Sentiment Analysis on Tweets

B.K.S. Ishanka
Computer Science Department
University Of Moratuwa
Katubedda, Sri Lanka
shehan.20@cse.mrt.ac.lk

E.G.D. Sandaruwan
Computer Science Department
University Of Moratuwa
Katubedda, Sri Lanka
dinesh.20@cse.mrt.ac.lk

Y.H.S. Amarasiri
Computer Science Department
University Of Moratuwa
Katubedda, Sri Lanka
amarasiriyhs.20@uom.lk

***Abstract*—Sentiment analysis is one of the interesting research areas in the text mining field. Sentiment analysis is very challenging because of sarcasm, types of negations, word ambiguity and multipolarity. Even though it is very challenging, It is very useful for applications such as Identifying brand reputation, conducting market research and social media monitoring etc. In this project, we aim to mine and to summarize online opinions in tweets. For our experiment, we have used several natural language processing techniques to extract the features from text and supervised machine learning algorithms to identify the sentiment about the text. Here, we present the comparison of the accuracy of several feature extraction and classification algorithm combinations. We also present sentiment analysis related literature.**

Keywords—NLP, sentiment analysis, text mining, opinion mining, supervised learning, text mining, classification, word embedding, bag of words, Head word, TF-IDF, Logistic regression, Support Vector Machine, SVM, Random Forest, Neural network, LSTM

I. Introduction

Sentiment analysis, also known as opinion mining, is one of the interesting research areas in the text mining field. Sentiment analysis is the process of identifying the emotional tone behind the body of text. It involves building a system that can collect the user's opinions and then examine and classify them according to the polarity of the text.

There are several sentiment analysis applications in various fields such as social media monitoring, customer support management, analysing customer feedback, product analysis, brand monitoring, reputation management, market research and competitive research. In this project, we mainly focus on social media monitoring applications, so the experiment has been done based on twitter data.

Polarity (positive, negative, neutral), emotions and feelings (angry, happy, sad) , intentions (interested v. not interested) and urgency (urgent, not urgent) are focused by sentiment analysis models. According to the required interpretation of text, we can categorize the text to meet our sentiment analysis needs. Fine-grained Sentiment Analysis , Emotion detection, Aspect-based Sentiment Analysis and Multilingual sentiment analysis are the most popular types of sentiment analysis and also sentiment analysis can be investigated at three different levels.

1) Document level : Determine the overall sentiment of the document
2) Sentence level : Determine the sentiment of each sentences
3) Aspect level : Determine the sentiment with respect to the specific aspects of entities

Machine learning and lexical based approaches are the existing techniques for sentiment analysis. All sentiment analysis algorithms can be divided into main three categories.

1) Rule-based : uses a set of manually derived rules to identify the polarity
2) Automatic : uses machine learning techniques and don't rely on manually derived rules
3) Hybrid : uses both rule based and machine learning techniques

The aim of this paper is presenting various types of techniques used for sentiment analysis. In this experiment, we mainly use machine learning approaches for sentiment analysis on twitter data. The labelled (positive, neutral and negative) dataset is used for our experiment. Our methodology mainly focuses on different feature extraction techniques and different classification techniques. Bag of words, bigrams, head words and the word embedding techniques are used to get the feature vectors. Logistic regression, SVC (support vector classifier), Random Forest and Neural Networks are used as classification techniques in our methodology[1].

The remaining paper is structured as follows. In section 2, we present and discuss related literature. In section 3, we present our methodology. In section 4 we present and discuss the experimental results. In section 5, we conclude the paper with a summary of our results.

II. Literature Review

In Sentiment analysis we use natural language processing and information to extract user's comments or reviews. There is a rich literature on sentiment analysis and we have only selected a few to analyse.

Alexander's and Patricks's Twitter as a Corpus for Sentiment Analysis and Opinion Mining [3] shows that microblogging today has become a very popular communication tool among internet users. As the microblogging platforms and services grow everyday for example manufacturing companies may be interested in how positive people are about their products and political parties may be interested to know if people are supporting them or not likewise. All this information can be obtained from microblogging services. In this paper, they have studied how microblogging can be used for sentiment analysis purposes. Using Twitter API they have collected a corpus of 300000 text posts and formed a dataset of three classes: positive sentiments, negative sentiments, and a set of objective texts (no sentiments). They have also used emoticons to mine the opinion of corpus. For the feature extraction, the presence of an n-gram has been used as a binary feature. They have built a sentiment classifier using the multinomial Naïve Bayes classifier. They have also tried SVM (Alpaydin, 2004) and CRF (Lafferty et al., 2001), however the Naïve Bayes classifier has yielded the best results. To increase the accuracy of the classification, they have discarded common n-grams that don't strongly indicate any sentiment nor indicate objectivity of a sentence.

[1]Code is available is at https://github.com/ShehanIshanka/sentiment-analysis

Content vs. context for sentiment analysis of Aisopos and Fotis[6] says that today with the advent of world wide web microblogging has become a popular service because it includes valuable information regarding from product marketing to politics and policy making. Sentiment analysis can be distinguished in three tasks; Document level sentiment analysis, Sentence level sentiment analysis and Feature level sentiment analysis. Here in this paper, they have focused on document level sentiment analysis. Among all the microblogging platforms in this paper, they have selected Twitter for developing and testing the approach due to following advantages it conveys, strict interaction, social graph, public content and timed activity. Also they have focused on intrinsic characteristics of Twitter like hashtags, mentions, external pointers and emoticons. As the textual patterns are typically captured through language specific representation models that detect frequent sequences of words here they have used character based models. Second, there is no standard tokenization approach for multilingual documents; words are typically identified through the white spaces that delimit them, but there are languages, such as Chinese, where different words can be concatenated in a single token.

Most importantly, though, term-based models depend on dictionary-based and language-specific techniques, such as stemming and lemmatization, to tackle synonymy words with the same meaning, but different syntactic form (e.g. quickly and rapidly), which are considered as distinct features, sophisticated methods for matching them are employed. Such techniques are inapplicable to the user generated, multilingual microblog content, whose inherent noise (i.e., spelling mistakes) and neologisms further aggravate synonymy. As a result, both the effectiveness and the efficiency of term-based models may be significantly degraded in this approach due to that reason they have mainly focused on character n-grams and the character n-gram graphs. In addition to the textual patterns, another reliable source of evidence for detecting a tweet's sentiment is its social context. As such, they have defined an indication that associates it directly or indirectly with other messages (i.e. hashtags and URLs) or with the members of the underlying social network (i.e. the author of the message, her friends as well as the users mentioned in it). To quantify the effect of the above issue, they have introduced a metric that estimates the aggregate sentiment of a set of tweets called the Polarity Ratio. To examine the performance of their models in practical settings, they have conducted a thorough experimental study on a large-scale multilingual collection of real Twitter messages comprising 476 million tweets posted in a period of 7 months, from June 2009 until December 2009. To evaluate the performance of their models, they have employed the 10-fold cross-validation approach. To thoroughly evaluate the performance of their models they have employed three of the most popular and established classification algorithms Naïve Bayes (NB), C4.5 and the SVM. All models and experiments have been fully implemented in Java, version 1.6.

Predicting Stock Market Behavior using Data Mining Technique and News Sentiment Analysis[7] paper shows that stock market decision making is a very difficult and important task due to the complex behavior and the unstable nature of the stock market. Understanding this complex behaviour will help to determine the best time to buy or sell stocks in order to achieve the best profit on their investments. It was found that news has an influence on the stock price behavior [8]. The study reveals that there is a relation between news sentiment and stock prices changes. In this paper, the proposed approach uses sentiment analysis for financial news, along with features extracted from historical stock prices to predict the future behavior of the stock market. The prediction model uses Naïve Bayes and K-NN algorithms. Here in this paper, proposed architecture combines the analysis of the stock market news and the historical prices together, in order to boost the classification accuracy of the stock market behavior. Two data sets are collected, news and numeric data. For news, data is collected from different data sources such as nasdaq.com, Reuters, Wall Street Journal, marketwatch.com, zacks.com, yahoo finance, Google finance, ecomomics.com. There are three main components in this architecture: the Sentiment Analysis Component, Stocks Numeric Data Preprocessing Component and Prediction Model component. Here for the Sentiment Analysis Component, they have used N-Gram, TF-IDF, Naïve Bayesian classifiers to complete the process. Stocks Numeric Data Preprocessing Component outputs positive or negative according to the news sentiment. In The Prediction Model component, they have used KNN classifier to predict the stock class based on the collected feature set. KNN classifier has been used to predict the stock trend; fall or raise.

III. METHODOLOGY

As proposed in the literature, there are several kinds of traditional and deep learning techniques to mine important information from textual data. For sentiment analysis, there are different kinds of techniques and approaches. Here we have experimented with three kinds of approaches.

1) Feature based approach
2) Word embedding approach
3) Neural Network approach

Before describing the approaches, the dataset has been preprocessed. Preprocessing steps are listed down below.

1. Some of the records are empty records. So that, those kind of records are removed.
2. Since data is collected from twitter users, most of the texts are spelled incorrectly. So that texts are spell corrected.
3. Since punctuations and numeric values do not provide any insights on opinions, they are removed.
4. Since the special characters do not provide any insights on opinions, they are removed.
5. Since single characters do not provide any insights on opinions, they are removed.
6. Since single characters from the start do not provide any insights on opinions, they are removed.
7. Extra multiple spaces can be troublesome when tokenizing. So that they are substituted with single space.
8. Byte characters are removed by removing prefixed 'b'.
9. All the words are converted to lowercase for a unique word format.
10. Since stop words do not provide any insights on opinions, they are removed.

The preprocessed data set is used for the approaches mentioned above. The approaches are described below.

1) *Feature based approach*

In this approach, several features are developed and then those features are used in classification tasks. The features are

1. Bag of Words
2. Bigrams
3. Head words

A bag-of-words model, or BoW for short, is a method of extracting features from text for use in modeling. A bag-of-words is a representation of text that describes the occurrence of words within a document. It involves two things:

a) A vocabulary of known words

b) A measure of the presence of known words

It is called a "bag" of words, because any information about the order or structure of words in the document is discarded. The model is only concerned with whether known words occur in the document, not where in the document. The intuition is that documents are similar if they have similar content. Bag of Words feature is developed by using the Python Sci-Kit Learn package.

Bigrams means two adjacent words in the words. So here bigrams of tweets are captured using the Sci-Kit Learn package, the same as above.

Head words are captured for sentences in tweets and then the extracted words are vectorized as the bag of words model. For head word identification, dependency parsing is needed. Dependency parsing is the process of analyzing the grammatical structure of a sentence based on the dependencies between the words in a sentence. This is implemented by using the Spacy package.

These features are aggregated and then experimented using 3 traditional machine learning classifiers which are logistic regression, Support Vector Machine and Random Forest.

2) *Word embedding approach*

A word embedding is a learned representation for text where words that have the same meaning have a similar representation. Here words or phrases from the vocabulary are mapped to vectors of real numbers. Word embeddings for English language are downloaded from the word2vec model trained on news articles provided by Google with 300 dimensions. Two kinds of methods are used with the help of word embeddings.

1. Mean embeddings

Mean embedding for a particular record is generated by extracting word embeddings for the words of the preprocessed record and then calculating the mean embedding of it.

2. TF-IDF mean embeddings

TF-IDF (term frequency-inverse document frequency) was originally used for document search and information retrieval. But in NLP, it plays a major role in certain

TABLE I. Dataset

	Positive	**Neutral**	**Negative**	**Total**
Train	8582	11118	7781	27481
Test	1001	1430	1103	3534

TABLE II. Accuracies of Models

Approaches			**Accuracy**
Feature based approach		Logistic Regression	0.41
		SVM	0.42
		Random Forest	0.4
Word embedding approach	Mean embeddings	Logistic Regression	0.66
		SVM	0.69
		Random Forest	0.63
	TF-IDF mean embeddings	Logistic Regression	0.61
		SVM	0.66
		Random Forest	0.58
Neural Network approach			**0.71**

TABLE III. Precision, Recall and F1 Score of Models

Approaches			**Precision**	**Recall**	**F1 score**
Feature based approach		Logistic Regression	0.38	0.38	0.37
		SVM	0.38	0.38	0.36
		Random Forest	0.35	0.36	0.33
Word embedding approach	Mean embeddings	Logistic Regression	0.68	0.65	0.66
		SVM	0.71	0.69	0.70
		Random Forest	0.67	0.61	0.62
	TF-IDF mean embeddings	Logistic Regression	0.63	0.60	0.61
		SVM	0.67	0.65	0.66
		Random Forest	0.62	0.56	0.57
Neural Network approach			**0.72**	**0.70**	**0.71**

applications. TF-IDF for a word in a document is calculated by multiplying two different metrics: term frequency and inverse document frequency.

There are several ways of calculating the term frequency of a word in a document, with the simplest being a raw count of instances a word appears in a document. Then, there are ways to adjust the frequency, by length of a document, or by the raw frequency of the most frequent word in a document.

The inverse document frequency of the word across a set of documents means how common or rare a word is in the entire document set. The closer it is to 0, the more common a word is. This metric can be calculated by taking the total number of documents, dividing it by the number of documents that contain a word, and calculating the logarithm.

TF-IDF mean embedding for a particular record is generated by extracting TF-IDF embeddings for the words of the preprocessed record and then calculating the TF-IDF mean embedding of it.

These two methods are experimented using 3 traditional machine learning classifiers which are logistic regression, Support Vector Machine and Random Forest as same as above feature based approach.

3) *Neural Network approach*

In this approach LSTM network is implemented with 128 LSTM cells. The final two layers of the network are dense layers with 64 units and 3 units (classes) respectively. Relu function is used as the activation function except in the last dense layer activation function is softmax. Categorical cross entropy is used as the loss function and Adam optimizer is used as the optimization function. Bag of words is used as features and classes are transformed to one hot encoded vector.

IV. EXPERIMENT

The experiment is done on a HP laptop with i5 processor and 16GB RAM. The dataset[2] used for training and testing is extracted from kaggle Tweet Sentiment Extraction competition. The data set includes separate datasets for training and tests. The output has 3 classes which are positive, neutral and negative. Table I shows the statistics in each set in the dataset.

Each of the approaches mentioned in the methodology section is tested and the results are provided in table II and table III.

From all the approaches attempted, the neural network approach has the highest accuracy with highest F1 score.

V. CONCLUSION

In this research, we attempted three separate approaches to derive sentiments of the tweets using neural networks and traditional machine learning approaches. Out of them, the neural network approach seems to provide the best results. Even though the feature based approach does not provide better results, with more features like POS tags it might be able to provide good results.

We have used several preprocessing steps to filter out non-essential textual parts from tweets. But we can further incorporate preprocessing steps like named entity recognition, tweet tokenization etc. Although we removed stop words, some stop words like ‘do not’,’does not’ resemble importance in identifying sentiments.

VI. ACKNOWLEDGMENT

This research is completed on behalf of the semester 3 module, CS5227 – Data Mining, of MSc in Computer Science specialized in Data Science Course of batch 2020. We would like to thank Dr. Charith Chitraranjan for the guidance to complete the research and the lectures on Data Mining.

VII. REFERENCES

[1] *https://osf.io/6xc4y/download*

[2] *Ortigosa, Alvaro, José M. Martín, and Rosa M. Carro. "Sentiment analysis in Facebook and its application to elearning." Computers in Human Behavior 31 (2014): 527-541*

[3] *Pak, Alexander, and Patrick Paroubek. "Twitter as a Corpus for Sentiment Analysis and Opinion Mining." LREC. Vol. 10. 2010.*

[4] *Agarwal, Apoorv, et al. "Sentiment analysis of twitter data." Proceedings of the Workshop on Languages in Social Media. Association for Computational Linguistics, 2011.*

[5] *Scholar, P. G. "Big-SoSA: Social Sentiment Analysis and Data Visualization on Big Data."*

[6] *[Aisopos, Fotis, et al. "Content vs. context for sentiment analysis: a comparative analysis over microblogs." Proceedings of the 23rd ACM conference on Hypertext and social media. ACM, 2012.*

[7] *Predicting Stock Market Behavior using Data Mining Technique and News Sentiment Analysis*

[8] *W. Walter, K. Ho, W. R. Liu, and K. Tracy, ―The relation between news events and stock price jump : an analysis based on neural network,‖ 20th Int. Congr. Model. Simulation, Adelaide, Aust. 1–6 December 2013 www.mssanz.org.au/modsim2013, no. December, pp. 1–6, 2013*

[2]Dataset is available is at https://www.kaggle.com/c/tweet-sentiment-extraction/data