SENTIMENT ANALYSIS FOR SOCIAL MARKETING

Pankaz Debnath, Mir Ummay Touhida, Md. Zakib Uddin Khan, Shahinuzzaman Shawon
Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh.
ponkaj3439@diu.edu.bd, touhida15-1910@diu.edu.bd, zakib287@diu.edu.bd, shawon3573@diu.edu.bd

Abstract: From the beginning of human race they can communicate with each other. In the very first time there was not any language. Human did it by signs and signals. By the revolution of era, humans got smart language to talk, to write and to read. Then they started using languages for gathering knowledge. In that time it was easy because there was a little amount of data to learn. But in course of time the amount of data rised to a huge number. Learning all of the data was quite impossible. As a result generating sentiment value from all these data become so challenging that it could take the lifetime to get the result. Then human started to think how they can increase gathering more knowledge by reducing the time than before. From that thinking AI came and use of machine has been increased and NLP based research started taking place. This work is not different and it aims to make a model for deriving the sentiment from any given English language text.

Keywords: Sentiment analysis, polarity, classification, positive sentiment, negative sentiment.

1. Introduction

In this era of modern technology, the necessity of machine is increasing. In this age which technology we use that is not a fact but, by that technology what we can do that is the main fact. Technologies are improving day by day. New technologies are added by the people, for making their life more comfortable. Not a single work of this day can be done without technology. All type of work has an easy way to do, that way had made by the technology. Every field of research and invention is an applied version of the technology. For humans satisfaction technology is improving in every day. All type of work got touch with technology. Technological parts are full of the application of AI. Automation, Text Mining, Robotics, Smart Service Systems are fully depends on Artificial Intelligence. This work is kind of applied application of Artificial Intelligence. It Supports ‘Text Mining’ directly. It is about analyzing the sentiment of a human from a text. By it one can understand the depth of context of a comment or a review or an article. No library was used for calculation or extraction.

It is all about the thinking of a human about a specific thing or subject. When a person write a review or a blog about a product or a scenario then he/she express his/her opinion about that thing. A man can’t understand the context of the writing without reading the text. Only after reading the text one can decide wether the writer take the thing as positively or negatively. But it is very time consuming and if the amount of data is too much then it is impossible to get the accurate context or the value of negativity or positivity. As a result, the service provider or producer didn’t get the actual reaction of the customer and writer. In order to get rid of this problem this work may help. It can extract all the positive and negative words and that can extract the special characters from the input text. And, from the extractions it can calculate the values by mathematical operators. By mathematical operation it will give us a value within the range of (-1 to 1). The values are categorized in several types to define the context by a single comment. One can decide the polarity or demand of his/her service or product. From the result he/she can understand which persons are going to take his/her service or buy his/her product. Then they can target the people with positive thought to sell the services. And by using an API one can run advertisement of a product in social sites and all types of AdWords.

2. Literature Review

Sentiment analysis is avery popular research domain in the field of Natural Language processing where Natural Language Processing (NLP) refers to the application of computerized technics to analysis and synthesis of natural language (text and speech). The procedure of NLP depends on most common patterns and cases. Taking the pattern as sample it do the analysis on both text and speech. By separating patterns from the data it can learn new patterns and signs to react against the data [1].

Pang et al. have done binary sentiment analysis based on term frequency of the positive and negative words and claimed that presence of sentiment bearing word changes the polarity of a sentence either positive or to negative which mostly depends on the polarity of the word. They have also claimed that most rare words contain more information then the frequently occurred words[2].

Researchers in [2,3] have claimed that postion of the words in a document signifyies the sentiment of of that documently notably. For example, a positive word in the title can help to polarize the sentiment of the document positively. On the other hand, a negative word in the conclusion often classify that document negatively.

N-gram techniques are proposed for sentiment analysis by [2,5] whereas [2] found that unigrams performs better than bigrams in classifying movie reviews and [5] proved that tweaking in some setting ensures that bigram and trigram better than unigram which finally raises a debate which N-gram techniques better in sentiment analysis.

Adjectives play essential role in deriving sentiment from text and as a result part of speech analysis helps in dealing word sense disambiguation [2,6]. Semantic orientation method is used by [7] to identify point wise mutual information which also helps in defining positive and negative sentiment.

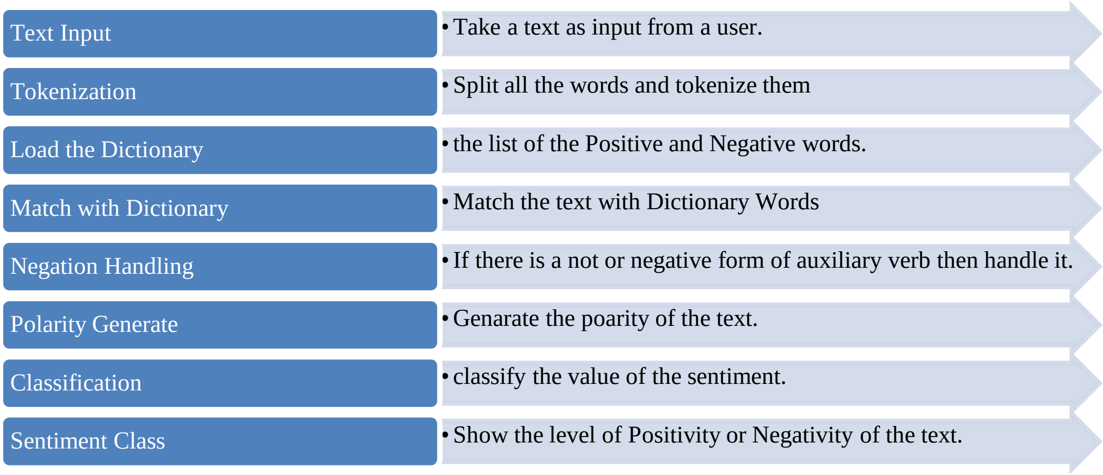
Machine approaches are very common in recent time sor sentiment analysis from huge amount of data. A few notable works are [8,9] where Latent Aspect Ratiing Analysis Model was used for sentiment analysis. In [10,11] a modified version of that work is used and in [12] authors have used topic modeling.

In recent times deep learning based approaches have become popular in sentiment analysis. In [13] authors have analyzed text in order to identify sentiment and topic and in [14] a corpuse based dictionary is proposed to analyze sentiment whereas in [15] a cognition based attention model is used for sentiment analysis.

It is observed from the above literature review that most of the works are based on machine learning models and also depends on huge amount of data. Our focus is to propose a method which will be able to classify data from any given text.

3. Research Methodology

The main procedure of the system is done by 8 separate steps. The details of the steps are given below and an overview of the steps are shown in Figure 1.



Text Input	• Take a text as input from a user.
Tokenization	• Split all the words and tokenize them
Load the Dictionary	• the list of the Positive and Negative words.
Match with Dictionary	• Match the text with Dictionary Words
Negation Handling	• If there is a not or negative form of auxiliary verb then handle it.
Polarity Generate	• Genarate the poarity of the text.
Classification	• classify the value of the sentiment.
Sentiment Class	• Show the level of Positivity or Negativity of the text.

Figure 1: Steps followed in the research (sequence from top to bottom)

This the sequence by which the program do its work to find the sentiment of the text. The procedural headings represent the word and, the side text represent what the process do.

Firstly, the system takes an input from user which is text type. Then the system read the text and tokenized the words from the text extracting and trimming the unexpected characters. Then the system start loading the dictionary of negative and positive words. After loading the dictionary the system start matching the words from the text with the words of the dictionary. Then the system will get some reading of the data and store it in variables. After that, it will check for negations. If it find any of them then is do some calculation with the storing data. Then, it will do the main calculation for generating the polarity of the text. From the polarity the system generate the actual sentiment value. Then, the system classify the value of the polarity and construct the sentiment class. The sentiment class will be the actual output of the system.

3.2 Dataset

Here, I gather three dictionaries as the dataset for the system. One dictionary for positive word one for negative word and another for negative auxiliary verb. In the list of the negative and positive dictionary there are some most used text imoji is available with the polarity values. Now here, I give a sample of three type of data.

Table 1: Sample of the word in dictionary

No	Word/Symbol	Type
1	Abound	Positive
2	Dependably	Positive
3	Complements	Positive
4	Suffice	Positive
5	Aloof	Negative
6	Authoritarian	Negative
7	Disservice	Negative
8	Ignoble	Negative
9	Don’t	Negative-auxiliary
10	☺	Positive[1]
11	☹	Negative[1]

If here I find this type (not, haven’t, won’t) of words then the system will check for the next words if it find positive then it took the value as negative and if it find negative word then it count the value as positive.

3.3 Text Input

First I defined here a variable to take the text inside it for farther use. And it assign to the built in input function ‘input()’ of python. It in the read mode to read the input text. Here I declare some special character that will trim when the system read the input. And one should give an input without having enter or newline comment. If the system find such thing then it will count or calculate only the portion before enter or newline.

3.4 Tokenization

For tokenize the words I used pythons built-in function ‘line.split()’. This function split the lines into words. And it stores the words as a list under a variable. The variable give the total number of the words in the text for calculation. It need for matching the words with the positive and negative dictionary.

3.5 Load the Dictionary

In the “About the Project” chapter I told about the dictionaries. Here, It has three different text documents with positive, negative and negative auxiliary words. In this process the system loads the files and take it inside three different variable as list. It is need to be ensured that, the dictionary words are correct and no nonrelated words stays in the dictionary.

3.6 Match the Text with Dictionary

This is the first logical part of the project. Here I used ‘for’ loop to run full text and full dictionary. After that, I use ‘if’ condition for matching or comparing the words of the text with the dictionary.

For counting the number of positive and negative word I declare here two variables. If words of the text get a match with positive dictionary then the positive count will increase by 1. And if the text get a match with negative dictionary then the negative count will increase by 1. It keeps the counter value into two different variables.

3.7 Negation Handling

If the text has not before a negative or positive word then in goes to the special case. We know that, if a positive starts with negation or negative auxiliary verb then the word will return negative value. And the word is negative then it return the positive meaning.

Here I set a filter for searching negative auxiliary verb. The filter separate the words which contains negation before it. If it find 'not' or such words before positive word then the negative prefix count increase by 1. If it find such words before negative word then the positive prefix count will increase by 1. After searching all the words of the text then the number of positive words will decrease by the negative prefix count and the number of negative words will decrease by the positive prefix count.

3.8 Sentiment Generate

The polarity will generate by the number of positive and negative words, the negative and positive prefix words and also the number of total words in the text. The calculation of the words will calculate in four part.

Firstly, the positive and negative words will calculate as below:

$$polarity = \frac{\sum positive\ words - \sum negative\ words}{\sum positive\ words + \sum negative\ words} \dots\dots\dots (1)$$

Secondly, the number positive and negative words with prefix words will calculate as below:

$$polarity\ against\ prefix = \frac{\sum positive\ prefix - \sum negative\ prefix}{\sum positive\ prefix + \sum negative\ prefix} \dots (2)$$

Thirdly, the number of positive and negative words and the total words will calculate as below:

$$polarity\ against\ total\ words = \frac{\sum positive\ word - \sum negative\ word}{\sum words} \dots (3)$$

After above calculation the system comes to our final calculation as below:

$$sentiment\ value = result\ of\ (3.1) + \frac{result\ of\ (3.2) + result\ of\ (3.3)}{2} \dots (4)$$

The resultant value of equation (5.4) will be the value of the sentiment. Then it will classified in the next process.

3.9 Classification

The value of sentiment will lie between the ranges from -1 to 1. The class of the value defined by some ranges. If the value lies between the range then the system return the level of the sentiment of the text. Using the range one can classify the result the result on his/her own purpose. Here I just divide the values into some parts how one can understand the meaning of the context.The value range divided as seven classes they are given below in the table 2.

Table 2: Classification of Sentiment value

Class	Range
A	-0.1 to 0.1
B	-0.3 to -0.1
C	0.3 to 0.1
D	-0.5 to -0.3
E	0.3 to 0.5
F	-1 to -0.5
G	0.5 to 1

The specific class of sentiment will return a specific comment about the text which was inputted. It is the most specified comments which has given against the values of sentiment. If one want another type of values like marketing related or product related values. Here I gave the general values in the table. The comments on the classes against the values are given below in the table 3:

Table 3: Comments against Classes

Class	Comment
A	Neutral
B	Lightly Negative
C	Lightly Positive
D	Mid-level Negative
E	Mid-level Positive
F	High-level Negative
G	High-level Positive

4. Experimental Result And Discussion

Firstly, the system take input from the user. And it load the dictionaries which given here as training data. Then the main function of the system began to search and calculate.

Here I take some simple text for calculating the experimental result of the sentiment.

Sample Text: “People like a good person and the people hate a two-faced person all time.”

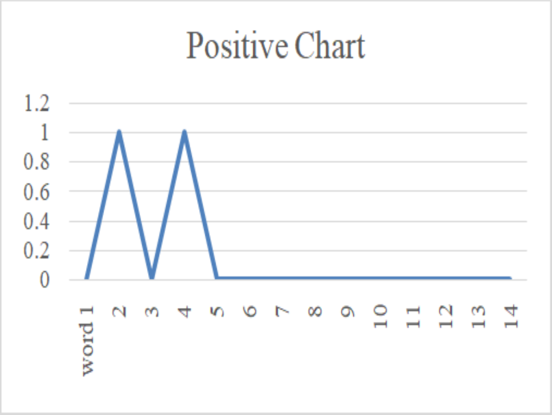


Figure 2: Positive Word Identify

From this graph we can understand there is two positive words in this piece of text.

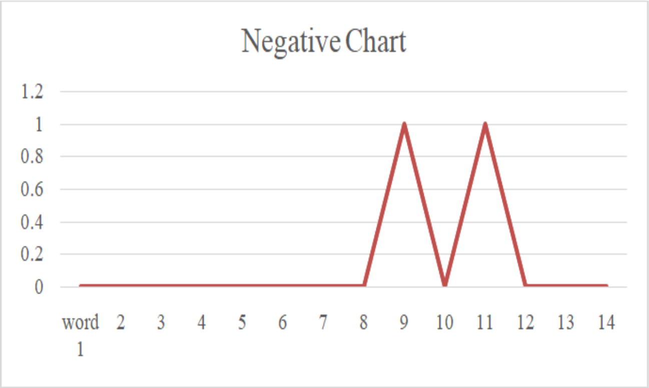


Figure 3: Negative Word Identify

From this graph we can understand there is two negative words in this piece of text.

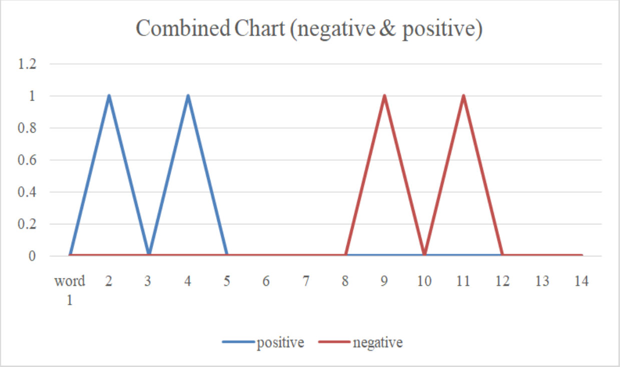


Figure 4: Combined chart of positive & negative word.

From this graph we can understand there is two positive and also two negative words in this piece of text.

If here, the system find negative prefix before the word then it will change the type of the word. Means if the word is positive word then it will give negative meaning and if it negative then that will give positive meaning.

The number of positive and negative will count in two different counter variable. And used in calculation:

Polarity = (2-2)/ (2+2) = 0/4= 0.

This polarity consider as the prime polarity. After thatthe positive and negative words calculated as,

Polarity against total word = (2 - 2)/14 = 0/14 = 0.

Then the prefixes are being calculated, here we don’t find any prefix so we don’t get any value from the prefix calculation here.

At last, the calculated values combined and give the final result.

Combined result = 0 + ((0 + 0)/2) = 0.

Which represent the texts experimental sentiment and that is neutral.

However, these polarity calculation methods were used on the dataset and accuracy of the system was finally calculayed by the following formula:

$$Accuracy = \frac{TP + FN}{TP + FN + FP + TN}$$

Where, TP, FN, FP and TN represent True Positive, False Negetive, False positive and True Negetive respectively. Based this formula our system showed an accuracy of 92.3%.

5. Conclusion

This system is going to replace the manual analysis on text. It can be used to define human personality from the text also. It will able to find how they feel about a topic or a product by separating the positive and negative words from the text. It will give the user a smooth result from text which may help users to understand about the context. The simplicity of the work would made it worthy to use in various sentiment analysis work. However the generated result of the system is not fully correct and may improve in fuure.

References

[1] Ibm.com. 2021. *What is Natural Language Processing?*. [online] Available at: https://www.ibm.com/cloud/learn/natural-language-processing.

[2] Pang B, Lee VS, Lillian (2002) Thumbs up?: sentiment classification using machine learning techniques. EMNLP ‘02.

[3] Mukherjee S, Joshi S (2014). Author-specific sentiment aggregation for polarity prediction of reviews. In: Proceedings of the ninth international conference on language resources and evaluation, LREC 2014, Reykjavik, 26–31 May 2014, pp 3092–3099

[4] Ramteke A, Malu A, Bhattacharyya P, Nath JS (2013) Detecting turnarounds in sentiment analysis: thwarting. In: ACL (2). The Association for Computer Linguistics, pp 860–865

[5] Dave K, Lawrence S, Pennock DM (2003) Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: Proceedings of the 12th international conference on world wide web, WWW '03. ACM, New York, pp 519–528

[6] Benamara F, Cesarano C, Picariello A, Reforgiato D, Subrahmanian V (2007) Sentiment analysis: adjectives and adverbs are better than adjectives alone. In: Proceedings of the international conference on weblogs and social media (ICWSM)

[7] Turney PD (2002) Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th annual meeting on Association for Computational Linguistics, ACL '02. Association for Computational Linguistics, Stroudsburg, pp 417–424

[8] Wang H, Lu Y, Zhai C (2010) Latent aspect rating analysis on review text data: a rating regression approach. In: KDD, pp 783–792

[9] Wang H et al (2011) Latent aspect rating analysis without aspect keyword supervision. In: KDD '11

[10] Yu J et al (2011) Aspect ranking: identifying important product aspects from online consumer reviews. ACL, pp 1496–1505

[11] Lu Y, Zhai C, Sundaresan N (2009) Rated aspect summarization of short comments. In: WWW, pp 131–140

[12] Titov I, McDonald RT (2008) A joint model of text and aspect ratings for sentiment summarization. ACL, Columbus, Ohio, pp 308–316

[13] Stappen, L., Baird, A., Cambria, E., & Schuller, B. W. (2021). Sentiment analysis and topic recognition in video transcriptions. *IEEE Intelligent Systems*, *36*(2).

[14] Rice, D. R., & Zorn, C. (2021). Corpus-based dictionaries for sentiment analysis of specialized vocabularies. *Political Science Research and Methods*, *9*(1), 20-35.

[15] Long, Y., Lu, Q., Xiang, R., Li, M., & Huang, C. R. (2021). A cognition based attention model for sentiment analysis. Association for Computational Linguistics (ACL).