Sentiment Analysis of tweets using Supervised Machine Learning

Talal Najam, Kadhar Batcha Nowshath, Vazeerudeen
Asia Pacific University Malaysia
chelseatalal@gmail.com, dr.nowshath@apu.edu.my, vazeerudeen@gmail.com

***Abstract*— This paper addresses the problem of analyzing sentiment on a social media platform called Twitter; that is to identify and classify whether a tweet expresses a positive sentiment or a negative sentiment. Twitter is an online social networking and micro-blogging website where people from around the world communicate and write short updates, called as "tweets" without exceeding the character limit of 140 characters. It is a continuously expanding web service, having an average of 330 million monthly active users as of the fourth quarter of 2017. This large number of users results in enormous amounts of publicly available data that may be used to gather insight and to reflect the public sentiment by analyzing the tweets. There are many applications to performing sentiment analysis in today's world in various fields and industries such as gathering market research insights, predicting political campaigns' outcomes etc. Performing sentiment analysis with traditional techniques of writing manual rules and conditions becomes complex as the number of rules to be written, keep increasing to tackle the raw natural language. The aim of this paper is to investigate further into the sub-domain of natural language processing called sentiment analysis, and develop a statistics based functional classifier and incorporate it with other machine learning models to create a vote based classifier algorithm, for automating and performing accurate sentiment classification of desired tweets.**

***Index Terms*— Image Processing, Biometrics, Emotion Recognition, Kanade-Lucas-Tomasi (KLT) algorithm, Viola-Jones algorithm**

1. Introduction

The evolution of internet and technological advancement throughout the years has changed the way information is obtained. The role of an ordinary person, from just being part of an audience, receiving information, has also changed to being a source of information and content provider with the rise of social media. In this technological era, with enormous amounts of data available digitally on the Internet today, a deep understanding of natural language by a computer is still an unresolved problem that results to become a major barrier to opportunities. These days, people rely heavily on opinions and reviews that online forum and social media provide. Similarly, organizations that wanted their customer's opinions and reviews, used methods like customer surveys, questionnaires and opinion polls to study the customer's opinions. Today, these organizations are evolving from manual, time-consuming methods, towards using and analyzing social media to gather insight about their customer base. This lack of understanding of human language by computers is a barrier to many opportunities in various domains. Sentiment Analysis, sometimes also referred to as 'Opinion mining' is a field of study that studies people's attitudes, sentiments and opinions towards a particular topic, brand, product etc. (Liu, 2012). Sentiment Analysis is a Natural Language Processing problem and throughout the recent years, has developed an active research area. It is a rising field and due to its significant role in businesses and active research area, it has expanded from just computer sciences to various other fields such as management sciences and it is applicable to various domains. Sentiment Analysis is very crucial in modern-world because of its impact and involvement within a society. For the proposed solution, the domain chosen for sentiment analysis is movie reviews extracted from a social media website called "Twitter".

2. Problem Context

Using the traditional methods of performing sentiment analysis of using lexicon-based approaches and dictionary-based approaches involves writing a lot of manual rules and conditions. As the internet is evolving, digitization has changed the way information is processed and analyzed. Due to that, these manually written rules start to stack up making the code base complex, and eventually break when tackling the ever-evolving natural language. Moreover, the manual means of performing text classification is comparatively inefficient and slower. A lexicon is a list of words and phrases such as "good, bad, happy", each having an assigned sentiment score value for every word to expresses its sentiment. Systems that perform sentiment analysis relying on using just lexicon to perform analysis, may express the whole sentence as positive if there is a positive connotation such as "excellent" in the sentence and vice versa, thus, producing unwanted results. This is because sentiment analysis applications that are lexicon based, corpus based or dictionary based, break down and tokenize the sentence into a "bag of words" model, and then compare the positive and negative words in a sentence to come up with a sentiment polarity score. Therefore, the phrase "this cake is not bad" would be interpreted negatively due to the word "bad" appearing in the sentence despite having neutral or arguably positive sentiment. Similarly, there are problems in processing phrases such as "is this car good?" where there is a neutral or no sentiment at all. These ambiguity problems are very significant in Sentiment Analysis and can be very crucial for businesses and organizations that rely heavily on customer's opinions for large-scale decision-making.

Alternatively, for organizations to use human employees for analyzing large amount of data would result in hiring large teams of people to work together in order to manage all the data. In this manual process, conflicts can arise on deciding the

analysis result of a sentiment. Furthermore, humans get tired, annoyed or bored when performing their job and this can pose a threat when there is a need to make quick decisions while processing data. Not only this, time is also a crucial factor when getting lots of data or sentiments analyzed by humans since an average human employee will find it difficult to analyze an enormous chunk of text and to summarize it effectively.

3. Proposed Solution

The aim is to propose a standalone web application system that analyzes and extracts subjective information from a given piece of text in natural language such as a sentiment or an opinion; either from social media (Twitter) or entered manually, to create resulting actionable knowledge which is to be used in decision-making by organization members or decision support systems.

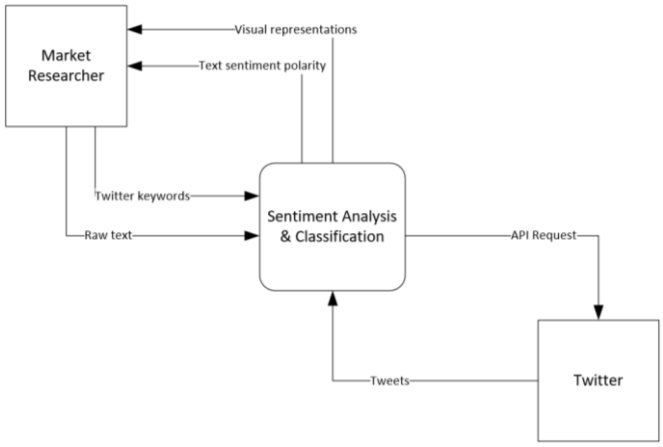

Figure 1. Context Diagram of the System

The context diagram in figure 1 is used to represent the information system as a single high-level process, to display the links and relationships that it has with external entities in the proposed system (Adams, 2018). The context diagram above shows the two entities that interact with the sentiment analysis and classification system. For instance, Market Researcher (one of the target users) and Twitter are the respective entities that provide inputs and receive an output from the system. The market researcher either provides raw text or Twitter keywords, the system then returns the text sentiment polarity as well as the visualized result to the user, which in this case is the market researcher. The other entity "Twitter" provides the system with relevant tweets upon receiving the API request from the system. The main algorithm of this system can be perceived from the data flow diagram level-0 shown in figure 2, as it expands on the context diagram shown earlier and illustrates the main processes of the system. It shows how the data is provided by the entities, received by the system, the processes it goes through, such as pre-processing, negation handing, voted classifier etc. and back to the user who in this case is the market researcher, in the form of visualized data and pie chart.

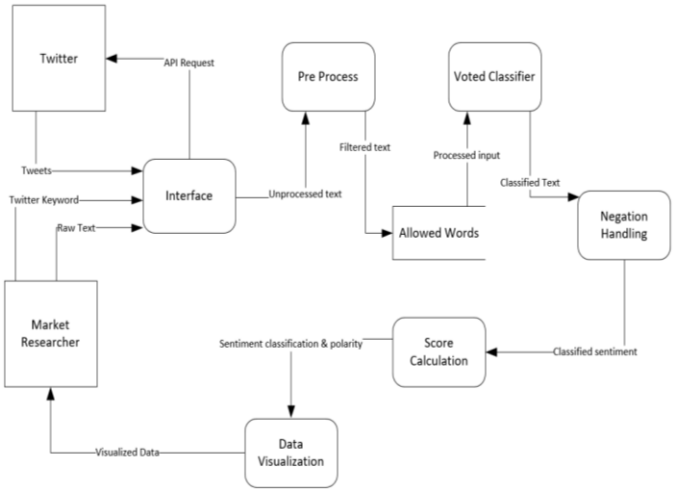

Figure 2. Data Flow Diagram Level- 0

The two main entities interacting with the system are the same as in the context diagram, the market researcher (user) and the Twitter API. Both the entities interact with the interface to input data, raw text in the case of the manual sentiment, and keywords and number of tweets in the case of twitter sentiment. In the latter case, an API request is generated and sent to the twitter API, which returns the tweet accordingly. After the interface, the raw input or the unprocessed data is sent to be preprocessed, in terms of encoding it with utf-8, tokenizing the words, removing unwanted words (stop words) etc. to filter and remove words that do express any sentiment, the output is compared to the list of allowed words, and filtered again. The final filtered text is then used to form labelled feature sets, which are then sent to the voted classifier. Voted classifier is a technique that is used commonly in natural language processing and text classification problems (Sriparna Saha, 2013). It is comprised of multiple classifiers, each trained and tested on the dataset. When all the classifiers in the voted classifier algorithm have successfully classified the text, the final result of all the classifiers is then determined using mode function, which returns the result that most classifiers "voted for" or classified. The result of the classifier is then sent to the negation handling process, which checks for the number of negations present in the input and adjusts the result accordingly. The output is sent to the score calculator, which uses a word dictionary to generate the resulting score manually. Lastly, after the completion of the whole process, the results are sent to the data visualization process that forms the pie chart and outputs to the user.

4. Evaluation

As for test results, the system was tested against the expected outputs, and due to the careful planning and training of the classifiers on large dataset consisting of 10000 movie reviews, the unit tests were successful and met the respective expected outputs successfully. The classifier was able to predict the sentences from Twitter as well as the manually written phrases and sentences correctly. Moreover, the manual rules, used for the negation handling, sentiment shifters and score calculation produced the right results as well. One thing to notice is that the

reason for these successful tests might be that the classifier did not encounter any tweet containing sarcasm or multiple entities, which is common in social media, and as these bring up completely new challenges, they were out of the scope of this project. The vote based classifier proved to be working correctly in case of classifying text from Twitter, and to be able to implement in a web application with optimal response time. This was due to using the "Pickle" module that serializes the python object so it can be used later without the need of training and testing again. Not only the classifiers, but also the document objects, training sets etc. were saved with the same method, resulting in producing the output in an optimal time of less than a second.

5. Conclusion

The task of sentiment analysis, especially in the domain of social media and micro-blogging, is still under progress and going through a 'developing' stage and thus, far from complete at this point. After reviewing and conducting research, it became clear that in this project, out of many sentiment analysis problems, the developer could only focus on a small portion of the existing problems. It still feels incomplete in terms of investigating the entire Natural Language Processing domain, since it is very large, developing and there are multiple sub-fields within itself that require more research and a longer time frame.

As for the scope of this project, the data source either was tweets from Twitter, or manually entered text. The voted classifier algorithm seemed to be working well as the tweets tested in the results stage were classified correctly by the classifier, and after passing through the manual rules, produced correct outputs as expected. However, for future, the developer would like to explore more into working with other social media websites such as Facebook, Reddit etc. This is because these websites have various methods of users-posts and various means for users to express their opinions and sentiments (e.g. Facebook post reactions), while also not having the input restriction of just 140 characters that Twitter currently has.

Moreover, the developer would like to explore the areas of sentiment analysis and NLP that focus on detecting sarcasm, and producing accurate results regardless of having informal language within text. This is because social media websites such as Twitter, are used by users that browse the website casually in their spare time and thus, make use of a lot of slangs while avoiding to write their sentences in formal English, which most datasets and normal Corpuses support. Lastly, the developer would like to explore and investigate other techniques such as unsupervised machine learning, and resources such as different data sets with multiple categories.

References

Adams, C., 2018. *What is a Context Diagram and what are the benefits of creating one?*. [Online]
Available at:
http://www.modernanalyst.com/Careers/InterviewQuestions/tabid/128/ID/1433/What-is-a-Context-Diagram-and-what-are-the-benefits-of-creating-one.aspx
[Accessed 28 July 2018].

Gil, P., 2018. *What Is Twitter & How Does It Work?*. [Online]
Available at: https://www.lifewire.com/what-exactly-is-twitter-2483331
[Accessed 3 July 2018].

Liu, B., 2012. *Sentiment Analysis and Opinion Mining.* 1st ed. Toronto: Morgan & Claypool Publishers.

Sriparna Saha, A. E., 2013. Combining multiple classifiers using vote based classifier ensemble technique for named entity recognition. *Data & Knowledge Engineering*, May, p. 85:15–39.