

Sentiment Analysis on Twitter

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY

IN

COMPUTER SCIENCE ENGINEERING

Submitted By:

RAJAT KUMAR ARYA

(2K18/CSE/14)

Under the supervision of

PROF. RAJNI JINDAL
(H.O.D. CSE)

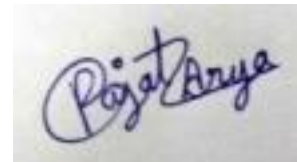


DEPARTMENT OF INFORMATION TECHNOLOGY
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

JUNE, 2020

CANDIDATE'S DECLARATION

I, Rajat Kumar Arya (2K18/CSE/14) student of M.Tech Computer Science Engineering, hereby declare that the project Dissertation titled "Sentiment Analysis on Twitter" which is submitted by me to the Department of Information Technology, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any degree, Diploma Associateship, Fellowship or other similar title or recognition.

A handwritten signature in blue ink that reads "Rajat Arya". The signature is written in a cursive style with a large initial 'R'.

Place: Delhi

Rajat Kumar Arya

Date:

CERTIFICATE

I hereby certify that the Project Dissertation titled “Sentiment Analysis on twitter” which is submitted by Rajat Kumar Arya, Roll No 2K18/CSE/14 Computer Science Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi

PROF. RAJNI JINDAL

Date:

SUPERVISOR

ACKNOWLEDGEMENT

I express my gratitude to my major project guide PROF. RAJNI JINDAL, Cse Dept., Delhi Technological University, for the valuable support and guidance she provided in making this major project. It is my pleasure to record my sincere thanks to my respected guide for her constructive criticism and insight without which the project would not have shaped as it has.

I humbly extend my words of gratitude to other faculty members of this department for providing their valuable help and time whenever it was required.

RAJAT KUMAR ARYA

Roll No. 2K18/CSE/14

M.Tech (Computer Science Engineering)

E-mail: -rajat1994arya@gmail.com

ABSTRACT

Be that as it may, In this current period, online life assumes a significant job in information trade, sharing their contemplations. Enthusiastic Effect of an individual keeps up a significant job on their everyday life. Assessment Analysis is a strategy of breaking down the feelings and extremity of considerations of the individual. Twitter is a primary stage on sharing the thought's, supposition and estimations on various events. Twitter Sentimental Analysis is strategy for investigating the feelings from tweets (message posted by client in twitter). Tweets are useful in extricating the Sentimental qualities from the client. The information give the Polarity sign like positive, negative or fair-minded qualities. It is centered around the individual's tweets and the hash labels for understanding the circumstances in every part of the rules. The paper is to investigate the celebrated individual's id's or hash labels for understanding the outlook of individuals in every circumstance when the individual has tweeted or has followed up on certain occurrences. The proposed framework is to break down the conclusion of the individuals utilizing python, twitter API, Unigrams + Bigrams + Trigrams, prepared on Naive Bayes Classifier. As the outcomes it serves to investigation the post with a superior precision.

Keywords: Unigrams ,Bigrams , Trigrams, prepared on Naive Bayes Classifier

.

CONTENTS

<u>CANDIDATE’S DECLARATION</u>	ii
<u>CERTIFICATE</u>	iii
<u>ACKNOWLEDGEMENT</u>	iv
<u>ABSTRACT</u>	v
<u>CONTENTS</u>	vi
<u>List of Figures</u>	viii
<u>List of Tables</u>	iError! Bookmark not defined.
<u>List of Formulas</u>	Error! Bookmark not defined.
<u>CHAPTER 1 INTRODUCTION</u>	ix
1.1 <u>APPLICATIONS OF SENTIMENT ANALYSIS</u>	1
1.7 <u>CHARACTERSISTICS FEATURES OF TWEETS</u>	2
<u>CHAPTER 2 RELATED WORK</u>	3
2.1 Go, Bhayani and Huang (2009)	4
<u>2.2</u> Pak and Paroubek (2010)	4
<u>2.3</u> Koulompis, Wilson and Moore (2011)	4
<u>2.4</u> Saif, He and alani (2012)	4
<u>CHAPTER 3 THE EXPERIMENTAL APPROACH</u>	6
<u>3.1</u> <u>DATASETS</u>	6
<u>3.1.1</u> TWITTER SENTIMENT CORPUS	9
<u>3.1.2</u> STANFORD TWITTER	10
<u>3.2</u> <u>PRE PROCESSING</u>	10
<u>3.2.1</u> HASTAGS	11
<u>3.2.2</u> HANDLES	11

<u>3.2.3</u>	URLS	11
<u>3.2.4</u>	EMOTICONS	Error! Bookmark not defined.
<u>3.2.5</u>	PUNCTUATIONS	14
<u>3.2.6</u>	REPEATING CHARACTERS	14
<u>3.3</u>	<u>STEMMING ALGORITHM</u>	15
<u>3.3.1</u>	PORTER STEMMER	16
<u>3.3.2</u>	LEMMATIZATION	16
<u>3.1</u>	<u>FEATURES</u>	16
<u>3.4.1</u>	UNIGRAMS	16
<u>3.4.2</u>	N-GRAMS	17
<u>3.4.3</u>	NEGATION HANDLING	19
	<u>CHAPTER 4 EXPERIMENTAL RESULTS</u>	16
<u>4.1</u>	<u>NAÏVE BAYES</u>	29
<u>4.2</u>	<u>MAXIMUM ENTROPY CLASSIFIER</u>	30
	<u>CHAPTER 5 CONCLUSION AND FUTURE WORK</u>	Error!
	Bookmark not defined.	
	<u>References</u>	33
	<u>LIST OF PUBLICATIONS OF THE CANDIDATE'S WORK</u>	34

List of Figures

<u>Figure 1.1</u>	Sentiment Analysis of influences political decision results	2
<u>Figure 3.1</u>	Schematic Block Representation of the Methodology	6
<u>Figure 3.2</u>	Histogram of the tweets as indicated by their estimation	8
<u>Figure 3.3</u>	Cumulative frequency Graph plot for 50 Most Frequent Unigrams	17
<u>Figure 3.4</u>	Number of n-grams vs. Number of Tweets	18
<u>Figure 3.5</u>	Number of rehashing n-grams vs Number of Tweets	18
<u>Figure 3.6</u>	Scope of Negation	20
<u>Figure 3.7</u>	K-fold cross-validation	21
<u>Figure 3.8</u>	Result of Naïve bayes Classifier and K-fold cross-validation	21
<u>Figure 3.9</u>	Example of Confusion Matrix	22
<u>Figure 3.10</u>	Color Map of the confusion matrix related to the Naïve Bayes	23
<u>Figure 3.11</u>	Tags in the data set with their corresponding count	24
<u>Figure 3.12</u>	Result of the naïve bayes classifier with stopwords removed	24
<u>Figure 3.13</u>	Result of the Naïve bayes classifier after stemming Error! Bookmark not defined.	
<u>Figure 3.14</u>	Result of the naïve bayes classifier with bigram features	28
<u>Figure 3.15</u>	Result of the naïve bayes classifier with unigram and bigram features	28
<u>Figure 4.1</u>	Accuracy for Naïve Thomas bayes Classifier	29
<u>Figure 4.2</u>	Precision vs Recall for Naïve Bayes Classifier	30
<u>Figure 4.3</u>	Precision vs Recall for Max Entropy Classifier	31

List of Tables

<u>Table 3.1:</u> Twitter posts commented on with their comparing supposition	7
<u>Table 3.2:</u> Twitter Sentiment Corpus	10
<u>Table 3.3:</u> Stanford Corpus	10
<u>Table 3.4:</u> Frequency of features per Tweet	11
<u>Table 3.5:</u> List of Emotions	12
<u>Table 3.6:</u> Before Processing Emotions	12
<u>Table 3.7:</u> After Processing Emotions	Error! Bookmark not defined.
<u>Table 3.8:</u> Tweets before processing URLs.	Error! Bookmark not defined.
<u>Table 3.9:</u> Tweets after Processing URLs	Error! Bookmark not defined.
<u>Table 3.10:</u> List of Punctuations	Error! Bookmark not defined.
<u>Table 3.11:</u> Number of Words Initial and Final pre-processing	Error! Bookmark not defined.
<u>Table 3.12:</u> Porter Stemming Steps	Error! Bookmark not defined.
<u>Table 3.13:</u> Explicit Negation Cues	20
<u>Table 3.14:</u> Most frequent words in the data set with their corresponding count	23

List of Formulas

<u>Formula 3.1:</u> F1 score	21
<u>Formula 3.2:</u> Precision	22
<u>Formula 3.3:</u> Recall	22
<u>Formula 3.4:</u> Perplexity and Entropy to evaluate language models	Error! Bookmark not defined.
<u>Formula 3.5:</u> General form of N-grams	25
<u>Formula 3.6:</u> MLE of N-grams	26
<u>Formula 3.7:</u> Unigram	26
<u>Formula 3.8:</u> Bigram	26
<u>Formula 3.9:</u> MLE for Unigram	26
<u>Formula 3.10:</u> MLE for Bigram	26
<u>Formula 3.11:</u> Objective Function of N-gram	27
<u>Formula 3.12:</u> Objective Function Rewritten using baye's rule of N-gram	27

CHAPTER 1 INTRODUCTION

1.1 Applications of Sentiment Analysis

Conclusion Analysis discovers its application in an assortment of spaces.

A. Online Commerce

The most broad utilization of slant examination is in web based business exercises. Sites permits their clients to present their experience about shopping and item characteristics. They give rundown to the item and various highlights of the item by appointing evaluations or scores. Clients can without much of a stretch view assessments and proposal data on entire item just as explicit item includes. Graphical outline of the general item and its highlights is introduced to clients. Well known dealer sites like amazon.com gives audit from editors and furthermore from clients with rating data. <http://tripadvisor.in> is a mainstream site that gives audits on inns, travel goals. They contain 75 millions conclusions and surveys around the world. Conclusion investigation helps such sites by changing over disappointed clients into advertisers by examining this enormous volume of feelings.

B. Voice of the Market (VOM)

Voice of the Market is tied in with figuring out what clients are feeling about items or administrations of contenders. Precise and ideal data from the Voice of the Market helps in increasing upper hand and new item improvement. Discovery of such data as right on time as potential aides in direct and target key showcasing efforts. Emotion Analysis encourages corporate to hear client point of view continuously. This constant data encourages them to structure new advertising procedures, improve item includes and can anticipate odds of item disappointment. Zhang et al proposed shortcoming discoverer framework which can assist makers with finding their item shortcoming from Chinese surveys by utilizing viewpoints based opinion investigation. There are some business and free notion examination administrations are accessible, Radiant6, Sysomos, Viralheat, Lexalytics, and so forth are business administrations. Some free apparatuses like www.tweetfeels.com, www.socialmention.com are additionally accessible.

C. Voice of the Customer (VOC)

Voice of the Customer is worry about what singular client is stating about items or administrations. It implies breaking down the surveys and input of the clients. VOC is a key component of Customer Experience Management. VOC helps in recognizing new open doors for item innovations. Separating client sentiments additionally recognizes utilitarian necessities of the items and some non-useful prerequisites like execution and cost.

D. Brand Reputation Management

Brand Reputation Management is worry about dealing with your notoriety in showcase. Conclusions from clients or some other gatherings can harm or improve your notoriety. Brand Reputation Management (BRM) is an item and friends concentrated as opposed to client.

Presently, one-to-numerous discussions are occurring on the web at a high rate. That makes open doors for associations to oversee and reinforce brand notoriety. Presently Brand discernment is resolved not just by publicizing, advertising and corporate informing. Brands are presently a whole of the discussions about them. Opinion investigation helps in deciding how organization's image, item or administration is being seen by network on the web.

E. Government

Feeling investigation helps government in surveying their quality and shortcomings by breaking down assessments from open. For instance, "If this is the state, how would you anticipate that fact should come out? The MP who is examining 2g trick himself is profoundly degenerate.". this model plainly shows negative supposition about government. Regardless of whether it is following residents' suppositions on another 108 framework, distinguishing qualities and shortcomings in an enlistment battle in government work, surveying accomplishment of electronic accommodation of assessment forms, or numerous different regions, we can see the potential for estimation investigation.



Figure 1.1: Sentiment Analysis can be valuable to see how the state of mind of the open influences political decision results

1.2 Characteristic features of Tweets

From the point of view of Sentiment Analysis, we talk about a couple of attributes of Twitter:

Length of a Tweet The most extreme length of a Twitter message is 140 characters. This implies we can essentially believe a tweet to be a solitary sentence, bereft of complex linguistic develops. This is an immense distinction from customary subjects of Sentiment Analysis, for example, film surveys.

Language utilized Twitter is utilized by means of an assortment of media including SMS and cell phone applications. Along these lines and the 140-character limit, language utilized in Tweets tend be increasingly informal, and loaded up with slang and incorrect spellings. Utilization of hashtags additionally picked up ubiquity on Twitter and is an essential component in some random tweet. Our examination shows that there are around 1-2 hashtags per tweet, as appeared in Table 3 .

Information accessibility Another distinction is the extent of information accessible. With the Twitter API, it is anything but difficult to gather a large number of tweets for preparing. There likewise exist a couple datasets that have consequently and physically named the tweets [2] [3].

Domain of topics People frequently post about their preferences via web-based networking media. This makes twitter an exceptional spot to display a conventional classifier rather than space explicit classifiers that could be construct datasets, for example, film reviews.

CHAPTER 2 RELATED WORK

In this section, we will present related work which addresses the classification and segmentation task and related work.

2.1 Go, Bhayani and Huang (2009)

They arrange Tweets for a question term into negative or positive supposition. They gather preparing dataset naturally from Twitter. To gather positive and negative tweets, they question twitter for upbeat and miserable emojis.

- Happy emojis are various adaptations of grinning face, as ":", ":-)", ":)", ":D", "=:)" and so forth.
- Sad emojis incorporate grimaces, as ":(", ":-(", ":((" and so on.

They attempt different highlights – unigrams, bigrams and Part-of-Speech and train their classifier on different AI calculations – Naive Bayes, Maximum Entropy and Scalable Vector Machines and analyze it against a pattern classifier by tallying the quantity of positive and negative words from an openly accessible corpus. They report that Bigrams alone and Part-of-Speech Tagging are not useful and that Naive Bayes Classifier gives the best outcomes.

2.2 Pak and Paroubek (2010)

They distinguish that utilization of casual and imaginative language make supposition investigation of tweets a somewhat unique errand . They influence past work done in hashtags and notion investigation to construct their classifier. They use Edinburgh Twitter corpus to discover most continuous hashtags. They physically group these hashtags and use them to thus arrange the tweets. Aside from utilizing n-grams and Part-of-Speech highlights, they likewise construct a list of capabilities from previously existing MPQA subjectivity vocabulary and Internet Lingo Dictionary. They report that the best outcomes are seen with n-gram highlights with dictionary highlights, while utilizing Part-of-Speech highlights causes a drop in precision.

2.3 Koulompis, Wilson and Moore (2011)

They explored the use of phonetic focal point for acknowledge the slant of Twitter text. They acquire the worth of existing lexical resources similarly as features that get information about the easygoing and imaginative language used in microblogging. They received a controlled methodology to the issue, yet impact existing hashtags in the Twitter data for building planning data.

2.4 Saif, He and Alani (2012)

They talk about a semantic based way to deal with distinguish the substance being examined in a tweet, similar to an individual, association and so on. They likewise show that evacuation of stop words is certainly not a vital advance and may have bothersome impact on the classifier.

The entirety of the previously mentioned methods depend on n-gram highlights. It is muddled that the utilization of Part-of-Speech labeling is helpful or not. To improve exactness, some utilize various strategies for include choice or utilizing information about miniaturized scale blogging. Conversely, we improve our outcomes by utilizing increasingly fundamental methods utilized in Sentiment Analysis, such as stemming, two-advance grouping and nullification location and extent of invalidation.

Nullification location is a procedure that has frequently been concentrated in opinion investigation. Refutation words like "not", "never", "no" and so on can definitely change the importance of a sentence and thus the opinion communicated in them. Because of quality of such words, the significance of close by words gets inverse. Such words are supposed to be in the extent of refutation. Numerous explores have chipped away at recognizing the extent of refutation.

The extent of refutation of a prompt can be taken from that word to the following after accentuation. Council, McDonald and Velikovich (2010) talk about a method to distinguish refutation signs and their degree in a sentence. They distinguish unequivocal refutation signals in the content and for each word in the degree. At that point they discover its good ways from the closest negative sign on the left and right.

CHAPTER 3 THE EXPERIMENTAL APPROACH

We utilize distinctive capabilities and AI classifiers to decide the best blend for slant examination of twitter. We likewise try different things with different pre-preparing steps like - accentuations, emojis, twitter explicit terms and stemming. We researched the accompanying highlights - unigrams, bigrams, trigrams and nullification identification. We at long last train our classifier utilizing different AI calculations - Naive Bayes, Decision Trees and Maximum Entropy.

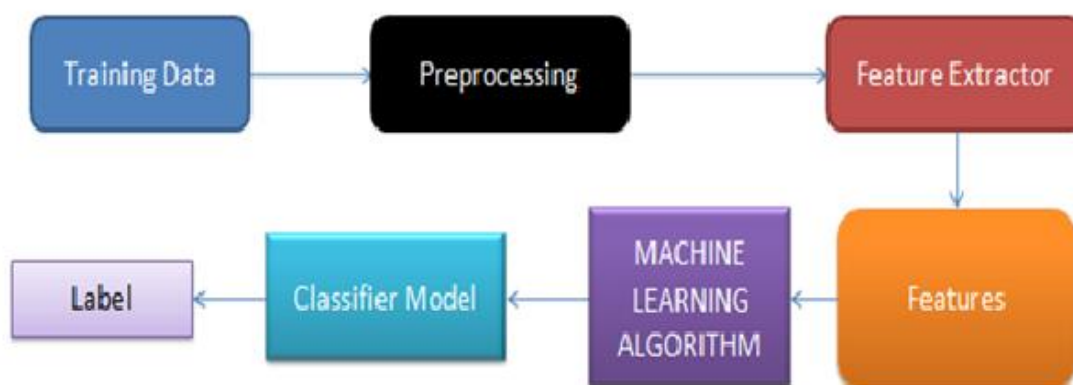


Figure 3.1: Schematic Block Representation of the Methodology

We utilize a modularized approach with include extractor and arrangement calculation as two free segments. This empowers us to explore different avenues regarding various alternatives for every part.

3.1 Datasets

To accumulate the information numerous alternatives are conceivable. In some past paper explores, they fabricated a program to gather consequently a corpus of tweets dependent on two classes, "positive" and "negative", by questioning Twitter with two sort of emojis:

- Happy emojis, for example, ":", ":P", ":-)" and so on.
- Sad emojis, for example, ":(", ":((", "=((".

Others make their own dataset of tweets my gathering and clarifying them physically which long and exacting.

Furthermore to discover a method of getting a corpus of tweets, we have to take of having a reasonable informational index, which means we ought to have an equivalent number of positive and negative tweets, yet it needs likewise to be sufficiently huge. In fact, more the information we have, more we can prepare our classifier and more the exactness will be.

After numerous investigates, I found a dataset of 1578612 tweets in english originating from two sources: Kaggle and Sentiment140. It is made out of four sections that are ItemID, Sentiment, SentimentSource and SentimentText. We are just intrigued by the Sentiment

segment comparing to our mark class taking a parallel worth, 0 if the tweet is negative, 1 if the tweet is certain and the SentimentText sections containing the tweets in a crude arrangement..

	ItemID	Sentiment	SentimentSource	SentimentText
0	1	0	Sentiment140	is so sad for my APL friend.....
1	2	0	Sentiment140	I missed the New Moon trailer...
2	3	1	Sentiment140	omg its already 7:30 :O
3	4	0	Sentiment140	.. Omgaga. Im sooo im gunna CRy. I've been at this dentist since 11.. I was supposed 2 just get a crown put on (30mins)...
4	5	0	Sentiment140	i think mi bf is cheating on me!!! T_T
5	6	0	Sentiment140	or i just worry too much?
6	7	1	Sentiment140	Juuuuuuuuuuuuuuuuuussst Chillin!
7	8	0	Sentiment140	Sunny Again Work Tomorrow :- TV Tonight
8	9	1	Sentiment140	handed in my uniform today . i miss you already
9	10	1	Sentiment140	hmmmm.... i wonder how she my number @-)

Table 3.1: Example of twitter posts commented on with their comparing supposition, 0 on the off chance that it is negative, 1 on the off chance that it is certain.

In the Table 3.1 indicating the initial ten twitter posts we would already be able to see a few particularities and troubles that we are going to experience during the preprocessing steps.

- The nearness of abbreviations "bf" or progressively confused "APL". Does it implies apple ? Apple (the organization) ? In this setting we have "companion" after so we could imagine that he alludes to his cell phone thus Apple, however shouldn't something be said about if "companion" was not here ?
- The nearness of groupings of rehashed characters, for example, "Juuuuuuuuuuuuuuuuuussst", "well". As a rule when we rehash a few characters in a word, it is to underscore it, to build its effect.
- The nearness of emojis, ":O", "T_T", ":-|" and significantly more, give bits of knowledge about client's states of mind.
- Spelling botches and "urban punctuation" like "im gunna" or "mi".
- The nearness of things, for example, "television", "New Moon".

Besides, we can likewise include,

- People likewise show their temperaments, feelings, states, between two, for example, *cries*, *hummin*, *sigh*.
- The refutation, "can't", "can't", "don't", "haven't" that we have to deal with as: "I don't care for chocolate", "like" for this situation is negative.

We could likewise be intrigued by the language structure of the tweets, or if a tweet is abstract/objective, etc. As should be obvious, it is incredibly intricate to manage dialects and

considerably more when we need to break down content composed by clients on the Internet since individuals.

Try not to deal with making sentences that are syntactically right and utilize a huge amount of abbreviations and words that are pretty much english for our situation.

We can picture more the dataset by making a diagram of what number of positive and negative tweets does it contains,

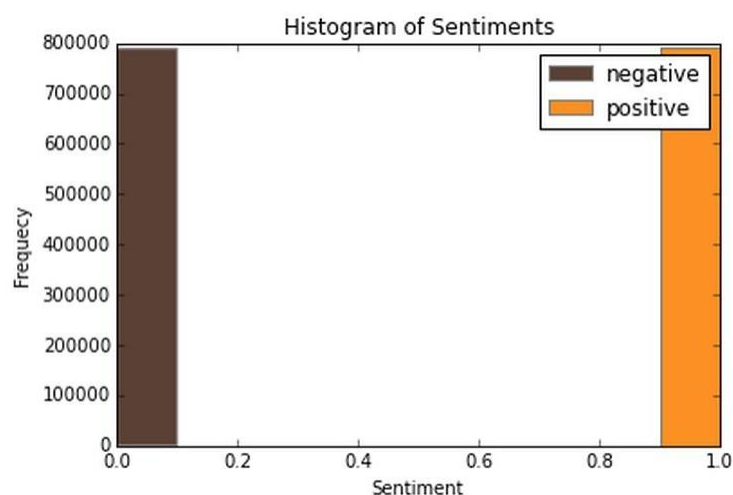


Figure 3.2: Histogram of the tweets as indicated by their estimation

We have precisely 790177 positive tweets and 788435 negative tweets which imply that the dataset is well-balanced. There is additionally no copies.

At long last, we should review the Twitter wording since we will need to manage in the tweets:

- **Hashtag:** A hashtag is any word or expression promptly went before by the # image. At the point when you click on a hashtag, you'll see different Tweets containing a similar catchphrase or theme.
- **@username:** A username is the way you're recognized on Twitter, and is constantly gone before quickly by the @ image. For example, Katy Perry is @katyperry.
- **MT:** Similar to RT (Retweet), a truncation for "Altered Tweet." Placed before the Retweeted text when clients physically retweet a message with changes, for instance shortening a Tweet.
- **Retweet:** RT, A Tweet that you forward to your devotees is known as a Retweet. Regularly used to go along news or other important disclosures on Twitter, Retweets consistently hold unique attribution.
- **Emoticons:** Composed utilizing accentuation and letters, they are utilized to communicate feelings succinctly, ";) :) ...".

Presently we have the corpus of tweets, we have to utilize different assets to make simpler the pre-processing step.

Resources

So as to encourage the pre-processing part of the information, we present five assets which are,

- An emoji word reference pulling together 132 of the most utilized emojis in western with their assessment, negative or positive.
- An abbreviation word reference of 5465 abbreviations with their interpretation.
- A stop word reference relating to words which are sifted through previously or in the wake of handling of regular language information since they are not helpful for our situation.
- A positive and negative word references given the extremity (feeling out-of-context) of words.
- A negative withdrawals and helpers word reference which will be utilized to identify nullification in a given tweet, for example, "don't", "can't", "can't", and so on.

The presentation of these assets will permit to uniform tweets and evacuate a portion of their complexities with the abbreviation word reference for example on the grounds that a great deal of abbreviations are utilized in tweets. The positive and negative word references could be helpful to increment (or not) the exactness score of the classifier. The emoji word reference has been worked from wikipedia with every emoji explained physically. The stop word reference contains 635 words, for example, "the", "of", "without". Ordinarily they ought not be helpful for characterizing tweets as indicated by their conclusion yet it is conceivable that they are.

Likewise we use Python 2.7 (<https://www.python.org/>) which is a programming language broadly utilized in information science and scikit-learn (<http://scikit-learn.org/>) an exceptionally complete and helpful library for AI containing each strategy, strategies we need and the site is additionally brimming with instructional exercises well-explained. With Python, the libraries, Numpy (<http://www.numpy.org/>) and Panda (<http://pandas.pydata.org/>) for controlling information effectively and instinctively are simply basic.

3.1.1 Twitter Sentiment Corpus

- It's a assortment of 5513 tweets gathered for 4 unique themes, to be specific, Apple, Google, Microsoft, Twitter they gathered and hand-ordered by Sanders Analytics LLC. Every section inside corpus having, Tweet id, Sentiment mark and a topic. We utilize Twitter-Python standard lib for improve there information by download and gather information like Tweet text, Creation, Creator Data and so on for each Tweet id. Every Tweet is hand arranged through American male into accompanying 4 classifications. With the end goal of our analyses, we believe Neutral and Irrelevant to be a similar class. Outline of Tweets in corpus is appear in Table 3.2.
- **Positive** For indicating positive estimation for concerning the subject
- **Positive** For demonstrating no or blended or frail assumptions concerning the subject
- **Negative** For demonstrating negative assumption concerning the subject
- **Irrelevant** For non English content or off-subject remarks

Classes	Counts	Example
negative	529	# Skype regularly smashing: #microsoft, what's going on with you ?
neutral	3770	How # Google Ventures select Which establishment achive \$200 Million http://t.co/FCWXoUd8 by means of @mashable @mashbusiness
positive	483	Currently each @Apple needs to do is get swype on the iphone and it would be break. Iphone that is

Table 3.2: Twitter Sentiment Corpus

3.1.2 Stanford Twitter

This corpus of tweets, created by Sanford's Natural Language preparing research gathering, is publically accessible. The preparation firm is gathered by questioning Twitter API for cheerful emojis like ":)" and tragic emojis like ":(" and naming those negative or positive. The emoji were going for Re-Tweets and disposes and copies evacuated. It likewise contains around 500 tweets physically gathered and named for testing purposes. We casually test and utilize 5000 tweets from data file. A case of Tweets in this corpus are appeared in Table 3.3.

Classes	Counts	Example
negative	2501	After playing anothers because of TV booking might good permit us to comprehend what's going trend, yet it creates things show terrible on Saturday evenings
positive	2499	@francescazurlo HAHA!!! to what extent have you been singing that melody now? It must be at any rate a day. I believe you're uncontrollably engaging!

Table 3.3: Stanford Corpus

3.2 Pre Processing

Client created content on the web is only from time to time present in a structure usable for learning. It gets critical to standardize the content by applying a progression of pre-preparing steps. We have applied a broad arrangement of pre-preparing steps to diminish the size of the list of capabilities to make it appropriate for learning calculations. Figure 3.2 shows different highlights seen in small scale blogging. Table 3.4 shows the recurrence of these highlights per tweet, cut by datasets. We likewise give a short depiction of pre-preparing steps taken.

	Twitter Sentiment		Stanford Corpus		Both	
	Average	Max.	Average	Max.	Average	Max.
Handles	0.6761	8	0.4888	10	0.5804	10
Hashtags	2.0276	13	0.0282	11	1.0056	13
Urls	0.4431	4	0.0452	2	0.2397	4
Emoticons	0.0550	3	0.0154	4	0.0348	4
Words	14.4084	31	13.2056	33	13.7936	33

Table 3.4: Frequency of Features per Tweet

3.2.1 Hashtags

A hashtag is a word or an un-separated expression prefixed with the hash image (#). These are utilized to both naming subjects and expressions that are right now in slanting points. For instance, #iPad, #news

Standard Expression: #(\w+)

Supplant Expression: HASH_\1

3.2.2 Handles

Each Twitter client has a novel username. Anything coordinated towards that client can be demonstrated be composing their username went before by '@'. In this way, these resemble formal people, places or things. For instance, @Apple

Standard Expression: @(\w+)

Supplant Expression: HNDL_\1

3.2.3 URLs

Employee continue share hyperlinks in their tweets. Twitter abbreviates them utilizing its in-house URL shortening administration, similar to <http://t.co/FCWXoUd8> - such connections likewise empowers Twitter to alarm clients if the connection leads out of its area. From the perspective of text arrangement, a specific URL isn't significant. Be that as it may, nearness of a URL can be a significant component. Ordinary articulation for identifying a URL is genuinely unpredictable in light of various kinds of URLs that can be there, but since of Twitter's shortening administration, we can utilize a moderately basic customary articulation.

Customary Expression: (http|https|ftp)://[a-zA-Z0-9\.\-]+

	ItemID	Sentiment	SentimentSource	SentimentText
0	1	0	Sentiment140	is so sad for my APL friend.....
1	2	0	Sentiment140	I missed the New Moon trailer...
2	3	1	Sentiment140	omg its already 7:30 pos
3	4	0	Sentiment140	.. Omgaga. Im sooo im gunna CRy. I've been at this dentist since 11.. I was supposed 2 just get a crown put on (30mins)...
4	5	0	Sentiment140	i think mi bf is cheating on me!!! neg
5	6	0	Sentiment140	or i just worry too much?
6	7	1	Sentiment140	Juuuuuuuuuuuuuuuussssst Chillin!!
7	8	0	Sentiment140	Sunny Again Work Tomorrow neg TV Tonight
8	9	1	Sentiment140	handed in my uniform today . i miss you already
9	10	1	Sentiment140	hmmmm.... i wonder how she my number pos

Table 3.7: After processing emoticons, they have been replaced by their corresponding tag.

The data set contains 19469 positive emoticons and 11025 negative emoticons.

URLs

We replace all URLs with the tag ||url||. There is about 73824 urls in the data set and we proceed as the same way we did for the emoticons.

	ItemID	Sentiment	SentimentSource	SentimentText
50	51	0	Sentiment140	baddest day ever.
51	52	1	Sentiment140	bathroom is clean..... now on to more enjoyable tasks.....
52	53	1	Sentiment140	boom boom pow
53	54	0	Sentiment140	but i'm proud.
54	55	0	Sentiment140	congrats to helio though
55	56	0	Sentiment140	David must be hospitalized for five days end of July (palatine tonsils). I will probably never see Katie in concert.
56	57	0	Sentiment140	friends are leaving me 'cause of this stupid love http://bit.ly/ZoxZC
57	58	1	Sentiment140	go give ur mom a hug right now. http://bit.ly/azFwv
58	59	1	Sentiment140	Going To See Harry Sunday Happiness
59	60	0	Sentiment140	Hand quilting it is then...

Table 3.8: Tweets before processing URLs.

	ItemID	Sentiment	SentimentSource	SentimentText
50	51	0	Sentiment140	baddest day eever.
51	52	1	Sentiment140	bathroom is clean..... now on to more enjoyable tasks.....
52	53	1	Sentiment140	boom boom pow
53	54	0	Sentiment140	but i'm proud.
54	55	0	Sentiment140	congrats to helio though
55	56	0	Sentiment140	David must be hospitalized for five days end of July (palatine tonsils). I will probably never see Katie in concert.
56	57	0	Sentiment140	friends are leaving me 'cause of this stupid love url
57	58	1	Sentiment140	go give ur mom a hug right now. url
58	59	1	Sentiment140	Going To See Harry Sunday Happiness
59	60	0	Sentiment140	Hand quilting it is then...

Table 3.9: Tweets after processing URL

3.2.5 Punctuations

Despite the fact that not all Punctuations are significant from the perspective of order however a portion of these, similar to question mark, outcry imprint can likewise give data about the assumptions of the content. We supplant each word limit by a rundown of important accentuations present by then. Table 3.10 records the accentuations at present distinguished. We likewise expel any single statements that may exist in the content.

Punctuations	Examples	
PUNC_DOT	.	
PUNC_EXCL	!	i
PUNC_QUES	?	¿
PUNC_ELLP

Table 3.10: List of Punctuations

3.2.6 Repeating Characters

Individuals frequently use rehashing characters while utilizing casual language, similar to "I'm in a hurrriyyyyy", "We won, yaaayyyyy!" As our last pre-handling step, we supplant characters rehashing more than twice as two Features.

Normal Expression: (.)\1{1,}

Supplant Expression: \1\1

Decrease in highlight space

It's critical to take note of that by applying these pre-preparing steps, we are diminishing our list of capabilities else it very well may be excessively inadequate. Table 3.11 records the lessening in include set because of preparing every one of these highlights.

Preprocessing	Twitter Sentiment		Stanford Corpus		Both	
	Words	Percentage	Words	Percentage	Words	Percentage
None	19128		15910		31832	
Hashtags	18649	97.50%	15550	97.74%	31223	98.09%
Handles	17118	89.49%	13245	83.25%	27383	86.02%
Urls	16723	87.43%	15335	96.39%	29083	91.36%
Emoticons	18631	97.40%	15541	97.68%	31197	98.01%
Punctuations	13724	71.75%	11225	70.55%	22095	69.41%
Repeatings	18540	96.93%	15276	96.02%	30818	96.81%
All	11108	58.07%	8646	54.34%	16981	53.35%

Table 3.11: Number of words Initial and Final pre-processing

3.3 Stemming Algorithms

All stemming approaches are of the accompanying significant sorts – append expelling, measurable and blended. The primary kind, Affix evacuation stemmer, is the most fundamental one. These apply a lot of change rules to each word trying to cut off generally known prefixes and/or additions [8]. A unimportant stemming calculation is shorten words at N-th image. In any case, this clearly isn't appropriate for reasonable purposes.

J.B. Lovins portrayed first stemming calculation in 1968. It characterizes 294 endings, each connected to one of 29 conditions, in addition to 35 change rules. For a word being stemmed, a closure with a wonderful condition is found and expelled. Another acclaimed stemmer utilized broadly is portrayed in the following area.

3.3.1 Porter Stemmer

Martin Porter composed a stemmer that was distributed in July 1980. This stemmer was broadly utilized and became and remains the true standard calculation utilized for English stemming. It offers amazing exchange off between speed, clarity, and exactness. It utilizes a lot of around 60 standards applied in 6 progressive advances [9]. A significant element to note is that it doesn't include recursion. The means in the calculation are portrayed in Table 3.12.

1.	Gets rid of plurals and -ed or -ing suffixes
2.	Turns terminal y to i when there is another vowel in the stem ^[10]
3.	Maps double suffixes to single ones: -ization, -ational, etc.
4.	Deals with suffixes, -full, -ness etc.
5. ^[10]	Takes off -ant, -ence, etc.
6.	Removes a final -e

Table 3.12: Porter Stemmer Steps

3.3.2 Lemmatization

Lemmatization is the way toward normalizing a word as opposed to simply discovering its stem. All the while, an addition may not exclusively be evacuated, yet may likewise be subbed with an alternate one. It might likewise include first deciding the grammatical form for a word and afterward applying standardization rules. It may likewise include word reference gaze upward. For instance, action word 'saw' would be lemmatized to 'see' and the thing 'saw' will remain 'saw'. For our motivation of arranging text, stemming should do the trick.

3.4 Features

A wide assortment of highlights can be utilized to assemble a classifier for tweets. The most generally utilized and fundamental list of capabilities is word n-grams. In any case, there's a great deal of space explicit data present in tweets that can likewise be utilized for characterizing them. We have explored different avenues regarding two arrangements of highlights:

3.4.1 Unigrams

Unigrams are the least complex highlights that can be utilized for text characterization. A Tweet could be spoken to by a multiple set of words included in it. We, notwithstanding, have utilized the nearness of unigrams in a tweet as a list of capabilities. Nearness of a word is a higher priority than how frequently it is rehashed. Ache et al. discovered that nearness of unigrams yields preferable outcomes over reiteration [1]. This additionally causes us to abstain from scaling the information, which can impressively diminish preparing time [2]. Figure 3.3 delineated the total circulation of words in our dataset.

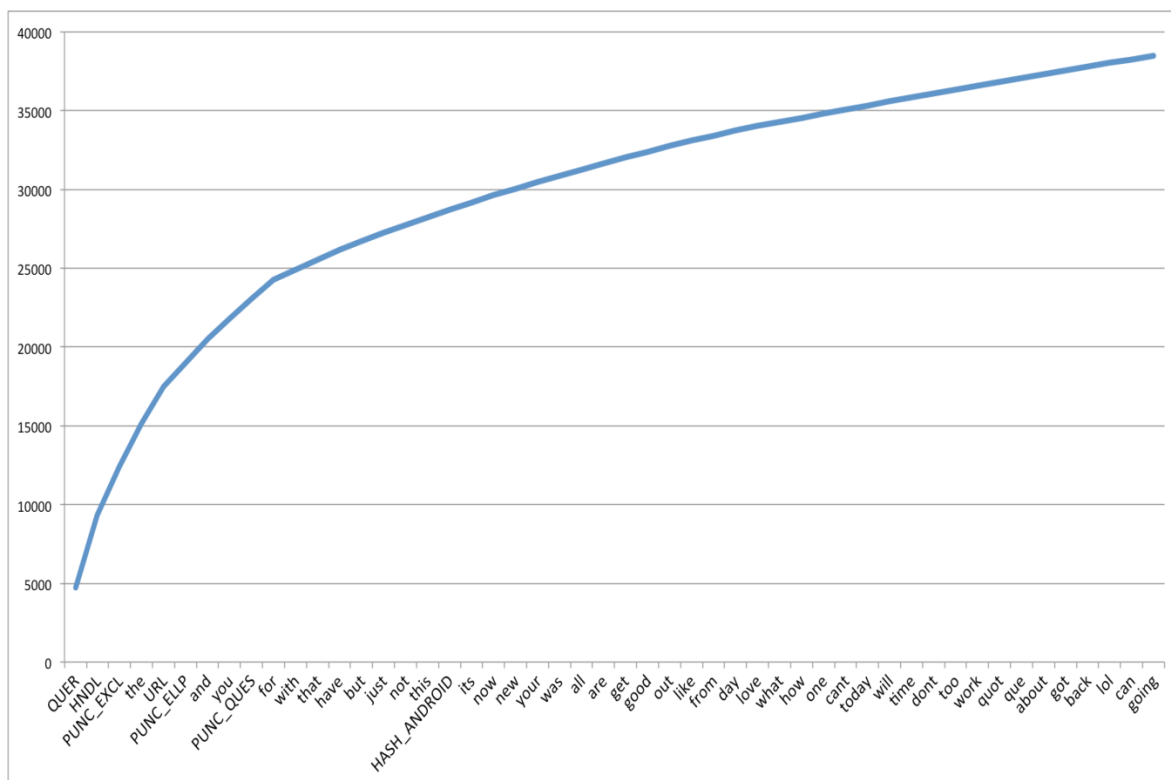


Figure 3.3: Cumulative Frequency Graph Plot for 50 Most Frequent Unigrams

We likewise see that the unigrams pleasantly adhere to Zipf's law. It expresses that in a corpus of regular language, the recurrence of any word is conversely corresponding to its position in the recurrence table. Figure 4 is a plot of log recurrence versus log rank of our dataset. A straight trendline fits well with the information.

3.4.2 N-grams

N-gram alludes to a n-long arrangement of words. Probabilistic Language Models dependent on Unigrams, Bigrams and Trigrams could be effectively utilized to foresee the following word giving a recent setting of words. In the area of notion investigation, the presentation of N-grams is indistinct. As indicated by Pang et al., a few analysts report whose unigrams alone are rise above than bigrams for order film audits, Although few another report that bigrams and trigrams yield rise better item survey extremity arrangement [1].

As the request for the n-grams expands, they will in general be increasingly inadequate. In light of our investigations, we locate that number of bigrams and trigrams increment substantially more quickly than the quantity of unigrams with the quantity of Tweets. Figure 3.4 showing the quantity of n-grams versus number of Tweets. We could able to see that bigrams and trigrams increment straightly where as unigrams are expanding logarithmically.

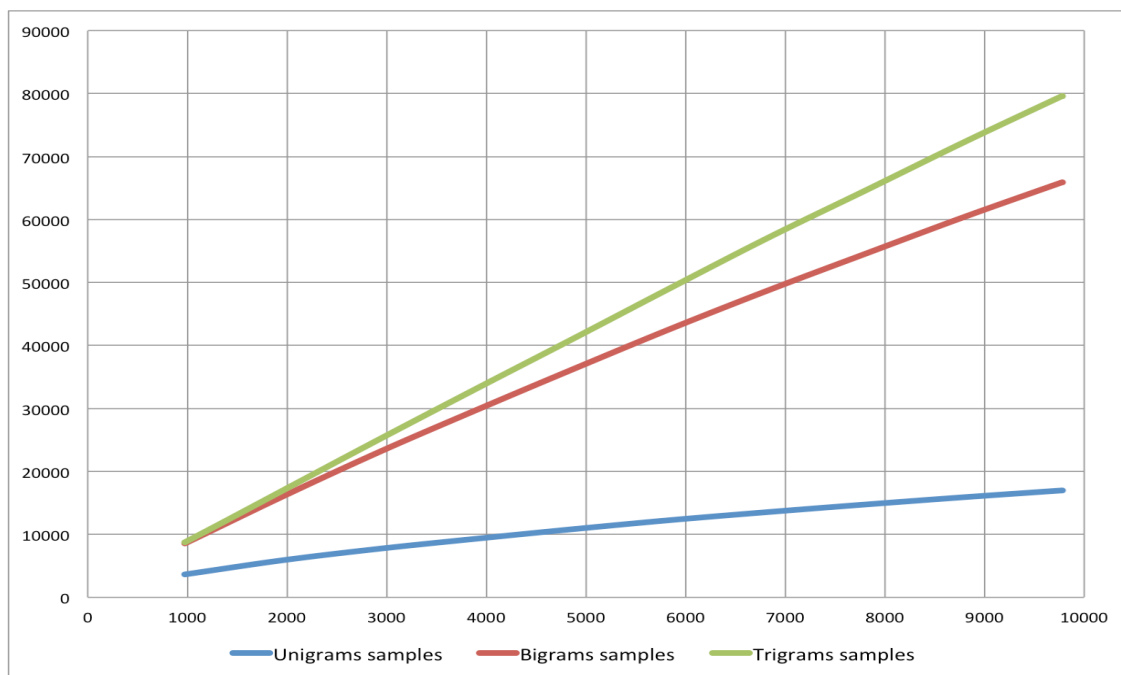


Figure 3.4: Number of n-grams vs. Number of Tweets

Since higher request n-grams are scantily populated, we choose to trim off the n-grams that are not seen more than once in the preparation corpus, since chances are that these n-grams are bad pointers of assumptions. After the sifting through non-rehashing n-grams, we see that the quantity of n-grams is significantly diminished and approaches the request for unigrams, as appeared in Figure 3.5.

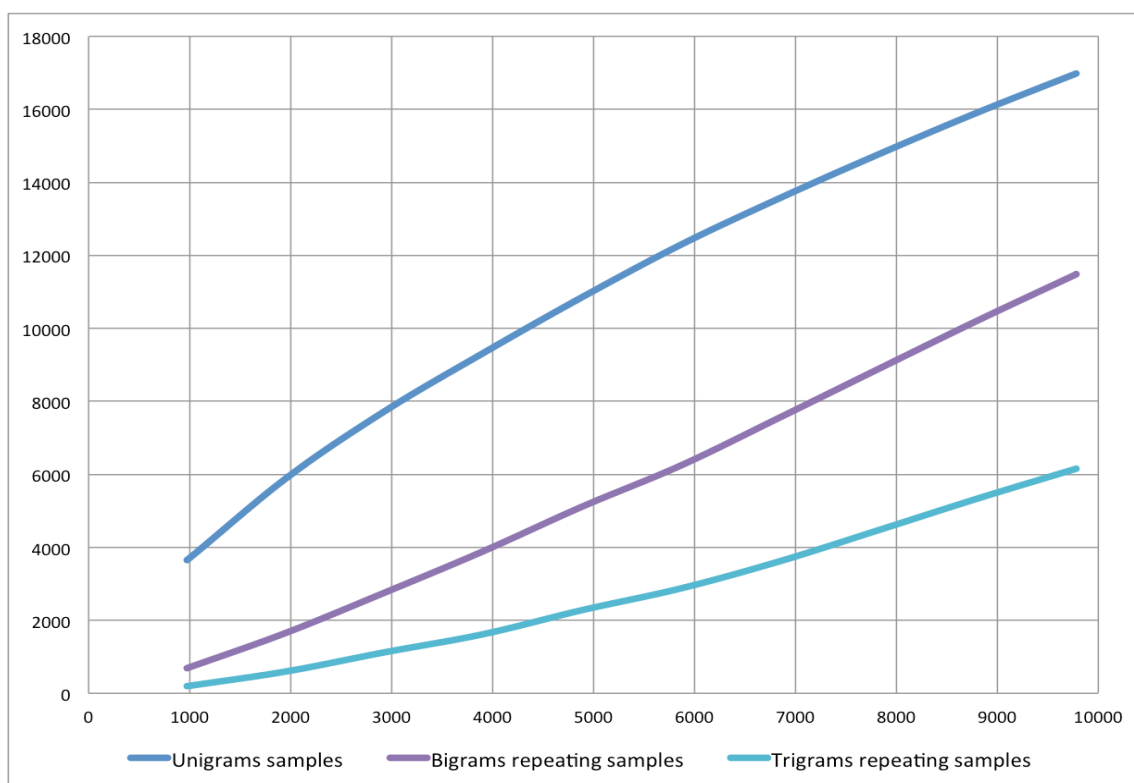


Figure 3.5: Number of rehashing n-grams vs. Number of Tweets

3.4.3 Negation Handling

The Demanded refutation recognition in assumption examination could be represented by the distinction in the significance of the expressions, "This is acceptable" versus "This isn't acceptable" Although, the invalidations happening in characteristic language are only from time to time so basic. Taking care of the refutation comprises of two undertakings – Detection of unequivocal nullification prompts and the extent of invalidation of these words.

Councill et al. take a gander at whether invalidation location is valuable for slant investigation and furthermore how much is it conceivable to decide the specific extent of a nullification in the content [7]. They depict a strategy for refutation recognition dependent on Left and Right path difference of a token to the closest unequivocal nullification sign.

Identification of Explicit Negation Cues

To distinguish express nullification signs, we are searching for the accompanying words in Table 3.11. The hunt is finished utilizing normal articulations.

S.No.	Negation Cues
1.	never
2.	no
3.	nothing
4.	nowhere
5.	no one
6.	none
7.	not
8.	haven't
9.	hasn't
10.	hadn't
11.	cant
12.	couldn't

13.	shouldn't
14.	wont
15.	wouldn't
16.	don't
17.	doesn't
18.	didn't
19.	isn't
20.	aren't
21.	aint
22.	Anything ending with "n't"

Table 3.13: Explicit Negation Cues

Extent of Negation

Words promptly going before and undergoing the invalidation signals are much negative and the words that getting forth onwards don't lie in the extent of refutation of that prompts. We characterize left and right cynicism of a word as the odds that importance of that word is really the inverse. Left cynicism relies upon the nearest nullification prompt on the left and correspondingly for Right antagonism. Figure 6 delineates the left and right cynicism of words in a tweet.

```
Words: ['HASH_Skype', 'crash', 'too', 'much', 'PUNC_EXCL',
        'not', 'expect', 'this', 'from', 'HASH_MICROSOFT']
Neg_l: [0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.9, 0.8, 0.7, 0.6]
Neg_r: [0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 0.0, 0.0, 0.0, 0.0]
```

Figure 3.6: Scope of Negation

To make the validation set, there are two main options:

- Split the training set into two parts (60%, 20%) with a ratio 2:8 where each part contains an equal distribution of example types. We train the classifier with the largest part, and make prediction with the smaller one to validate the model. This technique works well but has the disadvantage of our classifier not getting trained and validated on all examples in the data set (without counting the test set).
- The **K-fold cross-validation**. We split the data set into k parts, hold out one, combine the others and train on them, then validate against the held-out portion. We repeat that

process k times (each fold), holding out a different portion each time. Then we average the score measured for each fold to get a more accurate estimation of our model's performance.

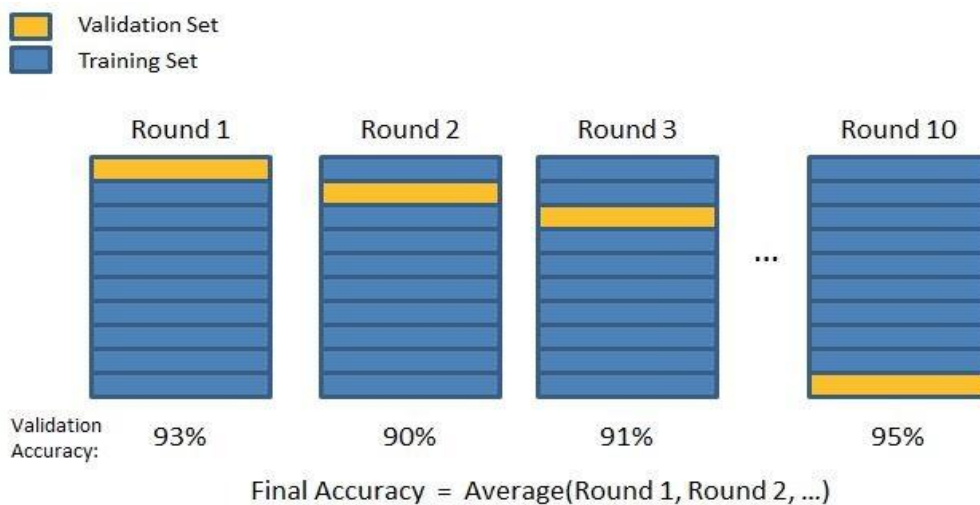


Figure 3.7: K-fold cross-validation

We split the training data into K folds and cross validate on them using scikit-learn as shown in the figure 7 above. The number of K -folds is arbitrary and usually set to 10 it is not a rule. In fact, determine the best K is still an unsolved problem but with lower K : computationally cheaper, less variance, more bias. With large K : computationally expensive, higher variance, lower bias.

We can now train the naive bayes classifier with the training set, validate it using the hold out part of data taken from the training set, the validation set, repeat this 10 times and average the results to get the final accuracy which is about **0.77** as shown in the screen results below,

```
Total tweets classified: 1183958
Score: 0.77653600187
Confusion matrix:
[[465021 126305]
 [136321 456311]]
```

Figure 3.8: Result of the naive bayes classifier with the score representing the average of the results of each 10-fold cross-validation, and the overall confusion matrix.

Notice that to evaluate our classifier we two methods, the F1 score and a confusion matrix. The **F1 Score** can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. It a measure of a **classifier's accuracy**. The F1 score is given by the following formula,

$$F1 = \frac{2 \times (\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})}$$

Formula 3.1: F1 score

where the precision is the number of true positives (the number of items correctly labeled as belonging to the positive class) divided by the total number of elements labeled as belonging to the positive class,

$$Precision = \frac{TP}{TP + FP}$$

Formula 3.2 : Precision

and the recall is the number of true positives divided by the total number of elements that actually belong to the positive class,

$$Recall = \frac{TP}{TP + FN}$$

Formula 3.3: Recall

A precision score of 1.0 means that every result retrieved was relevant (but says nothing about whether all relevant elements were retrieved) whereas a recall score of 1.0 means that all relevant documents were retrieved (but says nothing about how many irrelevant documents were also retrieved).

There is a **trade-off between precision and recall** where increasing one decrease the other and we usually use measures that combine precision and recall such as F-measure or MCC.

A **confusion matrix** helps to visualize how the model did during the classification and evaluate its accuracy. In our case we get about 156715 false positive tweets and 139132 false negative tweets. It is "about" because these numbers can vary depending on how we shuffle our data for example.

		Predicted Class	
		Yes	No
Actual Class	Yes	TP	FN
	No	FP	TN

Figure 3.9: Example of confusion matrix

Notice that we still didn't use our test set, since we are going to tune our classifier for improving its results.

The confusion matrix of the naive bayes classifier can be expressed using a color map where dark colors represent high values and light colors represent lower values as shown in the corresponding color map of the naive bayes classifier below,

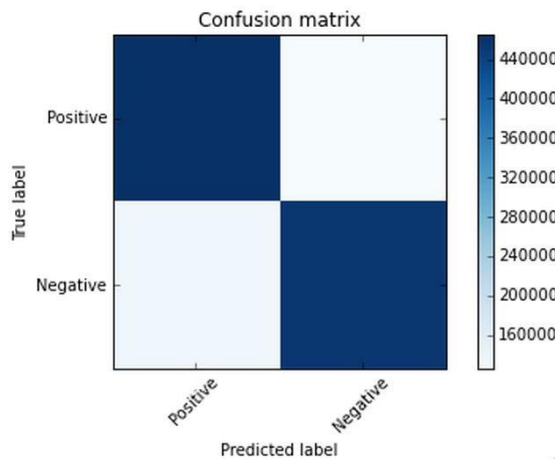


Figure 3.10: Color map of the confusion matrix related to the naive bayes classifier used.

Hopefully we can distinguish that the number of true positive and true negative classified tweets is higher than the number of false and positive and negative tweets. However from this result we try to improve the accuracy of the classifier by experimenting different techniques and we repeat the same process using the k-fold cross validation to evaluate its averaged accuracy.

Improvements

From the baseline, the goal is to improve the accuracy of the classifier, which is 0.77, in order to determine better which tweet is positive or negative. There are several ways of doing this and we present only few possible improvements (or not).

First we could try to removed what we called, stop words. Stop words usually refer to the most common words in the English language (in our case) such as: "the", "of", "to" and so on.

They do not indicate any valuable information about the sentiment of a sentence and it can be necessary to remove them from the tweets in order to keep only words for which we are interested. To do this we use the list of 635 stopwords that we found. In the table below, you can see the most frequent words in the data set with their counts,

```
[ ('||target||', 780664),
  ('i', 778070),
  ('to', 614954),
  ('the', 538566),
  ('a', 383910),
  ('you', 341545),
  ('my', 336980),
  ('and', 316853),
  ('is', 236393),
  ('for', 236018),
  ('it', 235435),
  ('in', 217350),
  ('of', 192621),
  ('on', 169466),
  ('me', 163900),
  ('so', 158457),
  ('have', 150041),
  ('that', 146260),
  ('out', 143567),
  ('but', 132969) ]
```

Table 3.14: Most frequent words in the data set with their corresponding count.

We can derive from the table, some interesting statistics like the number of times the tags used in the pre-processing step appear,

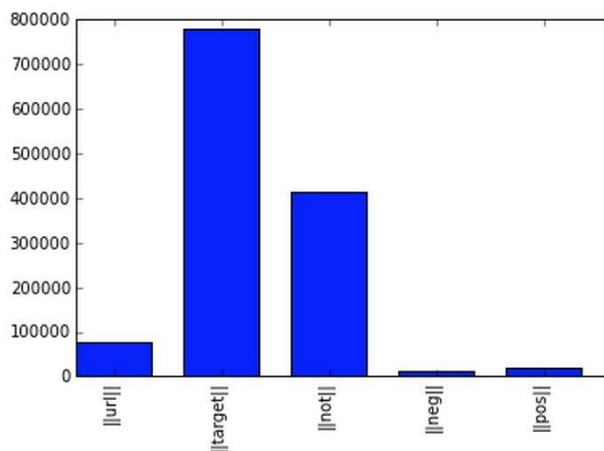


Figure 3.11: Tags in the data set with their corresponding count.

Recall that ||url|| corresponds to the URLs, ||target|| the twitter usernames with the symbol “@” before, ||not|| replaces the negation words, ||pos|| and ||neg|| replace the positive and negative smiley respectively. After removing the stop words we get the results below,

```
Total tweets classified: 1183958
Score: 0.758623708326
Confusion matrix:
[[437311 154015]
 [136343 456289]]
```

Figure 3.12: Result of the naive bayes classifier with stopwords removed.

Compared to the previous result, we lose 0.02 in accuracy and the number of false positive goes from 126305 to 154015 . We conclude that stop words seem to be useful for our classification task and remove them do not represent an improvement.

We could also try to stem the words in the data set. **Stemming** is the process by which endings are removed from words in order to remove things like tense or plurality. The stem form of a word could not exist in a dictionary (different from Lemmatization). This technique allows to unify words and reduce the dimensionality of the dataset. It's not appropriate for all cases but can make it easier to connect together tenses to see if you're covering the same subject matter. It is faster than **Lemmatization** (remove inflectional endings only and return the base or dictionary form of a word, which is known as the lemma). Using the library NLTK which is a library in Python specialized in natural language processing, we get the following results after stemming the words in the data set,

```

Total tweets classified: 1183958
Score: 0.773106857186
Confusion matrix:
[[462537 128789]
 [138039 454593]]

```

Figure 3.13: Result of the naive bayes classifier after stemming.

We actually lose 0.002 in accuracy score compared to the results of the baseline. We conclude that stemming words does not improve the classifier's accuracy and actually do not make any sensible changes.

Language Models

Let's introduce language models to see if we can have better results than those for our baseline. Language models are models assigning **probabilities to sequence of words**.

Initially, they are extensively used in speech recognition and spelling correction but it turns out that they give good results in text classification.

The quality of a language model can be measured by the empirical perplexity (or entropy) using:

$$Perplexity = T \sqrt{\frac{1}{P(w_1, \dots, w_T)}}$$

$$Entropy = \log_2 Perplexity$$

Formula 3.4: Perplexity and Entropy to evaluate language models.

The goal is to **minimize the perplexity which is the same as maximizing probability**.

An **N-Gram model** is a type of probabilistic language model for predicting the next item in such a sequence in the form of (n - 1) order Markov Model. The Markov assumption is the probability of a word depends only on the probability of a limited history (previous words).

$$P(w_i | w_1, \dots, w_{i-1}) = P(w_i | w_{i-n+1}, \dots, w_{i-1})$$

Formula 3.5: General form of N-grams.

A straightforward maximum likelihood estimate of n-gram probabilities from a corpus is given by the observed frequency,

$$P(w_i | w_{i-n+1}, \dots, w_{i-1}) = \frac{\text{count}(w_{i-n+1}, \dots, w_i)}{\text{count}(w_{i-n+1}, \dots, w_{i-1})}$$

Formula 3.6: MLE of N-grams.

There are several kind of n-grams but the most common are the unigram, bigram and trigram. The **unigram model** make the assumption that every word is independent and so we compute the probability of a sequence using the following formula,

$$P(w_1, w_2, \dots, w_n) = \prod_i P(w_i)$$

Formula 3.7: Unigram.

In the case of the **bigram model** we make the assumption that **a word is dependent of its previous word**,

$$P(w_i | w_1, w_2, \dots, w_{i-1}) \approx P(w_i | w_{i-1})$$

Formula 3.8: Bigram.

To estimate the n-gram probabilities, we need to compute the **Maximum Likelihood Estimates**.

For Unigram:

$$P(w_i) = \frac{C(w_i)}{N}$$

Formula 3.9: MLE for unigram.

For Bigram:

$$P(w_i, w_j) = \frac{\text{count}(w_i, w_j)}{N}$$

$$P(w_j | w_i) = \frac{P(w_i, w_j)}{P(w_i)} = \frac{\text{count}(w_i, w_j)}{\sum_w \text{count}(w_i, w)} = \frac{\text{count}(w_i, w_j)}{\text{count}(w_i)}$$

Formula 3.10.: MLE for bigram.

Where N is the Number of Word, C Mean Count, w_i and w_j are Words.

There are two main practical issues:

- We compute everything in log space (log probabilities) to avoid underflow (multiplying so many probabilities can lead to too small number) and because adding is faster than multiplying ($p1 \times p2 \times p3 = \log_p 1 + \log_p 2 + \log_p 3$)

- We use smoothing techniques such as Laplace, Witten-Bell Discounting, Good-Turing Discounting to deal with unseen words in the training occurring in the test set.

An N-gram language model can be applied to text classification like Naive Bayes model does. A tweet is categorized according to,

$$c^* = \arg \max_{c \in C} P(c|d)$$

Formula 3.11: Objective function of n-gram.

and using Baye's rule, this can be rewritten as,

$$c^* = \arg \max_{c \in C} \{P(c)P(d|c)\}$$

$$c^* = \arg \max_{c \in C} \left\{ P(c) \times \prod_{i=1}^T P(w_i | w_{i-n+1}, \dots, w_{i-1}, c) \right\}$$

$$c^* = \arg \max_{c \in C} \left\{ P(c) \times \prod_{i=1}^T P_c(w_i | w_{i-n+1}, \dots, w_{i-1}) \right\}$$

Formula 3.12: Objective function rewritten using baye's rule of n-gram.

$P(d|c)$ is the likelihood of d under category c which can be Computed by n-gram Language model.

An important note is that n-gram classifiers are in fact a generalization of Naive Bayes. A unigram classifier with Laplace smoothing corresponds exactly to the traditional naive Bayes classifier.

Since we use bag of words model, meaning we translate this sentence: "I don't like chocolate" into "I", "don't", "like", "chocolate", we could try to use bigram model to take care of negation with "don't like" for this example. Using bigrams as feature in the classifier we get the following results,

```
Total tweets classified: 1183958
Score: 0.784149223247
Confusion matrix:
[[480120 111206]
 [138700 453932]]
```

Figure 3.14: Results of the naive bayes classifier with bigram features.

Using only bigram features we have slightly improved our accuracy score about 0.01. Based on that we can think of adding unigram and bigram could increase the accuracy score more.

```
Total tweets classified: 1183958
Score: 0.795370054626
Confusion matrix:
[[486521 104805]
 [132142 460490]]
```

Figure 3.15: Results of the naive bayes classifier with unigram and bigram features.

and indeed, we increased slightly the accuracy score about 0.02 compared to the baseline.

CHAPTER 4 EXPERIMENTAL RESULTS

We train 90% of our data utilizing numerous mixes of highlights and take a look at them on the staying 10%. We tend to take the highlights within the incidental to blends.

- only unigrams, unigrams + Sifted bigrams and trigrams, unigrams + negation, unigrams + Separated bigrams and trigrams + negation. we tend to at that time train classifiers utilizing various arrangement calculations - Naive Thomas Bayes Classifier and most Entropy Classifier.

The assignment of grouping of a tweet ought to be potential in 2 stages - 1st, characterizing "impartial" (or "emotional") versus "objective" tweets and second, ordering target tweets into "positive" versus "negative" tweets. we tend to likewise ready a pair of stage classifiers. The exactnesses for each one among these style square measure appeared in Figure 4.1 , we tend to mention these intimately beneath.

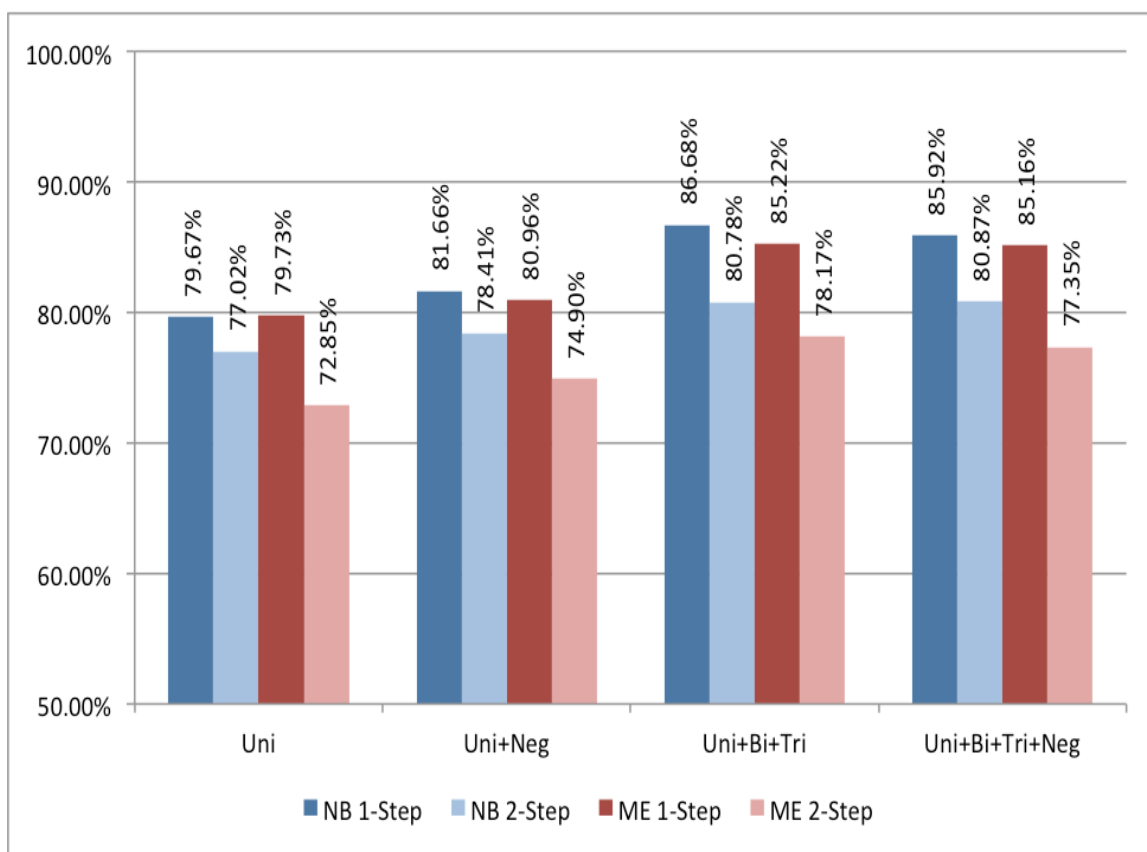


Figure 4.1: Accuracy for Naïve Thomas Bayes Classifier

4.1 Naive Bayes

Naive Bayes classifier is that the most simple and also the fastest classifier. Varied scientists [2], [4] guarantee to own gotten best outcomes utilizing this classifier.

For a given tweet, on the off probability that we've to find the name for it, we tend to discover the possibilities of the extensive variety of marks, as long as component and after choose the

name with greatest chance. The outcomes from getting ready the Naive Thomas Bayes classifier square measure appeared beneath in Figure 4.1 . The exactitude of Unigrams is that the most reduced at 79.67%. The preciseness increments within the event that we tend to in addition use Negation location (81.66%) or higher request n-grams (86.68%). we tend to see that within the event that we tend to utilize each Negation location and better request n-grams, the exactitude is probably not specifically merely utilizing higher request n-grams (85.92%). We can likewise observe of that exactnesses for twofold advance classifier are lesser than those for relating single step.

We have in addition indicated preciseness versus Recall esteems for Naive Thomas Bayes classifier regarding numerous categories – Negative, Neutral and Positive in Figure 4.2 . The sturdy markers show the P-R esteems for single step classifier and empty markers show the impact of utilizing twofold advance classifier. numerous focuses square measure for numerous capabilities. we can see that each accuracy even as review esteems are higher for single step than that for twofold advance.

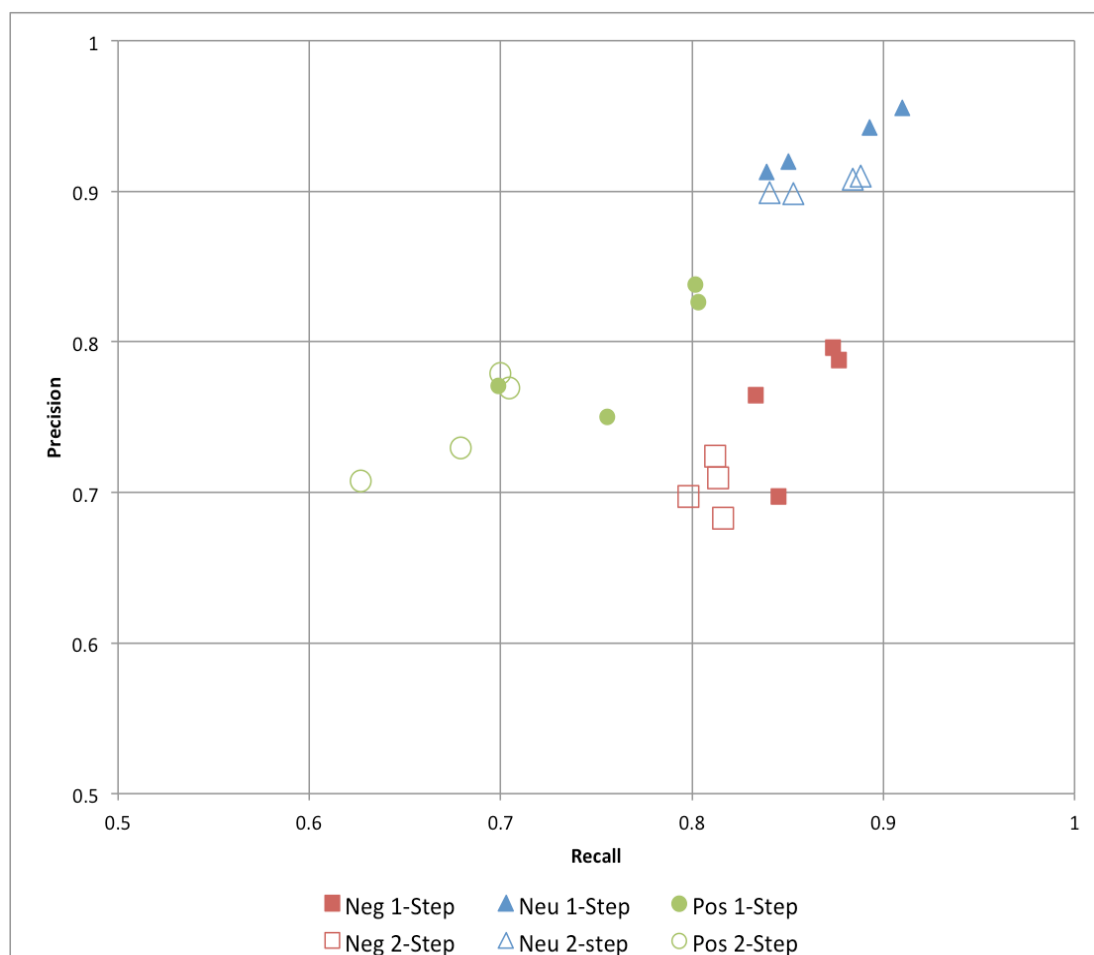


Figure 4.2: Precision vs. Recall for Naive Bayes Classifier

4.2 Maximum Entropy Classifier

This classifier works by finding a chance circulation that augments the likelihood of testable data. This chance work is outlined by weight vector. the perfect estimation of which may be discovered utilizing the strategy for Lagrange multipliers.

The outcomes from getting ready the most Entropy Classifier square measure appeared beneath in Figure 4.3 . Correctnesses follow a comparative pattern once contrasted with Naive Thomas Bayes classifier. Unigram is that the most bottom at 79.73% and that we see associate growth for refutation identification at 80.96%. the best is accomplished with unigrams, bigrams and trigrams at 85.22% firmly followed by n-grams and breakup at 85.16%. By and by, the exactnesses for twofold advance classifiers square measure imposingly lower.

Precision versus Recall map is likewise appeared for many extreme entropy classifier in Figure 4.3 . Here we tend to see that exactitude of "impartial" category increment by utilizing a twofold advance classifier, nonetheless with a formidable decline in its review and slight fall in accuracy of "negative" and "positive" categories.

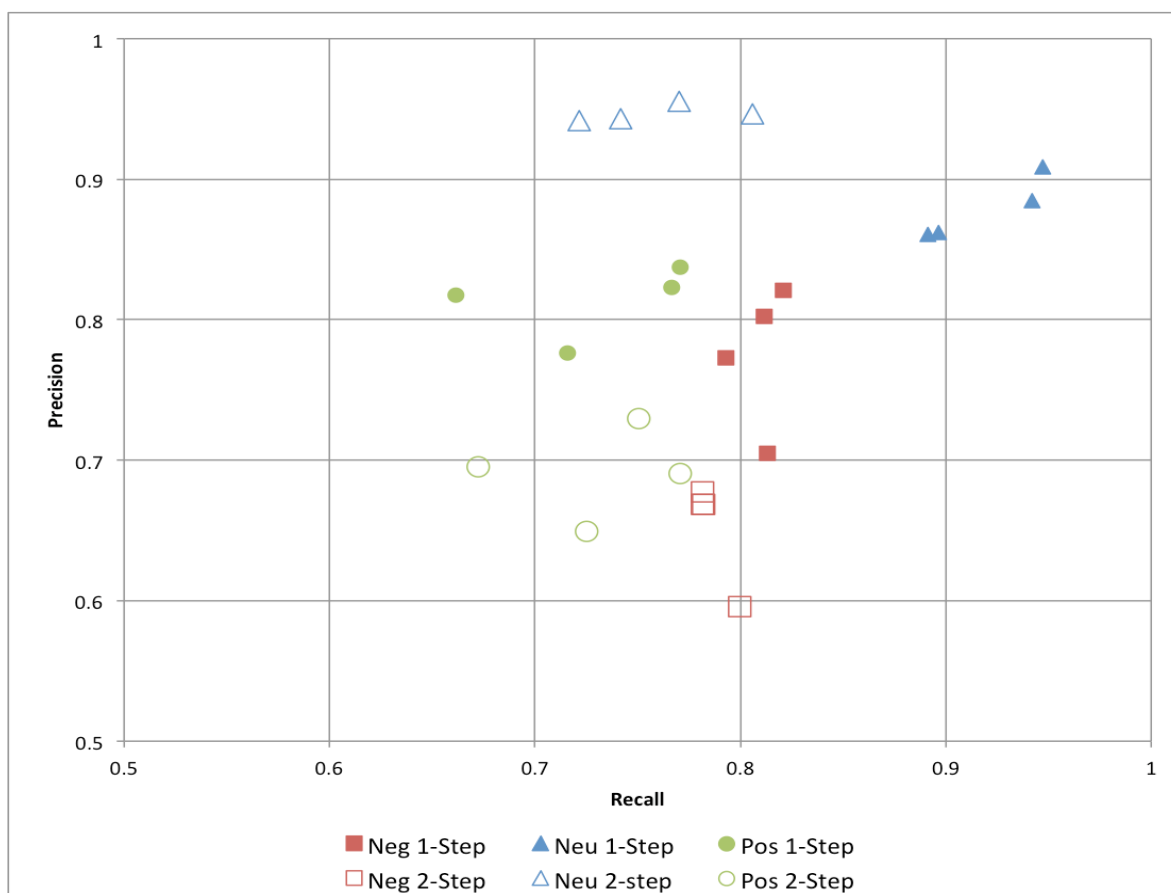


Figure 4.3: Precision vs. Recall for Max Entropy Classifier

CHAPTER 5 FUTURE WORK AND CONCLUSION

5.1 Future Work

Researching Support Vector Machines Many papers have talked regarding the outcomes utilizing Support Vector Machines (SVMs) too. the next stage is take a look at our methodology on SVMs. nonetheless, Go, Bhayani and Huang have careful that SVMs do not build the exactitude [2].

Building a classifier for Hindi tweets There area unit various shoppers on Twitter that utilization essentially Hindi language. The methodology talked regarding here will be used to form a Hindi language assessment classifier.

Improving Results utilizing Semantic Analysis Understanding the task of the items being mentioned will assist USA with ameliorative organize a given tweet. for example, "Skype often smashing: microsoft, what is happening with you?" Here Skype is associate degree item and Microsoft is a corporation. we will utilize linguistics labellers to accomplish this. Such a technique is talked regarding by Saif, He and Alani [6]

5.2 Conclusion

These days, supposition investigation or conclusion mining is a noteworthy issue in AI. we tend to area unit still so much to differentiate the estimations of s corpus of writings exactly in lightweight of the multifarious nature within the West Germanic and significantly additional within the event that we tend to think about different dialects, as an example, Chinese.

In this task we tend to tried to point out the essential methodology of ordering tweets into positive or negative category utilizing Naive Bayes as gauge and the way language models area unit known with the Naive Bayes and may deliver higher outcomes. we tend to may to boot improve our classifier by trying to get rid of additional highlights from the tweets, trying varied kinds of highlights, standardisation the boundaries of the guileless Thomas Bayes classifier, or trying another classifier all at once.

We build a notion classifier for twitter utilizing named informational indexes. we tend to to boot examine the applicability of utilizing a twofold advance classifier and refutation recognition with the top goal of supposition investigation.

Our normal classifier that utilizes solely the unigrams accomplishes a preciseness of around 80.00%. Exactitude of the classifier increments within the event that we tend to use breakup discovery or gift bigrams and trigrams. during this approach. we can reason that each Negation Detection and better request n-grams are valuable with the top goal of text grouping. In any case, within the event that we tend to utilize each n-grams and nullification recognition, the preciseness falls hardly. we tend to likewise note that Single step classifiers out perform twofold advance classifiers. once all is claimed in done, Naive Thomas Bayes Classifier performs superior to most Entropy Classifier.

We accomplish the most effective exactitude of 86.68% on account of Unigrams + Bigrams + Trigrams, ready on Naive Thomas Bayes Classifier.

References

- [1]. Pak, Alexander, and St. Patrick Paroubek. "Twitter as a Corpus for Sentiment Analysis and Opinion Mining." LREc. Vol. 10. 2010.
- [2]. Alec Go, Richa Bhayani, and Lei Huang. Twitter notion arrangement utilizing inaccessible management. Preparing, pages 1-6, 2009.
- [3]. Niek Sanders. Twitter notion corpus. <http://www.sananalytics.com/lab/twitter-estimation/>. Sanders Analytics.
- [4]. Alexander Pak and St. Patrick Paroubek. Twitter as a corpus for notion examination and feeling mining. volume 2010, pages 1320-1326, 2010.
- [5]. Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore. Twitter notion examination: the good the awful and also the omg! ICWSM, 11:pages 538-541, 2011.
- [6]. Hassan Saif, Yulan He, and Harith Alani. linguistics notion examination of twitter. within the linguistics Web-ISWC 2012, pages 508-524. Springer, 2012.
- [7]. Isaac G Councill, Ryan McDonald, and Leonid Velikovich. what is unimaginable then forth: working out a way to order the extent of nullification for improved assessment investigation. In Proceedings of the workshop on dissolution and theory in common language making ready, pages 51-59. Relationship for linguistics, 2010.
- [8]. Ilia Smirnov. Diagram of stemming calculations. Mechanical Translation, 2008.
- [9]. Martin F Porter. A calculation for suffix baring. Program: electronic library and information frameworks, 40(3):pages 211-218, 2006.
- [10]. Balakrishnan Gokulakrishnan, P Priyanthan, T Ragavan, N Prasath, and A Perera. Supposition mining and slant investigation on a twitter info stream. In Advances in ICT for rising Regions (ICTer), 2012 International Conference on. IEEE, 2012.
- [11]. John Ross Quinlan. C4. 5: programs for AI, volume 1. Morgan kaufmann, 1993.
- [12]. Steven Bird, Ewan Klein, and Edward Loper. Regular language Processing with Python. " O'Reilly Media, Inc.", 2009.

LIST OF PUBLICATIONS OF THE CANDIDATE'S WORK

- [1]. Rajat Kumar Arya, Dr. Rajni Jindal. Sentiment Analysis on Online Product Review. International Journal of New Era Research, (IJNER- UGC Approved) March 2020.
- [2]. Rajat Kumar Arya, Dr. Rajni Jindal. Twitter Sentiment Analysis using N-gram with Naïve Bayes Classifier. International Journal of Advanced Research in Engineering and Technology (IJARET- Scopus Indexed). June 2020.
- [3]. Rajat Kumar Arya, Dr. Rajni Jindal. Integrating E-Governance with Data Analytics using Spark. International Journal of Engineering Science and Computing (Peer Reviewed). June 2020.

