches of SA include lexicon based approach, rule based approach, ML based approach and hybrid approach. There are challenges in SA, that may classify the text incorrectly and can affect the accuracy of the overall classification also, we have discuss the various applications of SA. Here, from the survey we can conclude that RF, ME and SVM are the most accurate approach. This is an overview of SA and any of the techniques/approaches can be used, according to the suitability of the application. We saw that in FSP approach, accuracy can be increased with increased window size and sarcasm detection. We have also increased the scope of aspect based sentiment analysis with working on all the present aspect in the sentence rather than pre-defined aspects. We have also improve the accuracy of enhanced NB model by using pre-processing techniques. Table 10.1 shows the summary and result of all the proposed approach. Here the limitation describes the limitation in the actual approach.

Approach	Reference	Method	Dataset	Accuracy	Simulated accuracy	Improved accuracy	Advantages	Limitations
Fuzzy Sentiment phrase	Phan et al. [85]	Divide and conquer	Twitter dataset	80%	79.8% with 10 fold-78.6%	With window approach 80.9%,with 10 fold - 79.3% with sarcasm approach-81.9% (with 10 fold)	Focus on each type of sentence, use of intensifier and diminisher	Fuzzy feature extraction
Aspect based sentiment analysis	Sun et al. [109]	Utilizing BERT via constructing auxiliary sentence	SentiHood dataset	85.2% accuracy	85.2% accuracy(10 fold)	83.7% accuracy(10 fold)	Auxiliary sentence reduces the computational cost	Target and aspects are fixed
Enhanced NB classification	Narayanan et al. [77]	Naive Bayes	IMDB dataset	88.8%	88.4% (10-fold)	90.6%(10-fold)	Efficiency of the existing approach is increased by using bernoulli NB, laplacian smoothing, negation handling, n-gram features and feature selection with mutual information	Only negation handling is used as a preprocessing technique

Table 10.1: Summary and result of the proposed approaches

Bibliography

[1] AM Abirami and V Gayathri. "A survey on sentiment analysis methods and approach". In: *2016 Eighth International Conference on Advanced Computing (ICoAC)*. IEEE. 2017, pp. 72–76.

[2] MJ Adarsh and Pushpa Ravikumar. "Sarcasm detection in Text Data to bring out genuine sentiments for Sentimental Analysis". In: *2019 1st International Conference on Advances in Information Technology (ICAIT)*. IEEE. 2019, pp. 94–98.

[3] M. Adnan, R. Sarno, and K. R. Sungkono. "Sentiment Analysis of Restaurant Review with Classification Approach in the Decision Tree-J48 Algorithm". In: *2019 International Seminar on Application for Technology of Information and Communication (iSemantic)*. 2019, pp. 121–126.

[4] Apoorv Agarwal et al. "Sentiment analysis of twitter data". In: *Proceedings of the Workshop on Language in Social Media (LSM 2011)*. 2011, pp. 30–38.

[5] Alaa F Alsaqer and Sreela Sasi. "Movie review summarization and sentiment analysis using rapidminer". In: *2017 International Conference on Networks & Advances in Computational Technologies (NetACT)*. IEEE. 2017, pp. 329–335.

[6] Giulio Angiani et al. "A Comparison between Preprocessing Techniques for Sentiment Analysis in Twitter." In: *KDWeb*. 2016.

[7] Akshat Bakliwal et al. "Sentiment analysis of political tweets: Towards an accurate classifier". In: Association for Computational Linguistics. 2013.

[8] Alexandra Balahur. "Sentiment analysis in social media texts". In: *Proceedings of the 4th workshop on computational approaches to subjectivity, sentiment and social media analysis*. 2013, pp. 120–128.

[9] Jorge A Balazs and Juan D Vel squez. "Opinion mining and information fusion: a survey". In: *Information Fusion* 27 (2016), pp. 95–110.

[10] Shri Bharathi and Angelina Geetha. "Sentiment analysis for effective stock market prediction". In: *International Journal of Intelligent Engineering and Systems* 10.3 (2017), pp. 146–153.

[11] Parminder Bhatia, Yangfeng Ji, and Jacob Eisenstein. "Better document-level sentiment analysis from rst discourse parsing". In: *arXiv preprint arXiv:1509.01599* (2015).

[12] Maryam Bibi. "Sentiment Analysis at Document Level". In: (Oct. 2017).

[13] Sasha Blair-Goldensohn et al. "Building a sentiment summarizer for local service reviews". In: (2012).

[14] Marina Boia et al. "A:) is worth a thousand words: How people attach sentiment to emoticons and words in tweets". In: *2013 International Conference on Social Computing*. IEEE. 2013, pp. 345–350.

[15] Kalina Bontcheva et al. "Twitie: An open-source information extraction pipeline for microblog text". In: *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*. 2013, pp. 83–90.

[16] Josemar A Caetano et al. "Using sentiment analysis to define twitter political users' classes and their homophily during the 2016 American presidential election". In: *Journal of Internet Services and Applications* 9.1 (2018), pp. 1–15.

[17] Jose Camacho-Collados and Mohammad Taher Pilehvar. "On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis". In: *arXiv preprint arXiv:1707.01780* (2017).

[18] Erik Cambria. "An introduction to concept-level sentiment analysis". In: *Mexican international conference on artificial intelligence*. Springer. 2013, pp. 478–483.

[19] Iti Chaturvedi et al. "Distinguishing between facts and opinions for sentiment analysis: Survey and challenges". In: *Information Fusion* 44 (2018), pp. 65–77.

[20] Jinhyuck Choi, Jin-Hee Song, and Yanggon Kim. "An analysis of music lyrics by measuring the distance of emotion and sentiment". In: *2018 19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*. IEEE. 2018, pp. 176–181.

[21] Anaıs Collomb et al. "A study and comparison of sentiment analysis methods for reputation evaluation". In: *Rapport de recherche RR-LIRIS-2014-002* (2014).

[22] Isaac G Councill, Ryan McDonald, and Leonid Velikovich. "What's great and what's not: learning to classify the scope of negation for improved sentiment analysis". In: *Proceedings of the workshop on negation and speculation in natural language processing*. Association for Computational Linguistics. 2010, pp. 51–59.

[23] Dimitrios Effrosynidis. *Mapping of slangs and abbreviation*. Last Accessed:28-04-2019. URL: https://github.com/Deffro/text-preprocessing-techniques/blob/master/slang.txt (visited on 09/30/2010).

[24] Tarek Elghazaly, Amal Mahmoud, and Hesham A Hefny. "Political sentiment analysis using twitter data". In: *Proceedings of the International Conference on Internet of things and Cloud Computing*. 2016, pp. 1–5.

[25] Nikos Engonopoulos et al. "ELS: a word-level method for entity-level sentiment analysis". In: *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*. ACM. 2011, p. 12.

[26] Umar Farooq et al. "Negation Handling in Sentiment Analysis at Sentence Level." In: *JCP* 12.5 (2017), pp. 470–478.

[27] Usama M Fayyad, Gregory Piatetsky-Shapiro, and Ramasamy Uthurusamy. "Summary from the KDD-03 panel: data mining: the next 10 years". In: *ACM Sigkdd Explorations Newsletter* 5.2 (2003), pp. 191–196.

[28] Ronen Feldman. "Techniques and applications for sentiment analysis." In: *Commun. ACM* 56.4 (2013), pp. 82–89.

[29] Geetika Gautam and Divakar Yadav. "Sentiment analysis of twitter data using machine learning approaches and semantic analysis". In: *2014 Seventh International Conference on Contemporary Computing (IC3)*. IEEE. 2014, pp. 437–442.

[30] Geeeks for Geeks. *Supervised and unsupervised techniques of machine learning.* URL: https://www.geeksforgeeks.org/supervised-unsupervised-learning/ (visited on).

[31] Anastasia Giachanou and Fabio Crestani. "Like it or not: A survey of twitter sentiment analysis methods". In: *ACM Computing Surveys (CSUR)* 49.2 (2016), p. 28.

[32] Sunir Gohil, Sabine Vuik, and Ara Darzi. "Sentiment analysis of health care tweets: review of the methods used". In: *JMIR public health and surveillance* 4.2 (2018), e43.

[33] Balakrishnan Gokulakrishnan et al. "Opinion mining and sentiment analysis on a twitter data stream". In: *International Conference on Advances in ICT for Emerging Regions (ICTer2012)*. IEEE. 2012, pp. 182–188.

[34] Seth Grimes. "Voice of the Customer, Text Analytics for the Responsive Enterprise". In: *Business Intelligence Network* (2008).

[35] Shailja Gupta, Sachin Lakra, and Manpreet Kaur. "Sentiment Analysis using Partial Textual Entailment". In: *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*. IEEE. 2019, pp. 51–55.

[36] Emitza Guzman and Walid Maalej. "How do users like this feature? a fine grained sentiment analysis of app reviews". In: *2014 IEEE 22nd international requirements engineering conference (RE)*. IEEE. 2014, pp. 153–162.

[37] Zhang Hailong, Gan Wenyan, and Jiang Bo. "Machine learning and lexicon based methods for sentiment classification: A survey". In: *2014 11th Web Information System and Application Conference*. IEEE. 2014, pp. 262–265.

[38] Mark A Hall and Lloyd A Smith. "Practical feature subset selection for machine learning". In: (1998).

[39] Yulan He, Chenghua Lin, and Harith Alani. "Automatically extracting polarity-bearing topics for cross-domain sentiment classification". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics. 2011, pp. 123–131.

[40] Judy Hoffman et al. “Asymmetric and category invariant feature transformations for domain adaptation”. In: *International journal of computer vision* 109.1-2 (2014), pp. 28–41.

[41] Robert C Holte. “Very simple classification rules perform well on most commonly used datasets”. In: *Machine learning* 11.1 (1993), pp. 63–90.

[42] Minqing Hu and Bing Liu. “Mining and summarizing customer reviews”. In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2004, pp. 168–177.

[43] Doaa Mohey El-Din Mohamed Hussein. “A survey on sentiment analysis challenges”. In: *Journal of King Saud University-Engineering Sciences* 30.4 (2018), pp. 330–338.

[44] Hasna Ighazran, Larbi Alaoui, and Tarik Boujiha. “Metaheuristic and Evolutionary Methods for Feature Selection in Sentiment Analysis (a Comparative Study)”. In: *2018 International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*. IEEE. 2018, pp. 1–6.

[45] Zhao Jianqiang. “Pre-processing boosting Twitter sentiment analysis?” In: *2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*. IEEE. 2015, pp. 748–753.

[46] P Karthika, R Murugeswari, and R Manoranjithem. “Sentiment Analysis of Social Media Network Using Random Forest Algorithm”. In: *2019 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*. IEEE. 2019, pp. 1–5.

[47] Farhan Hassan Khan, Saba Bashir, and Usman Qamar. “TOM: Twitter opinion mining framework using hybrid classification scheme”. In: *Decision Support Systems* 57 (2014), pp. 245–257.

[48] Vishal Kharde, Prof Sonawane, et al. “Sentiment analysis of twitter data: a survey of techniques”. In: *arXiv preprint arXiv:1601.06971* (2016).

[49] MOHAMMED AL-KHARUSI, Abubakar Usman, and Jamilu Awwalu. “Application of Sentiment Analysis in Business Intelligence”. In: *International Journal of Knowledge, Innovation and Entrepreneurship* 3 (Jan. 2015), p. 51.

[50] Soo-Min Kim and Eduard Hovy. "Automatic identification of pro and con reasons in online reviews". In: *Proceedings of the COLING/ACL on Main conference poster sessions*. Association for Computational Linguistics. 2006, pp. 483–490.

[51] Filip Knyszewski. *Implementing Naive Bayes for Sentiment Analysis*. URL: https://medium.com/datadriveninvestor/implementing-naive-bayes-for-sentiment-analysis-in-python-951fa8dcd928 (visited on).

[52] Olga Kolchyna et al. "Twitter Sentiment Analysis: Lexicon Method, Machine Learning Method and Their Combination". In: July 2015.

[53] Seema Kolkur, Gayatri Dantal, and Reena Mahe. "Study of different levels for sentiment analysis". In: *International Journal of Current Engineering and Technology* 5.2 (2015), pp. 768–770.

[54] Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore. "Twitter sentiment analysis: The good the bad and the omg!" In: *Fifth International AAAI conference on weblogs and social media*. 2011.

[55] Akrivi Krouska, Christos Troussas, and Maria Virvou. "The effect of preprocessing techniques on Twitter sentiment analysis". In: *2016 7th International Conference on Information, Intelligence, Systems & Applications (IISA)*. IEEE. 2016, pp. 1–5.

[56] Chenghua Lin and Yulan He. "Joint sentiment/topic model for sentiment analysis". In: *Proceedings of the 18th ACM conference on Information and knowledge management*. ACM. 2009, pp. 375–384.

[57] Bing Liu. "Sentiment analysis: A multi-faceted problem". In: *IEEE Intelligent Systems* 25.3 (2010), pp. 76–80.

[58] Bing Liu and Lei Zhang. "A survey of opinion mining and sentiment analysis". In: *Mining text data*. Springer, 2012, pp. 415–463.

[59] Huan Liu and Rudy Setiono. "Chi2: Feature selection and discretization of numeric attributes". In: *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence*. IEEE. 1995, pp. 388–391.

[60] Aditya Mandhare. "Application of Decision Tree to Predict Gross Income of a Movie". In: *International Journal of Computer Applications* 106.5 (2014).

[61] Tejas Mankar et al. “Stock Market Prediction based on Social Sentiments using Machine Learning”. In: *2018 International Conference on Smart City and Emerging Technology (ICSCET)*. IEEE. 2018, pp. 1–3.

[62] Ruli Manurung et al. “Machine learning-based sentiment analysis of automatic indonesian translations of english movie reviews”. In: *Proceedings of the International Conference on Advanced Computational Intelligence and Its Applications (ICACIA)*. 2008, pp. 1–6.

[63] FJAP Mattosinho. “Mining Product Opinions and Reviews on the Web”. In: ().

[64] DG Maynard and Mark A Greenwood. “Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis”. In: *LREC 2014 Proceedings*. ELRA. 2014.

[65] Walaa Medhat, Ahmed Hassan, and Hoda Korashy. “Sentiment analysis algorithms and applications: A survey”. In: *Ain Shams engineering journal* 5.4 (2014), pp. 1093–1113.

[66] Walaa Medhat, Ahmed H Yousef, and Hoda K Mohamed. “Component analysis of a Sentiment Analysis framework on different corpora”. In: *2014 9th International Conference on Computer Engineering & Systems (ICCES)*. IEEE. 2014, pp. 300–306.

[67] R Meena and V Thulasi Bai. “Study on Machine learning based Social Media and Sentiment analysis for medical data applications”. In: *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*. IEEE. 2019, pp. 603–607.

[68] Yelena Mejova and Padmini Srinivasan. “Exploring feature definition and selection for sentiment classifiers”. In: *Fifth international AAAI conference on weblogs and social media*. 2011.

[69] George A Miller. “WordNet: a lexical database for English”. In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[70] Gary Miner et al. *Practical text mining and statistical analysis for non-structured text data applications*. Academic Press, 2012.

[71] Samaneh Moghaddam and Martin Ester. "Opinion digger: an unsupervised opinion miner from unstructured product reviews". In: *Proceedings of the 19th ACM international conference on Information and knowledge management*. ACM. 2010, pp. 1825–1828.

[72] Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. "Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets". In: *arXiv preprint arXiv:1308.6242* (2013).

[73] Saloni Mohan et al. "Stock Price Prediction Using News Sentiment Analysis". In: *2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService)*. IEEE. 2019, pp. 205–208.

[74] Tareq Al-Moslmi et al. "Approaches to cross-domain sentiment analysis: A systematic literature review". In: *IEEE Access* 5 (2017), pp. 16173–16192.

[75] Tony Mullen and Robert Malouf. "A Preliminary Investigation into Sentiment Analysis of Informal Political Discourse." In: *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*. 2006, pp. 159–162.

[76] Sana Nabil, Jaber Elbouhdidi, and Mohamed Yassin. "Recommendation system based on data analysis-application on tweets sentiment analysis". In: *2018 IEEE 5th International Congress on Information Science and Technology (CiSt)*. IEEE. 2018, pp. 155–160.

[77] Vivek Narayanan, Ishan Arora, and Arjun Bhatia. "Fast and accurate sentiment classification using an enhanced Naive Bayes model". In: *International Conference on Intelligent Data Engineering and Automated Learning*. Springer. 2013, pp. 194–201.

[78] R Narmadha and P Perumal. "An Extensive Study On Automated Aspect And Aspect Category Summarization Technique To Influence On Sentimental Analysis Of Co-Occurrence Data". In: *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*. IEEE. 2019, pp. 267–271.

[79] Arjun Srinivas Nayak et al. "Survey on pre-processing techniques for text Mining". In: *International Journal Of Engineering And Computer Science, ISSN* (2016), pp. 2319–7242.

[80] Rafeeque Pandarachalil, Selvaraju Sendhilkumar, and GS Mahalakshmi. "Twitter sentiment analysis for large-scale data: an unsupervised approach". In: *Cognitive computation* 7.2 (2015), pp. 254–262.

[81] Seongik Park and Yanggon Kim. "Building thesaurus lexicon using dictionary-based approach for sentiment classification". In: *2016 IEEE 14th International Conference on Software Engineering Research, Management and Applications (SERA)*. IEEE. 2016, pp. 39–44.

[82] Hitesh Parmar, Sanjay Bhanderi, and Glory Shah. "Sentiment Mining of Movie Reviews using Random Forest with Tuned Hyperparameters". In: *International Conference on Information Science. Kerala*. 2014.

[83] Minlong Peng et al. "Cross-domain sentiment classification with target domain specific information". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2018, pp. 2505–2513.

[84] Jacob Perkins. *Python text processing with NLTK 2.0 cookbook*. 1. Packt Publishing Ltd, 2010.

[85] Huyen Trang Phan et al. "A Method for Detecting and Analyzing the Sentiment of Tweets Containing Fuzzy Sentiment Phrases". In: *2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*. IEEE. 2019, pp. 1–6.

[86] Ana-Maria Popescu and Orena Etzioni. "Extracting product features and opinions from reviews". In: *Natural language processing and text mining*. Springer, 2007, pp. 9–28.

[87] Paitoon Porntrakoon and Chayapol Moemeng. "Thai sentiment analysis for consumer's review in multiple dimensions using sentiment compensation technique (sensecomp)". In: *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. IEEE. 2018, pp. 25–28.

[88] Suhaas Prasad. "Micro-blogging sentiment analysis using bayesian classification methods". In: *Technical Report*. Stanford University, 2010.

[89] Bella Azis Dewanti Putri, Annisa Uswatun Khasanah, and Abdullah Azzam. "Sentiment Analysis on Grab User Reviews Using Support Vector Machine and Maximum Entropy Methods". In: *2019 International Conference on Information and Communications Technology (ICOIACT)*. IEEE. 2019, pp. 468–473.

[90] Guang Qiu et al. "Opinion word expansion and target extraction through double propagation". In: *Computational linguistics* 37.1 (2011), pp. 9–27.

[91] Benaissa Azzeddine Rachid., Harbaoui Azza., and Ben Ghezala Henda. "Sentiment Analysis Approaches based on Granularity Levels". In: *Proceedings of the 14th International Conference on Web Information Systems and Technologies - Volume 1: WEBIST,* INSTICC. SciTePress, 2018, pp. 324–331. ISBN: 978-989-758-324-7. DOI: 10.5220/0007187603240331.

[92] Rahul Rajput and Arun Kumar Solanki. "Review of Sentimental Analysis methods using lexicon based approach". In: *IJCSMC* 5.2 (2016), pp. 159–166.

[93] Vallikannu Ramanathan and T Meyyappan. "Twitter Text Mining for Sentiment Analysis on People's Feedback about Oman Tourism". In: *2019 4th MEC International Conference on Big Data and Smart City (ICBDSC)*. IEEE. 2019, pp. 1–5.

[94] Ankit Ramteke et al. "Detecting turnarounds in sentiment analysis: Thwarting". In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 2013, pp. 860–865.

[95] Megha Rathi et al. "Sentiment Analysis of Tweets Using Machine Learning Approach". In: *2018 Eleventh International Conference on Contemporary Computing (IC3)*. IEEE. 2018, pp. 1–3.

[96] Shubham Rendalkar and Chaitali Chandankhede. "Sarcasm Detection of Online Comments Using Emotion Detection". In: *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*. IEEE. 2018, pp. 1244–1249.

[97] Ellen Riloff, Janyce Wiebe, and Theresa Wilson. "Learning subjective nouns using extraction pattern bootstrapping". In: *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Association for Computational Linguistics. 2003, pp. 25–32.

[98] Lina Maria Rojas-Barahona. "Deep learning for sentiment analysis". In: *Language and Linguistics Compass* 10.12 (2016), pp. 701–719.

[99] Òscar Romero Llombart. "Using machine learning techniques for sentiment analysis". In: (2017).

[100] Pedro Aniel Sánchez-Mirabal et al. "UMCC_DLSI: Sentiment Analysis in Twitter using Polirity Lexicons and Tweet Similarity". In: *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. 2014, pp. 727–731.

[101] Asad B Sayeed et al. "Grammatical structures for word-level sentiment detection". In: *Proceedings of the 2012 Conference of the North American Chapter of the Association for computational Linguistics: Human language technologies*. Association for Computational Linguistics. 2012, pp. 667–676.

[102] USC Marshall Business School. *How do Neural networks mimic the human brain?* Last Accessed:28-04-2019. URL: https://www.marshall.usc.edu/blog/how-do-neural-networks-mimic-human-brain (visited on 09/30/2010).

[103] Kim Schouten and Flavius Frasincar. "Finding Implicit Features in Consumer Reviews for Sentiment Analysis". In: July 2014, pp. 130–144. ISBN: 978-3-319-08244-8. DOI: 10.1007/978-3-319-08245-5_8.

[104] Kim Schouten and Flavius Frasincar. "Finding implicit features in consumer reviews for sentiment analysis". In: *International Conference on Web Engineering*. Springer. 2014, pp. 130–144.

[105] Kim Schouten and Flavius Frasincar. "Survey on aspect-level sentiment analysis". In: *IEEE Transactions on Knowledge and Data Engineering* 28.3 (2015), pp. 813–830.

[106] Chirag Sehra. *Decision Trees Explained Easily*. Last Accessed:10-04-2020. URL: https://medium.com/@chiragsehra42/decision-trees-explained-easily-28f23241248.

[107] Shubham. *Decision Tree*. Last Accessed:10-04-2020. URL: https://www.hackerearth.com/practice/machine-learning/machine-learning-algorithms/ml-decision-tree/tutorial/.

[108] V Srividhya and R Anitha. "Evaluating preprocessing techniques in text categorization". In: *International journal of computer science and application* 47.11 (2010), pp. 49–51.

[109] Chi Sun, Luyao Huang, and Xipeng Qiu. "Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence". In: *arXiv preprint arXiv:1903.09588* (2019).

[110] Sachin N. Deshmukh Supriya B. Moralwar. "Different Approaches of Sentiment Analysis". In: *International Journal of Computer Sciences and Engineering (JCSE)* 3.3 (2015).

[111] Symeon Symeonidis, Dimitrios Effrosynidis, and Avi Arampatzis. "A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis". In: *Expert Systems with Applications* 110 (2018), pp. 298–310.

[112] Yongcai Tao et al. "Joint Embedding of Emoticons and Labels Based on CNN for Microblog Sentiment Analysis". In: *2019 IEEE Fourth International Conference on Data Science in Cyberspace (DSC)*. IEEE. 2019, pp. 168–175.

[113] Ankita Tripathi and Shrawan Kumar Trivedi. "Sentiment analyis of Indian movie review with various feature selection techniques". In: *2016 IEEE International Conference on Advances in Computer Applications (ICACA)*. IEEE. 2016, pp. 181–185.

[114] Oren Tsur, Dmitry Davidov, and Ari Rappoport. "ICWSM—a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews". In: *Fourth International AAAI Conference on Weblogs and Social Media*. 2010.

[115] Anthony K. H. Tung. "Rule-based Classification". In: *Encyclopedia of Database Systems*. Ed. by LING LIU and M. TAMER ÖZSU. Boston, MA: Springer US, 2009, pp. 2459–2462. ISBN: 978-0-387-39940-9. DOI: 10.1007/978-0-387-39940-9_559. URL: https://doi.org/10.1007/978-0-387-39940-9_559.

[116] Peter D Turney. "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews". In: *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics. 2002, pp. 417–424.

[117] Diego Uribe. “Optimizing feature selection techniques for sentiment classification”. In: *2011 IEEE Electronics, Robotics and Automotive Mechanics Conference*. IEEE. 2011, pp. 103–107.

[118] Keisuke Utsu, Junki Saito, and Osamu Uchida. “Sentiment Polarity Estimation of Emoticons by Polarity Scoring of Character Components”. In: *2018 IEEE Region Ten Symposium (Tensymp)*. IEEE. 2018, pp. 237–242.

[119] Raisa Varghese and M Jayasree. “A survey on sentiment analysis and opinion mining”. In: *International Journal of Research in Engineering and Technology* 2.11 (2013), pp. 312–317.

[120] S Vijayarani, Ms J Ilamathi, and Ms Nithya. “Preprocessing techniques for text mining-an overview”. In: *International Journal of Computer Science & Communication Networks* 5.1 (2015), pp. 7–16.

[121] G Vinodhini and RM Chandrasekaran. “Sentiment analysis and opinion mining: a survey”. In: *International Journal* 2.6 (2012), pp. 282–292.

[122] Mr Saifee Vohra and Jay Teraiya. “Applications and challenges for sentiment analysis: A survey”. In: *International journal of engineering research and technology* 2.2 (2013), pp. 1–6.

[123] SM Vohra and JB Teraiya. “A comparative study of sentiment analysis techniques”. In: *Journal JIKRCE* 2.2 (2013), pp. 313–317.

[124] Wei Wang, Hua Xu, and Wei Wan. “Implicit feature identification via hybrid association rule mining”. In: *Expert Systems with Applications* 40.9 (2013), pp. 3518–3531.

[125] Yaser Maher Wazery, Hager Saleh Mohammed, and Essam Halim Houssein. “Twitter Sentiment Analysis using Deep Neural Network”. In: *2018 14th International Computer Engineering Conference (ICENCO)*. IEEE. 2018, pp. 177–182.

[126] Janyce Wiebe and Ellen Riloff. “Creating subjective and objective sentence classifiers from unannotated texts”. In: *International conference on intelligent text processing and computational linguistics*. Springer. 2005, pp. 486–497.

[127] Janyce Wiebe, Theresa Wilson, and Claire Cardie. "Annotating expressions of opinions and emotions in language". In: *Language resources and evaluation* 39.2-3 (2005), pp. 165–210.

[128] Wikipedia. *Probabilistic classification*. URL: https://en.wikipedia.org/wiki/Probabilistic_classification (visited on).

[129] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. "Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis". In: *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*. Vancouver, British Columbia, Canada: Association for Computational Linguistics, Oct. 2005, pp. 347–354. URL: https://www.aclweb.org/anthology/H05-1044.

[130] Katarzyna Wójcik and Janusz Tuchowski. "Ontology Based Approach to Sentiment Analysis". In: *KNOWLEDGE ECONOMY SOCIETY* (2014), p. 267.

[131] Wei Wu, Bin Zhang, and Mari Ostendorf. "Automatic generation of personalized annotation tags for twitter users". In: *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics*. 2010, pp. 689–692.

[132] Nisha Yadav et al. "Extraction-Based Text Summarization and Sentiment Analysis of Online Reviews Using Hybrid Classification Method". In: *2019 Sixteenth International Conference on Wireless and Optical Communication Networks (WOCN)*. IEEE. 2019, pp. 1–6.

[133] Lin Yue et al. "A survey of sentiment analysis in social media". In: *Knowledge and Information Systems* (2018), pp. 1–47.