

An approach to analyze suicidal tendency in blogs and tweets using Sentiment Analysis

Shrija Madhu

Dept. of CSE, GIET, Rajahmundry(A.P),India

Available online at: www.isroset.org

Accepted: 20/Aug/2018, Online: 31/Aug/2018

Abstract— Sentiment Analysis(SA) or Opinion Mining is done to find the opinion of the users and customers to review and analyse their opinions on various products and services. It is one of the major tasks in business these days for knowing demands of customers. In this paper a sentiment analysis method for analyzing the suicidal tendencies in blogs and tweets is proposed. The proposed method uses concepts like Bag of words ,Part of Speech and Natural Language Processing for analyzing the text. In life sometimes such a situation arises where the person finds himself trapped and suicide only seems to be the ultimate respite from all problems. Such a person may not share his mental condition verbally with anyone but may share it through tweets and messages. This paper is an approach to reach out a helping hand to such people by analyzing such tendencies in their messages. The proposed method uses Python Language module Textblob for performing different analysis task on the text.

Keywords— Sentiment Analysis; Suicide; Textblob; blogs; tweets.

I. INTRODUCTION

With advancement in technology and development of social media, information dispersion and content transmission online takes place in no time. Opinions on a new product can be gathered by stakeholders in fraction of seconds. There is a large growth of user-generated content in social networks, such as Twitter, Facebook and Whatsapp . Businesses are using this content for getting review for their products and services. This massive information is analyzed using varied technological tools from Artificial Intelligence (AI), Machine Learning, Natural Language Processing (NLP) and many others .

Internet has become a platform for sharing opinions and exchanging ideas. Social networking sites and apps like Whatsapp, Twitter, Facebook, Google+ have become very popular and fast mode of communicating views on topics and products, sharing ideas, pictures and messages.

Opinion Mining is commonly used for finding opinion of customers to know their view on products and services provided by different companies. It helps businesses in improving their products and services by conducting online reviews.

Some of the earlier works in the area of Sentiment Analysis used the Bag Of Words (BOW) model. In this model a document is mapped to a feature vector and which is then

classified by machine learning techniques. This method is simple but leads to a lot of loss of information(Xia & Zong, 2010). Another feature that is popularly used for syntax analysis in Sentiment Analysis is the Part Of Speech (POS) tagging(Gimpel et al. (2011). There are some approaches where the concept of n-grams is used for reviewing the opinions (Pak & Paroubek, 2010).

Lexicon-based techniques are also used which has a number of advantages (Taboada, Brooke, Tofiloski, Voll, & Stede, 2011).The dominant approaches in sentiment analysis are based on machine learning techniques (Pang, Lee, & Vaithyanathan, 2002)

Traditional approaches are time consuming as they rely on manual feature extraction. Some approaches using deep learning has shown excellent performance in NLP tasks which is a promising alternative to traditional methods.(Collobert et al., 2011). A study of different techniques for analyzing twitter data was also done which highlights all possible methods of Sentiment Analysis. (Kharde & Sonawane,2016).

In this paper, we propose sentiment analysis approach for analyzing the suicidal tendencies in blogs and tweet messages. Suicidal tendency is a condition where a person is unable to find out a way and finds him to be trapped in a particular situation. Suicide only seems to be the ultimate respite from all problems. Such a person may not share his

mental condition verbally with anyone but may share it through tweets and messages.

The rest of the paper is organized as follows. Section II describes the proposed Sentiment Analysis methods and the functions of Python that can be used for analysis. Experimental results are presented along with the methods. Finally, Section III draws conclusion and future scope of the work.

II. SENTIMENT ANALYSIS

Sentiment analysis uses Natural Language Processing (NLP) for analyzing attitudes, opinions, views and emotions from text, speech, tweets and database sources in the publicly available social networks. In Sentiment analysis the opinions in text are classified as positive, negative or neutral.

Sentiment Analysis includes many tasks such as sentiment extraction, classification, and summarization. It aims at analyzing sentiments, views, attitudes, opinions, emotions of people towards products, services, topics, organizations and also individuals.

Bag of Words (BOW) Analysis

In this a set of words pertaining to the area concerned is selected. The text to be checked is verified against the selected words to identify the extent to which the text matches the words and the area on the whole.

Table 1 shows the classification of sentiments based on bag of words concept. All possible words that can be used by people who have a suicidal tendency are selected and classified as positive, negative and neutral.

Table 1. Classification of words and phrases

	POSITIVE	NEGATIVE	NEUTRAL
I			√
WE			√
YOU			√
MYSELF			√
WANT	√		
KILL		√	
DIE		√	
WANT TO			√

WANT TO KILL		√	
WANT TO DIE		√	
DON'T WANT TO		√	
DON'T WANT		√	
DON'T WANT TO LIVE		√	
COMMIT	√		
DON'T COMMIT		√	
SUICIDE		√	
COMMIT SUICIDE		√	

N-GRAMS ANALYSIS

Sentiment Analysis of text can be done at different levels like word level, sentence level and document level. In Python the Textblob.ngrams() method returns a list of tuples of n successive words. This can be used for getting Unigrams(single words), Bigrams(two words) or n-grams (multiple successive words).

Unigram: Textblob.ngrams(n=1)

Bigram: Textblob.ngrams(n=2)

N-grams: Textblob.ngrams(n=3or more)

```
b = TextBlob("I don't want to live any more.")
b.ngrams(n=3)
[WordList(['I', 'don't', 'want']), WordList(['don't', 'want', 'to']),
 WordList(['want', 'to', 'live'])]
```

TOKENIZATION ANALYSIS

Text can be broken into words and sentences. In Textblob, there are functions like words and sentences which can be used for splitting words in a sentence or splitting sentences in a paragraph.

```
t = TextBlob("I don't want to live anymore.I want to die")

t.words
WordList(['I','don't', 'want','to','live','anywhere','I','want','to','die'])

t.sentences
[Sentence("I don't want to live anymore."),Sentence("I want to die ")]
```

WORD FREQUENCY ANALYSIS

There are two ways to get the frequency of a word or noun phrase using Textblob. The first is through the word_counts dictionary and the other is by use of words.count function.

```
t = TextBlob("I don't want to live anymore. I want to die")

t.word_counts['want']
2

t.words.count('want')
2
```

POLARITY AND SUBJECTIVITY ANALYSIS

In Python the sentiment property of Textblob returns a named tuple of the form Sentiment(polarity, subjectivity). The polarity score is a float in the range -1.0 and 1.0 while the subjectivity is a float in the range 0.0 and 1.0 where 0.0 is very objective and 1.0 is very subjective.

```
t = TextBlob("I don't want to live anymore.I want to die.
I want to commit suicide.")

t.sentiment
Sentiment(Polarity = 0.136364, Subjectivity = 0.500000)

t.sentiment.polarity
```

III. CONCLUSION

In this paper a sensitive topic is taken into consideration for analysis that is "Suicide". Few ways of identifying the crucial words in the blogs and tweets is suggested. This will help in saving life of a person. A person with suicidal tendency needs help to recover from that situation. The help should be provided at the right time or else it will be of no use. Python language provides a Textblob module for Sentiment Analysis which is used for extracting and analyzing data from messages. Different ways of checking and analyzing the text is provided.

REFERENCES:

- [1]. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. "Natural language processing (almost) from scratch". The Journal of Machine Learning Research, 12, 2493–2537, 2011.
- [2]. Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., et al. "Part-of-speech tagging for twitter: Annotation, features, and experiments". In Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies: Short papers - volume 2 HLT '11, pp. 42–47, 2011.
- [3]. Pak, A., & Paroubek, P. "Twitter as a corpus for sentiment analysis and opinion mining". In LREc: vol. 10, pp. 1320–1326, 2010.
- [4]. Pang, B., Lee, L., & Vaithyanathan, S. "Thumbs up?: sentiment classification using machine learning techniques". In Proceedings of the ACL-02 conference on empirical methods in natural language processing-Volume 10, pp. 79–86, 2002.
- [5]. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. "Lexicon-based methods for sentiment analysis". Computational linguistics, 37, 267–307, 2011.
- [6]. Xia, R., & Zong, C. "Exploring the use of word relation features for sentiment classification". In Proceedings of the 23rd international conference on computational linguistics: Posters COLING '10, pp. 1336–1344, 2010.
- [7]. Kharde, A.V., & Sonawane, S.S. "Sentiment Analysis of Twitter Data: A Survey of Techniques". International Journal of Computer Applications, Volume 139, No.11, 2016.

Author Profiles

Ms. Shrija Madhu did her MCA from MP Bhoj University Bhopal and M.Tech in CSE from Jawaharlal Nehru Technological University Kakinada. She is having Seventeen years of teaching experience. At present she is working as an Associate Professor in the Department of CSE at GIET, Rajahmundry. She is pursuing Ph.D from Bharathiar University, Coimbatore in the area of Information Security. She has published many research papers in various International, National Conferences and Journals. Her research interest includes Image Encryption, Image Processing, Cloud Computing and Bioinformatics.

