Towards Better Decision-making with Twitter Sentiment Analysis

Wedjdane Nahili*, Khaled Rezeg**, Lyna Miloudi***

*LINFI Laboratory, Computer Science, Mohamed Khider University, Biskra
wd.n.zemmouri@gmail.com,
** LINFI Laboratory, Computer Science, Mohamed Khider University, Biskra
rezeg_khaled@yahoo.fr
***LINFI Laboratory, Computer Science, Mohamed Khider University, Biskra
lynamiloudi07@gmail.com

Abstract. Due to the short and simple way of expression on social media platforms such as Facebook and Twitter, millions of people share daily real-time opinions about everything in an informal way due to the use of short language (slang) and emoticons, which generates an increasing availability of unstructured and yet valuable information to data science researchers. Traditional approaches such as paper-based surveys are not the wisest path for collecting and studying consumer behavior because they are time-consuming which leads to considerable losses for companies around the world. In this paper, we develop a hybrid system to identify and classify sentiment represented in an electronic text from Twitter where users post real-time reactions about everything to improve the decision-making process for companies. To do so, we used tweepy to access Twitter's Streaming API, we combined natural language processing techniques with Naive Bayes to classify users data, we used the Python library Matplotlib to display the results. The purpose of this paper is to propose an efficient and accurate approach for predicting sentiment from raw unstructured data in order to extract opinions from the Internet and predict online customer's preferences, which could be valuable and crucial for economic and marketing researchers.

1 Introduction

During the decision-making process 'What other people think' is a precious piece of information for the majority of us. The rise of microblogging and social networking websites fueled the interest in sentiment analysis. Since an increasing number of people are willing to post their opinions about different topics, products, companies, and anything else that is part of their daily life on Twitter, which is now considered a valuable online source for opinions. Sentiment analysis on Twitter is a rapid and effective way for analyzing public opinion for business marketing or social studies. For example, a business can retrieve timely feedback on a new product in the market by evaluating people's opinions on Twitter. As people often talk about various entities (e.g., products, organizations, people, etc.) in a tweet, we perform sentiment analysis at the entity level; that is, we mine people's opinions on specific entities in each tweet rather than the opinion about each whole sentence or whole tweet. We assume that the entities are provided by the user, e.g., he/she is interested in opinions on iPhone (an entity). The task of sentiment analysis, also known as opinion mining, is detecting the polarity of these opinions. The term opinion in itself has many definitions in the literature. However, the focus of sentiment analysis is mainly the opinions, which express or imply positive or negative statements. In various analytical domains, sentiment analysis is significantly becoming a popular tool, especially on the Web and social media. One approach to performing a context sentiment analysis is based on a set of frequently used words to express positive, negative and neutral sentiment, such as "good" and "bad". The approach generally uses a dictionary of opinion words; each word is associated with a score determining how positive or negative a word is in order to identify and determine sentiment orientation (positive, negative or neutral). The approach was introduced by Ding (2008); Taboada (2011), it consists on using opinion words to determine opinion orientations and is called the lexicon-based approach to sentiment analysis. The advantage of this approach is its efficiency and speed, in addition, it can be used for text analysis at the document, sentence or entity level. Thus it is applicable to our task as well. Although, Twitter data has expanded its own characteristics with the use of emoticons, colloquial expressions, abbreviations, slang, etc. in tweets. Unfortunately, they are unfavorable to the lexicon-based approach because they do not exist in a general opinion lexicon although they may possess semantic/sentiment orientation. Take the following tweet example, "I bought an iPhone today, and I just looovvee it! :)". Since "looovvee" is not a general opinion word to the lexicon-based method, as a result, the tweet is expressing no/neutral opinion on iPhone. Due to the fact that this method entirely depends on the presence of opinion words to determine the sentiment orientation, it leads to the lack of accuracy. Maybe the obvious solution is dictionary enrichment with additional expressions, but the issue is the constant change and the appearance of new ones all the time following the trends and fashions on the Internet. But the main problem is, the opinion words polarities are domain dependent and the sentiment analysis results may suffer without an adaptive lexicon. Otherwise, we can apply a machine learning based method to perform sentiment analysis Pang (2002). Where a sentiment classifier is trained to determine sentiment orientation whether is it positive, negative or neutral. Machine learning methods have been frequently used for sentiment classification of documents or sentences. However, in our case, its application is not easy because manually labeling a large set of tweet examples is labor-intensive and time-consuming. Moreover, manual labeling needs to be done for each application domain, as it is well known that a sentiment classifier may perform very well in the domain that it is trained, but performs poorly when it is applied to a different domain Aue and Gamon (2005). Thus, the learning-based method suffers the lack of scalability when applied to Twitter sentiment analysis because it holds opinions about almost all domains as people can express opinions about everything on Twitter. The rest of the paper is organized as follows: Section 2 discusses some of the related work, section 3 presents a background of sentiment analysis, section 4 summarizes our proposed approach and the results obtained, and finally, in section 5 we conclude our paper with a conclusion and references at the end.

2 Related work

The proposed approach is in the field of sentiment analysis. To determine whether a piece of text expresses a positive or negative sentiment, two main approaches are commonly used: the lexicon-based approach and the machine learning-based approach. The lexicon-based approach M and B (2004); SooMin and Eduard (2004) determines the sentiment polarity using a dictionary to capture the set opinion words in the document or the sentence. The machine learning-based approach typically trains sentiment classifiers using features such as unigrams or bigrams Pang (2002); Khanaferov (2014); Zhang (2011); Debjyoti (2017). Most techniques use some form of supervised learning by applying different learning techniques such as Naive Bayes, Maximum Entropy and Support Vector Machines. Although machine learning methods proved high accuracy in previous work, unfortunately, manual labeling for training examples are needed for every application domain. There are also some approaches that utilize both the opinion words/lexicon and the learning approach. For example, Riloff (2005) used a subjectivity lexicon to identify training data for supervised learning for subjectivity classification. Our work does not do subjectivity classification. In the work of Tan and Songo (2008) applied sentiment analysis to classify reviews into two classes, positive and negative, but no neutral class, which is an easier approach that does not reflect reality. The previously mentioned approaches are different from ours: First, we perform sentiment analysis at the entity level, thus the sentiment polarities assigned at a much accurate, precise level. Second, our approach for polarity assignment is also different since we deal with three classes of sentiment (positive, negative and neutral). For us, this step is crucial because it reflects real life scenarios. While most sentiment analysis methods were proposed for large opinionated documents (e.g. reviews, blogs), some recent work has addressed microblogs. Third, many research Zhang (2015, 2011); Debjyoti (2017); Khanaferov (2014); Ostrowski (2013) has been done on sentiment analysis and opinion mining from social media using either dictionary-based methods or machine learning algorithms, most of which focus on people?s sentiment towards various topics. The issue in analyzing social media unstructured data in this manner lacks accuracy and gives a generalized idea. In order to make it more specific, we propose to combine lexicon-based methods and machine learning algorithms to perform a location-based sentiment analysis on tweets since they are a reliable source of information, mainly because people tweet about anything and everything, either it is about buying new products or reviewing them.

Year	App	Domain/Authors
2011	L	Movie reviews. Taboada (2011)
2013	ML	Topic trending. Ostrowski (2013)
2014	ML	Healthcare. Khanaferov (2014)
2015	ML	Text document. Zhang (2011)
	L	Movie, hotel and product reviews. Vilares (2015)
2016	L	Movie reviews. Cambria (2016)
2017	ML	US Elections 2016. Debjyoti (2017)

TAB. 1 – *Previous related works.*

decision-making with sentiment analysis

3 Sentiment Analysis

Sentiment analysis is an active research area in natural language processing that analyzes peoples opinions, sentiments, emotions toward entities (products, services, organizations, individuals, events, issues, or topics) expressed in written text. Sentiment analysis goal is the identification, extraction, and organization of sentiments from user-generated texts in social networking platforms, blogs or product reviews site. With the rapid growth of social media such as Twitter, Facebook, and online review sites such as IMDB, Amazon, Yelp, sentiment analysis draws growing attention from both research and industry communities. Subjectivity classification is a process, to separate subjective from objective sentences, or distinguish opinions from facts. While, Sentiment classification is a process, to determine sentiment orientation whether that sentence expressed positive or negative feeling. Also, some researchers are interested in determining the intensity (strength) of sentiment polarity to measure the semantic intensity. Feature-based sentiment analysis is an in-depth study that refers to, the determining of the expressed sentiments on different features of entities. For example, the feature based sentiment analysis of smartphone screen is the study of people expression on screen whether it is positive or negative. Sentiment analysis tasks can be done at several levels, word level, phrase or sentence level, document level and even feature level. Twitter is one of the most used social networking platforms to share short pieces of information limited to 280 characters known as 'tweets'. In this case, the word level granularity suits its setting. To automate the process of sentiment analysis, different approaches have been applied to predict the sentiment of words, expressions or documents. These, include lexicon-based (dictionary) and machine earning (ML) techniques. In our attempt to mine the sentiment from Twitter data, we propose an approach that combines the advantages of available supervised techniques, along with unsupervised techniques. The following section illustrates our proposed approach.

4 Proposed approach

In this paper, we propose a real time localized entity-level Twitter sentiment analysis approach using Python, where we analyze a dataset (tweets) expressing reactions and opinions. Our analysis is done as follows: we establish connexion with Twitter's streaming API using tweepy, and take advantage of the geo-localization feature on Twitter to perform localized sentiment analysis. The extracted tweets contain a lot a noise so preliminary processing is needed beforehand then the tweets are stored. In our approach, we apply a lexicon-based method. Although this method gives good precision, to improve accuracy we do the following: We first extract tweets that contain opinionated terms (words and tokens) in the dictionary. Secondly, we use Naive Bayes classifier to assign tweets polarities. Finally, we take insight of the results using Matplolib to derive high-quality information that was not that obvious beforehand and visually display the results. Our proposed approach is illustrated in figure 1.

4.1 Dictionary creation

This is the most crucial phase in our sentiment analysis model. It is divided into three steps: the first step is manually creating a dictionary of frequently used words on social media to describe and review products. The second step, we labeling each term with a score depending

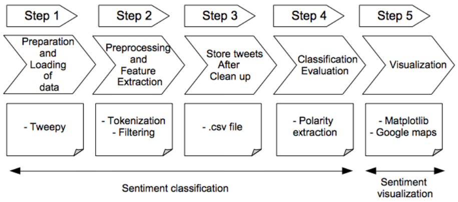
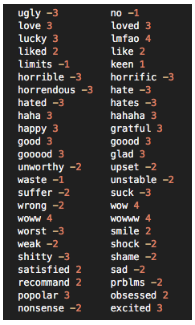


FIG. 1 – *Proposed approach.*

on how positive or negative it can be and finally we trained a Naive Bayes classifier on our pre-labeled dataset because it is easy to build and particularly functional for large data sets, and it is known to outperform even highly sophisticated classification methods Ray (2017).



```
ugly -3          no -1
love 3           loved 3
lucky 3          lmfao 4
liked 2          like 2
limits -1        keen 1
horrible -3      horrific -3
horrendous -3    hate -3
hated -3         hates -3
haha 3           hahaha 3
happy 3          gratful 3
good 3           goood 3
gooood 3         glad 3
unworthy -2      upset -2
waste -1         unstable -2
suffer -2        suck -3
wrong -2         wow 4
woww 4           wowww 4
worst -3         smile 2
weak -2          shock -2
shitty -3        shame -2
satisfied 2      sad -2
recommand 2      prblms -2
popolar 3        obsessed 2
nonsense -2      excited 3
```

FIG. 2 – *A sample from the dictionary.*

4.2 Data extraction

When performing Twitter sentiment analysis data is collected from Twitter, using the public Twitter API which allows developers to extract tweets from Twitter programmatically. For our approach, we prioritize a list of different query keywords relevant to the searched product to query the Twitter API against. A search query is based on a random keyword such as "iPhone", "samsung", "lenovo", "porsche"... etc.

4.3 Preliminary processing and storing tweets

It is often necessary to normalize the text for any NLP task. Since the tweets are often represented in a cryptic and informal way, systematic preprocessing of tweets is required to enhance the accuracy of the sentiment analyzer. The tweets are pre-processed to extract all valid terms that have immense significance to determine the polarity. At this level, we analyze the extracted tweets relevant to the prefixed keyword inserted in the search query. This analysis is done according to two following phases:

decision-making with sentiment analysis

- **Tokenization:** is the process of converting the sequence of 280 characters composing the tweets, into a sequence of words (tokens), in our case only the tweets containing matching terms with the predefined set of opinion words are kept for further processing. In simple terms, tokenization means dividing a given text into smaller and meaningful elements like sentences and words. In our approach this step is done using the Python implemetation of The Treebank tokenizer that used regular expressions to tokenize text assuming that the text has already been segmented into sentences. For example, let us assume we have the following review: The battery of the iPhone is awful and the display as well. After tokenization, the sentence would take the following form: "The", "battery", "of","the","iPhone", "is", "awful", "and", "the", "display", "as", "well"
- **Filtering:** the dataset obtained obviously contains a lot of non relevant data (noise). Therefore, very basic and rudimentary cleanup needs to be performed. Arbitrary characters and other useless information such as punctuation, emoticons substitution, stopwords, special characters and text normalization are applied using regular expressions, finally links/URL, hashtags and words that start with '@' character were removed since we found no significance in our scoring approach. When these two steps are completed, the processed tweets are then stored in a comma-separated values (.csv) file for further processing.

4.4 Polarity Calculation

In order to perform any sentiment analysis approach, a list of positive and negative words or phrases is required, in our case such a list of words is referred to as a manually defined dictionary which is an important lexical resource for sentiment analysis. It is difficult to generate a single dictionary for all domains because of the domain specificity of words, certain words convey different sentiments in different domains. For our approach, we focused on the set of opinion words commonly used to express sentiment towards an entity on social media. In case of a presence of a nonrelevant term to the predefined dictionary, the statement (tweet) is rejected. To do so, for each subjective tweet we determine its polarity and assign a score, after that we calculate the sum of polarities to find the average polarity later on. The distinction between positive, negative and neutral statements is calculated as follows:

- if sentiment.polarity = 0 then neutral
- if sentiment.polarity is between 0 and 0.3 then weak positive
- if sentiment.polarity is between 0.3 and 0.6 then positive
- if sentiment.polarity is between 0.6 and 1 then strong positive
- if sentiment.polarity is between -0.3 and 0 then weak negative
- if sentiment.polarity is between -0.6 and -0.3 then negative
- if sentiment.polarity is between -1 and -0.6 then strong negative

The average reaction of users is calculated as follows:

$$AVGpolarity = \frac{Polarity}{Numberoftweets} \tag{1}$$

4.5 Results and data visualization

The methodology was applied to latest 800 tweets extracted from Twitter and several products were considered. In order to make more sense out of the results, we used the Matplotlib Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter notebook, web application servers, and for graphical user interface toolkits matplotlib (2017). Sentiment categories for iPhone 8 are shown in figure 3. As we mentioned in previous sections, a search query could contain several parameters. A large set of filtering parameters is provided by Twitter, so that, a previously defined set of tweets can be obtained. For more accurate results, we take advantage of the location tag available in tweets. The purpose behind this, is the segmentation of data according to the location feature, where we will be able to visualize the different densities of sentiment and opinion (positive and negative) in several areas by using geo-visualization tools like Google Maps to display the results, and density variations of opinions concerning a specific product, will help make better decisions in various domains such as Marketing, Elections, Business management...etc.

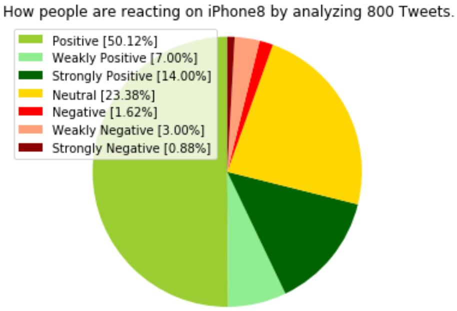


FIG. 3 – *Pie Chart representing the results of 'iPhone 8' as a search keyword.*

5 Conclusion

Sentiment analysis has provided more importance to the mass of opinions that leads to better product development and good business management. Companies can take advantage of this understanding and therefore, they will position themselves as they want it to be, and improve their decision-making process. In this paper, we introduced a sentiment analysis methodology for automatically extracting and analyzing opinion from tweets. To do so, we have considered mobile phone brands specifically (iPhone 8) and the latest 800 tweets were extracted. Due to the fact that in our methodology location was only used for user segmentation, further perspectives would be working on a combination of language (idiomatic expressions, slang, cultural differences) and location to actually improve the result's accuracy of sentiment analysis itself. Another perspective is using Vader which is the newest and higher performing lexicon in sentiment analysis for better NLP filtering techniques for sarcasm detection, seeing that it is frequently used in natural language, in addition to expanding the terms domain in order to reflect real-life complex scenarios.

References

Aue, A. and M. Gamon (2005). Customizing sentiment classifiers to new domains: a case study. In *Proceedings of Recent Advances in Natural Language Processing RANLP 2005*, Volume 292.

Bollen, J. (2011). Twitter mood predicts the stock market. *Jounal of Computational Science 2*(1), 01–08.

Cambria, E. (2016). Affective computing and sentiment analysis. In *IEEE Intelligent Systems 31*, Volume 2, pp. 102–107.

Debjyoti, P. (2017). Compass: Spatio temporal sentiment analysis of us election. In *Knowledge Discovery from Data (KDD 17).*

Ding, X. (2008). A holistic lexicon-based approach to opinion mining. In *ACM International Conference on Web Search and Data Mining, WSDM 2008.*

G.Vinodhini and RM.Chandrasekaran (2012). Sentiment analysis and opinion mining: A survey. *International Journal of Advanced Research in Computer Science and Software Engineering 2*(6), 282–292.

Khanaferov, D. (2014). Social network data mining using natural language processing and density based clustering. In *In IEEE International Conference on Semantic Computing (ICSC)*, pp. 151–250.

Kharde, V. A. (2016). Sentiment analysis of twitter data: A survey of techniques. *(2016) International Journal of Computer Applications 139*(11), 0975–8887.

Liu and Bing (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.

M, H. and L. B (2004). Mining and summarizing customer reviews. In *In Knowledge Discovery from Data (KDD) 2004*, pp. 168–177.

matplotlib (2017). https://matplotlib.org/. In *https://matplotlib.org/.*

Ostrowski, D. (2013). Semantic social network analysis for trend identification. In *In IEEE Sixth International Conference on Semantic Computing*, pp. 215–222.

Pang, B. (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Philadelphia, Association for Computational Linguistics*, pp. 79–86.

Poria, S. (2016). A deeper look into sarcastic tweets using deep convolutional neural networks. In *In COLING, ACL*, pp. 1601–1612.

Ray, S. (2017). Essentials of machine learning algorithms(with python and r codes). In *https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms.*

Riloff, E. (2005). Exploiting subjectivity classification to improve information extraction. In *In Proceedings of the 20th National Conference on Artificial Intelligence (AAAI 05).*

Shaalan, S. S. M. A.-E. K. (2017). A survey of text mining in social media: Facebook and twitter perspectives. *Advances in Science, Technology and Engineering Systems Journal 2*(1), 127–133.

SooMin, K. and H. Eduard (2004). Determining the sentiment of opinions. In *Proceedings of the 20th International Conference on Computational Linguistics, COLING 04*.

Taboada, M. (2011). Lexicon-based methods for sentiment analysis. *Journal of Computational Linguists 2011 37*(2), 267–307.

Tan and Songo (2008). Combining learn-based and lexicon-based techniques for sentiment detection without using labeled examples. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Volume SIGIR 08, pp. 743–744.

Vilares, D. (2015). A syntactic approach for opinion mining on spanish reviews. *Natural Language Engineering 21 1*, 139–163.

Wang, B. and L. Min (2015). Deep learning for aspect based sentiment analysis. In *https://cs224d.stanford.edu/reports/WangBo.pdf.*

Zhang (2011). Combining lexicon-based and learning-based methods for twitter sentiment analysis. In *In HP Laboratories technical reports 89.*

Zhang, L. (2015). Twitter sentiment analysis with deep convolutional neural networks. In *In proceedings Severyn 2015 Twitter SA, SIGIR*.

Résumé

En raison de la manière courte et simple d'expression sur les plateformes des réseaux sociaux tels que Facebook et Twitter, des millions de personnes partagent des opinions quotidiennes en temps réel sur tout de manière informelle en raison de l'utilisation du langage court (argot) et émoticônes, ce qui génère une disponibilité croissante d'informations non structurées et pourtant précieuses pour les chercheurs en science des données. Les approches traditionnelles telles que les sondages ne sont pas le meilleur moyen pour recueillir et étudier le comportement des consommateurs, car elles prennent beaucoup de temps, ce qui entraîne des pertes considérables pour les entreprises. Dans cet article, nous développons un système hybride pour identifier et classer les sentiments représentés dans un texte électronique provenant de Twitter pour améliorer le processus de prise de décision pour les entreprises. Pour ce faire, nous avons utilisé tweepy pour accéder l'API de Twitter, nous avons combiné les techniques de traitement du langage naturel avec les réseaux bayésiens pour classer les données des utilisateurs, nous avons utilisé la bibliothèque Python Matplotlib pour illustrer les résultats. Le but de cet article est de proposer une approche efficace et précise pour prédire le sentiment à partir de données brutes non structurées afin d'extraire des opinions et de prédire les préférences des clients en ligne, ce qui pourrait être précieux pour les chercheurs en économie et marketing.