# Effective Segmentation of Consumer Feedback Analysis using Machine Learning and Deep Learning Algorithms

[1]B. Sangeetha, [2]K Sandhya Rani, [3]Dr. Pradeep Venuthurumilli, [4]K. Praveen Kumar

[1]Assistant Professor, Department of CSE (DS), Vignan's Institute of Management and Technology for Women, Kondapur, Ghatkesar, Telangana

[2]Assistant Professor, Department of AI&DS, Vignan's Institute of Management and Technology for Women, Kondapur, Ghatkesar, Telangana

[3]Associate Professor& HOD, Department of CSE (DATA SCIENCE), Mallareddy Engineering College for Women, Maisammaguda, Secunderabad

[4]Assistant professor, Department of CSE, CMR technical campus, Kandlakoya, Hyderabad.

## Abstract

This Paper discourses the delinquent of consumer reviews analysis in Social Media Like twitter, Facebook, Instagram etc, that is classifying tweets according to the feedback expressed in them: positive, negative or neutral. Twitter is an online micro-blogging and social networking platform which allows users to write short status updates of maximum length 140 characters. It is a rapidly expanding service with over 200 million users, out of which half of them log in on a daily basis - generating nearly 500 million tweets per day. Due to this large amount of usage we hope to achieve a reflection of public opinion by analysing the opinions expressed in the tweets. Analysing the public opinion is important for many applications such as firms trying to find out the response of their products in the market, predicting political elections and predicting socioeconomic phenomena like stock exchange. The aim of this Paper is to develop a functional classifier for accurate and automatic opinion classification of an unknown tweet stream.

***Keywords: Consumer feedback, Analysis, Machine Learning, Deep Learning, Social Media.***

## 1. Introduction

This Paper statements the tricky of consumer reviews analysis in twitter; that is classifying tweets according to the feedback expressed in them: positive, negative or neutral. Twitter is an online micro-blogging and social-networking platform which allows users to write short status updates of maximum length 140 characters. It is a rapidly expanding service with over 200 million registered users - out of which 100 million are active users and half of them log on twitter on a daily basis - generating nearly 250 million tweets per day. Due to this large amount of usage we hope to achieve a reflection of public opinion by analyzing the opinions expressed in the tweets. Analyzing the public opinion is important for many applications such as firms trying to find out the response of their products in the market, predicting political elections and predicting socioeconomic phenomena like stock exchange. The aim of this Paper is to develop a functional classifier for accurate and automatic opinion classification of an unknown tweet stream. In the past few years, there has been a huge growth in the use of microblogging platforms such as Twitter. Spurred by that growth, companies and media organizations are increasingly seeking ways to mine Twitter for information about what people think and feel about their products and services. Companies such as Twittar (Twittar.com), tweetfeel (www. tweetfeel.com), and Social Mention (www.socialmention.com) are just a few who advertise Twitter sentiment analysis as one of their services.

While there has been a fair amount of research on how sentiments are expressed in genres such as online reviews and news articles, how sentiments are expressed given the informal language and message-length constraints of microblogging has been much less studied. Features such as automatic part-of-speech tags and resources such as sentiment lexicons have proved useful for sentiment analysis in other domains, but will they also prove useful for sentiment analysis in Twitter? In this paper, we begin to investigate this question.

Table-1: List of Notations Used

| | |
|---|---|
| OM | Open Mining |
| SA | Sentiment Analysis |
| ML | Machine Learning |
| MPQA | Multi Perspective Question Answering |
| OAuth | Open Authorization Protocol |
| NLP | Natural Language Processing |
| POS | Parts Of Speech |
| SVM | Support Vector Machine |

The online medium has become a significant way for people to express their opinions and with social media, there is an abundance of opinion information available. Using sentiment analysis, the polarity of opinions can be found, such as positive, negative, or neutral by analyzing the text of the opinion.

Sentiment analysis has been useful for companies to get their customer's opinions on their products predicting outcomes of elections , and getting opinions from movie reviews. The information gained from sentiment analysis is useful for companies making future decisions. Many traditional approaches in sentiment analysis uses the bag of words method. The bag of words technique does not consider language morphology, and it could incorrectly classify two phrases of having the same meaning because it could have the same bag of words. The relationship between the collection of words is considered instead of the relationship between individual words. When determining the overall sentiment, the sentiment of each word is determined and combined using a function. Bag of words also ignores word order, which leads to phrases with negation in them to be incorrectly classified. Other techniques discussed in sentiment analysis include Naive Bayes, Maximum Entropy, and Support Vector Machines. In the Literature Survey section, approaches used for sentiment analysis and text classification are summarized. Sentiment analysis refers to the broad area of natural language processing which deals with the computational study of opinions, sentiments and emotions expressed in text. Sentiment Analysis (SA) or Opinion Mining (OM) aims at learning people's opinions, attitudes and emotions towards an entity. The entity can represent individuals, events or topics. An immense amount of research has been performed in the area of sentiment analysis. But most of them focused on classifying formal and larger pieces of text data like reviews. With the wide popularity of social networking and microblogging websites and an immense amount of data available from these resources, research Papers on sentiment analysis have witnessed a gradual domain shift. The past few years have witnessed a huge growth in the use of microblogging platforms. Popular microblogging websites like Twitter have evolved to become a source of varied information. This diversity in the information owes to such microblogs being elevated as platforms where people post real time messages about their opinions on a wide variety of topics, discuss current affairs and share their experience on products and services they use in daily life.

While reviews are characterized by formal text patterns and are summarized thoughts of authors, tweets are more casual and restricted to 140 characters of text. Tweets offer companies an additional avenue to gather feedback. Sentiment analysis to research products, movie reviews etc. aid customers in decision making before making a purchase or planning for a movie. Enterprises find this area useful to research public opinion of their company and products, or to analyze customer satisfaction. Organizations utilize this information to gather feedback about newly released products which supplements in improving further design. Different approaches which include machine learning(ML) techniques, sentiment lexicons, hybrid approaches etc. have been proved useful for sentiment analysis on formal texts. But their effectiveness for extracting sentiment in microblogging data will have to be explored. A careful investigation of tweets reveals that the 140 character length text restricts the vocabulary which imparts the sentiment. The hyperlinks often present in these tweets in turn restrict the vocabulary size. The varied domains discussed would surely impose hurdles for training.

The proposed work attempts a novel approach on twitter data by aggregating an adapted polarity lexicon which has learnt from product reviews of the domains under consideration, the tweet specific features and unigrams to build a classifier model using machine learning techniques.

## 2. Related Work

### 2. 1 Limitations of Prior Works

Feedback analysis of in the domain of micro-blogging is a relatively new research topic so there is still a lot of room for further research in this area. Decent amount of related prior work has been done on feedback analysis of user reviews [x], documents, web blogs/articles and general phrase level feedback analysis [1]. These differ from twitter mainly because of the limit of 140 characters per tweet which forces the user to express opinion compressed in very short text. The best results reached in feedback classification use supervised learning techniques such as Naive Bayes and Support Vector Machines, but the manual labelling required for the supervised approach is very expensive. Some work has been done on unsupervised [2] and semi-supervised approaches, and there is a lot of room of improvement. Various researchers testing new features and classification techniques often just compare their results to base-line performance. There is a need of proper and formal comparisons between these results arrived through different features and classification techniques in order to select the best features and most efficient classification techniques for particular applications.

2. 2 Recent Work

The bag-of-words model is one of the most widely used feature model for almost all text classification tasks due to its simplicity coupled with good performance. The model represents the text to be classified as a bag or collection of individual words with no link or dependence of one word with the other, i. e. it completely disregards grammar and order of words within the text. This model is also very popular in feedback analysis and has been used by various researchers. The simplest way to incorporate this model in our classifier is by using unigrams as features. Generally speaking, n-grams is a contiguous sequence of “n” words in our text, which is completely independent of any other words or grams in the text. So unigrams is just a collection of individual words in the text to be classified, and we assume that the probability of occurrence of one word will not be affected by the presence or absence of any other word in the text. This is a very simplifying assumption but it has been shown to provide rather good performance [3]. One simple way to use unigrams as features is to assign them with a certain prior polarity, and take the average of the overall polarity of the text, where the overall polarity of the text could simply be calculated by summing the prior polarities of individual unigrams. Prior polarity of the word would be positive if the word is generally used as an indication of positivity, for example the word “sweet”; while it would be negative if the word is generally associated with negative connotations, for example “evil”. There can also be degrees of polarity in the model, which means how much indicative is that word for that particular class. A word like “awesome” would probably have strong subjective polarity along with positivity, while the word “decent” would although have positive prior polarity but probably with weak subjectivity.

There are three ways of using prior polarity of words as features. The simpler un- supervised approach is to use publicly available online lexicons/dictionaries which map a word to its prior polarity. The Multi-Perspective-Question-Answering (MPQA) is an online resource with such a subjectivity lexicon which maps a total of 4, 850 words according to whether they are "positive" or "negative" and whether they have "strong" or "weak" subjectivity [4]. The SentiWordNet 3. 0 is another such resource which gives probability of each word belonging to positive, negative and neutral classes. Kouloumpis et al. noted a decrease in performance by using the lexicon word features along with custom n-gram word features constructed from the training data, as opposed to when the n-grams were used alone [5]. The third approach is a middle ground between the above two approaches. In this approach we construct our own polarity lexicon but not necessarily from our training data, so we don't need to have labelled training data. One way of doing this as proposed by Turney et al. is to calculate the prior semantic orientation (polarity) of a word or phrase by calculating it's mutual information with the word "excellent" and subtracting the result with the mutual information of that word or phrase with the word "poor" [6]. They used the number of result hit counts from online search engines of a relevant query to compute the mutual information. The final formula they used is as follows:

$$Polarity(phrase) = \log_2 \frac{hits(phrase\ NEAR\ \text{"excellent"}).hits(\text{"poor"})}{hits(phrase\ NEAR\ \text{"poor"}).hits(\text{"excellent"})}$$

Where hits (phrase NEAR "excellent") means the number documents returned by the search engine in which the phrase (whose polarity is to be calculated) and word "excellent" are co-occurring. While hits("excellent") means the number of documents retuned which contain the word "excellent". Prabowo et al. have gone ahead with this idea and used a seed of 120 positive words and 120 negative to perform the internet searches [7]. So the overall semantic orientation of the word under consideration can be found by calculating the closeness of that word with each one of the seed words and taking and average of it. Another graphical way of calculating polarity of adjectives has been discussed by Hatzivassiloglou et al. [8]. The process involves first identifying all conjunctions of adjectives from the corpus and using a supervised algorithm to mark every pair of adjectives as belonging to the same semantic orientation or different. A graph is constructed in which the nodes are the adjectives and links indicate same or different semantic orientation. Finally, a clustering algorithm is applied which divides the graph into two subsets such that nodes within a subset mainly contain links of same orientation and links between the two subsets mainly contain links of different orientation. One of the subsets would contain positive adjectives and the other would contain negative.

Many of the researchers in this field have used already constructed publicly available lexicons of feedback bearing words and while many others have also explored building their own prior polarity lexicons.

The basic problem with the approach of prior polarity approach has been identified by Wilson et al. who distinguish between prior polarity and contextual polarity. They say that the prior polarity of a word may in fact be different from the way the word has been used in the particular context. The paper presented the following phrase as an example:

Philip Clapp, president of the National Environment Trust, sums up well the general thrust of the reaction of environmental movements: “There is no reason at all to believe that the polluters are suddenly going to become reasonable. ” In this example all of the four underlined words “trust”, “well”, “reason” and “reasonable” have positive polarities when observed without context to the phrase, but here they are not being used to express a positive feedback. This concludes that even though generally speaking a word like “trust” may be used in positive sentences, but this doesn’t rule out the chances of it appearing in non-positive sentences as well.

Henceforth prior polarities of individual words (whether the words generally carry positive or negative connotations) may alone not enough for the problem. The paper explores some other features which include grammar and syntactical relationships between words to make their classifier better at judging the contextual polarity of the phrase. The task of twitter feedback analysis can be most closely related to phrase level feedback analysis. A seminal paper on phrase level feedback analysis was presented in 2005 by Wilson et al. [16] which identified a new approach to the problem by first classifying phrases according to subjectivity (polar) and objectivity (neutral) and then further classifying the subjective-classified phrases as either positive or negative. The paper noticed that many of the objective phrases used prior feedback bearing words in them, which led to poor classification of especially objective phrases. It claims that if we use a simple classifier which assumes that the contextual polarity of the word is merely equal to its prior polarity gives a result of about 48%. The novel classification process proposed by this paper along with the list of ingenious features which include information about contextual polarity resulted in significant improvement in performance (in terms of accuracy) of the classification process. The results from this paper are presented in the table below:

Table 2: Accuracy for different features 1

| Features | Accuracy | Subjective F. | Objective F. |
|---|---|---|---|
| Word tokens | 73.6 | 55.7 | 81.2 |
| Words + prior polarity | 74.2 | 60.6 | 80.7 |
| 28 features | 75.9 | 63.6 | 82.1 |

Table 3: Accuracy for different features 2

| Features | Accuracy | Positive F. | Negative F. | Both F. | Objective F. |
|---|---|---|---|---|---|
| Word tokens | 61.7 | 61.2 | 73.1 | 14.6 | 37.7 |
| Word + prior | 63.0 | 61.6 | 75.5 | 14.6 | 40.7 |
| 10 features | 65.7 | 65.1 | 77.2 | 16.1 | 46.2 |

One way of alleviating the condition of independence and including partial context in our word models is to use bigrams and trigrams as well besides unigrams. Bigrams are collection of two contiguous words in a text, and similarly trigrams are collection of three contiguous words. So we could calculate the prior polarity of the bigram / trigram - or the prior probability of that bigram / trigram belonging to a certain class – instead of prior polarity of individual words. Many researchers have experimented with them with the general conclusion that if we have to use one of them alone unigrams perform the best, while unigrams along with bigrams may give better results with certain classifiers. However trigrams usually result in poor performance as reported by. The reduction in performance by using trigrams is because there is a compromise between capturing more intricate patterns and word coverage as one goes to higher-numbered grams. Besides from this some researchers have tried to incorporate negation into the unigram word models. Pang et al. and Pakl et al. used a model in which the prior polarity of the word was reversed if there was a negation (like “not”, “no”, “don’t”, etc. ) next to that word. In this way some contextual information is included in the word models.

2. 3 about the dataset

To gather the data many options are possible. In some previous paper researches, they have used datasets from Kaggle and other online sources. In this Paper, we will extract live tweets from Twitter using Authentication keys like Consumer key, Consumer Secret Key, Access Token, Access Token Secret key.

## 3. Methodology

The bag-of-words model is one of the most widely used feature model for almost all text classification tasks due to its simplicity coupled with good performance. The model represents the text to be classified as a bag or collection of individual words with no link or dependence of one word with the other, i. e. it completely disregards grammar and order of words within the text. This model is also very popular in feedback analysis and has been used by various researchers. The simplest way to incorporate this model in our classifier is by using unigrams as features.

Sentiment analysis of in the domain of micro-blogging is a relatively new research topic so there is still a lot of room for further research in this area. Decent amount of related prior work has been done on sentiment analysis of user reviews , documents, web blogs/articles and general phrase level sentiment analysis. These differ from twitter mainly because of the limit of 140 characters per tweet which forces the user to express opinion compressed in very short text. The best results reached in sentiment classification use supervised learning techniques such as Naive Bayes and Support Vector Machines, but the manual labelling required for the supervised approach is very expensive. Some work has been done on unsupervised and semi-supervised approaches, and there is a lot of room of improvement. Various researchers testing new features and classification techniques often just compare their results to base-line performance. There is a need of proper and formal comparisons between these results arrived through different features and classification techniques in order to select the best features and most efficient classification techniques for particular applications.

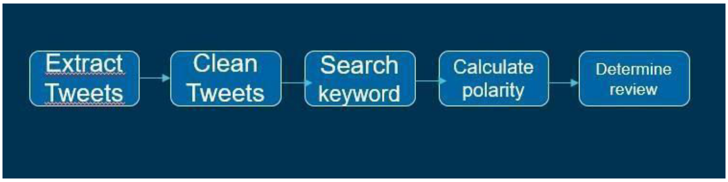


Figure 1: Tweet Pre-processing

Using machine learning techniques and natural language processing we can extract the subjective information of a document and try to classify it according to its polarity such as positive, neutral or negative. It is a really useful analysis since we could possibly determine the overall opinion about a selling objects, or predict stock markets for a given company like, if most people think positive about it, possibly its stock markets will increase, and so on. In this Paper I choose to try to classify tweets from Twitter into “positive” or “negative” feedback by building a model based on probabilities. The end data is visualized using matplotlib for clean and clear presentation to the user.

**3. 1 Input(KeyWord):**

Data in the form of raw tweets is acquired by using the Python library “tweepy” which provides a package for simple twitter streaming API. This API allows two modes of accessing tweets: SampleStream and FilterStream. SampleStream simply delivers a small, random sample of all the tweets streaming at a real time. FilterStream delivers tweet which match a certain criteria. It can filter the delivered tweets according to three criteria:

• Specific keyword to track/search for in the tweets

• Specific Twitter user according to their name

• Tweets originating from specific location(s) (only for geo-tagged tweets).

A programmer can specify any single one of these filtering criteria or a multiple combination of these. But for our purpose we have no such restriction and will thus stick to the SampleStream mode. Since we wanted to increase the generality of our data, we acquired it in portions at different points of time instead of acquiring all of it at one go. If we used the latter approach then the generality of the tweets might have been compromised since a significant portion of the tweets would be referring to some certain trending topic and would thus have more or less of the same general mood or sentiment. This phenomenon has been observed when we were going through our sample of acquired tweets. For example the sample acquired near Christmas and New Year's had a significant portion of tweets referring to these joyous events and were thus of a generally positive sentiment. Sampling our data in portions at different points in time would thus try to minimize this problem. Thus forth, we acquired data at four different points which would be 17th of December 2015, 29th of December 2015, 19th of January 2016 and 8th of February 2016. A tweet acquired by this method has a lot of raw information in it which we may or may not find useful for our particular application. It comes in the form of the python "dictionary" data type with various key-value pairs.

### 3. 2 Tweets Retrieval:

Since human labelling is an expensive process we further filter out the tweets to be labelled so that we have the greatest amount of variation in tweets without the loss of generality. The filtering criteria applied are stated below:

• Remove Retweets (any tweet which contains the string "RT")

• Remove very short tweets (tweet with length less than 20 characters)

• Remove non-English tweets (by comparing the words of the tweets with a list of 2, 000 common English words, tweets with less than 15% of content matching threshold are discarded)

• Remove similar tweets (by comparing every tweet with every other tweet, tweets with more than 90% of content matching with some other tweet is discarded) After this filtering roughly 30% of tweets remain for human labelling on average per sample, which made a total of 10, 173 tweets to be labelled.

### 3. 3 Data Preprocessing:

Data preprocessing consists of following Steps:

The raw data that is extracted from Twitter converted into proper format. Fromatted data is stored in dataframe of pandas library. Then, tweets are cleaned such that anything like @hello or @upgrade are deleted and also everything other than text, numbers, space, tabspace are also deleted. RT is also removed if the tweet is retweeted. Anyother hyperlinks are also removed.

**3. 4 Classified Tweets:**

We labelled the tweets in three classes according to sentiments expressed/observed in the tweets: positive, negative and neutral We gave the following guidelines to our labellers to help them in the labelling process:

Positive: If the entire tweet has a positive/happy/excited/joyful attitude or if something is mentioned with positive connotations. Also if more than one sentiment is expressed in the tweet but the positive sentiment is more dominant. Example: “4 more years of being in shithole Australia then I move to the USA! :D”.

Negative: If the entire tweet has a negative/sad/displeased attitude or if something is mentioned with negative connotations. Also if more than one sentiment is expressed in the tweet but the negative sentiment is more dominant. Example: “I want an android now this iPhone is boring :S”.

Neutral: If the creator of tweet expresses no personal sentiment/opinion in the tweet and merely transmits information. Advertisements of different products would be labelled under this category. Example: “US House Speaker vows to stop Obama contraceptive rule... http://t. co/cyEWqKlE”.

**4. Experiments and Results**

The task of feedback analysis, especially in the domain of micro-bloging, is still in the developing stage and far from complete. So we propose a couple of ideas which we feel are worth exploring in the future and may result in further improved performance.

Right now we are exploring Parts of Speech separate from the unigram models, we can try to incorporate POS information within our unigram models in future. So say instead of calculating a single probability for each word like P(word | obj) we could instead have multiple probabilities for each according to the Part of Speech the word belongs to. For example we may have P(word | obj, verb), P(word | obj, noun) and P(word | obj, adjective). Pang et al. used a somewhat similar approach and claims that appending POS information for every unigram results in no significant change in performance (with Naive

Bayes performing slightly better and SVM having a slight decrease in performance), while there is a significant decrease in accuracy if only adjective unigrams are used as features. However these results are for classification of reviews and may be verified for feedback analysis on micro blogging websites like Twitter.

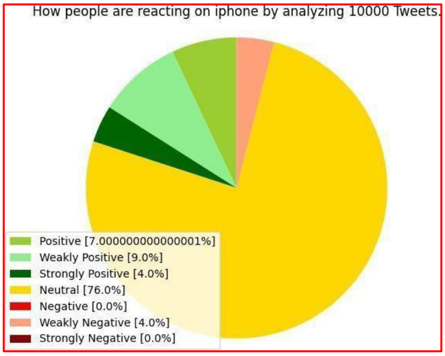


Figure 2: Figure Feedback for iPhone

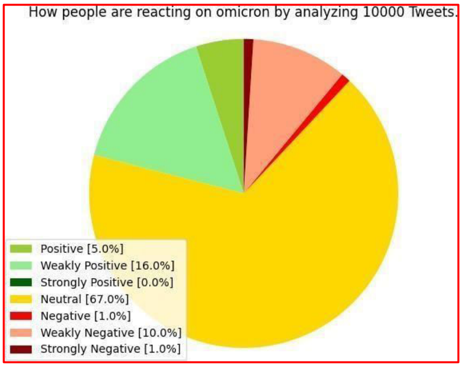


Figure 3: Feedback for omicron

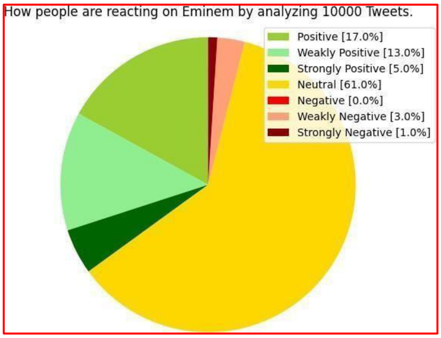


Figure 5: Feedback for Eminem

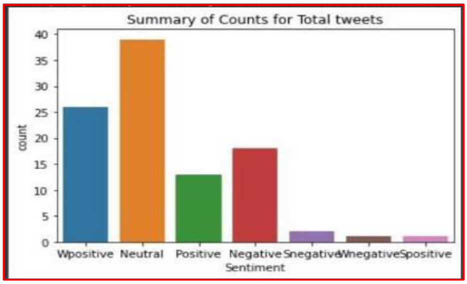


Figure 6: Tweet Summary Count

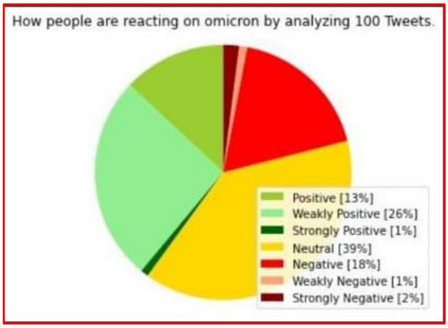


Figure 7: 100Tweets analysis

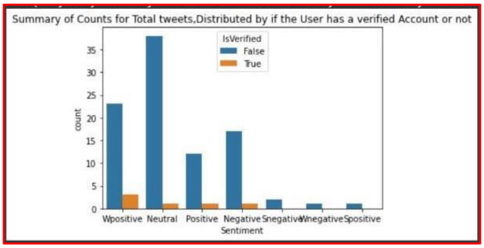


Figure 8: Tweet Count Account Verification

## 5. Conclusions and Future Work

In this research we focused on general feedback analysis. There is potential of work in the field of feedback analysis with partially known context. For example we noticed that users generally use our website for specific types of keywords which can divided into a couple of distinct classes, namely: politics/politicians, celebrities, products/brands, sports/sportsmen, media/movies/music. So we can attempt to perform separate feedback analysis on tweets that only belong to one of these classes (i. e. the training data would not be general but specific to one of these categories) and compare the results we get if we apply general feedback analysis on it instead.

Right now we have worked with only the very simplest unigram models; we can improve those models by adding extra information like closeness of the word with a negation word. We could specify a window prior to the word (a window could for example be of 2 or 3 words) under consideration and the effect of negation may be incorporated into the model if it lies within that window. The closer the negation word is to the unigram word whose prior polarity is to be calculated, the more it should affect the polarity.

## References

[1] Rahul, S.; Laxmiputra, S.; Saraswati, J. Customer segmentation using rfm model and k-means clustering. Int. J. Sci. Res. Sci. Technol. 2021, 8, 591–597

[2] Jinfeng, Z.; Jinliang, W.; Bugao, X. Customer segmentation by web content mining. J. Retail. Consum. Serv. **2021**, 61, 102588.

[3] Himanshu, S.; Shubhangi, N. Improving Customer Segmentation in E-Commerce Using Predictive Neural Network. Int. J. **2020**, 9, 2326–2331.

[4] Ge, X.; Jin, Y.; Ren, J. A Data-Driven Approach for the Optimization of Future Two-Level Hydrogen Supply Network Design with Stochastic Demand under Carbon Regulations. Sensors **2021**, 21, 5728.

[5] Joshua, E.S.N.; Vardhan, K.A.; Rao, N.T.; Bhattacharyya, D. An Enhanced K-Means Clustering Algorithm to Improve the Accuracy of Clustering Using Centroid Identification Based on Compactness Factor; Springer: Singapore, 2021; pp. 59–67.

[6] Xin, S.Y.; Xing, S.H. Nature-Inspired Computation in Data Mining and Machine Learning; Springer Nature: Berlin/Heidelberg, Germany, 2020; Volume 855, pp. 225–248

[7] Seongbeom, H.; Yuna, L. Identifying Customer Priority for New Products in Target Marketing: Using RFM Model and Textrank. Innov. Mark. **2021**, 17, 125–133.

[8] Sukru, O.; Leonardo, O.I. Artificial Neural Networks in Customer Segmentation. In Proceedings of the ResearchGate, Sivas, Turkey, 10 September 2019; pp. 1–5.