B.E. PROJECT
ON
DISCLOSING LATENT MOCKERY SENSE OF SENTENCES : SARCASM DETECTION USING NATURAL LANGUAGE PROCESSING

Submitted by:

ADITI KUMAR (706/IT/14)

ARUSHI BHATT (718/IT/14)

Under the Guidance Of

MR. DEVENDER KUMAR

A Project in Partial Fulfillment of Requirements for the award of
Bachelor of Engineering in
Information Technology Engineering
(2014 - 2018)

DIVISION OF INFORMATION TECHNOLOGY ENGINEERING

NETAJI SUBHAS INSTITUTE OF TECHNOLOGY

(UNIVERSITY OF DELHI)

NEW DELHI - 110078

2018

SELF-DECLARATION

This is to certify that the project entitled "**Disclosing Latent Mockery Sense of Sentences: Sarcasm Detection using Natural Language Processing"** by **Aditi Kumar** and **Arushi Bhatt** is a record of bonafide work carried out by us, in the Division of Information Technology, Netaji Subhas Institute of Technology, University of Delhi, New Delhi, in partial fulfillment of requirements for the award of the degree of Bachelor of Engineering in Information Technology Engineering in the academic year 2014-2018.

The results presented in this thesis have not been submitted to any other university in any form for the award of any degree.

Aditi Kumar
(706/IT/14)

Arushi Bhatt
(718/IT/14)

Division of Information Technology Engineering
Netaji Subhas Institute of Technology
(University of Delhi)
New Delhi - 110078
2018

नेताजी सुभाष प्रौद्योगिकी संस्थान
NETAJI SUBHAS INSTITUTE OF TECHNOLOGY
(An Institution of Govt. of NCT of Delhi Formerly, Delhi Institute of Technology)
Azad Hind Fauj Marg, Sector-3, Dwarka, New Delhi-110045
Telephone : 564 6638-41, 5568701-03 Fax : 564 6635 Website : http : // www.nsit.ac.in

CERTIFICATE

This is to certify that the project entitled "**Disclosing Latent Mockery Sense of Sentences: Sarcasm Detection using Natural Language Processing**" submitted by **Aditi Kumar (706/IT/14) and Arushi Bhatt (718/IT/14)** for the partial fulfillment of Award of Bachelor of Engineering is a bonafide record of the work done by the candidates under my guidance. To the best of my knowledge, this work has not been submitted for the award of any other degree.

Date:

Place:

Signature

Mr. Devender Kumar
Assistant Professor, IT
NSIT, New Delhi

ACKNOWLEDGEMENT

Setting an endeavor may not always be an easy task: obstacles are bound to come in its way and when this happens, help is welcome and without help of the people whom we are mentioning here, this endeavor would not have been successful. The completion of any project brings with it a sense of satisfaction, but it is never complete without thanking those people who made it possible and whose constant support has crowned our efforts with success. We would like to extend our sincere thanks to all of them.

We take immense pleasure in thanking **Asst. Prof. Devender Kumar** for his abled guidance and useful suggestions, which helped us in completing the project work in time. He was the one who inspired us with her dynamic energy and her proficiency in technical aspect made us spellbound.

Also we would like to express our sincere thanks to our senior ***Abhinav Sharma***, and our friend ***Harsh Kumar*** for their immense support and help.

Our thanks and appreciations also go to our colleagues in developing the project and people who have willingly helped us out with their abilities.

Aditi Kumar (706/IT/14)

Arushi Bhatt (718/IT/14)

ABSTRACT

For the past 20 years, researchers from linguists to psychologists to neurologists have been studying our ability to perceive snarky remarks and gaining new insights into how the mind works. Sarcasm is a latent form of language where usually, the speaker explicitly states the opposite of what the intended meaning is. Inculcated with this kind of ambiguity and subtlety, detecting sarcasm is a difficult task, even for humans. While a fair amount of work has been done on automatically detecting emotion in human speech, there has been little research on sarcasm detection. Current works approach this challenging problem primarily from a linguistic perspective, focussing on the lexical and syntactic aspects of sarcasm. Further studies referred for the presented thesis, explore the possibility of using social traits intrinsic to users of sarcasm to detect sarcastic tweets. Based on these, we have worked in the direction of improvisation in feature extraction pertaining to social traits of users. After rigorous pre-processing, computation of group of features like Contrasting Connotation, Transition Probability, Pragmatic Features and so on has been done. We have shown a comparative study of performance of different features by feeding different combination of features to the supervised learning model consisting of thirteen classifiers. All the results that have been analyzed with the help of graphs, indicates the relevance of study carried out by us. Different social traits covered in this thesis are of greater significance for tackling problems of sarcasm detection.

List of Figures and Tables

LIST OF CONTENTS

CHAPTER 1

OVERVIEW

1.1 Introduction

In recent years, social networking sites such as Twitter have grown immensely in popularity and importance. According to a Pew Research Center study, as of September 2013, 74% of all online adults use social networking sites, up from less than 30% in 2008. These sites have not only gained users but also multiple functionalities. They have become an ad-hoc source of entertainment, news, information etc. . These sites have evolved from simple platforms where users connect to each other and keep in touch, to large ecosystems where users, among other things, express their ideas and opinions uninhibitedly. Nowadays, with social media forming a part of our everyday lives, users candidly share a wide breadth of information, from the relatively general to the highly personal. From a sales and marketing perspective, companies have unbridled access to this unique ecosystem to gain critical insights into the mindset and thought process of their customers and to better serve their needs. They can tap into public opinion on their products or services and even provide real-time customer assistance through social media. Not surprisingly, most large companies have a social media presence and a dedicated social media team working on marketing, after-sales service, and consumer assistance.

Given the high velocity and volume of social media data, companies rely on automated social media management tools such as HootSuite, to analyze data and to provide customer service. These tools perform tasks such as content management, sentiment analysis and extraction/filtering of relevant messages for the company's customer service representatives to take action. While these tools perform well for basic tasks, they lack the necessary sophistication to decipher more nuanced forms of language such as sarcasm, in which the meaning of a message is not always obvious and explicit. This is quite a handicap, especially in the context of social media where the relative ambiguity and the ability to hide behind computer screens often encourages snarky, rude and sarcastic posts. The lack of a viable sarcasm detection mechanism imposes an extra burden on the company's social media team, who are already inundated with customer messages, to identify these sarcastic messages and respond appropriately. We have seen many examples where the customer service representatives fail to detect sarcasm. Such public gaffes not only upset the already disgruntled customers but also ruin the public images of companies.

Interestingly in June 2014, the United States Secret Service also issued a work order seeking social media software capable of detecting sarcasm 3, explicitly stating that social media tools currently in the market do not have the capability of detecting nuanced forms of language such as sarcasm.

The Free Dictionary defines sarcasm as a form of verbal irony that is intended to express contempt or ridicule. The figurative nature of sarcasm makes it an often-quoted challenge for sentiment analysis [1]. Sarcasm has a negative implied sentiment, but may not have a negative surface sentiment. A sarcastic sentence may carry positive surface sentiment (for example, 'Visiting dentists is so much fun!'), negative surface sentiment (for example, 'His performance in Olympics has been terrible anyway' as a response to the criticism of an Olympic medalist) or no surface sentiment (for example, the idiomatic expression 'and I am the Queen of England is used to express sarcasm). Since sarcasm implies sentiment, detection of sarcasm in a text is crucial to predicting the correct sentiment of the text.

Our goal in this study is to tackle the challenging problem of sarcasm detection on Twitter. While sarcasm detection is inherently complex and difficult, the style and nature of content on Twitter further complicate the process. Compared to other, more conventional sources such as news articles and novels, Twitter [i] is more informal in nature with an evolving vocabulary of slang words and abbreviations and [ii] has a limit of 140 characters per tweet which provides fewer word-level cues thus adding more ambiguity. However, Twitter provides other information such as social graphs, past tweets and profile bio details, which when used effectively, may help overcome the aforementioned challenges.

Current research on sarcasm detection on Twitter (Tsur et al., 2010; González-Ibáñez et al., 2011; Liebrecht et al., 2013; Riloff et al., 2013[2]) primarily analyze information obtained only from the text of tweets. These techniques treat sarcasm as a linguistic phenomenon, with limited emphasis on the psychological aspects of sarcasm. However, sarcasm has been extensively studied in the psychological and behavioral sciences and theories explaining when, why, and how sarcasm is expressed have been established. These theories can be extended and employed to automatically detect sarcasm on

Twitter. For example, Rockwell (Rockwell, 2007) identified a positive correlation between cognitive complexity and the ability to produce sarcasm. A high cognitive complexity of an individual may be manifested in the language complexity of her tweets on Twitter.

In this thesis, we have focused on different features and their contribution to sarcasm detection. The major contributions of this thesis are:

1. We identify several new forms forms of sarcasm and demonstrate how these forms may be manifested on Twitter.
2. We introduce a model with rigorous work of feature extraction for 10 features using natural language processing methods.
3. We investigate and demonstrate the importance of social traits in sarcasm detection with the help of 13 classifiers.
4. We compare the results and analyse them with the help of graphs generated through code.

1.2 Background

With the increasing relevance of social media platforms in all fields, ranging from politics to customer service, almost all major MNCs have specialized teams dedicated to handling their social media outreach. Many customers prefer to send their feedbacks, comments, and opinions via these platforms, especially Twitter, because of the instantaneity provided. With the increasing customer expectations, minor goof-ups on the part of the customer response team results in tarnishing the reputation of the company. Hence there is a need of a robust sarcasm detection system.

Starting with the earliest known work by [2] which deals with sarcasm detection in speech, the area has seen wide interest from the natural language processing community as well. Following that, sarcasm detection from text has extended to different data forms (tweets, reviews, TV series dialogues), and spanned several approaches (rule-based, supervised, semi-supervised). This synergy has resulted in interesting innovations for automatic sarcasm detection.

1.3 Problem Statement

While researchers in linguistics and psychology debate on what exactly constitutes sarcasm, for the sake of clarity, we use the Oxford dictionary's definition of sarcasm as a way of using words that are the opposite of what you mean in order to be unpleasant to somebody or to make fun of them.
The challenges of sarcasm and the benefit of sarcasm detection to sentiment analysis have led to interest in automatic sarcasm detection as a research problem. Automatic sarcasm detection refers to computational approaches that predict if a given text is sarcastic. Like, the sentence '*I love it when my son rolls his eyes at me*' should be predicted as sarcastic, while the sentence *'I love it when my son gives me a present*' should be predicted as non-sarcastic. This problem is difficult because of nuanced ways in which sarcasm may be expressed.

We now look at how the problem of automatic sarcasm detection has been defined, in past work. The most common formulation for sarcasm detection is a classification task. Given a piece of text, the goal is to predict whether or not it is sarcastic. us, the sentence '*I love being ignored*' is to be predicted as sarcastic while the sentence 'I love being pampered' is to be predicted as non-sarcastic. Past work varies in terms of what these output labels are.

1.4 Objective

Primarily, the project aims at analysing, implementing, remodelling and verifying the results published in Identifying Sarcasm in Twitter: A Closer Look. The problem statement has been reproduced here for the sake of clarity:

"Given an unlabeled tweet t from user U along with a set of U's past tweets T, a solution to sarcasm detection aims to automatically detect if it is sarcastic or not."

CHAPTER 2

LITERATURE REVIEW

2.1 Earlier Works

Automatic detection of sarcasm is a relatively new, less researched topic and is deemed a difficult problem . While works on automatic detection of sarcasm in speech (Tepperman et al., 2006) utilizes prosodic, spectral and contextual features, sarcasm detection in text has relied on identifying text patterns (Davi- dov et al., 2010)[3] and lexical features (Gonz ́alez-Iba ́n ̃ez et al., 2007).

Davidov et al. (Davidov et al., 2010) devised a semi-supervised technique to detect sarcasm in Amazon product reviews and tweets. They used an interesting pattern-based (using high frequency words and content words) and punctuation-based features to build a classification model using a weighted k-nearest neighbor classifier to perform sarcasm detection. Gonza ́lez-Ib ́anez et al. (Gonz ́alez-Iba ́n ̃ez et al., 2011) devised a detection technique using numerous lexical features (derived from LWIC (Pennebaker et al., 2001)[4], Wordnet Affect (Strapparava and Valitutti, 2004)) and pragmatic features such as emoticons and replies. Reyes et al. (Reyes et al., 2012)[5] focussed on developing classifiers to detect verbal irony based on ambiguity, polarity, unexpectedness and emotional cues derived from text. Liebrecht et al. (Liebrecht et al., 2013)[6] used unigrams, bigrams and trigrams as features to detect sarcastic dutch tweets using a Balanced Winnow classifier. More recently, Riloff et al. (Riloff et al., 2013)[7], used a well constructed lexicon-based approach to detect sarcasm based on an assumption that sarcastic tweets are a contrast between a positive sentiment and a negative situation. Table 2.1 gives a brief overview of the aforementioned current research related to automatic sarcasm detection.

Table 1: Overview of Related Work

Author and Year	Overview and Methodology
PUSHPAK BHATTACHARYYA (2017)	Automatic Sarcasm Detection semi-supervised pa ern extraction to

	identify implicit sentiment, use of hashtag-based supervision, and incorporation of context beyond target text.
Riloff et al. (2013)	Lexicon-based approach contrasting positive sentiment and negative situation.
Liebrecht et al. (2013)	Unigram,bigram and trigram features used to train a Balanced Winnow Classifier
Reyes et al. (2012)	Ambiguity, Polarity, emotional cues , etc. , to train decision trees
Gonzalez-Ibanez et al. (2011)	Lexical and pragmatic features to train SMO classifier
Davidov et al. (2010)	Patterns and punctuation based features used to train weighted k-nearest neighbour classifier

As described above, current works on sarcasm detection have heavily focussed on sarcasm's linguistic aspects and utilized primarily, the content of the tweet. In contrast, we believe that our framework provides a systematic approach towards better sarcasm detection by not only analyzing the content of tweets but by also exploiting the social traits and cognitive traits of users derived from their past activities. Furthermore, the user's past activities also aid in incorporating contextual awareness to our character modeling framework to improve the classification process. Contextual awareness has been acknowledged within psychology research as being a necessary condition for identifying sarcasm . We map research on (1) what makes people use sarcasm, (2) when they use it and (3) how they use it, to observable user behavior on Twitter and build a comprehensive supervised framework to detect sarcasm.

2.2 Sarcasm Studies in Linguistics

Before we begin with approaches to automatic sarcasm detection, we discuss linguistic studies pertaining to sarcasm.

Sarcasm is a form of figurative language where the literal meaning of words does not hold, and instead the opposite interpretation is intended [8]. Sarcasm is closely related

to irony - in fact, it is a form of irony. [9] state that ‘verbal irony is recognized by literary scholars as a technique of using incongruity to suggest a distinction between reality and expectation’. ey de ne two types of irony: verbal and situational. Verbal irony is irony that is expressed in words. For example, the sentence ‘Y*our paper on grammar correction contains several grammatical errors.’* is ironic. On the other hand, situational irony is irony that arises out of a situation. For example, a situation where a scientist discovers the cure for a disease but herself succumbs to the disease before being able to apply the cure, is a situational irony.
[9] refer to sarcastic language as ‘irony that is especially bi er and caustic’. ere are two components of this de nition: (a) presence of irony, (b) being bi er. Both together are identifying features of sarcasm. For example, ‘I could not make it big in Hollywood because my writing was not bad enough’. is example from [34] is sarcastic, because: (a) it contains an ironic statement that implies a writer in Hollywood would need to be bad at writing, (b) the appraisal in the statement is in fact bi er/contemptuous towards the entity ‘Hollywood’. Several linguistic studies describe different aspects of sarcasm:

(1) Characteristics of sarcasm: [10] state that sarcasm occurs along several dimensions, namely, failed expectation, pragmatic insincerity, negative tension, and the presence of a victim. [11] state that sarcasm can be understood in terms of the response it elicits. ey observe that the responses to sarcasm may be laughter, no response4, smile, sarcasm (in retort), a change of topic5, literal reply and non-verbal reactions (a popular non-verbal reaction would be rolling one’s eyes). According to [12], sarcasm arises when there is situational disparity between text and contextual information. For example, the sentence ‘I love being ignored’ is understood as sarcastic due to the disparity between the contextual information that being ignored is an undesirable situation, and that the speaker claims to love it in the given sentence.

(2) Types of sarcasm: [13] show that there are four types of sarcasm: (i) Propositional: In such situations, the statement appears to be a proposition but has an implicit sentiment involved. For example ‘Your plan sounds fantastic!’. is sentence may be interpreted as non-sarcastic, if the context is not understood. (ii) Embedded: is type of sarcasm has an embedded incongruity in the form of words and phrases themselves. For example ‘John has turned out to be such a diplomat that no one takes him seriously’. e incongruity is embedded in the meaning of the word ‘diplomat’ and rest of the sentence. (iii) Like-pre xed: A like-phrase provides an implied denial of the argument being made. For example, ‘Like you care!’ is a common sarcastic retort. (iv) Illocutionary: is kind of sarcasm involves non-textual clues that indicate an a itude opposite to a sincere u erance. For example, rolling one’s eyes when saying ‘Yeah right’. In such cases, prosodic variations play a role. e examples above are from [14].

(3) Tuple-representation of sarcasm: [15] represent sarcasm as a 6-tuple consisting of <S, H, C, u, p, p’> where: S = Speaker, H = Hearer/Listener, C = Context, u = U erance, p = Literal Proposition, and p’ = Intended Proposition.

The tuple can be read as ‘Speaker S generates an utterance u in Context C meaning proposition p but intending that hearer H understands p’. For example, if a teacher says to a student, “ at’s how assignments should be done!” and if the student knows that they have barely completed the assignment, they would understand the sarcasm. In context of the 6-tuple above, the properties of this sarcasm would be:
S: Teacher
H: Student

C: The student has not completed his/her assignment. u: “That’s how assignments should be done!”
p: The student has done a good job at the assignment. p’: e student has done a bad job at the assignment.

(4) Echoic mention theory: A contrary view is described in [16]. e theory states that a literal proposition may not always be intended. is can be understood with the help of the sentence ‘I love it when I do not forward a chain mail and I die the next day’. The intention of the speaker is to remind the listener of situations where chain mails do not have any result. It is through highlighting this fact that the speaker’s intended ridicule of chain mails is understood. e echoic reminder theory also o ers a similar perspective [17]. e echoic reminder theory states that a sarcastic statement reminds the listener of a situation they have encountered, but with clues to indicate the sarcasm. For example, the sentence ‘Visits to a dentist are fun’ reminds the listener of the situation of visiting a dentist, and the popular appraisal that it is an unpleasant event.

(5) Sarcasm as a dropped negation: [18] states that irony/sarcasm is a form of negation in which an explicit negation marker is lacking. In other words, when one expresses sarcasm, a negation is intended, despite the lack of a negation word like ‘not’. For example, the sarcastic sentence ‘Being awake at 4 am with a headache is fun’ is equivalent to the non-sarcastic sentence ‘Being awake at 4am with a headache is not fun’. is results in the possibility that many sarcastic sentences could be converted to non-sarcastic by simply applying an appropriate negation.

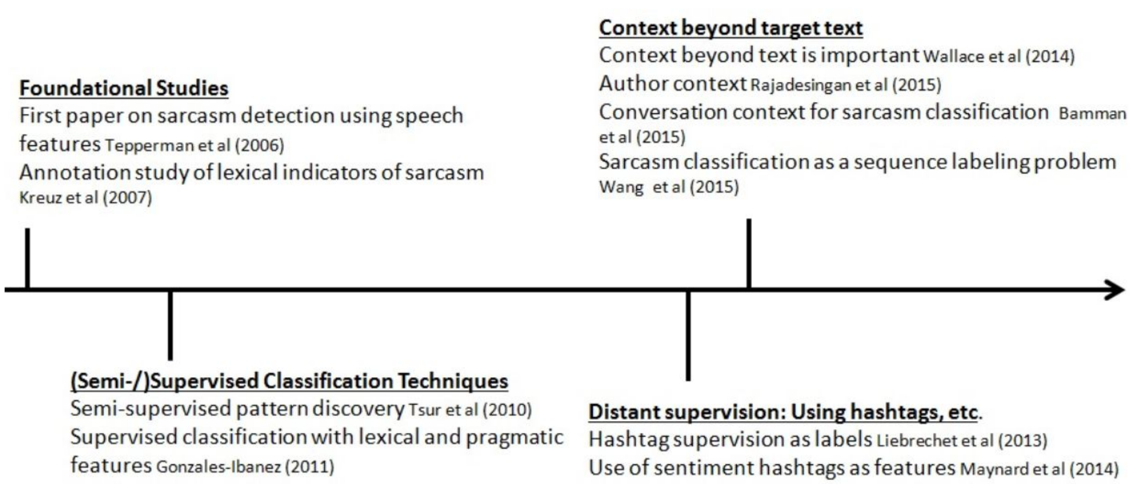
(6) Understanding sarcasm: [19] describe how sarcasm may be understood. They state that violation of truthfulness maxims is a key for a listener to understand sarcasm. For

example, ‘I love being ignored’ is understood as sarcastic by a listener who believes that being ignored is not a pleasant state to be in. However, ‘I love your new shirt!’ may or may not be sarcastic. e intended sarcasm cannot be understood until the listener observes that the literal meaning of the text violates truthfulness. To understand sarcasm, if any, in the sentence above, it would be essential to know information that would violate the truthfulness.

2.3 Trends in Sarcasm Detection

In this section, we detail interesting trends observed in sarcasm detection research. These trends are represented in Figure 1. Representative work in each area are indicated in the figure, around four key milestones. Following fundamental studies, supervised/semi-supervised sarcasm classification approaches were explored.These approaches focused on using specific patterns or novel features. Then, as Twitter emerged as a viable source of data, hashtag-based supervision became popular. Recently, there is an emerging trend to use context beyond the text being classified.

Figure 1. Trends in Sarcasm Detection



CHAPTER 3

USER CHARACTER MODELING FRAMEWORK

3.1 Character Modeling Approach

In Twitter, tweets are not always created in isolation. When posting a sarcastic tweet, the user makes a conscious choice to express her thoughts through sarcasm. The user may decide to use sarcasm as a response to a certain situation, observation or emotion. This behavior is informed by the user's individual characteristics, moods etc., which may be observed and analyzed through her activities on Twitter.

Further, it is observed that some people have more difficulty in generating and recognizing sarcasm than others due to cultural differences, language barriers etc. Therefore, some individuals have a higher propensity to use sarcasm than others. Hence, we factor in the user's likelihood of being a sarcastic person or otherwise, by analyzing historical data in the form of the user's past tweets.

Using existing research on sarcasm and our observations on Twitter, we find that sarcasm generation can be characterized as one (or a combination) of the following:

3.1.1 Sarcasm as a contrast of sentiments

A popular perception of sarcasm among researchers is that sarcasm is a contrast of sentiments. A classical view of sarcasm, argues that sarcastic utterances are first processed in the literal sense and if the literal sense is found incompatible with the present context, only then is the sentence processed in its opposite (ironic) form. This perceived contrast may be expressed through multiple facets such as mood, affect or sentiment.

3.1.2. Sarcasm as a complex form of expression

There is a small but significant correlation between cognitive complexity and the ability to produce sarcasm. A high cognitive complexity involves understanding and taking into account, multiple perspectives to make cogent decisions. Furthermore, expressing sarcasm requires determining if the environment is suitable for sarcasm, creating an appropriate sarcastic phrase and assessing if the receiver would be capable of recognizing sarcasm.

3.1.3 Sarcasm as a means of conveying emotion

Sarcasm is primarily a form of conveying one's emotions. While sarcasm is sometime interpreted as aggressive humor and as form of verbal aggression ,it also functions as a tool of self expression. Past studies, recognize that sarcasm is usually expressed in situations with negative emotions and attitudes.

3.1.4 Sarcasm as a function of familiarity

Friends and relatives are found to be better at recognizing sarcasm than strangers. Further, it has been demonstrated that the familiarity of language and cultural factors also play an important role in the recognition and usage of sarcasm.

3.1.5 Sarcasm as form of written expression

In psychology, sarcasm has been studied primarily as a spoken form of expression. However, sarcasm is quite prevalent in the written context as well,especially with the advent of online social networking sites. Through time, users have become more adept at conveying sarcasm in writing by including subtle markers that indicate to the unassuming reader, that the phrase is sarcastic. For example, while "you're so smart" does not hint at sarcasm, "Woowwww you are SOOOO cool" elicits some doubts on the statement's sincerity.

We believe that when expressing sarcasm, the user would invariably exhibit one or more of the aforementioned forms of sarcasm. Therefore, we build a behavior modeling framework for sarcasm detection that utilizes features which model these different forms. These extracted features are used to train a supervised classification model to determine if the tweet is sarcastic or not. As the novelty of approach lies in the behavior modeling and not the actual classifier itself, we explain more in detail on how sarcasm is modeled and incorporated into the framework. If the reader is unfamiliar with Twitter, a brief introduction of Twitter is included in the Appendix section.

CHAPTER 4

REPRESENTING FORMS OF SARCASM

Users' efforts in generating sarcasm are manifested in many ways on Twitter. In this section, we describe how different forms of sarcasm are realized in Twitter and how one can construct relevant features to capture these forms in the context of Twitter.

4.1 Sarcasm as a Contrast of Sentiments

4.1.1 Contrasting Connotations

A common means of expressing sarcasm is to employ words with contrasting connotations within the same tweet. For example, in I love getting spam emails!, spam has an obvious negative connotation while love is overwhelmingly positive. To model such occurrences, we construct features based on (1) affect and (2) sentiment scores. We obtain affect score of words from a dataset compiled by Warriner et al. (Warriner et al., 2013). This dataset contains affect (valence) scores for 13,915 English lemmas which are on a 9-point scale, with 1 being the least pleasant.

The sentiment score is calculated using SentiStrength . SentiStrength is a lexicon-based tool optimized for tweet sentiment detection based on sentiments of individual words in the tweet. Apart from providing a ternary sentiment result {positive, negative, neutral} for the whole tweet, SentiStrength outputs two scores for each tweet. A negative sentiment score from -1 to -5 (not-negative to extremely-negative) and a positive sentiment score from 1 to 5 (not-positive to extremely-positive). Here, we use SentiStrength's lexicon to obtain word-level sentiment scores. From these sentiment and affect scores, we calculate different scores as follows:

$$A = \{affect(w) | w \in t\}$$

$$S = \{ sentimentw) \mid w \in t\}$$

$$\Delta affect = max(A) - min(A)$$

$$\Delta sentiment = max(S) - min(S)$$

where t is the tweet and w is a word in t.
The affect(w) outputs the affect score of w.
The sentiment(w) outputs the sentiment score of w.
Δaffect and Δsentiment indicate the level of contrast in terms of sentiment and affect infused into the tweet by the user.
We use Δaffect and Δsentiment as features (2 features).

1. From these tweets, we extracted bigrams and trigrams along with their respective frequencies. We filter out bigrams and trigrams with frequencies less than 10.

2. For each bigram or trigram b, we find its associated sentiment score Sb,

$$Sb = \frac{POS(b) - NEG(b)}{POS(b) + NEG(b)}$$

where POS(b) is the number of occurrences of b in the positive tweets dataset and NEG(b) is the number of occurrences of b in the negative tweets dataset. We filter out bigrams or trigrams with marginal sentiment scores $\in (-0.1, 0.1)$.
Using the generated lexicon, we include as features, the number of bigrams and trigrams with positive sentiment scores, negative sentiment scores and their respective sum of scores (4 features).

4.1.2 Contrasting Present with the Past

While users often use contrasting words in the same tweet to express sarcasm, often times, a user may set up a contrasting context in her previous tweet and then, choose to use a sarcastic remark in her current tweet. This behavior may be more prevalent on Twitter as a result of the 140 character limit.

To model such behavior, we obtain the sentiment expressed by the user (i.e., positive, negative, neutral) in the previous tweet and the current tweet using SentiStrength. Then, we include the type of sentiment transition taking place from the past tweet to the current tweet (for example, positive → negative, negative → positive) as a feature (1 feature). In total, there are nine such transitions involving the combinations of positive, negative and neutral sentiments.

To provide a historical perspective on the user’s likelihood for such sentiment transitions, we compute the probability for all nine transitions using the user’s past tweets. The transition probabilities along with the probability score of the current transition are included as features

4.2 Sarcasm as a Complex Form of Expression

4.2.1 Readability

As sarcasm is widely acknowledged to be hard to read and understand, we adapt standardized readability tests to measure the degree of complexity and understandability of tweets. We use as features: number of words, number of syllables and number of syllables per word in the tweet . We also include number of polysyllables (words containing three or more syllables) and the number of polysyllables per word in the tweet derived from SMOG grade for readability as features

Inspired by average word length feature, we formulate a more comprehensive set of features involving the word length distribution L = {li} i=1 to 19 constructed from tweet t as follows:

1. For each word w in t, we compute its character length |w|. For convenience, we ignore words of length 20 or more. We construct a word length distribution

2. L may be represented succinctly using the following 6-tuple presentation:

$$< E[lw] , med[lw] , mode[lw] , \sigma[lw] , \min_{W \in t} lw , \max lw >$$

where E is the mean, med is the median, mode is the mode and σ is the standard deviation of word length distribution L. We include the 6-tuple representation as features (6 features).

Further, given the availability of the user’s past tweets, we examine if there is a noticeable difference in the word length distribution between the user’s current tweet and her past tweets. It must be noted that while sarcastic tweets may also be present in the user’s past tweets, because of their relative rarity, the past tweets when taken in entirety, would average out any influence possibly introduced by a few past sarcastic tweets. Therefore, any difference from the norm in the word length distribution of the current tweet can be captured. To capture differences in word length distribution, we perform the following steps:

1. From the user's current tweet, we construct a probability distribution D1 over length of words in the tweet.
2. From the user's past tweets, we construct a probability distribution D2 over length of words in all the past tweets.
3. To calculate the difference between the world length distribution of the current tweet and the past tweets, we calculate the Jenson-Shannon (JS) divergence between D1 and D2:

$$JS(D1||D2) = \tfrac{1}{2}KL(D1||M) + \tfrac{1}{2}KL(D2||M)$$

where $M = \frac{D1+D2}{2}$

and KL is the KL-divergence: $KL(T1||T2) = \ln\left(\frac{T1(i)}{T1(i)}\right)T1(i)$

We include the JS-divergence value also as a feature .

4.3 Sarcasm as a Means of Conveying Emotion

4.3.1 Emoji

We analysed the use of emoticons. A dictionary of emoticons has been used. Each emoticon is valued from -5 to 5. A total of 15 features have been extracted.

4.3.2 Affect and Sentiment

As sarcasm is a combination of affect and sentiment expression, we explore the possibility of observing differences with respect to how affect and sentiment is expressed in a sarcastic tweet. To this end, we construct a sentiment score distribution SS in which each count is the number of words in the tweet with sentiment score i where $i \in [-5, 5]$. We also construct an affect score distribution AS in which each count is the number of words in the tweet of affect score j where $j \in [1,9]$. We normalize counts in SS and AS. We include as features both these distributions .We represent these distributions as 6-tuples and include them as features . We also included the

number of affect words, number of sentiment words and the sentiment expressed (positive, negative and neutral) which are obtained from SentiStrength as features
To capture the difference in sentiment expression, we compare the sentiment score distribution of the user's past tweets to that of her current tweet. We calculate the JS-divergence between the past and current sentiment score distributions and include it as a feature .

In order to gain insights into the range of sentiments expressed by the user to gauge how she uses Twitter as a tool to express emotion, we construct a normalized distribution over the sentiment score [−5,5] of each word of her past tweets and include the distribution as a feature. This distribution given a perception of how expressive the user is, on Twitter. This is crucial as different users use Twitter for different reasons. Some Twitter users tweet objective facts, news articles and are generally information while other users are quite informal and tweet personal issues, emotions, opinions etc.

4.4 Sarcasm as a Form of Written Expression

4.4.1 Prosodic Variations

Prosody has been studied and identified as one of the major cues of sarcasm. Prosodic variations refer to changes made to writing styles in order to express intonation and stress. Language in social media is continuously evolving as users find simple, yet effective ways to better express themselves within the constraints imposed by the social networking site.

Users often repeat letters in words to stress and over-emphasize certain parts of the tweet (for example, sooooo, awesomeeee) to indicate that they mean the opposite of what is written. We capture such usage by including as boolean features, the presence of repeated characters and the presence of repeated characters (3 or more) in sentiment-loaded words (such as, loveeee). We also include as features, the number of characters used, and the ratio of the number of distinct characters to the total characters used in the tweet.

4.4.2 Pragmatic Features

Furthermore, users also use certain punctuations to express non-verbal cues that are crucial for sarcasm deliverance in speech. For example, users use “*” to indicate emphasis, “...” to indicate pause, “!!!” for exclamations (sometimes overdone to indicate

sarcasm). Therefore, we include as features, the normalized distribution of common punctuation marks(.,!?’*”) . \
We also observe that users often capitalize certain words to emphasize changes in tone (if the tweet were to be read out loud). We account for such changes by including as features, number of capitalized words in the tweet . It is also commonly observed that some users capitalize certain parts-of-speech(POS) to exaggerate or to vent their frustration. Using TweetNLP, we obtain the POS tag for each capitalized word in the tweet. Then, we compute the probability of observing such tags and include the same as features.
Existing works on quantifying linguistic style use lexical density, intensifiers and personal pronouns as important measures to gauge the writing style of the user. Lexical density is the fraction of information carrying words present in the tweet (nouns, verbs, adjectives and adverbs). Intensifiers are words that maximize the effect of adverbs or adjectives (for example, so, very). Personal pronouns are pronouns denoting a person or group (for example, me, our, her). We include as features the lexical density, the number of intensifiers used and the number of first- person singular, first-person plural, second-person and third-person pronouns present in the text (6 features).
In total we construct 327 features based on the behavioral aspects of sarcasm.

4.4.3 Structural Variations

Structural variations are inadvertent variations in the POS composition of tweets to express sarcasm. We observe that sarcastic tweets sometimes have a certain structure wherein the user’s views are expressed the first few words of the tweet, while in the later parts, a description of a particular scenario is put forth *(for example, I love it when my friends ignore me*). To capture possible syntactic idiosyncrasies arising from such tweet construction, we use as features, the POS tags of the first three words and the last three words in the tweet. We also include the position of the first sentiment-loaded word (0 if not present) and the first affect-loaded word (0 if not present) as a feature.

4.4.4 Polarity and Subjectivity

This feature checks the polarity and subjectivity, which amounts to Incongruity feature have been analysed. This is done by dividing the sentence in one-half, one-third,.....and so on. to extract 23 features.

4.4.5 Parse Tree

Parse Tree are syntactic structures which depict to a particular type of formation of sentences. Similar points of parse trees can be used for sarcasm detection. We use particular formation in the forms of parse Trees and validate the tweets in the dataset with them.

Table 2. Salient Features

	Salient Features
	Sarcastic patterns, Punctuations User mentions, emoticons, unigrams, sentiment-lexicon- based features Ambiguity-based, semantic relatedness N-grams, POS N-grams N-grams, emotion marks, intensifiers Sarcastic patterns (Positive verbs, negative phrases) Skip-grams, Polarity skip-grams Freq. of rarest words, max/min/avg # synsets, max/min/avg # synonyms Synonyms, Ambiguity, Written-spoken gap Interjection, ellipsis, hyperbole, imbalance-based POS sequences, Semantic imbalance. Chinese-specific features such as homophones, use of honorifics Word shape, Pointedness, etc. Length, capitalization, semantic similarity Unigrams, Implicit incongruity-based, Explicit incongruity- based Readability, sentiment flips, etc. Pattern-based features along with word-based, syntactic, punctuation-based and sentiment-related features Affect-based features derived from multiple emotion lexicons Features based on word embedding similarity Cognitive features derived from eye-tracking experiments

CHAPTER 5

PROPOSED WORK

5.1 Overview

Our model basically extracts tweets of both kinds (normal and sarcastic) using Twitter API. Extracted tweets are then pre-processed using natural language processing techniques. 10 feature groups have been extracted from the data and corresponding .csv files are generated. These files are then fed to the classifier algorithms. Data Visualization of the scores generated is done to capture a better view. Also we have done 10-fold cross validation to improve the scores generated. A comparative study has been conducted to find out the efficiency of features in sarcasm detection.
Let's discuss each step :

5.2 Data Source and Preprocessing

We validate our framework using a dataset1 of tweets from Twitter. To obtain a set of sarcastic tweets, we query the Streaming API using keywords #sarcasm and #not filtering out non-english tweets and retweets. We also remove tweets containing mentions and URLs as obtaining information from media and URLs is computationally expensive. We limit our analysis to tweets which contain more than three words as we found that tweets with fewer words were very noisy or clich ́ed (e.g., yeah, right! #sarcasm). Davidov et al. (Davidov et al., 2010) noted that some tweets containing the #sarcasm hashtag were about sarcasm and that the tweets themselves were not sarcastic. To limit such occurrences, we include only tweets that have either of the two hashtags as its last word; this reduces the chance of obtaining tweets that are about sarcasm but are themselves not sarcastic. After preprocessing, we obtained about 9104 sarcastic tweets which were self described by the user as being sarcastic using the appropriate hashtags. We remove the #sarcasm and #not hashtags from the tweets before proceeding with the evaluation.

In order to collect a set of general tweets (not sarcastic), we used Twitter's Sample API which provides a random sampling of tweets. We remove tweets that contain #sarcasm or #not from this random sample. It is true that this random sample may yet contain tweets that are sarcastic (but without the sarcasm hashtags) and fully acknowledge that the random dataset collected may not be pure. However, we believe that the possible proportion of sarcastic tweets in the random sample is extremely low and that when these tweets are taken in entirety, its effect would be miniscule. These tweets were subjected to the same aforementioned preprocessing technique.

Finally, for each tweet in the collected dataset, we extract the user who posted the tweet and then, we obtained that user's past tweets (we obtain the past 80 tweets for each user).

Some examples of tweets in the dataset are:
1. This paper is coming along... #not
2. Finding out your friends' lives through tweets is really the greatest feeling. #sarcasm

The above examples illustrate the difficulty of the task at hand. The first tweet may or may not be sarcastic purely depending on the context (which is not available in the tweet). Even if some background is available to us, as in the case of the second tweet, clearly, it is still a complicated task to map that information to sarcasm.

It must also be noted that, to avoid confusion and ambiguity when expressing sarcasm in writing, the users choose to explicitly mark the sarcastic tweets with appropriate hashtags. The expectation is that these tweets, if devoid of these hashtags, might be difficult to comprehend as sarcasm, even for humans. Therefore, our dataset might be biased towards the hardest forms of sarcasm. Using this dataset, we evaluate our framework and compare it with existing baselines.
So our dataset consists of following .csv files :

Normal tweets	2855 users
Sarcastic tweets	3219 users

After this data is preprocessed in the following ways :

- Mentions and URLs were removed : @ , http, www, etc
- Lemmatization :**Lemmatization** usually refers to doing things properly with the use of a vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the lemma .
- Stemming : **stemming** is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form—generally a written word

form. The stem need not be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem, even if this stem is not in itself a valid root.

- Stop word were removed : Ex. a, the, an, is, etc.
- Punctuations were also removed : . , ’ , ” ,; , : etc.

Preprocessing generated new dataset which is next used for feature extraction

```
"username","time","tweet"
"leavewithasong","Thu Oct 03 15:09:37 +0000 2013","Time to study. #yay #not"
"leavewithasong","Thu Oct 03 12:18:51 +0000 2013","@proudjonasgirl are you basically telling
yourself you love you? :D"
"leavewithasong","Wed Oct 02 20:34:16 +0000 2013","@coleyy2 you go girl!"
"leavewithasong","Wed Oct 02 19:37:26 +0000 2013","I'm trying to watch X-Factor but this stream
takes forever to load."
"leavewithasong","Wed Oct 02 19:35:57 +0000 2013","@taylaxo_ if people actually retweet or
believe that there is a 11/31 I'm gonna burry myself"
"leavewithasong","Wed Oct 02 19:34:34 +0000 2013","Can you STOP to fucking SPAM my timeline. I
don't have time for your shit to get noticed by some irrelevant person."
"leavewithasong","Wed Oct 02 18:58:28 +0000 2013","@someoneoutwhere yes it is"
"leavewithasong","Wed Oct 02 18:21:55 +0000 2013","Omg why is tumblr so slow"
"leavewithasong","Wed Oct 02 18:08:52 +0000 2013","I have so much work to do over the weekend
urg"
"leavewithasong","Wed Oct 02 18:01:28 +0000 2013","@JoickAddict love you more and you're
welcome!"
"DwarvenWidow","Wed Oct 02 17:55:40 +0000 2013","Sarah is the freaking best. Helping me with my
homework. I love you so much @leavewithasong"
"leavewithasong","Wed Oct 02 17:53:59 +0000 2013","Doing homework with @JoickAddict yep that's
how our friendship works hahaha"
"JoBros_Germany","Wed Oct 02 17:47:38 +0000 2013","Ab heute gibt es endlich #FirstTime auf
#iTunes zu kaufen.
Falls ihr es euch noch nicht vorbestellt habt, schaut... http://t.co/XKeLalluTR"
"leavewithasong","Wed Oct 02 15:21:22 +0000 2013","It's so cool that I can text with
@TimberWolf0108 every day now :)"
```

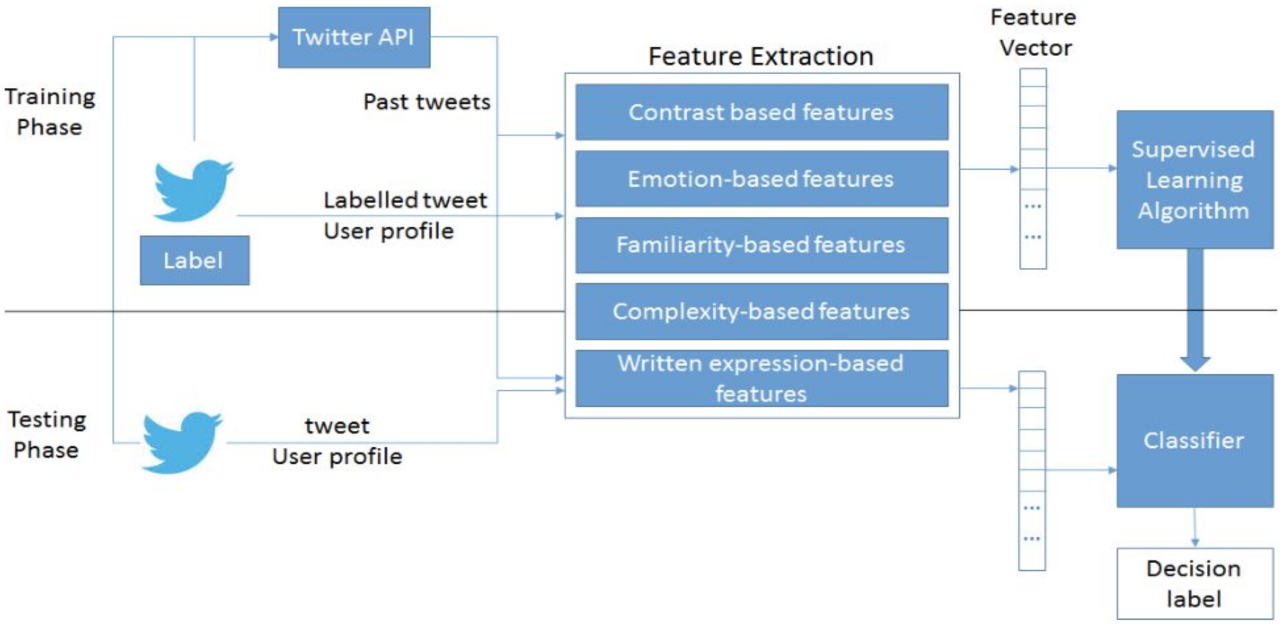
```
username,time,tweet
leavewithasong,Thu Oct 03 15:09:37 +0000 2013,Time study yay
leavewithasong,Thu Oct 03 12:18:51 +0000 2013, basically telling yourself love D
leavewithasong,Wed Oct 02 20:34:16 +0000 2013, go girl
leavewithasong,Wed Oct 02 19:37:26 +0000 2013,Im trying watch X-Factor stream takes forever
load
leavewithasong,Wed Oct 02 19:35:57 +0000 2013, people actually retweet believe 11/31 Im gonna
burry myself
leavewithasong,Wed Oct 02 19:34:34 +0000 2013,Can STOP fucking SPAM timeline I dont time shit
noticed irrelevant person
leavewithasong,Wed Oct 02 18:58:28 +0000 2013, yes
leavewithasong,Wed Oct 02 18:21:55 +0000 2013,Omg tumblr slow
leavewithasong,Wed Oct 02 18:08:52 +0000 2013,I much work over weekend urg
leavewithasong,Wed Oct 02 18:01:28 +0000 2013, love more youre welcome
DwarvenWidow,Wed Oct 02 17:55:40 +0000 2013,Sarah freaking best Helping homework I love much
leavewithasong,Wed Oct 02 17:53:59 +0000 2013,Doing homework  yep thats friendship works hahaha
JoBros_Germany,Wed Oct 02 17:47:38 +0000 2013,Ab heute gibt es endlich FirstTime auf iTunes zu
kaufenFalls ihr es euch noch nicht vorbestellt habt schaut
leavewithasong,Wed Oct 02 15:21:22 +0000 2013,Its cool I text  day now
leavewithasong,Wed Oct 02 15:14:54 +0000 2013,Thank god finally German edition  harpersbazaar
leavewithasong,Wed Oct 02 15:11:06 +0000 2013,Hey favorite magazine ❤️
leavewithasong,Wed Oct 02 14:48:32 +0000 2013,Just figured out  real Gossip Girl They lied
leavewithasong,Wed Oct 02 14:47:42 +0000 2013,“  xoxoxox” know love xoxo Gossip Girl
leavewithasong,Wed Oct 02 14:35:35 +0000 2013,“ remember I made people think Nick coming study
college rip” meany thats much
```

Figure 2. Sarcastic unprocessed and preprocessed data

5.3 Experiment Setup

As seen in the previous section, we have labeled data in the form of tweets with and without the sarcasm hashtags. Using this labeled data, we model our sarcasm detection problem as a supervised classification problem. A schematic diagram of the experiment set up is given in figure 3.

Figure 3. Experimental Setup



5.4. Feature Generation

(i) Feature 1 : Contrasting Connotation

No. of Features: 6

Using Bigrams and Trigrams of already labeled data, bigrams scores and trigram scores were generated. Now using these affect scores were calculated, along with sentiment strength.

(ii) Feature 2 : Transition Probability

No. of Features : 10

This feature checks the relation of transition of sentiment with respect to past tweets posted by the user.

(iii) Feature 3 : Prosodic Variation

No. of Features : 2

This checks for the repeated characters along with the sentiments.

(iv) Feature 4 : Structural Variation

No. of Features : 5

This feature extracts lexical density, hereby looking at the complexity issue ogf the sentences

(v) Feature 5 : Probability

No. of Features : 5
This calculates the probability by the ratio of length of letters and total letters.

(vi) Feature 6 : Word Length Distribution

No. of Features : 12
This includes calculation of word length distribution with the help of polysyllables and syllables. Using these, median, mode, standard deviation were calculated for word length distribution.

(vii) Feature 7 : Emoji

No. of Features : 16
This feature calculates the sentiments presented by emoticons, using emoji dictionary (values ranging from -5 to 5)

(viii) Feature 8 : Polarity

No. of Features : 23
This pertains to incongruity of feature by dividing the sentences into different parts(1-4). Polarity and subjectivity of every part is calculated.

(ix) Feature 9 : Pragmatic Features

No. of Features : 10
This includes capitalized words, interjections, intensifiers, negations and affirmations.

(x) Feature 10: Parse Trees

No. of Features : 1
This includes sarcasm detection by checking the occurrence of certain parse trees in the sentence.

5.5. Classification

After this 13 classifiers were fed with the features extracted. Scores were generated on Google cloud as we needed more computation power which were out of the limitation of our personal machines.

1. Gaussian Naive Bayes

2. Stochastic Gradient Descent
3. Decision Tree
4. Random Forest
5. Adaboost
6. L1 regularised Logistic Regression
7. L2 regularised Logistic Regression
8. Gradient Boosting
9. Bagging Model
10. SVM (rbf kernel)
11. K-neighbours hybrid
12. SVM (polynomial kernel)

Figure 4. Scores of all the features

Algorithm	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5	Feature 6	Feature 7	Feature 8	Feature 9	Feature 10	Feature 1 to 6	Feature 1 to 8	Feature 1 to 10	10 Fold
Gaussian Naive Bayes	0.6842105	0.560855263	0.4851973684	0.53782894	0.536184	0.597039	0.4835526315	0.534539	0.570723684	0.5509868421	0.66118421	0.6381578	0.63651315	0.64
SVM rbf	0.6891447	0.544407894	0.53618421	0.53947368	0.5457697	0.588815	0.5641447368	0.53125	0.5970394736	0.5509868421	0.59703947	0.6151315	0.59703947	0.61
Decision Tree	0.6644736	0.5361842105	0.53618421	0.527960526	0.534539	0.5625	0.5328947368	0.5444078	0.55098684	0.5509868	0.65625	0.6332236	0.63980263	0.65
SVM polynomial	0.602649	0.542763157	0.53618421	0.518092105	0.541118	0.574013	0.5625	0.5460526	0.585526315	0.55098684	0.62608418	0.615894	0.63980263	0.63
Random Forest	0.6759868	0.541118421	0.53618421	0.53453947	0.552631	0.623355	0.5625	0.5674342	0.6085526315	0.55098684	0.68092105	0.6907894	0.69572368	0.71
Adaboost	0.6907894	0.534539473	0.53618421	0.519736842	0.521381	0.5838815	0.56085526	0.5493421	0.54276315	0.55098684	0.71052631	0.7072368	0.73190789	0.73
Bagging Classifier	0.680921	0.534539473	0.53618421	0.5296052631	0.532894	0.557565	0.559210526	0.544407	0.5625	0.55098684	0.69243421	0.6940789	0.69572368	0.7
Extra trees	0.6332236	0.5509868421	0.53618421	0.5345394736	0.569078	0.629933	0.56414473	0.5625	0.628289473	0.55098684	0.66776315	0.6381578	0.69572368	0.72
Gradient boosting	0.7006578	0.5509868421	0.53618421	0.53782894	0.550986	0.629934	0.5542763157	0.527968	0.63157894	0.55098684	0.71546052	0.7368421	0.75657894	0.76
L1 Logistic Regression	0.6595394	0.554276315	0.53618421	0.53782894	0.550986	0.633223	0.554276315	0.5279605	0.634868421	0.55098684	0.64802631	0.6776315	0.75822368	0.77
L2 Logistic Regression	0.6595394	0.465460526	0.53618421	0.546052631	0.470394	0.463815	0.5263157894	0.555921	0.48026315	0.53618421	0.64473684	0.6776315	0.76151315	0.77
Stochastic Gradient Descent	0.5921052	0.563758389	0.5231788079	0.54304635	0.557046	0.5625	0.5328947368	0.5751633	0.6298701298	0.538961	0.48519736	0.4884868	0.53618421	0.52
MLP Classifier	0.6924342	0.5509868421	0.53618421	0.53947368	0.521381	0.565789	0.5625	0.5526315	0.6299342105	0.5509868421	0.5805921	0.6299342	0.61348684	0.63

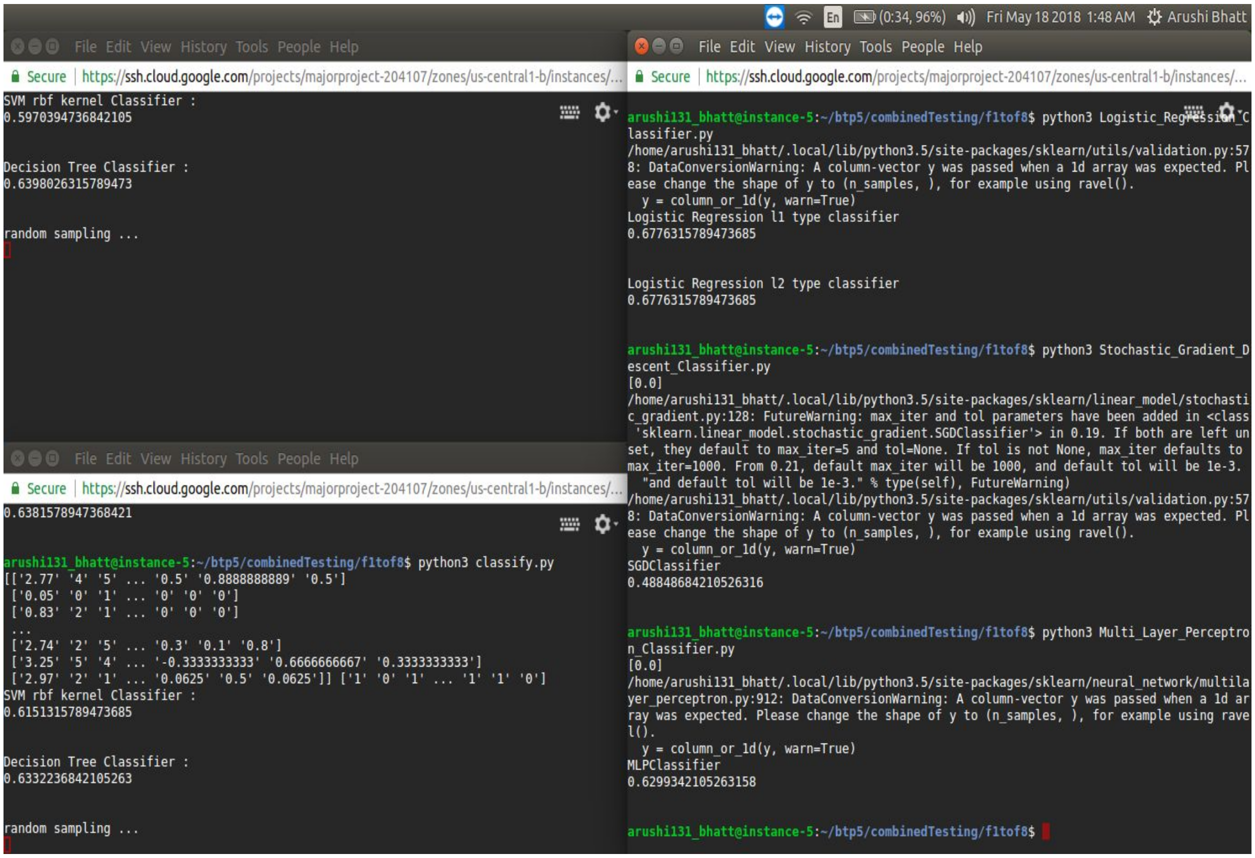
CHAPTER 6

RESULTS AND ANALYSIS USING DATA VISUALIZATION

6.1 Checking Results for different classifiers:

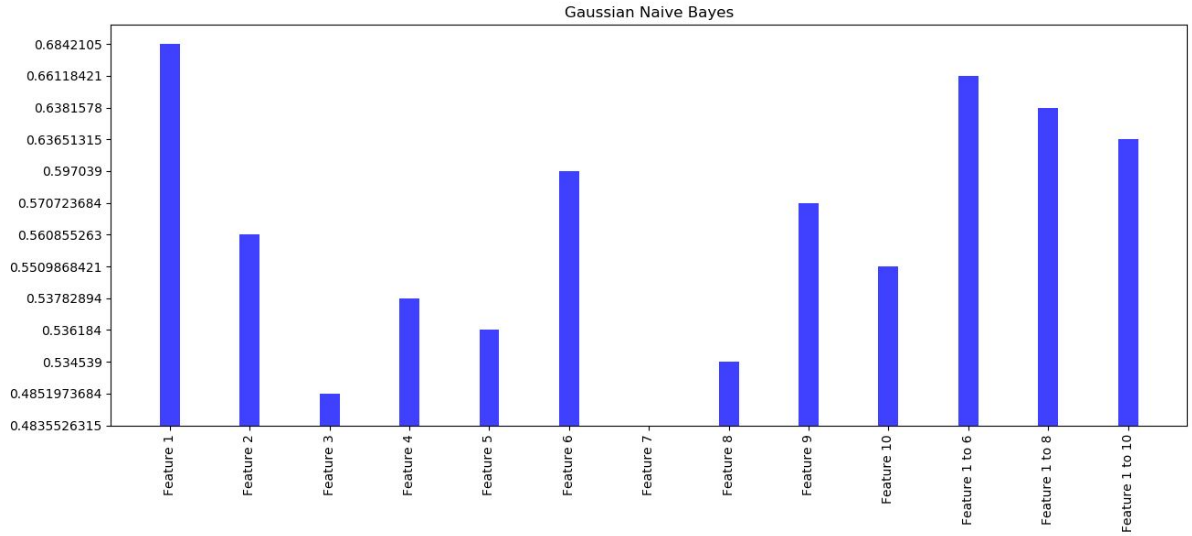
Following figure is the sample of program running on google cloud :

Figure 5. Google Cloud



a. Naive Bayes

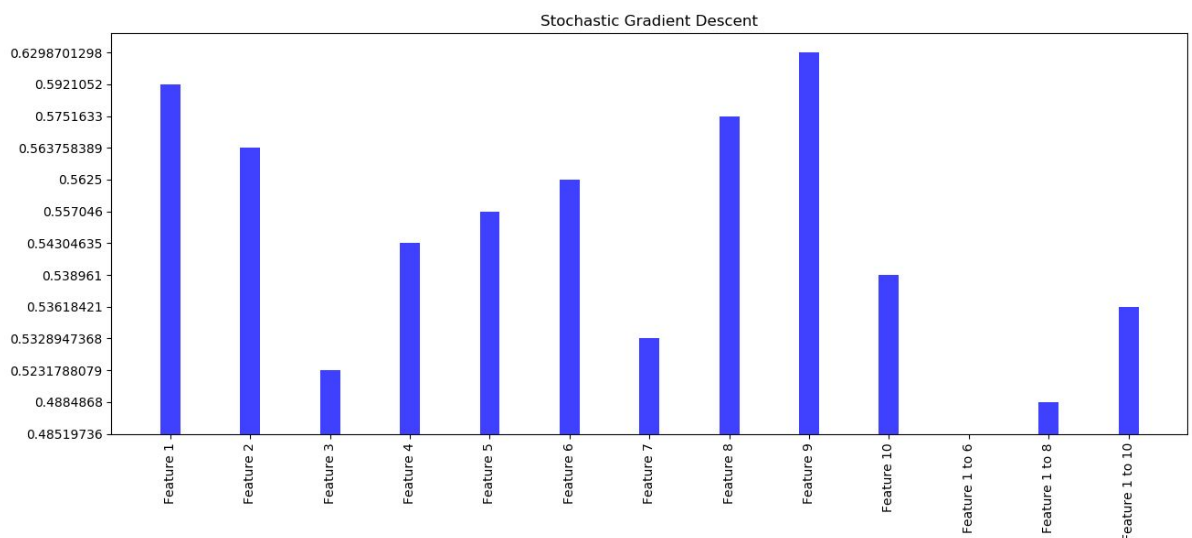
Figure 6.Naive Bayes Classifier



Here we can see that feature 1 Contrasting Connotation has performed well. And feature 7 has underperformed.

b. Stochastic Gradient Descent

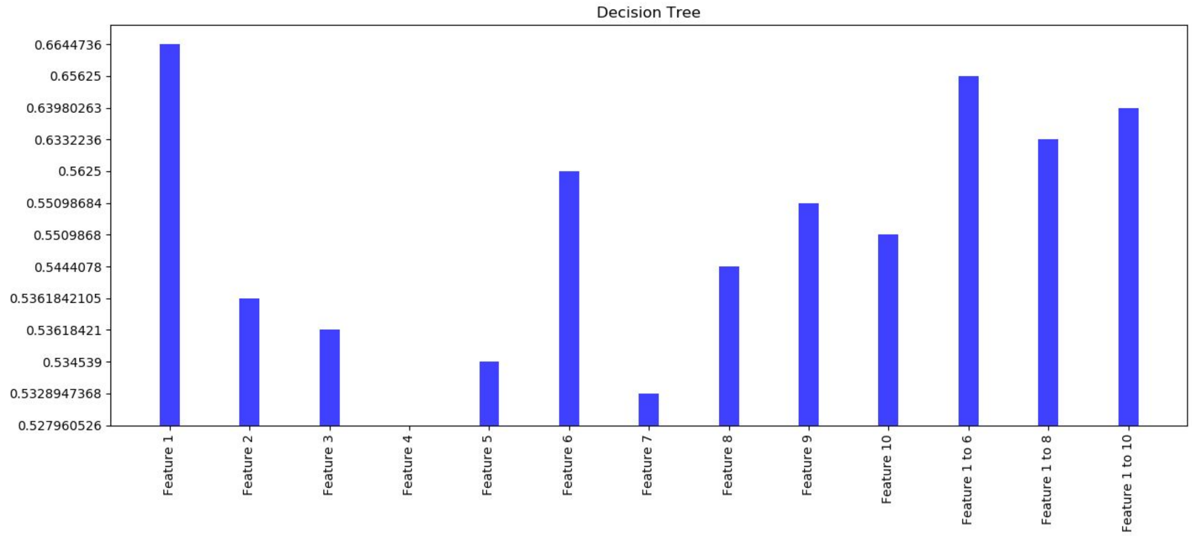
Figure 7.Stochastic gradient Classifier



Here we can see that the combination of feature 1 to 6 has underperformed but feature 9 has performed well.

c. Decision Tree

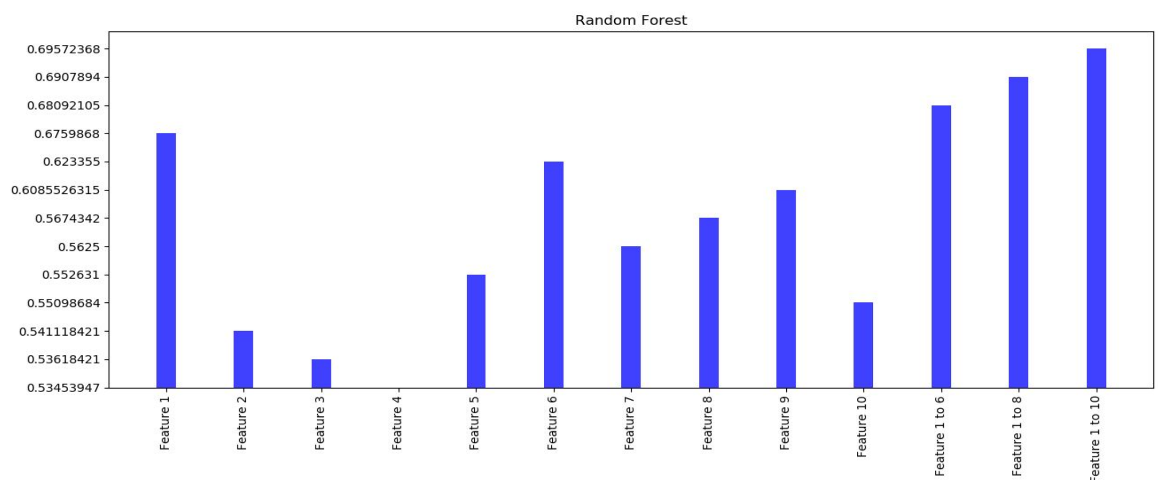
Figure 8. Decision Tree Classifier



Here we can see that feature 1 Contrasting Connotation has performed well. And feature 4 has underperformed.

d. Random Forest

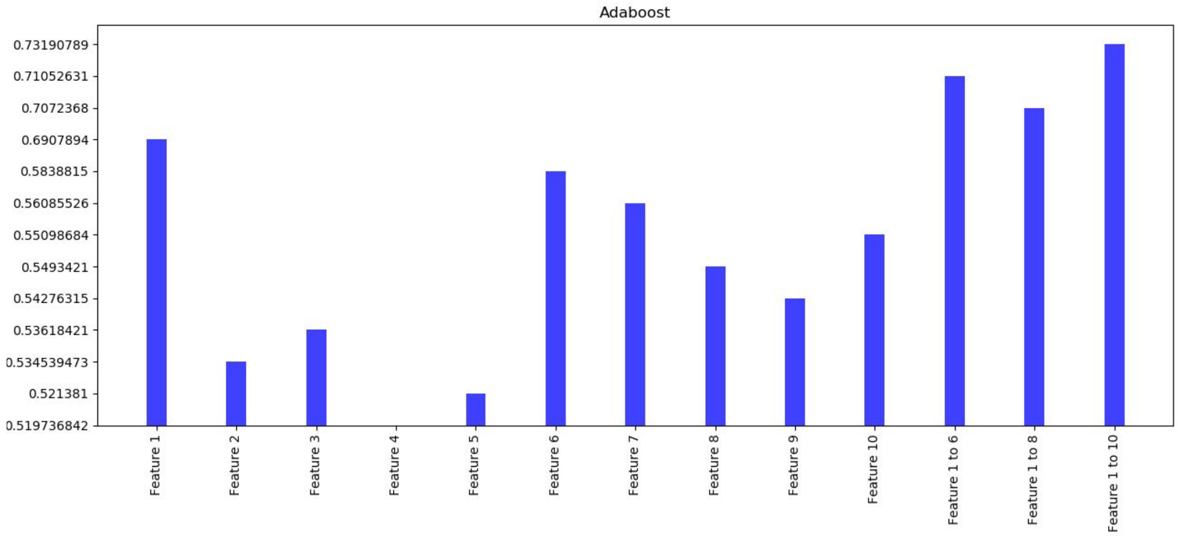
Figure 9 : Random Forest



Here feature 4 has underperformed and combined features 1 to 10 has performed well.

e. Adaboost

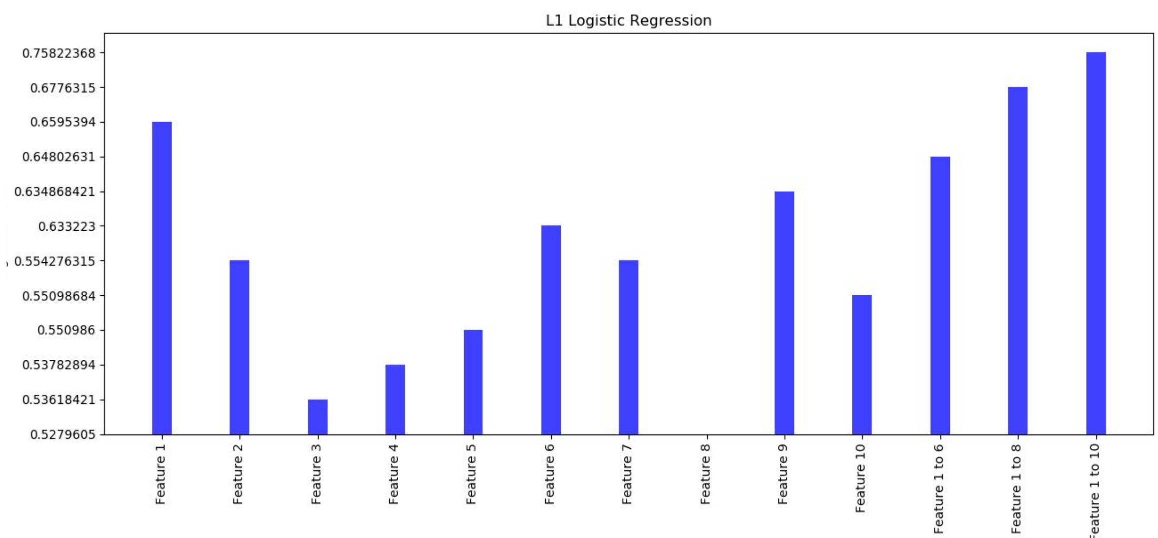
Figure 10. Adaboost Classifier



Here we can see that feature 1 to 10 has performed well. And feature 4 Structural Variation has underperformed.

f. L1 regularised Logistic Regression

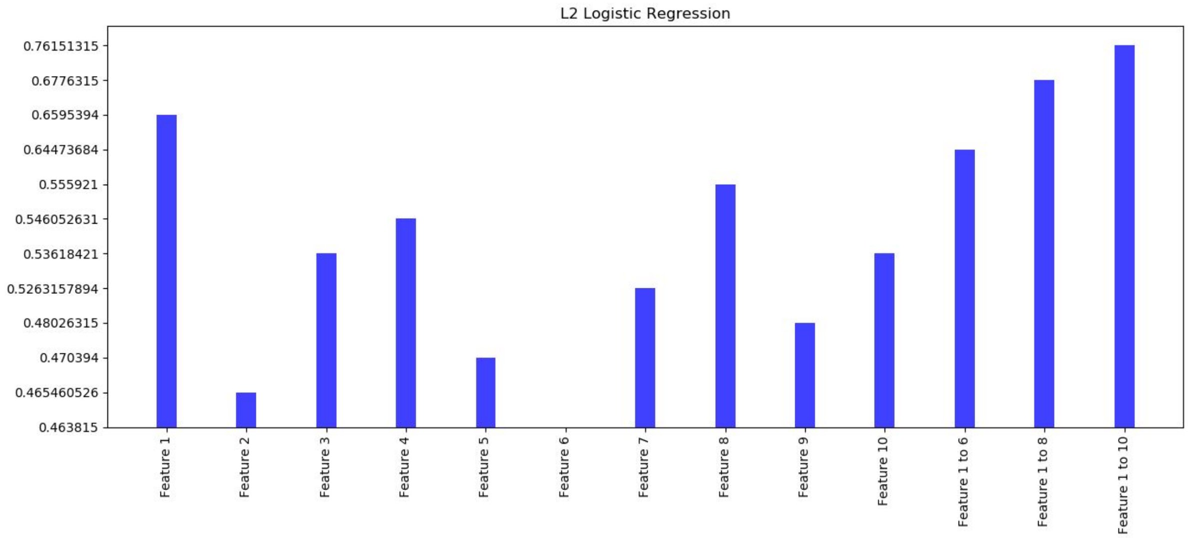
Figure 11. Logistic Regression L1 Classifier



Here we can see that feature 1 to10 has performed well. And feature 8 Polarity and Subjectivity has underperformed.

g. L2 regularised Logistic Regression

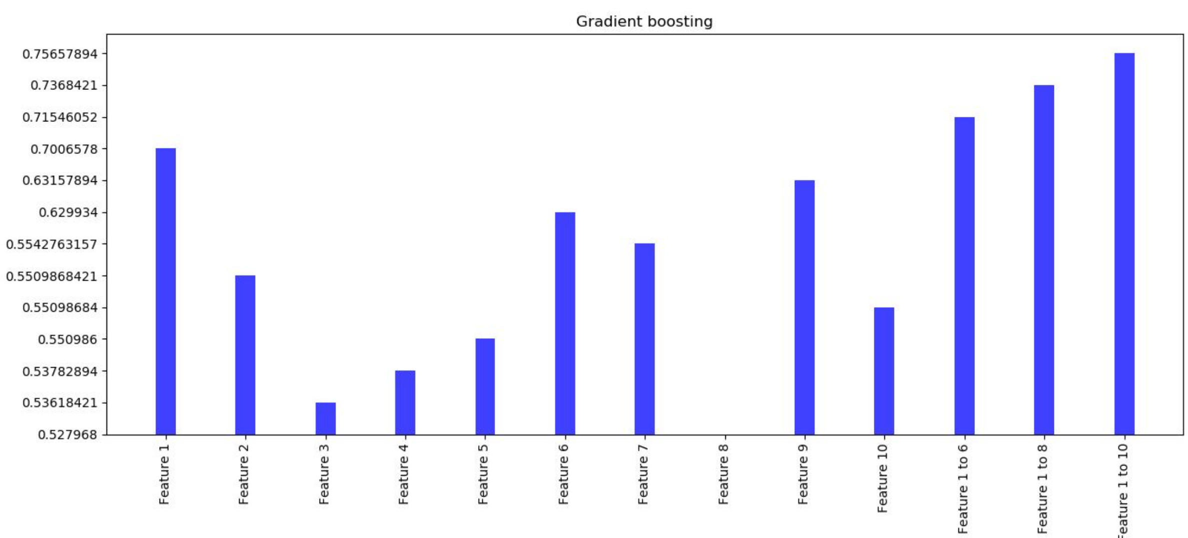
Figure 12 Logistic Regression L2 classifier



Here we can see that feature 1 to 10 has performed well. And feature 6 Word Length Distribution has underperformed.

h. Gradient Boosting

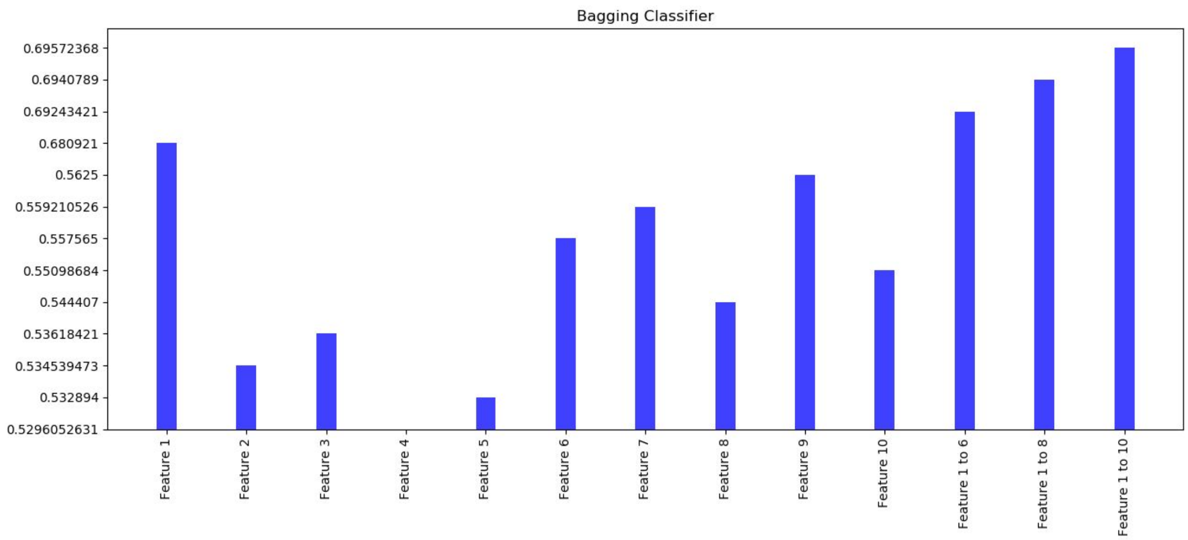
Figure 13. Gradient Boosting Classifier



Here we can see that feature 1 to 10 has performed well. And feature 8 Polarity and Subjectivity has underperformed.

i. Bagging Model

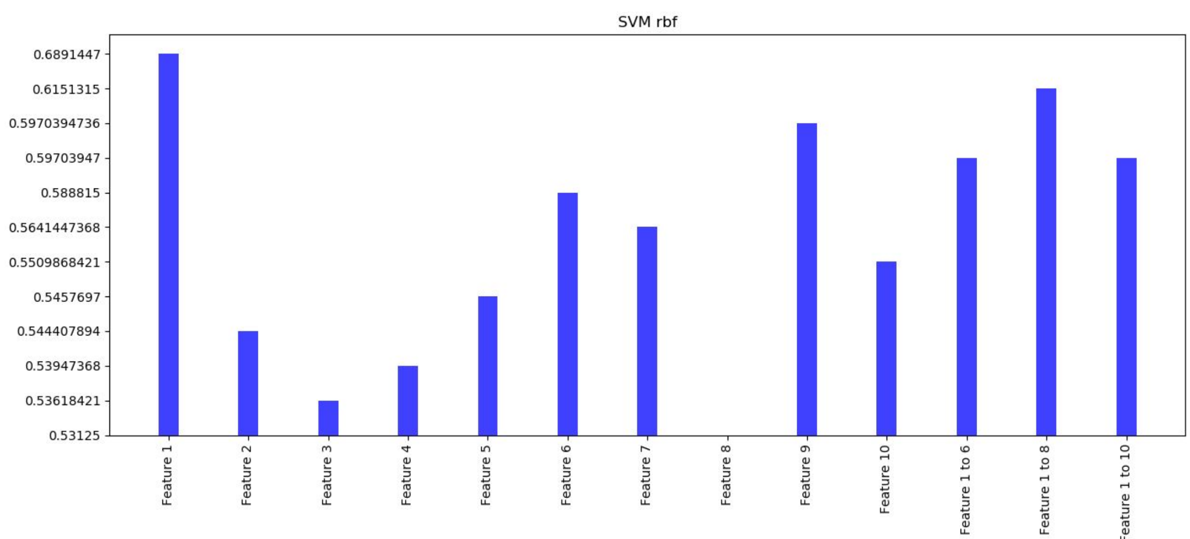
Figure 14.Bagging Classifier



Here we can see that feature 1 to 10 has performed well. And feature 4 Structural Variation has underperformed.

j. SVM (rbf kernel)

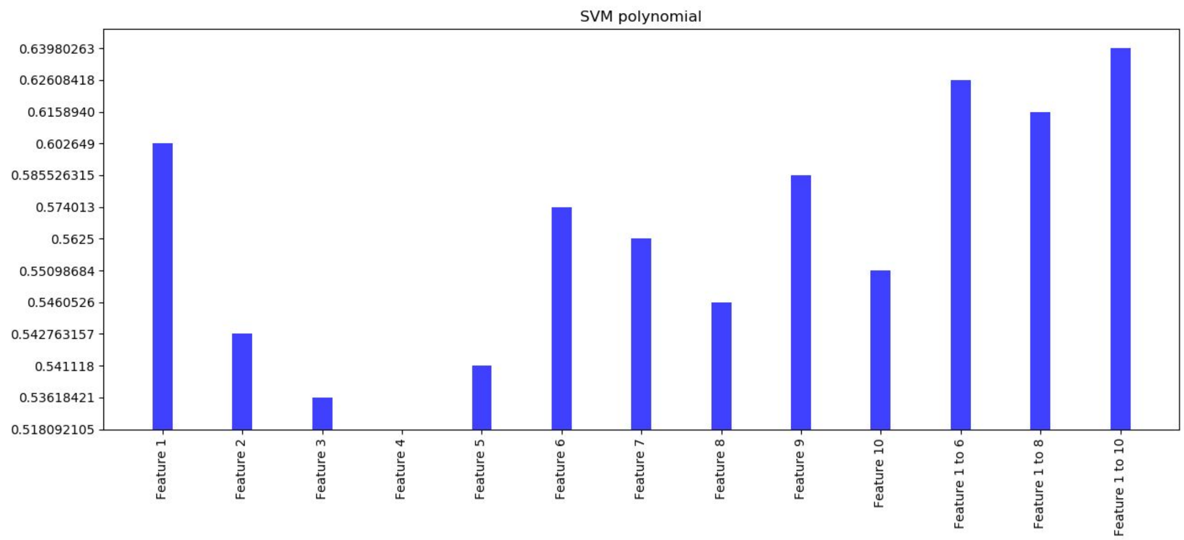
Figure 15. SVM(rbf) Classifier



Here we can see that feature 1 Contrasting Connotation has performed well. And feature 8 has underperformed.

k. SVM (polynomial kernel)

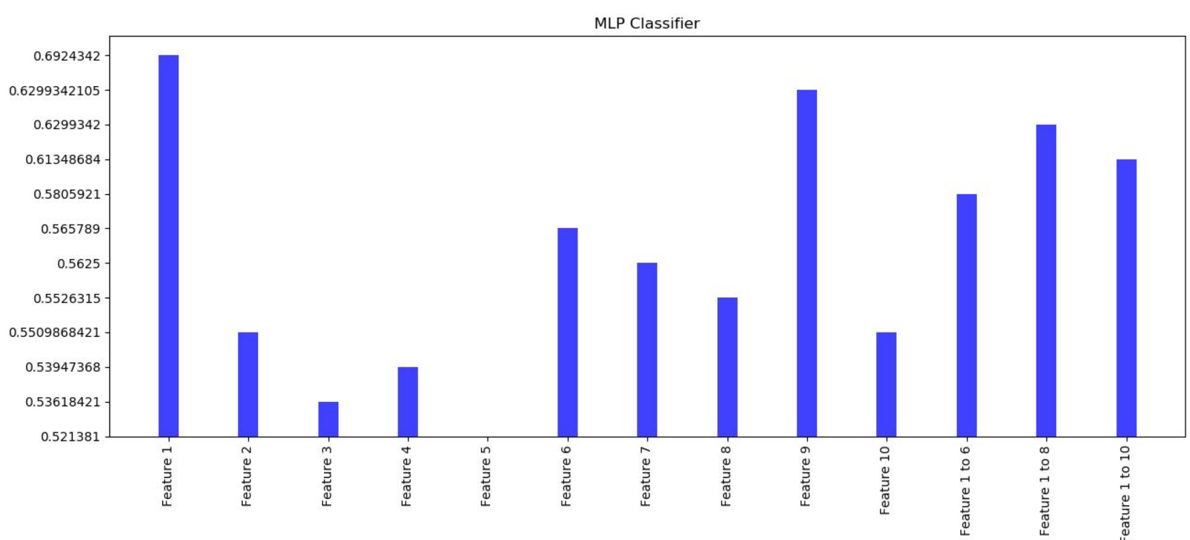
Figure 16. SVM(poly) Classifier



Here we can see that feature 1 to 10 has performed well. And feature 4 has underperformed.

I. Multi Layer Perceptron

Figure 17. MultiLayer Perceptron Classifier



Here we can see that feature 1 Contrasting Connotation has performed well. And feature 5 has underperformed.

6.2 10-Cross Fold Validation

Figure 18. Comparison with 10- cross fold validation

Cross-validation is a technique to evaluate predictive models by partitioning the riginal sample into a training set to train the model, and a test set to evaluate it. In k-fold **Cross-validation**, the original sample is randomly partitioned into k equal size subsamples. Here we have taken k as 10.

Here we note that 10 -cross fold validation technique has improved slightly in cases of logistic regression l1 regularization, Random Forest, MLP Classifier but there are some cases where we can’t see any improvement as evident in the case of SVM polynomial.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1. Conclusion :

As is evident from the results described above, that including social traits for sarcasm detection framework gives pretty robust results. This clearly indicates that incorporating features that describe the psychological and behavioral aspects of the user goes a long way in helping the process of automatic identification of sarcasm.

Improved preprocessing definitely helped in better feature extraction. Also different features have helped in better training of classifier models. We can easily see the what type of feature outperformed of underperformed using Data Visualization Techniques.

Optimization done through 10-fold cross validation improvised the scores of most of the algorithms.

7.2. Future Work

Because of the encouraging results obtained, future work could definitely be pursued in the direction of expanding the feature set in order to include more features that are expressive of the user's behaviour. Moreover, the detection of sarcasm in tweets is limited, in the sense that the sentences are of limited sizes (Twitter has a limit of 140 characters per tweet). The same approach could be expanded to detect sarcasm on other social media platforms (To classify posts as sarcastic on facebook for example). This could go a long way in restricting the spread of fake news on account of people not recognizing the posts as sarcastic. Given more time and more computational power, a greater range of algorithms could be applied, more features could be generated and processed, and visualization too could be employed in order to get a better grasp of the results obtained.

Finally, as the approach of incorporating behavioral analysis successfully improves sarcasm detection, the same could be employed in order to identify equally (or perhaps even more)complex forms of expression such as humor, satire, etc. Moreover, the dataset of sarcastic tweets has been created by extracting tweets having #sarcasm and #not tags. The expectation of the users, while writing this tweet, might have been that without these hashtags, identifying the tweets as sarcastic might be very difficult. Hence, there is a strong possibility that our dataset of sarcastic tweets may be biased towards the hardest forms of sarcasm. Given more time, a better approach to building an unbiased dataset of sarcastic tweets can be developed. On an ending note, the

current approach works on a static dataset. The possibility of adding incremental classification capabilities could also be attempted in the future.
We observe the following emerging directions:

(1)Sarcasm annotation:Sarcasm is understood on the basis of shared knowledge. As shown in [43], sarcasm is closely related to language/culture-speci c traits. Future approaches to sarcasm detection in new languages will benefit from understanding such traits, and incorporating them into their classification frameworks. [30] show that American and Indian annotators may have substantial disagreement in their sarcasm annotations. However, this sees a non-signi cant degradation in the performance of sarcasm detection. Since crowd-sourcing may be used for sarcasm annotation, the quality of this annotation and its impact on sarcasm classi cation must be evaluated on the basis of critical parameters such as cultural backgrounds.

(2) Extraction of implicit sentiment in patterns:Basedonpastwork,it is well-established that sarcasm is closely linked to sentiment incongruity [32]. Several related works exist for detection of implicit sentiment in sentences, as in the case of ' e phone gets heated quickly' v/s ' e induction cooktop gets heated quickly'. is will help sarcasm detection, following the line of semi-supervised pa ern discovery.

(3) Analysis based on types of sarcasm: As noted in the survey, past work does not report which of the types of sarcasm are correctly handled by existing systems. A dataset which labels sarcastic sentences into one of the four types, and then studies the performance of various systems on each of these types will be helpful. Future work can benefit from reporting which types of sarcasm are proving to be di cult for diffierent approaches.

(4) Sarcasm versus irony classi cation: Sarcasm and irony are closely related and most work so far considers them to be the same. However, some recent work has dealt with understanding the differences between the two. [74] present endings of a data analysis to understand differences between sarcasm and irony. According to them, aggressiveness is the distinguishing factor between the two. [66] present a set of classi ers that distinguish between sarcasm and irony. ey describe an analysis of structural and a ective features in tweets. An important observation that they make is the peculiarity of the hashtag '#not' as a negation marker for sarcasm.

(5) Linguistic basis for sarcasm detection: Many sarcasm theories, except the theory of dropped negation (described in Section 2) have not been explored as means for sarcasm detection. [66] show that the hashtag '#not' plays a distinct role in sarcastic tweets. is may have correlations with this theory of dropped negation. Approaches grounded in linguistic theories may yield good results.

(6) Coverage of different forms of sarcasm: In Section 2, we described four species of sarcasm: propositional, lexical, like-pre xed and illocutionary sarcasm. We observe that current approaches are limited in handling the last two forms of sarcasm: like-pre xed and illocutionary. Future work may focus on these forms of sarcasm.

(7) Extraction of Contextual Information using Deep learning-based architectures: Very few approaches have explored deep learning-based architectures so far. As shown in [63], context embeddings can be captured. Embeddings derived from other forms of context may be useful to capture the additional shared knowledge (say, user or conversation- speci c knowledge) that is required to understand certain forms of sarcasm.

Appendix

A. TWITTER FUNDAMENTALS

Twitter is an online social networking site where users may connect with each other and post messages called “tweets”. Each tweet may have a maximum of 140 characters. In Twitter, the social network is directed, that is, a user may connect to other users by simply following them. Any user may follow any other user (unless the account is protected) without explicit consent. The users following a user are called followers, while users that are being followed by a user are called friends in Twitter parlance.

Each user has a “timeline” which contains tweets from accounts that the user follows in reverse chronological order. It is important to note that Twitter, unlike Facebook, does not filter or algorithmically curate timelines. A user may share tweets from other users with their followers by retweeting. This feature is similar to sharing posts on Facebook. Similar to the “Like” feature in Facebook, users may “favorite” a tweet. A user may also reply to tweets and mention other users using the “@” symbol.

An interesting feature which was quite unique to Twitter but is now prevalent throughout the online social networking sphere is the usage of hashtags. Hashtags, though quite common in Internet Relay Channels, were not initially part of Twitter’s design. They were conceived by Twitter users as a way to group tweets and users on topics of interest. Nowadays, hashtags are ubiquitous and function as a simple mechanism to converse with specific groups, search for specific topics, make tweets more visible etc. More recently, hashtagged tweets also function as cheap, readily available labelled data for supervised learning algorithms.

A more detailed introduction to Twitter is available here1. The book “Twitter Data Analytics” (Kumar et al., 2014) is an excellent resource detailing how to obtain data from Twitter using their APIs. Shown in figure 5.1, a sample tweet for illustration purposes.

B. Google Cloud

With Google Cloud Platform (GCP), you can build, test, and deploy applications on Google's highly-scalable and reliable infrastructure for your web, mobile, and backend solutions

GCP resources

GCP consists of a set of physical assets, such as computers and hard disk drives, and virtual resources, such as virtual machines (VMs), that are contained in Google's data centers around the globe. Each data center location is in a global *region*. Regions include Central US, Western Europe, and East Asia. Each region is a collection of *zones*, which are isolated from each other within the region. Each zone is identified by a name that combines a letter identifier with the name of the region. For example, zone `a` in the East Asia region is named `asia-east1-a`.

This distribution of resources provides several benefits, including redundancy in case of failure and reduced latency by locating resources closer to clients. This distribution also introduces some rules about how resources can be used together.

Accessing resources through services

In cloud computing, what you might be used to thinking of as software and hardware products, become *services*. These services provide access to the underlying resources. The list of available GCP services is long, and it keeps growing. When you develop your website or application on GCP, you mix and match these services into combinations that provide the infrastructure you need, and then add your code to enable the scenarios you want to build

The Google Cloud Platform Console provides a web-based, graphical user interface that you can use to manage your GCP projects and resources. When you use the GCP Console, you create a new project, or choose an existing project, and use the resources that you create in the context of that project. You can create multiple projects, so you can use projects to separate your work in whatever way makes sense for you. For example, you might start a new project if you want to make sure only certain team members can access the resources in that project, while all team members can continue to access resources in another project.

REFERENCES

1.BingLiu.2010.Sentimentanalysisandsubjectivity.Handbookofnaturallanguageprocessing2(2010),627–666.

2. González-Ibánez, Roberto, Smaranda Muresan, and Nina Wacholder. "Identifying sarcasm in Twitter: a closer look." *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers-Volume 2*.

3. Davidov, D., O. Tsur and A. Rappoport, “Semi-supervised recognition of sarcastic sentences in twitter and amazon”, in “Proceedings of the Fourteenth Conference on Computational Natural Language Learning”, pp. 107–116 (Association for Computational Linguistics, 2010).

4. Pennebaker, J. W., M. E. Francis and R. J. Booth, “Linguistic inquiry and word count: Liwc 2001”, Mahway: Lawrence Erlbaum Associates p. 71 (2001).

5.Reyes, A., P. Rosso and D. Buscaldi, “From humor recognition to irony detection: The figurative language of social media”, Data & Knowledge Engineering 74, 1–12 (2012).

6. Liebrecht, C., F. Kunneman and A. van den Bosch, “The perfect solution for detecting sarcasm in tweets #not”, WASSA 2013 p. 29 (2013).

7.Riloff, E., A. Qadir, P. Surve and Silva, “Sarcasm as contrast between a positive sentiment and negative situation.”, in “EMNLP”, pp. 704–714 (ACL2013)URLhttp://dblp.uni-trier.de/db/conf/emnlp/emnlp2013.html#RiloffQSSGH13.

8.HPaulGrice,PeterCole,andJerryLMorgan.1975.Syntaxandsemantics.Logicand conversation3(1975),41–58.

9.RaymondWGibbs.1994.epoeticsofmind:Figurativethought,language,andu nderstanding.CambridgeUniversityPress.

10.OhnDCampbellandAlbertNKatz.2012.Aretherenecessaryconditionsforind ucingasenseofsarcasticirony?Discourse Processes 49, 6 (2012), 459–480.

11. Jodi Eisterhold, Salvatore A ardo, and Diana Boxer. 2006. Reactions to irony in discourse: Evidence for the least disruption principle. Journal of Pragmatics 38, 8 (2006), 1239–1256.

12. Deirdre Wilson. 2006. e pragmatics of verbal irony: Echo or pretence? Lingua 116, 10 (2006), 1722–1743.

13. ElisabethCamp.2012.Sarcasm,Pretense,andeSemantics/PragmaticsDistinc tion*.Nouˆs46,4(2012),587–634.

14.DmitryDavidov,OrenTsur,andAriRappoport.2010.Semi-supervisedrecog nitionofsarcasticsentencesintwi er and amazon. In Proceedings of the Fourteenth Conference on Computational Natural Language Learning. Association for Computational Linguistics, 107–116

15. Stacey L Ivanko and Penny M Pexman. 2003. Context incongruity and irony processing. Discourse Processes 35, 3 (2003), 241–279.

16. Dan Sperber. 1984. Verbal irony: Pretense or echoic mention? Journal of Experimental Psychology: General 113, 1 (1984), 130–136.

17.RogerJKreuzandSamGlucksberg.1989.Howtobesarcastic: eechoicremindertheoryofverbalirony.Journalof Experimental Psychology: General 118, 4 (1989), 374.

18.RachelGiora.1995.Onironyandnegation.Discourseprocesses19,2(1995), 239–264.

19.RaymondW.GibbsandJenniferO'Brien.1991.Psychologicalaspectsofirony understanding.JournalofPragmatic,16, 6 (1991), 523 – 530. DOI:h p://dx.doi.org/10.1016/0378-2166(91)90101-3

20.Yann LeCun,Yoshua Bengio& Geoffrey Hinton
https://www.nature.com/articles/nature14539

21. Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Prateek Vij **A Deeper Look into Sarcastic Tweets Using Deep Convolutional Neural Networks :** http://arxiv.org/abs/1610.08815

22. Alex Morales and ChengXiang Zhai
Identifying Humor in Reviews using Background Text Sources http://aclweb.org/anthology/D17-1051

23. Aditya Joshi, Pushpak Bhattacharyya, Mark James Carman **Automatic Sarcasm Detection: A Survey** https://arxiv.org/abs/1602.03426

24. Aditya Joshi Vaibhav Tripathi, Kevin Patel,Pushpak Bhattacharyya Mark Carman **Are Word Embedding-based Features Useful for Sarcasm Detection?** https://www.cse.iitb.ac.in/~pb/papers/emnlp16-sarcasm.pdf