Sentiment Analysis on Tweets

Amer Elsheikh
The American University in Cairo
Cairo, Egypt
amer.elsheikh@aucegypt.edu
900196010

Abdallah Abdelaziz
The American University in Cairo
Cairo, Egypt
Abdallah_taha@aucegypt.edu
900196083

1. INTRODUCTION

The data in the world is constantly increasing in an unprecedent rate; however, more than 80% of the world's data is unstructured which is the data that does not fit easily in a table, and it is hard to be queried (May, 2020). Unstructured data come in a lot of forms like emails, blogs, videos, and text. One big source of this data is social media posts like Twitter, Facebook, and Reddit: the posts there include different types of information, and they are not easy to computationally deal with. Unfortunately, this huge amount of data was not used effectively until the fields of **machine learning** and **natural language processing** has been advanced to deal with such complex inputs in an effective manner. This advancement made companies think about how to use the huge amount of unstructured data they were collecting over the years. One important use that these companies implemented is doing **Sentiment Analysis, SA,** on their product reviews: now, a company has got many reviews on a specific product, and they want to know how satisfied the customers are. It is nearly impossible to go over each review manually if it is a text-based review. To solve this problem, machine learning and natural language processing are used to **classify** the reviews into either positive, neutral, or negative. Therefore, **Sentiment Analysis** is a classification problem described in Fig. 1 where the subjective statements are differentiated from objective statements in **Sentiment Identification** then right features are selected from the subjective statements which are then classified to the required classes (Medhat et al., 2014).

After classifying the reviews, the company can use this data easily to make informed decisions about that specific product or one of its aspects. That is absolutely important as it makes the company follow customers' desires and adjust their products accordingly.

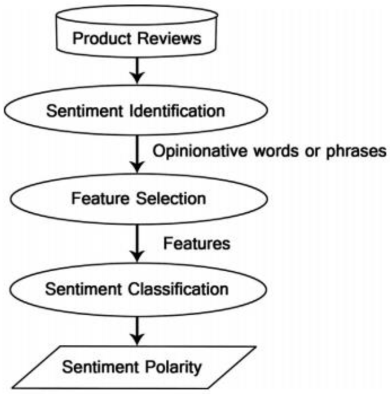

Figure 1 Sentiment analysis process on product reviews.

As said, the reviews about a specific product can come from different sources, one of which is social media. Thus, we chose Twitter, a famous social media network, to perform sentiment analysis on the users' tweets there. Twitter was specifically chosen as it is very popular that it would have all different kinds of data. Moreover, a single tweet, post, has a maximum size of 140 characters which will be helpful in data preprocessing. We aim to make a high accuracy model to classify these tweets which will in turn help any growing business understand users' feedback.

2. LITERATURE REVIEW

To design a sustained high performing solution, we researched previous solutions and models that already tackled the problem, read their models, and compared their performance. That would help us build upon their results to get an even better solution. In that process, we found many models used in sentiment classification Fig. 2. As stated by Medhat et al. (2014), the models mainly fall into three categories:

- Machine Learning Approach: it uses well known ML algorithms that work on natural language processing to get the best model. It can be divided into two main areas: supervised learning where we give the model the text (tweets) and its label (positive, negative, or neutral) and begin the training until it can make correct predictions on its own and unsupervised learning where the model classifies tweets into similar categories without any labels. In our case, supervised learning will be more effective and reasonable.

- Lexicon-based Approach: it uses a lexicon sentiment which is an already known sentiment terms to deduce whether the given text is positive or negative. To get the lexicon, it is first seeded manually then computational procedures are used to expand the seed set into a bigger set of words that we know it is either positive or negative. That can be done in either a dictionary-based approach or a Corpus-based approach.
- The hybrid Approach: it combines the above two approaches trying to use of the advantages of each.

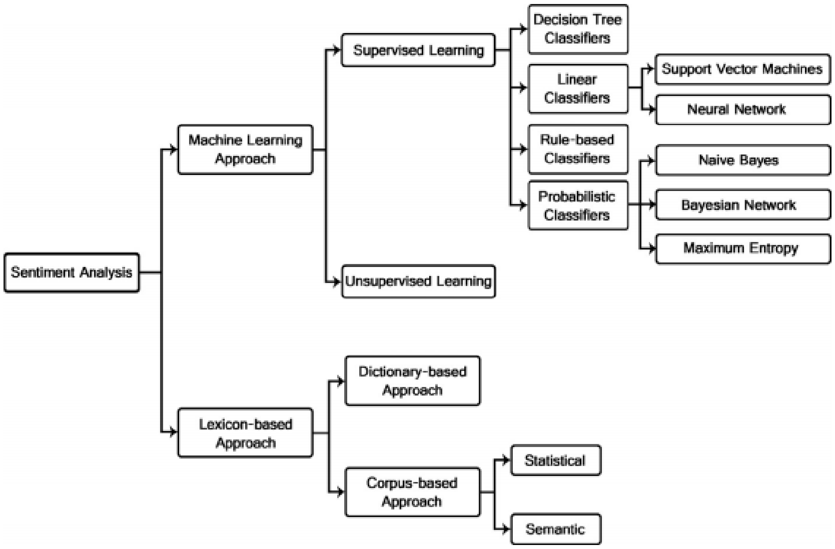

Figure 2 Sentiment classification techniques.

To continue the review, we will compare different approaches used in different papers:

2.1 Data Collection :

Collecting tweets is an easy task using Twitter API. However, labeling these tweets is rather tedious, and it is impossible to be done manually with large set of data. So, different papers use different automatic methods to label theses data as accurate as possible. Barbosa and Feng (2010) collected their data from three websites: TweetFeel, Twendz, and Twitter Sentiment. These websites provide the tweets along with their predicted sentiment; however, the data was rather noisy that it required a lot of preprocessing. On the other hand, Go et al. (2009) used a novel way to label the tweets. They only extracted tweets with emoticons, which are then used as noisy labels to determine the sentiment of the tweet. For example, :) specifies the tweet has positive sentiment, but :(specifies the tweet has negative sentiment. The data would still be noisy, but it is much better than the first case since emoticons are usually associated with tones. Also, the authors compensated for that by using more than 1.6M tweets and doing preprocessing to ensure quality of the data

2.2 Subjectivity Classification

Before extracting the features, the subjectivity of the tweets needs to be checked as it is not logical to give a sentiment for an objective statement that does not include any opinion. Thus, each paper needed a subjectivity classifier. To do so, Barbosa and Feng (2010) firstly used a subjectivity lexicon to make choose the tweets that have subjective words. They also noticed that users with the highest number of tweets usually post objective messages, i.e., they might be working in product advertisement, so they decided to remove the tweets posted by these users which eventually improved the classification. On the other hand, Go et al. (2009) used a smart test which is to search of the tweet, or most of it, appeared in a frontpage newspaper headline or as a sentence in Wikipedia, then it is objective, and it will not be used in the data. The test is of course not fully accurate, but it helps filtering the data.

2.3 Feature Selection

Before going into different models, we establish different methods used in different papers to do **Feature Selection** from the text being analyzed. Some of the features selected are:

- **Terms Frequency**: here, the features consist of single words and their associated frequency in the sentence/document.
- **Parts of Speech (POS)**: here, the words are mapped to a part of speech, i.e., noun, adjective, verbs, etc. using a pos dictionary and then relevant parts are only used. In sentiment analysis, adjectives, interjections, and negations are most important.
- **Opinion words and phrases:** here, the features are words that are usually used to express opinion i.e., awesome or awful. Some phrases might also convey opinions like "easy on the eyes."

2.4 Different Models

Go et al. (2009) tried different models to choose the best one. Out of those are:

2.4.1 Baseline Model

The approach here is use a precompiled list of positive and negative keywords, and then for a specific tweet, count occurrences of positive and negative words in the tweet then return the polarity depending on the higher count. However, this model is ineffective compared to the other ML and NLP models as it is strongly affected by the list of precompiled words

2.4.2 Naïve Bayes Model

Naïve bayes is a simple yet famous model that is used with classification of text. Go et al. (2009) tested multidimensional Naïve Bayes model which assigns class $\boldsymbol{c*}$ (positive or negative) to a tweed d such that:

$$c* = argmac_c P_{NB}(c|d)$$

$$P_{NB}(c|d) := \frac{(P(c) \sum_{i=1}^{m} P(f|c)^{n_i(d)})}{P(d)}$$

where $\boldsymbol{f}$ is a feature. We have a total of $\boldsymbol{m}$ features such that $\boldsymbol{n_i(d)}$ is the count of the feature $\boldsymbol{f_i}$ in the tweet $\boldsymbol{d}$. Moreover, $\boldsymbol{P(c)}$ and $\boldsymbol{P(f|c)}$ are calculated using maximum likelihood estimates.

Despite of its simplicity, the model achieved **81.3%** accuracy with **Unigram[1]** feature extractor. Moreover, the accuracy improved to **82.7%** when both **Unigrams and Bigrams[2]** were used as feature extractors.

2.4.3 Support Vector Machines Model

Support Vector Machines, a.k.a. SVMs, is a famous classification method where each data member is plotted in an n-dimensional space where n is the number of features we have. Now, SVM will use supervised learning to draw a hyper plane or line that classifies the two classes (positive and negative). Go et al. (2009) used $\boldsymbol{SVM^{light}}$ software along with a linear kernel to test the model. Their input consisted of two sets of vectors of size m, the number of features such that each entry in the vector indicates the presence of that specific feature with either 0 or 1, not the count of the feature as that speeds the preprocessing process.

As for the results, the SVM model achieved an accuracy of **82.9%** along with **Unigram** feature extractor; however, it has declined to **81.6%** when used with **Unigrams and Bigrams** feature extractors.

3. Solution

In this project, we build a machine learning solution to the problem of sentiment analysis. The model should predict the sentiment of a given text with reasonable accuracy. The prediction of the sentiment is done by assigning a label to the text that is either positive if the sentiment of the text is positive and negative if it is not. Model will be trained to detect the sentiment of tweets and hence will be suitable for social media monitoring applications. This solution can also be used in other

[1] Unigram feature extraction takes every single word as a feature.

[2] Bigram feature extraction takes every two words as a feature.

area such as brand monitoring and reputation management. To build this model, a dataset is needed. In the next section, candidate databases are discussed.

4. Datasets

Datasets related to the topic of sentiment analysis are quite available and relevant given how established the field is. Due to the nature of the topic, data available range from reviews and social media posts. Examples of these datasets include:

4.1 Stanford Sentiment Treebank:

The dataset mainly consists of around 240K expression extracted from 11K sentences. These sentences are scrapped from Rotten Tomatoes reviews. The expressions are the only features provided in the data set. Examples of expressions are: “good”, “a good movie” and “you can do no wrong with Jason X”. Data are given a number from 1 to 25 indicating the polarity of the expression 1 being the most negative and 25 being the most positive. This number acts as a label for each expression.

The dataset contains two files. One contains a mapping from each expression to a distinct index, and the other contains for each index 3 to 6 scores, each of them from 1 to 25 assigned by human judges. These scores can then be averaged to get a final label for each expression.

The **advantage** of the dataset is that it provides sentiment for expressions and words which gives more value to the data. To illustrate, instead of having the sentiment of each individual word or that of a whole paragraph, we have the sentiment of expressions which is more meaningful than the other two. The **disadvantage** of this set is that it is fetched from a movie review website, therefore, it might not be suitable for analyzing text from other fields that are distant from movies.

The dataset can be found and downloaded from Stanford’s Sentiment Treebank website [https://nlp.stanford.edu/sentiment/index.html].

4.2 Paper Reviews Dataset

This dataset includes reviews of academic papers from an international computing and informatics conference. The dataset has 172 papers with total of 405 review instances. Each of the 405 instances has 10 attributes.

Features include *Timespan* which corresponds to the time of the review, *Paper ID*, *Preliminary decision* which is the acceptance or rejection of the paper, *Review ID*, *Text* which is the actual review, *Remarks*, and *Language*. The remaining features include *Orientation* which represents the subjective perception of each review. This attribute can be considered the **label** for a sentiment analysis of the reviews. Finally, there is *Evaluation* which is represents the evaluation given to the paper and *Confidence* which describes the confidence of the reviewer.

It should be noted that orientation and evaluation are integers from -2 to 2 with -2 being the most negative and 2 being the most positive. Confidence is also a numeric measure that spans from 1 to 5 with 5 being the most confidence. It also should be noted that most of the instances are in Spanish.

One **advantage** of this dataset is that it has a variety of attributes which can used alongside the actual text of the reviews to get better results. However, the dataset is very small and might not be very suitable for machine learning purposes. It also has some reviews written in English while some are written in Spanish, so it will need more complicated preprocessing before applying any sentiment analysis to the text. One last **disadvantage** is that the data is field specific and is related to computer and informatics academic papers.

The dataset comes from Department of Computing & Systems Engineering, Universidad CatÃ³lica del Norte and can be downloaded from UCI Machine Learning Repository [https://archive.ics.uci.edu/ml/datasets/Paper+Reviews]

4.3 Sentiment140 Dataset

The Sentiment140 Dataset is an established dataset with more than 1.6M instances. The dataset contains tweets extracted from twitter using its API. The dataset is labeled based on the emoticons present in the tweets. Tweets with positive emoticons such as :-) are labeled positive tweets, and tweets with negative emoticons such as :-(are labeled negative tweets. The emoticons are then removed from the text of the tweet.

Features, other than the text of the tweet, include id of the tweet, the date of the tweet, the query, and the username. The tweets are **labeled** 4 if they are positive, 2 if they are neutral and 0 if they are negative.

The huge size of the dataset is a big **advantage** and makes the dataset suitable for machine learning applications. However, the automatic labeling of the data decreases the reliability of the labeling. This is compensated for by the huge size of the dataset which should allow the main pattern to still be unaffected.

The dataset was created by Alec Go, Richa Bhayani, and Lei Huang, who were Computer Science graduate students at Stanford University, and it can be accessed and downloaded from [http://help.sentiment140.com/for-students] or through Kaggle [https://www.kaggle.com/kazanova/sentiment140].

REFERENCES

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *Technical report, Stanford*.

Bo Pang, Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *In ACL*, pp 115‑124.

Keith, B., Fuentes, E., & Meneses, C. 2017. A Hybrid Approach for Sentiment Analysis Applied to Paper Reviews.

Luciano Barbosa, Junlan Feng. 2010. Robust sentiment detection on twitter from biased and noisy data. Proceedings of the 23rd international conference on computational linguistics: posters (COLING’10). *Association for Computational Linguistics, Stroudsburg*, pp 36‑44.

May. (2020, May 15). Is Twitter Structured Data or Unstructured Data? Mathamagicians. https://mathamagicians.co/is-twitter-structured-data-or-unstructured-data/

Walaa Medhat, Ahmed Hassan, and Hoda Korashy. 2014. Sentiment analysis algorithms and applications: A survey, *Ain Shams Engineering Journal*, pp 1093-1113, https://doi.org/10.1016/j.asej.2014.04.011.