Automated analysis techniques to extract sentiments and opinions conveyed in the user comments on social media.

Vikrant Waghmare, Mahesh Pimpalkar, Prof. Vaishali Londhe
1,2,3Yadavrao Tasgaonkar Institute of Engineering & Technology
University of Mumbai

Abstract - *Ubiquitous presence of internet, advent of web 2.0 has made social media tools like blogs, Facebook, Twitter very popular and effective. People interact with each other, share their ideas, opinions, interests and personal information. These user comments are used for finding the sentiments and also add financial, commercial and social values. However, due to the enormous amount of user generated data, it is an expensive process to analyze the data manually. Increase in activity of opinion mining and sentiment analysis, challenges are getting added every day. There is a need for automated analysis techniques to extract sentiments and opinions conveyed in the user comments.*
Sentiment analysis is the automated process of understanding an opinion about a given subject from written or spoken language. In a world where we generate 2.5 quintillion bytes of data every day, sentiment analysis has become a key tool for making sense of that data. This has allowed companies to get key insights and automate all kind of processes.

***Key Words*: Ubiquitous, machine learning; opining mining; sentiment analysis; sentiment classification**

1. INTRODUCTION

Sentiment Analysis also known as *Opinion Mining* is a field within Natural Language Processing (NLP) that builds systems that try to identify and extract opinions within text. Usually, besides identifying the opinion, these systems extract attributes of the expression e.g.:

- *Polarity*: if the speaker express a *positive* or *negative* opinion,
- *Subject*: the thing that is being talked about,
- *Opinion holder*: the person, or entity that expresses the opinion.

Currently, sentiment analysis is a topic of great interest and development since it has many practical applications. Since publicly and privately available information over Internet is constantly growing, a large number of texts expressing opinions are available in review sites, forums, blogs, and social media.
With the help of sentiment analysis systems, this unstructured information could be automatically transformed into structured data of public opinions about products, services, brands, politics, or any topic that people can express opinions about. This data can be very useful for commercial applications like marketing analysis, public relations, product reviews, net promoter scoring, product feedback, and customer service.
Sentiment analysis can be applied at different levels of scope:

- Document level sentiment analysis obtains the sentiment of a complete document or paragraph.
- Sentence level sentiment analysis obtains the sentiment of a single sentence.
- Sub-sentence level sentiment analysis obtains the sentiment of sub-expressions within a sentence.

2. PROPOSED SYSTEM

2.1 Data Collection:

The data collection step is the initial phase in the research, where data is collected from twitter. There are two methods on how to connect and collect tweets from Twitter. The first method is by searching tweets matching to the keywords. The second method is by collecting all the tweets provided by Twitter through streaming API, or all the tweets in a specific language, or all the tweets in a specific location then put all of them into the database.
Both methods have their own advantages and disadvantages. For example, the first method requires only small storage as the data are relatively small. The downside is that researcher cannot get data from other keywords (if he needs to) from an earlier time. Twitter allows the search API only for 7 days backwards. This data collection method is suitable if the focus of the research is on the feature extraction or the prediction method. With the second method, researcher can apply many set of keywords to get the best result.

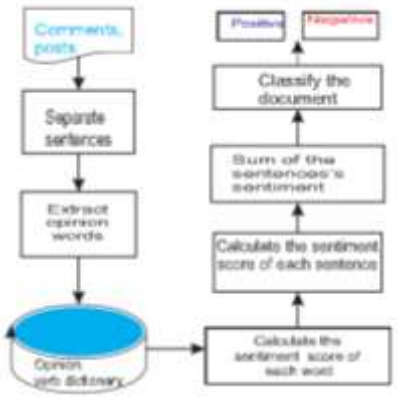

Fig1.Architecture of Sentiment Analysis

2.2 Pre-processing

Many current methods for text sentiment analysis contain various preprocessing steps of text. One of the most important goals of preprocessing is to enhance the quality of the data by removing noise. Another point is the reduction of the feature space size.

a) Lower Case Conversion:
Because of the many ways people can write the same things down, character data can be difficult to process. String matching is another important criterion of feature selection. For accurate string matching we are converting our complete text into lower case.

b) Removing Punctuations and Removing Numbers:
All punctuations, numbers are also need to remove from reviews to make data clean and neat. Unnecessary commas, question marks, other special symbols get removed in this case. Here, we are not removing dot (.) symbol from our reviews because are splitting our text into sentences.

c) Stemming:
Stemming is that the method of conflating the variant styles of a word into a standard illustration, the stem. For example, the words: "presentation", "presented", "presenting" could all be reduced to a common representation "present". This is a widely used procedure in text processing for information retrieval (IR) based on the assumption that posing a query with the term presenting implies an interest in documents containing the words presentation and presented. Stemming in our case helpful in correct words matching and counting case.

d) Striping White Spaces:
In this pre-processing step all text data is cleansed off. All unnecessary white spaces, tabs, newline character get removed from the text.

2.3 Sentiment Analysis:
a) Machine Learning Approach:

There are two approaches of machine learning, supervised and unsupervised. In our research we used supervised machine learning approach.
In supervised machine learning approach there is finite set of classes for classification. Training dataset is also available. Most research papers do not use the neutral class, which makes the classification problem considerably easier, but it is possible to use the neutral class. Given the training data, the system classifies the document by using one of the common classification algorithms such as Support Vector Machine, Naïve Bayes etc. We used naive bays for classification of tweets. We classified tweets into polarity and emotion also using naive bays classifier.
Naive Bayes is a machine learning algorithm for classification problems. It is based on Bayes' probability theorem. It is primarily used for text classification that involves high dimensional knowledge sets. A few examples are spam filtration, sentimental analysis, and classifying news articles.

b) Lexicon Based Approach:
There three main approaches to compile sentiment words. Three main approaches are: manual approach, dictionary-based approach, and corpus-based approach.in our research we used dictionary based approach. We used eleven different variables for classification, that variables are sadness, tentativeness, anxiety, work, anger, certainty, achievement, positive words, negative words, positive hashtag and negative hashtag. We collected various word related to that eleven variable and classified them.

3. METHODOLOGY
There are many methods and algorithms to implement sentiment analysis systems, which can be classified as:

- **Rule-based** systems that perform sentiment analysis based on a set of manually crafted rules.
- **Automatic** systems that rely on machine learning techniques to learn from data.
- **Hybrid** systems that combine both rule based and automatic approaches.

3.1 Rule-based Approaches

Usually, rule-based approaches define a set of rules in some kind of scripting language that identify subjectivity, polarity, or the subject of an opinion.

The rules may use a variety of inputs, such as the following:

- Classic NLP techniques like *stemming*, *tokenization*, *part of speech tagging* and *parsing*.
- Other resources, such as lexicons (i.e. lists of words and expressions).

A basic example of a rule-based implementation would be the following:

1. Define two lists of polarized words (e.g. negative words such as *bad*, *worst*, *ugly*, etc. and positive words such as *good*, *best*, *beautiful*, etc.).
2. Given a text:
   i. Count the number of positive words that appear in the text.
   ii. Count the number of negative words that appear in the text.

If the number of positive word appearances is greater than the number of negative word appearances return a positive sentiment, conversely, return a negative sentiment. Otherwise, return neutral.

This system is very naïve since it doesn't take into account how words are combined in a sequence. A more advanced processing can be made, but these systems get very complex quickly. They can be very hard to maintain as new rules may be needed to add support for new expressions and vocabulary. Besides, adding new rules may have undesired outcomes as a result of the interaction with previous rules. As a result, these systems require important investments in manually tuning and maintaining the rules.

3.2 Automatic Approaches

Automatic methods, contrary to rule-based systems, don't rely on manually crafted rules, but on machine learning techniques. The sentiment analysis task is usually modeled as a classification problem where a classifier is fed with a text and returns the corresponding category, e.g. positive, negative, or neutral (in case polarity analysis is being performed).

Said machine learning classifier can usually be implemented with the following steps and components:

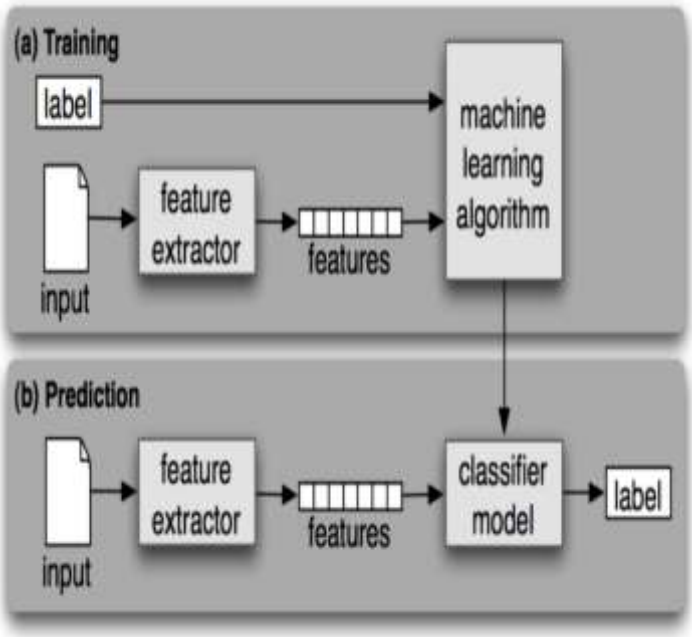

Fig2.Machine Learning Classifier

3.2.1 The Training and Prediction Processes

In the training process (a), our model learns to associate a particular input (i.e. a text) to the corresponding output (tag) based on the test samples used for training. The feature extractor transfers the text input into a feature vector. Pairs of feature vectors and tags (e.g. *positive*, *negative*, or *neutral*) are fed into the machine learning algorithm to generate a model.

In the prediction process (b), the feature extractor is used to transform unseen text inputs into feature vectors. These feature vectors are then fed into the model, which generates predicted tags (again, *positive*, *negative*, or *neutral*).

3.2.2 FEATURE EXTRACTION FROM TEXT

The first step in a machine learning text classifier is to transform the text into a numerical representation, usually a vector. Usually, each component of the vector represents the frequency of a word or expression in a predefined dictionary (e.g. a lexicon of polarized words). This process is known as feature extraction or text vectorization and the classical approach has been bag-of-words or bag-of-ngrams with their frequency.

More recently, new feature extraction techniques have been applied based on word embeddings (also known as *word vectors*). This kind of representations makes it possible for words with similar meaning to have a similar representation, which can improve the performance of classifiers.

3.3 CLASSIFICATION ALGORITHMS

The classification step usually involves a statistical model like Naïve Bayes, Logistic Regression, Support Vector Machines, or Neural Networks:

- **Naïve Bayes:** a family of probabilistic algorithms that uses Bayes's Theorem to predict the category of a text.
- **Linear Regression:** a very well-known algorithms in statistics used to predict some value (Y) given a set of features (X).
- **Support Vector Machines:** a non-probabilistic model which uses a representation of text examples as points in a multidimensional space. These examples are mapped so that the examples of the different categories (sentiments) belong to distinct regions of that space.. Then, new texts are mapped onto that same space and predicted to belong to a category based on which region they fall into.
- **Deep Learning:** a diverse set of algorithms that attempts to imitate how the human brain works by employing artificial neural networks to process data.

4. CHALLENGES

Most of the work in sentiment analysis in recent years has been around developing more accurate sentiment classifiers by dealing with some of the main challenges and limitations in the field.

4.1 Subjectivity and Tone

The detection of subjective and objective texts is just as important as analyzing their tone. In fact, so called *objective* texts do not contain explicit sentiments. Say, for example, you intend to analyze the sentiment of the following two texts:

The package is nice.

The package is red.

Most people would say that sentiment is positive for the first one and neutral for the second one, right? All *predicates* (adjectives, verbs, and some nouns) should not be treated the same with respect to how they create sentiment. In the examples above, *nice* is more *subjective* than *red*.

4.2 Context and Polarity

All utterances are uttered at some point in time, in some place, by and to some people, you get the point. All utterances are uttered in context. Analyzing sentiment without context gets pretty difficult. However, machines cannot learn about contexts if they are not mentioned explicitly. One of the problems that arise from context is changes in polarity. Look at the following responses to a survey:

Everything of it.

Absolutely nothing!

Imagine the responses above come from answers to the question *What did you like about the event?* The first response would be positive and the second one would be negative, right? Now, imagine the responses come from answers to the question *What did you DISlike about the event?* The negative in the question will make sentiment analysis change altogether.

A good deal of preprocessing or post processing will be needed if we are to take into account at least part of the context in which texts were produced. However, how to preprocess or post process data in order to capture the bits of context that will help analyze sentiment is not straightforward.

4.3 Irony and Sarcasm

Differences between literal and intended meaning (i.e. *irony*) and the more insulting or ridiculizing version of irony (i.e. *sarcasm*) usually change positive sentiment into negative whereas negative or neutral sentiment might be changed to positive. However, detecting irony or sarcasm takes a good deal of analysis of the context in which the texts are produced and, therefore, are really difficult to detect automatically.

For example, look at some possible answers to the question *Have you had a nice customer experience with us?* below.

Yeah. Sure.

Not one, but many!

What sentiment would you assign to the responses above? Probably, you have listened to the first response so many times, you would have said negative, right? The problem is there is no textual cue that will make a machine learn that negative sentiment since most often, *yeah* and *sure* belong to positive or neutral texts.

How about the second response? In this context, sentiment is positive, but we're sure you can come up with many different contexts in which the same response can express negative sentiment.

4.4 Comparisons

How to treat comparisons in sentiment analysis is another challenge worth tackling. Look at the texts below:
This product is second to none.
This is better than old tools.
This is better than nothing.
There are some comparisons like the first one above that do not need any contextual clues in order to be classified correctly.
The second and third texts are a little more difficult to classify, though. Would you classify them as *neutral* or *positive*? Probably, you are more likely to choose *positive* for the second one and *neutral* for the third, right? Once again, context can make a difference. For example, if the *old tools* the second text talks about were considered useless in context, then the second text turns out to be pretty similar to the third text. However, if no context is provided, these texts feel different.

4.5 Emojis

There are two types of emojis according to Guibon et al.. *Western emojis* (e.g. :D) are encoded in only one character or in a combination of a couple of them whereas *Eastern emojis* (e.g. ¯ \ _ (ツ) _ / ¯) are a longer combination of characters of a vertical nature. Particularly in tweets, emojis play a role in the sentiment of texts.
Sentiment analysis performed over tweets requires special attention to character-level as well as word-level. However, no matter how much attention you pay to each of them, a lot of preprocessing might be needed. For example, you might want to preprocess social media content and transform both Western and Eastern emojis into tokens and whitelist them (i.e. always take them as a feature for classification purposes) in order to help improve sentiment analysis performance.
Here's a quite comprehensive list of emojis and their unicode characters that may come in handy when preprocessing.

4.6 Defining Neutral

Defining what we mean by *neutral* is another challenge to tackle in order to perform accurate sentiment analysis. As in all classification problems, defining your categories -and, in this case, the *neutral* tag- is one of the most important parts of the problem. What *you* mean by *neutral*, *positive*, or *negative* does matter when you train sentiment analysis models. Since tagging data requires that tagging criteria be consistent, a good definition of the problem is a must.
Here's some ideas on what a *neutral* tag might contain:

- Objective texts. As we say here, so called *objective* texts do not contain explicit sentiments, so you should include those texts into the neutral category.
- Irrelevant information. If you haven't preprocessed your data to filter out irrelevant information, you can tag it neutral. However, be careful! Only do this if you know how this could affect overall performance. Sometimes, you will be adding noise to your classifier and performance could get worse.
- Texts containing wishes. Some wishes like I wish the product had more integrations are generally neutral. However, those including comparisons, like I wish the product were better are pretty difficult to categorize

5. CONCLUSIONS

In Social media, people usually follow various events such as natural disasters, political issues, sports events and posts there comments / sentiments about the particular event. By applying sentiment and opinion mining techniques, it can be deduced that whether the sentiment is in denial or in favor of the event. But the most important point is how many people in total support and broadcast this particular sentiment, and how to identify the relationship amongst the supporters of a specific sentiment. This approach will be able to predict any uncertainty which may be either harmful or beneficial for the institutions.
There are several work which has been done on sentiment and opinion mining of social media but this mining technique is currently unformed. There are various research gaps in sentiment and opinion mining which are explained in the work done by the researchers. Many researchers are still researching on sentiment and opinion mining and how to improve this particular mining techniques. This paper is based on a comprehensive study on sentiment and opinion mining, and discuss the limitations and address the new areas in the domain of sentiment and opinion mining.

6. REFERENCES

[1] Hillygus, D. S. (2011). The evolution of election polling in the United States. Public opinion quarterly, 75(5),, 962-981.
[2] Lewis Beck, M. S. (2005). Election forecasting: principles and practice. The British Journal of Politics & International Relations, 7(2), 145-164.
[3] Fumagalli, L. &. (2011). The total survey error paradigm and pre-election polls: The case of the 2006 Italian general elections. ISER Working Paper Series. 2011-29.
[4] Pak, A. &. (2010). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. LREC.
[5] Dann, S. (2010). Twitter content classification. First Monday, 15(12).
[6] Wong, F. M. (2013). Quantifying Political Leaning from Tweets and Retweets. ICWSM.

[7] Boutet, A. K. (2012). What's in your Tweets? I know who you supported in the UK 2010 general election. Proceedings of the International AAAI Conference on Weblogs and Social Media.

[8] Golbeck, J. &. (2011). Computing political preference among twitter followers. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.
[9] Pennacchiotti, M. &. (2011). Democrats, republicans and starbucks afficionados: user classification in twitter. Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining , 430-438.
[10] Tumasjan, A. S. (2010). Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. ICWSM, 10, 178-185.
[11] O'Connor, B. B. (2010). From tweets to polls: Linking text sentiment to public opinion time series. ICWSM, 11, 122-129
[12] Gayo-Avello, D. M. (2011). Limits of electoral predictions using twitter. ICWSM.
[13] Bermingham, A. &. (2011). On using Twitter to monitor political sentiment and predict election results.
[14] Ceron, A. C. (2014). Using Sentiment Analysis to Monitor Electoral Campaigns: Method Matters—Evidence From the United States and Italy. Social Science Computer Review.
[15] Ceron, A. C. (2013). Every tweet counts? How sentiment analysis of social media can improve our knowledge of citizens' political preferences with an application to Italy and France. New Media & Society, 16(2), 340-358.
[16] Sang, E. T. (2012). Predicting the 2011 dutch senate election results with twitter. the Workshop on Semantic Analysis in Social Media (pp. 53-60). Association for Computational Linguistics.
[17] Choy, M. C. (2012). US Presidential Election 2012 Prediction using Census Corrected Twitter Model. arXiv preprint arXiv:1211.0938.
[18] Gaurav, M. S. (2013). Leveraging candidate popularity on Twitter to predict election outcome.Proceedings of the 7th Workshop on Social Network Mining and Analysis. ACM., 7.
[19] Makazhanov, A. R. (2014). Predicting political preference of Twitter users. Social Network Analysis and Mining, 1-15.
[20] Cameron, M. P. (2013). Can Social Media Predict Election Results? Evidence from New Zealand. No. 13/08.
[21] Jungherr, A. J. (2012). Why the pirate party won the german election of 2009 or the trouble with predictions: A response to tumasjan, a., sprenger, to, sander, pg, & welpe,"predicting elections with twitter: What 140 characters reveal about political sentiment". Social Science Computer Review, 30(2),, 229-234.
[22] Gayo-Avello, D. (2012). No, you cannot predict elections with twitter. Internet Computing, IEEE, 16(6), 91-94.
[23] Gayo-Avello, D. (2013). A meta-analysis of state-of-the-art electoral prediction from Twitter data. Social Science Computer Review.
[24] https://monkeylearn.com/sentiment-analysis/