REPORT

Twitter Tweet Analysis (Project ID: SPS_PRO_1249)

DIXITA SHUKLA

1. INTRODUCTION

1.1 OVERVIEW

This project addresses the problem of sentiment analysis in twitter; that is classifying tweets according to the sentiment expressed in them: positive, negative or neutral. Twitter is an online micro-blogging and social-networking platform which allows users to write short status updates of maximum length 140 characters. It is a rapidly expanding service with over 200 million registered users [24] - out of which 100 million are active users and half of them log on twitter on a daily basis - generating nearly 250 million tweets per day [20]. Due to this large amount of usage we hope to achieve a reflection of public sentiment by analysing the sentiments expressed in the tweets. Analysing the public sentiment is important for many applications such as firms trying to find out the response of their products in the market, predicting political elections and predicting socioeconomic phenomena like stock exchange. The aim of this project is to develop a functional classifier for accurate and automatic sentiment classification of an unknown tweet stream.

1.2 PURPOSE

We have chosen to work with twitter since we feel it is a better approximation of public sentiment as opposed to conventional internet articles and web blogs. The reason is that the amount of relevant data is much larger for twitter, as compared to traditional blogging sites. Moreover the response on twitter is more prompt and also more general (since the number of users who tweet is substantially more than those who write web blogs on a daily basis). Sentiment analysis of public is highly critical in macro-scale socioeconomic phenomena like predicting the stock market rate of a particular firm. This could be done by analysing overall public sentiment towards that firm with respect to time and using economics tools for finding the correlation between public sentiment and the firm's stock market value. Firms can also estimate how well their product is responding in the market, which areas of the market is it having a favourable response and in which a negative response (since twitter allows us to download stream of geo-tagged tweets for particular locations. If firms can get this information they can analyze the reasons behind geographically differentiated response, and so they can market their product in a more optimized manner by looking for appropriate solutions like creating suitable market segments. Predicting the results of popular political elections and polls is also an emerging application to sentiment analysis. One such study was conducted by Tumasjan et al. in Germany for predicting the outcome of federal elections in which concluded that twitter is a good reflection of offline sentiment

2. LITERATURE SURVEY

2.1 Existing Problem

Sentiment analysis of in the domain of micro-blogging is a relatively new research topic so there is still a lot of room for further research in this area. Decent amount of related prior work has been done on sentiment analysis of user reviews [x], documents, web blogs/articles and general phrase level sentiment analysis [16]. These differ from twitter mainly because of the limit of 140 characters per tweet which forces the user to express opinion compressed in very short text. The best results reached in sentiment classification use supervised learning techniques such as Naive Bayes and Support Vector Machines, but the manual labelling required for the supervised approach is very expensive. Some work has been done on unsupervised (e.g., [11] and [13]) and semi-supervised (e.g., [3] and [10]) approaches, and there is a lot of room of improvement. Various researchers testing new features and classification techniques often just compare their results to base-line performance. There is a need of proper and formal comparisons between these results arrived through different features and classification techniques in order to select the best features and most efficient classification techniques for particular applications.

2.2 Proposed Solution

The bag-of-words model is one of the most widely used feature model for almost all text classification tasks due to its simplicity coupled with good performance. The model represents the text to be classified as a bag or collection of individual words with no link or dependence of one word with the other, i.e. it completely disregards grammar and order of words within the text. This model is also very popular in Project Thesis Report 14 sentiment analysis and has been used by various researchers. The simplest way to incorporate this model in our classifier is by using unigrams as features. Generally speaking n-grams is a contiguous sequence of "n" words in our text, which is completely independent of any other words or grams in the text. So unigrams is just a collection of individual words in the text to be classified, and we assume that the probability of occurrence of one word will not be affected by the presence or absence of any other word in the text. This is a very simplifying assumption but it has been shown to provide rather good performance (for example in [7] and [2]). One simple way to use unigrams as features is to assign them with a certain prior polarity, and take the average of the overall polarity of the text, where the overall polarity of the text could simply be calculated by summing the prior polarities of individual unigrams. Prior polarity of the word would be positive if the word is generally used as an indication of positivity, for example the word "sweet"; while it would be negative if the word is generally associated with negative connotations, for example "evil". There can also be degrees of polarity in the model, which means how much indicative is that word for that particular class. A word like "awesome" would probably have strong subjective polarity along with positivity, while the word "decent" would although have positive prior polarity but probably with weak subjectivity.

3. Theoretical Analysis

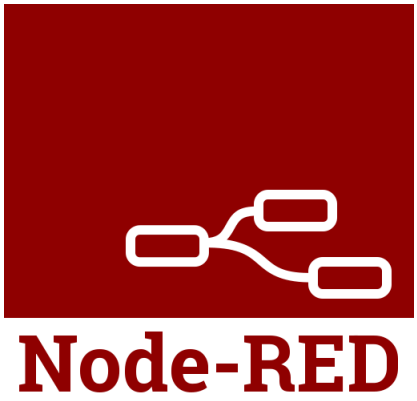
4.

4.1 Software Designing

IBM cloud *computing is a set of cloud computing services for business offered by the information technology company IBM.*

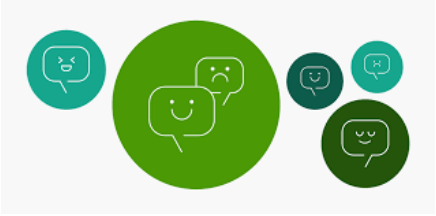
It provides many services like Node-Red, Watson Studio, etc for storing and processing data.

Node Red *is used for creating the User Interface (UI) application. Node-RED is a flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things. Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions.*



Watson Studio *helps data scientists and analysts prepare data and build models at scale across any cloud. With its open, flexible multicloud architecture,* **Watson Studio** *provides capabilities that empower businesses to simplify enterprise data science and AI: Automate AI lifecycle management with AutoAI.*

The ***IBM*** *Watson* ***Tone Analyzer*** *service uses linguistic analysis to detect emotional and language* ***tones*** *in written text. The service can analyze* ***tone*** *at both the document and sentence levels. ... You submit JSON, plain text, or HTML input that contains your written content to the service.*



5. Experimental Investigation

The process of designing a functional classifier for sentiment analysis can be broken down into five basic categories. They are as follows:

I. Data Acquisition
II. Human Labelling III.
III. Feature Extraction
IV. Classification V.
V. TweetMood

Web Application Data Acquisition: Data in the form of raw tweets is acquired by using the python library "tweestream" which provides a package for simple twitter streaming API [26]. This API allows two modes of accessing tweets: SampleStream and FilterStream. SampleStream simply delivers a small, random sample of all the tweets streaming at a real time. FilterStream delivers tweet which match a certain criteria. It can filter the delivered tweets according to three criteria:

- *Specific keyword(s) to track/search for in the tweets*
- *Specific Twitter user(s) according to their user-id's*
- *Tweets originating from specific location(s) (only for geo-tagged tweets).*

A programmer can specify any single one of these filtering criteria or a multiple combination of these. But for our purpose we have no such restriction and will thus stick to the SampleStream mode. Project Thesis Report 23 Since we wanted to increase the generality of our data, we acquired it in portions at different points of time instead of acquiring all of it at one go. If we used the latter approach then the generality of the tweets might have been compromised since a significant portion of the tweets would be referring to some certain trending topic and would thus have more or less of the same general mood or sentiment. This phenomenon has been observed when we were going through our sample of acquired tweets. For example the sample acquired near Christmas and New Year's had a significant portion of tweets referring to these joyous events and were thus of a generally positive sentiment. Sampling our data in portions at different points in time would thus try to minimize this problem. Thus forth, we acquired data at four different points which would be 17th of December 2011, 29th of December 2011, 19th of January 2012 and 8th of February 2012. A tweet acquired by this method has a lot of raw information in it which we may or may not find useful for our particular application. It comes in the form of the python "dictionary" data type with various key-value pairs. A list of some key-value pairs are given below:

- *Whether a tweet has been favourited*
- *User ID*
- *Screen name of the user*
- *Original Text of the tweet*

- *Presence of hashtags*
- *Whether it is a re-tweet*
- *Language under which the twitter user has registered their account*
- *Geo-tag location of the tweet*
- *Date and time when the tweet was created*

Since this is a lot of information we only filter out the information that we need and discard the rest. For our particular application we iterate through all the tweets in our sample and save the actual text content of the tweets in a separate file given that Project Thesis Report 24 language of the twitter is user's account is specified to be English. The original text content of the tweet is given under the dictionary key "text" and the language of user's account is given under "lang". Since human labelling is an expensive process we further filter out the tweets to be labelled so that we have the greatest amount of variation in tweets without the loss of generality. The filtering criteria applied are stated below:

- *Remove Retweets (any tweet which contains the string "RT")*
- *Remove very short tweets (tweet with length less than 20 characters)*
- *Remove non-English tweets (by comparing the words of the tweets with a list of 2,000 common English words, tweets with less than 15% of content matching threshold are discarded)*
- *Remove similar tweets (by comparing every tweet with every other tweet, tweets with more than 90% of content matching with some other tweet is discarded)*

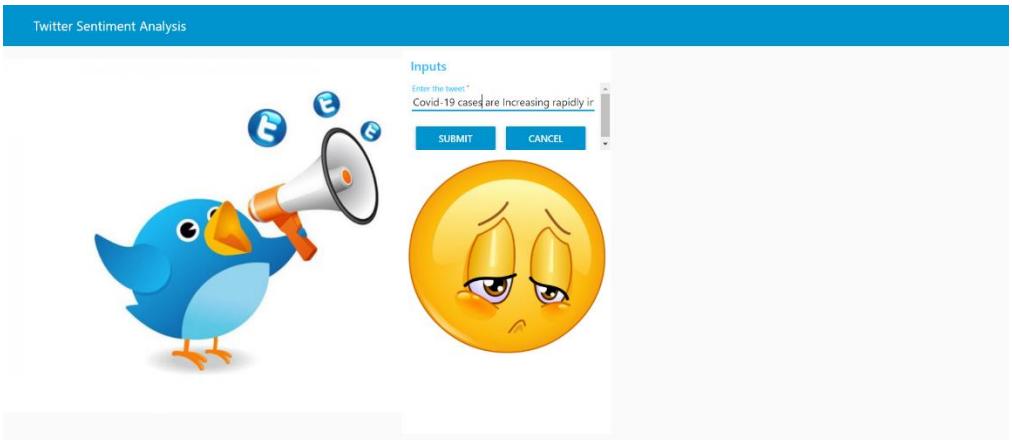
After this filtering roughly 30% of tweets remain for human labelling on average per sample, which made a total of 10,173 tweets to be labelled. Human Labelling: For the purpose of human labelling we made three copies of the tweets so that they can be labelled by four individual sources. This is done so that we can take average opinion of people on the sentiment of the tweet and in this way the noise and inaccuracies in labelling can be minimized. Generally speaking the more copies of labels we can get the better it is, but we have to keep the cost of labelling in our mind, hence we reached at the reasonable figure of three. We labelled the tweets in four classes according to sentiments expressed/observed in the tweets: positive, negative, neutral/objective and ambiguous. We gave the following guidelines to our labellers to help them in the labelling process: Project Thesis Report 25

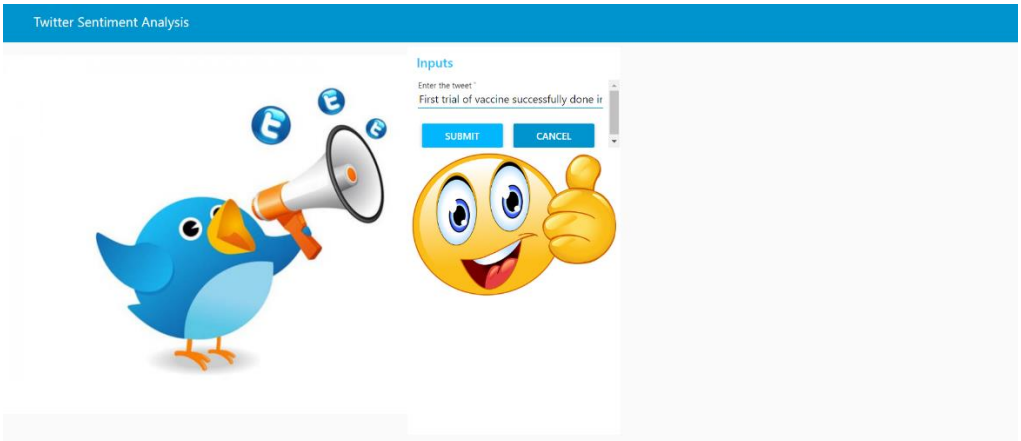
- *Positive: If the entire tweet has a positive/happy/excited/joyful attitude or if something is mentioned with positive connotations. Also if more than one sentiment is expressed in the tweet but the positive sentiment is more dominant. Example: "4 more years of being in shithole Australia then I move to the USA! :D".*
- *Negative: If the entire tweet has a negative/sad/displeased attitude or if something is mentioned with negative connotations. Also if more than one sentiment is expressed in the tweet but the negative sentiment is more dominant. Example: "I want an android now this iPhone is boring :S".*

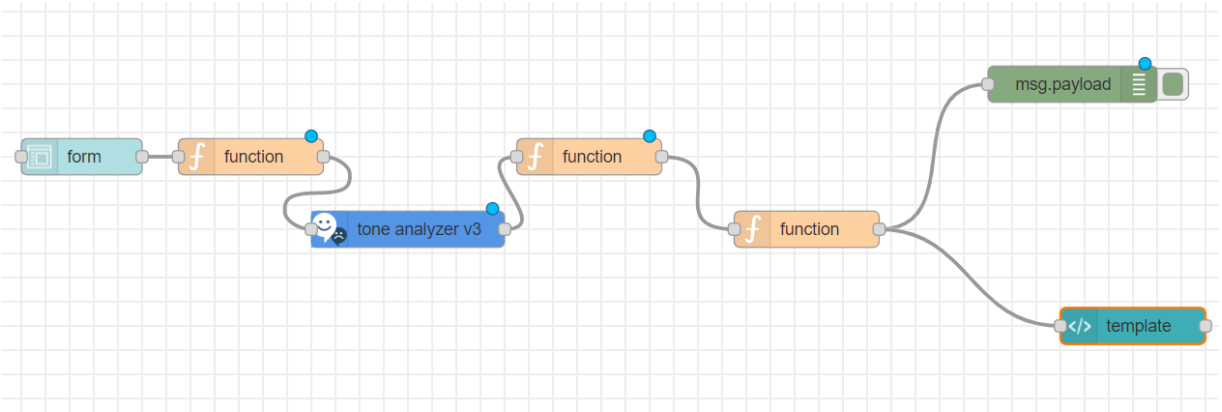
• Neutral/Objective: If the creator of tweet expresses no personal sentiment/opinion in the tweet and merely transmits information. Advertisements of different products would be labelled under this category. Example: "US House Speaker vows to stop Obama contraceptive rule...".

• Ambiguous: If more than one sentiment is expressed in the tweet which are equally potent with no one particular sentiment standing out and becoming more obvious. Also if it is obvious that some personal opinion is being expressed here but due to lack of reference to context it is difficult/impossible to accurately decipher the sentiment expressed. Example: "I kind of like heroes and don't like it at the same time...". Finally if the context of the tweet is not apparent from the information available. Example: "That's exactly how I feel about avengers haha". • : Leave the tweet unlabelled if it belongs to some language other than English so that it is ignored in the training data. Besides this labellers were instructed to keep personal biases out of labelling and make no assumptions, i.e. judge the tweet not from any past extra personal information and only from the information provided in the current individual tweet.

6. RESULT







Node Red Flow

According to the sentiment of the tweet, the emoji appears on the screen.

Smiling emoji appears for the good news and sad emoji for the sad ones.

7. Conclusion and Future Scope

The task of sentiment analysis, especially in the domain of micro-bloging, is still in the developing stage and far from complete. So we propose a couple of ideas which we feel are worth exploring in the future and may result in further improved performance. Right now we have worked with only the very simplest unigram models; we can improve those models by adding extra information like closeness of the word with a negation word. We could specify a window prior to the word (a window could for example be of 2 or 3 words) under consideration and the effect of negation may be incorporated into the model if it lies within that window. The closer the negation word is to the unigram word whose prior polarity is to be calculated, the more it should affect the polarity. For example if the negation is right next to the word, it may simply reverse the polarity of that word and farther the negation is from the word the more minimized ifs effect should be. Apart from this, we are currently only focusing on unigrams and the effect of bigrams and trigrams may be explored. As reported in the literature review section when bigrams are used along with unigrams this usually enhances performance. However for bigrams and trigrams to be an effective feature we need a much more labeled data set than our meager 9,000 tweets. Right now we are exploring Parts of Speech separate from the unigram models, we can try to incorporate POS information within our unigram models in future. So say instead of calculating a single probability for each word like P(word | obj) we could instead have multiple probabilities for each according to the Part of Speech the word belongs to. For example we may have P(word | obj, verb), P(word | obj, noun) and P(word | obj, adjective). Pang et al. used a somewhat similar approach and claims that appending POS information for every unigram results in no significant change in performance (with Naive Bayes performing slightly better and SVM having a slight decrease in performance), while there is a significant decrease in accuracy if only adjective unigrams are used as features. However these results are for classification of reviews and may be verified for sentiment analysis on micro blogging websites like Twitter.

One more feature we that is worth exploring is whether the information about relative position of word in a tweet has any effect on the performance of the classifier. Although Pang et al. explored a similar feature and reported negative results, their results were based on reviews which are very different from tweets and they worked on an extremely simple model. In this research we are focussing on general sentiment analysis. There is potential of work in the field of sentiment analysis with partially known context. For example we noticed that users generally use our website for specific types of keywords which can divided into a couple of distinct classes, namely: politics/politicians, celebrities, products/brands, sports/sportsmen, media/movies/music. So we can attempt to perform separate sentiment analysis on tweets that only belong to one of these classes (i.e. the training data would not be general but specific to one of these categories) and compare the results we get if we apply general sentiment analysis on it instead.

8. Bibliography

- *Albert Biffet and Eibe Frank. Sentiment Knowledge Discovery in Twitter Streaming Data. Discovery Science, Lecture Notes in Computer Science, 2010, Volume 6332/2010, 1-15, DOI: 10.1007/978-3-642-16184-1_1Introduction*
- *Alec Go, Richa Bhayani and Lei Huang. Twitter Sentiment Classification using Distant Supervision. Project Technical Report, Stanford University, 2009.*
- *SmartInternz Webinars*
- *Mentors*