Journal of the Nigerian Association of Mathematical Physics
Volume 53 **(November** ***2019 Issue*****), pp67 – 74**
© J. of NAMP

EFFECT OF PREPROCESSING ON SENTIMENT ANALYSIS USING NAÏVE BAYESIAN AND STOCHASTIC DESCENT GRADIENT ON N-GRAMS

Abdullah K-K. A, Sodimu S. M., Efuwape B. T., Solanke O. O., Olubanwo O. O. and Olasupo A. O.

Department of Mathematical Sciences, Olabisi Onabanjo University, Ago Iwoye, Ogun State, Nigeria

Abstract

Social media data are unstructured, so there is need for preprocessing to prevent unsatisfactory classification accuracy. Sentiment classification is affected by noisy nature of social media data, thus, to reduce the noise of textual data is to remove stop words and others unwanted words so as to help with the efficiency in the accuracy of sentiment classification. The appropriate selection and extraction of words/features has a huge impact on the classifier. In this work, feature representation is done in unigram, bi-gram and tri-gram using term frequency-inverse document frequency (TF-IDF) and chi-square as feature reduction. In order to analyse tweets data, Naive Bayesian and Stochastic Gradient Descent classifiers are used on term weighting scheme. This paper examines the effect of stopwords with or without on data sparsity, feature space and performance of the classifiers, however, the performance of classifiers are evaluated on the basis of the accuracy.

Keywords**:** Classification, Chi Square, TF-IDF, Naive bayes, Stochastic gradient descent.

1.0 Introduction

The huge amount of data stored online can be mined effectively to extract valuable information and make decision based on extracted information, thus, this information plays important role in sentiment classification task. Therefore, sentiment analysis is treated as a classification task as it classifies the tweets into different classes or polarity [1, 2]. So, the sentiment analysis work becomes popular since it is capable of analysing thousands of reviews and presents the output to the user in a simple and understandable manner. Sentiment classification methods can be classified into machine learning, lexicon based methods, and linguistic methods [3]. Many researchers claimed that lexicon-based methods and linguistic methods do not perform well on sentiment classification due to nature of an opinionated text which requires more understanding of text [4, 5]. Hence, Twitter is one of the most popular micro-blogging social-media platforms that provide a platform for millions of people to share their daily opinions or thoughts using real-time status update [6]. The Twitter data are predominantly unstructured in nature and also contains a large amount of noisy data, such as URLs, user names, punctuations symbols, stopwords etc. These characters make sentiment classification a bit difficult and challenging and thus preprocessing play a vital role in Twitter sentiment analysis. Preprocessing is carried out to convert unstructured data to structured form and undesirable information is filtered out.

People depend upon user generated online content to a great extent for decision making. The amount of content generated by users is too vast for a normal user to analyze, so there is a need for automatic sentiment analysis techniques. Machine learning classifiers deal with large amount of data which is not possible by traditional techniques. Therefore, the information collected from social media can serve as an important parameter for online enterprises if the information is properly dealt with for knowledge discovery purposes [7, 8]. Twitter sentiment analysis using machine learning techniques comprises of tasks such as preprocessing, feature representation and extraction, classification and evaluation. Preprocessing of stopwords in the literature is used in document classification and retrieval. This has been applied to Twitter in the

Corresponding Author**:** Abdullah K.K.A., Email: uwaizabdullah9@gmail.com, Tel: +2348060046592

context of sentiment analysis but obtained contradictory results. Although, some works are in support of removal of stopwords [9, 10] while others claimed that stopwords indeed carry sentiment information and removing it harms the performance of Twitter sentiment classifiers [11, 12]. This can be evaluated in this work by considering the effect of stopwords in the Twitter text using Naïve Bayes and Stochastic gradient Descent (SGD) classifiers. Preprocessed tweets are represented using N-gram representation model [13] which is a contiguous sequence of n number of words. Bag of words [14] is the one of the simplest representation of textual data and Vector Space Model (VSM)[15] is mostly used in document classification system. Thousands of term word occurs in the text document, so it is important to reduce the dimensionality of feature using feature selection process [16], to resolve this problem different techniques can be used. Researchers have used different feature selection methods such as Chi-Square (χ^2), Information Gain (IG), Mutual Information (MI), term strength, Term Frequency Inverse Document Frequency (TFIDF). With the help of these approaches, it is possible to reduce the high dimensionality of features. The main aim of this study was to examine the effect of stopwords preprocessing with the help of two different term weighting scheme that is tfidf and chi-square in analysing online tweets datasets. Subsequently, VADER (Valence Aware Dictionary and Sentiment Reasoner) lexicon [17] is employed for polarity classification taking into consideration the contextual emotion such as punctuation, slang, modifiers. This mapped the intensity values and performs normalisation to strengthen the sentiment rather than positive, negative or neutral labels. Then a weight values are associated with each pair using the TFIDF and Chi-Square schemes specifically using unigram, bi-gram and tri-gram as the feature representation and feature reduction respectively. Finally, the study for polarity classification is done in terms of accuracy, data sparsity and size of features in Twitter sentiment classifiers using Naïve Bayesian (NB) and Stochastic Gradient Descent (SGD).

The rest of the paper is organized as follows: Section 2 explains related work on sentiment analysis on Twitter data, Section 3 focuses on the methodology. Section 4 contains experimentation, results and discussion and finally, concludes the work with outlining future work in Section 5.

2.0 Background of the Study and Related Work

Microblogs are used for expressing sentiments on an event or topic, hence, many researchers concentrated their studies to understand sentiments expressed in Twitter. Twitter is one of the most commonly used micro-blog to express sentiment over the current issues. Research related to sentiment analysis can be done at three different levels: document level, sentence level and feature level. According to [18, 19], document level sentiment analysis classify the entire document either as positive or negative which is done using supervised learning. Sentiment analysis at sentence level applied the syntactic and semantic frames to detect the subjective sentences but failed to discover the sentiments about an entity and its aspects [20, 21]. Finally, feature level sentiment analysis performs the analysis at finer level of granularity [22]. Dimensionality reduction technique can be classified into feature extraction (FE) and feature selection (FS) approaches. Feature extraction is the first step of pre-processing used to presents the text documents into clear word format, therefore, removing stopwords and all other forms of preprocessing tasks is important [23]. The Twitter data are represented by a great amount of features and predominantly unstructured in nature, subsequently, contains large amount of noisy data [24]. Dimensionality reduction is based on large number of keywords, preferably on a statistical process, to create a low dimension vector. In [25], Haddi et al; described sentiment analysis with different preprocessing methods to reduce noise in the text. The results of preprocessing techniques show that data transformation and filtering can significantly enhance the performance of classifier on sentiment identification while Uysal and Gunal [26] explore the impact of preprocessing on text classification. In sentiment analysis, choosing appropriate preprocessing task such as tokenization, stopword removal, lowercase conversion, and stemming significant improve classification accuracy whereas inappropriate combinations resulted in degrading the accuracy. Many dictionaries have been created manually such as ANEW (Affective Norms for English Words) or automatically such as SentiWordNet [27], TextBlob [28] that allow estimating a score of the negativity, positivity and objectivity of the tweets, their polarity and subjectivity but neglect the aspect of the context especially in the area of domain. VADER algorithm is used to categorise the tweets into positive, negative and neutral. This evaluates the effect on detection of events from Twitter, hence, preprocessing, feature extraction and features selection as well as WordNet semantic similarity for improving the vocabulary of the tweet. Finally, different machine learning models are used for validation.

Classification is a machine learning model for solving large amount of different predictive and analytical problems such as text categorization, fraud detection, natural language processing, market segmentation and recommender systems [29]. Traditionally, sentiment classification is regarded as a binary-classification task [30], Tripathy et al; [31] used machine learning techniques such as Naive Bayes (NB), maximum Entropy (ME), SVM, and stochastic gradient descent (SGD) classification using N-gram approach but did not use Chi-square as feature reduction. In [32], structured reviews for SVM and Naıve Bayes Classification Ensemble Method were used for Sentiment Analysis, identifying appropriate features and scoring methods from information retrieval for determining whether reviews are positive or negative. These results perform as well as traditional machine learning method then use the classifier to identify and classify tweets generated, where classification is more difficult [33], this can be optimised with stochastic gradient classifier. Naïve Bayes has proved to be optimal and efficient in machine learning text classification, according to pang et al; [30] categorized tweets on the basis of N-gram technique and categorize the polarity of sentence as either positive or negative. Po-Wei Liang et al; [34] used unigram Naive Bayes model on tweets and eliminated unwanted features using the Mutual Information and Chi square feature extraction method.

Reducing the sparsity in Twitter sentiment analysis was done by[36],experiment illustrated that appropriate text preprocessing techniques can significantly reduce sparsity and increases the classificationaccuracyThis study makes contribution by hybridizing the machine learning techniques with lexicon based method to improve classification accuracy.
.

3.0 Methodology Approach

This paper presents an approach based on Naïve Bayes (NB) and Stochastic Gradient Descent (SGD) to detect tweets categories. The approach involves tweet acquisition and streaming using Tweepy API, preprocessing to remove unwanted parts of speech with stopwords and without stopwords using unigram, bi-gram and trigram. The main objective is to determine effect of stopwords on machine learning algorithms. This approach employs the use of dimensionality reduction which reduces the feature space and removes the unimportant features in a feature representation. Consequently, machine learning algorithms are used on the selected feature space to determine the classification as shown in the figure 1 below.

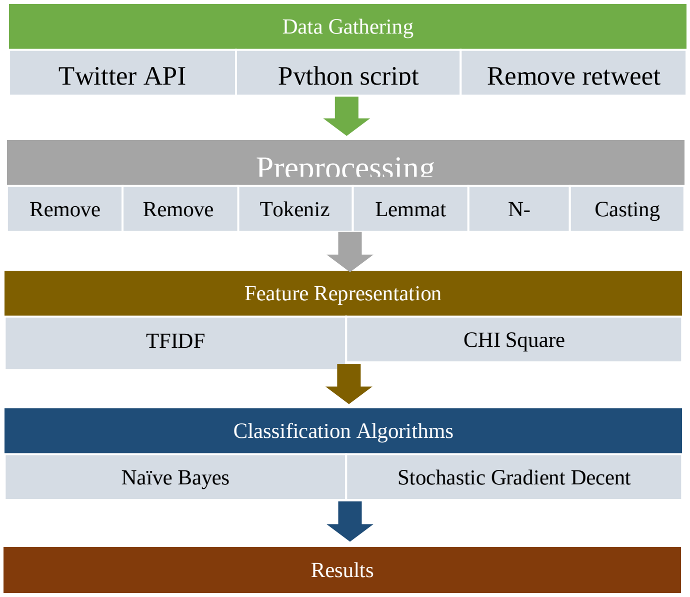


Figure 1: Implementation of the Proposed Model.

3.1.1 Data Collection and Preprocessing

Collection of data were done using Twitter API (tweepy) to extract streaming tweets, Twitter dataset is taken several times for more tweets are necessary from the Twitter page. The downloaded tweets is labelled using VADER algorithm to extract polarity and result of a sentiment from tweet.Preprocessing involves cleaning, filtering of data and making sure that the data is well prepared to be fed to feature extraction engineering techniques. In the preprocessing stage, data filtering removes noise from the datasets such as special characters (! , @, #, % e.t.c), Non ASCII characters, URL address. This does not add meaning to the dataset. Normalisation converts all characters to a lower case, this remove duplicate of same word with different case and also reduce the feature dimension. Consequently, tokenization breaks down each sentence into chunks and tokens. Finally, lemmatization groups together the inflected forms of a word into a single form or the root form.

3.1.2 Feature Extraction : Stopword Removal

The Baseline for this analysis is taken as non removal of stopwords, however, eliminating stopwords contribute less to the sentiment of tweet.Stopwords removal can be thought of as a feature selection routine, where features that do not contribute toward making correct classification decisions are considered stopwords, hence, removed from the feature space consequently. The proposed method makes use of term frequency inverse document frequency (TFIDF) in unigram, bigram, and trigram feature representations to represent the preprocessed tweets. After preprocessing, the remaining data are subjected to different features methods in order to improve the accuracy of sentiment classification, for the purpose of finding strongly related words for relevant documents and dimensionality reduction of features. The extraction of each feature is transformed into feature vector in binary form. The following feature extraction and selections are used for this study and the dataset was reduced into different training set for maximum optimisation.

i. **Term Frequency (TF):** This normalised the length to 1, no bias for short or longer words.

$$TF(t) = 1 + \log(f[t,d] \qquad (1)$$

where, $f[t,d]$ is the count of term t in document d

ii. **Term Frequency-Inverse Document Frequency (TFIDF):** This determines important word in the Twitter dataset. IDF put less weight on common terms by normalising each word with the inverse in corpus frequency. Then, adjust using Inverse Document Frequency (IDF) as expressed in equation (2)

$$TFIDF = TF * IDF \qquad (2)$$

$$idf[t] = \log(N/df[t])$$

Where, N is the total number of document in the dataset,

$df[t]$ is the number of documents containing the term t

iii. **Chi-Square** (χ^2) **:** This is use to improve classification performance and efficiency. It normalized the values by removing out words that are independent of class, hence, irrelevant for classification. It represents the degree of relationship of features, however, use for finding spam tweets. It measures how much expected counts and observed counts deviate from each other as described in equation (3).

$$X^2 = \sum_{i=0}^{n} \frac{observed - \exp ected)^2}{\exp ected}$$

$$\chi^2(f,t) = \frac{N(AD - CB)^2}{(A+C)(B+D)(C+D)} \qquad (3)$$

Where, f is a feature (a term in the Twitter),

B is the number of times f occurs without t,

t is a target variable for prediction,

C is the number of times t occurs without f,

N is the number of observation,

D is the number of times neither t and f occurs,

A is the number of times f and t co-occur.

3.2 Classification Model

The transformed features of $tfidf$ and Chi-square are sent to classification models: Naïve Bayes and Stochastic Gradient Descent (SGD) are used in prediction tasks. The description of each model and reasons for selecting these models are as follows:

Given a polarity label y where, y = {positive, negative and neutral}, and features vector x, target function f and Probability P.

i. Nave Bayesian:A Naive Bayes is a simple probabilistic classifier based on applying Bayes theorem (from Bayesian statistics) with strong (naive) independence assumptions. Naive Bayes classifier requires a small amount of training data to estimate the parameters necessary for classification as im equation (4).When assumption of independence hold

$$p(y|X) = \frac{P(X|y)P(y)}{P(X)} \tag{4}$$

where y is the label and x is a dependent feature vector of size n and $X = (x_{1,} x_{2,} ... x_n)$.

ii. **Stochastic Gradient Descent**

SGD is a simple yet very efficient approach to discriminative learning of linear classifiers under convex loss functions. It has been successfully applied to large-scale and sparse machine learning problems often encountered in text classification and sensitive to feature scaling.

In a set of training samples (x_i, y_i) where $x_i \in \Re^m$ and $y_i \in \{-1,1\}$

To learn a linear scoring function $f(x) = w^T x + b$ with model parameter $w \in \Re^m$ and intercept $b \in \Re$

Therefore to make predictions, $f(x)$ is considered by minimizing the regularized training error given by equation (5).

$$\text{E}(w,b) = \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(x_i) + \alpha\Re(w) \tag{5}$$

Where L is a loss function,

$\Re$ is a regularization term that penalized model parameter

$\alpha > 0$ is a non-negative hyperparameter

Hence, using SGD approximate the gradient of $\text{E}(w,b)$ considering a single training at a time, therefore, for each samples, SGD update the model parameter as in equation 6:

Hence, let learning rate $\eta_t = \frac{1}{\lambda t}$ for convergence analysis of stochastic approximations which will satisfy the conditions $\sum_{t=0}^{\infty} \eta_t < \infty$ and $\sum_{t=0}^{\infty} \eta = \infty$ as t increases, the weight of

$$\text{Using } \eta_t = \frac{1}{\lambda t} \quad \Rightarrow w_{t+1} = w_t - \frac{1}{\lambda t} w_t - \frac{1}{\lambda(t+1)} x_t \tag{6}$$

For $w_{t+1} = w_t - \frac{1}{\lambda t} w_t - \frac{1}{\lambda(t+1)} x \le 1$

$$w_{t+1} = w_t - \frac{1}{\lambda t} w_t$$

4.0 Experimental Results

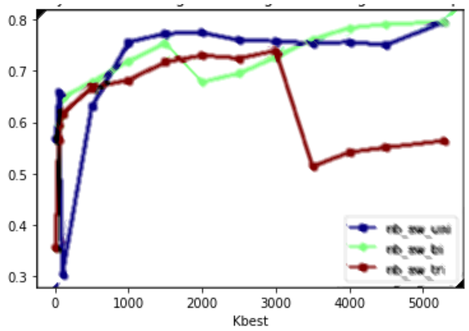
The datasets generated is based on keywords such as “Nigeria security” in which the kbest represents the number of reduced features and not the number of tweets feed into the model. The total number of generated tweets is 1959 and total features extracted are 5283.

Table 1: Naïve Bayes Accuracy with stopwords

Naïve Bayes (with stop word)			
KBest	Unigram	Bi-Gram	Tri-Gram
10	0.5656	0.3548	0.3548
50	0.6598	0.5933	0.56 4
100	0.3031	0.6487	0.6173
500	0.6303	0.6783	0.6672
1000	0.756	0.719	0.682
1500	0.7726	0.7541	0.7171
2000	0.7744	0.6783	0.7301
2500	0.7597	0.695	0.7245
3000	0.7578	0.7264	0.7393
3500	0.7541	0.7615	0.5138
4000	0.756	0.7837	0.5415
4500	0.7504	0.7911	0.5508
5283	0.7948	0.7948	0.5637

Table 2: Naïve Bayes Accuracy without stopwords

Naïve Bayes (without stop word)			
KBest	Unigram	Bi-Gram	Tri-Gram
10	0.5656	0.3548	0.3548
50	0.6598	0.5933	0.5674
100	0.3031	0.6487	0.6173
500	0.6303	0.6783	0.6672
1000	0.756	0.719	0.682
1500	0.7726	0.7541	0.7171
2000	0.7744	0.6783	0.7301
2500	0.7597	0.695	0.7245
3000	0.7578	0.7264	0.7393
3500	0.7541	0.7615	0.5138
4000	0.756	0.7837	0.5415
4500	0.7504	0.7911	0.5508
5283	0.7948	0.7948	0.5637

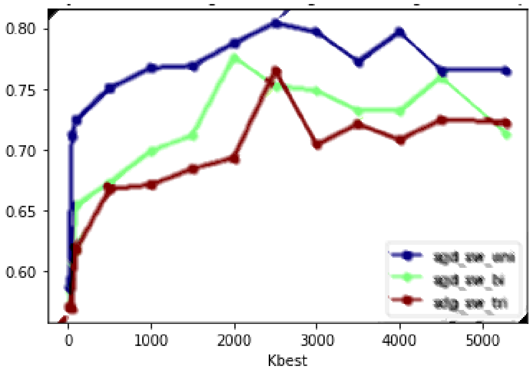


Accuracy chart NB (unigram vs bigram vs tri-gram) + stop word

Figure 2: Naïve Bayes Accuracy with stop word

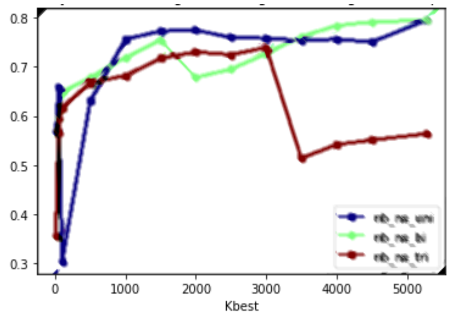
Table 3: SGD Accuracy with Stopwords

SGD Classifier (with stop word)			
KBest	Unigram	Bi-Gram	Tri-Gram
10	0.5859	0.5711	0.5693
50	0.7116	0.6062	0.5693
100	0.7245	0.6543	0.6173
500	0.7504	0.6728	0.6672
1000	0.767	0.6987	0.6709
1500	0.7689	0.7116	0.6839
2000	0.7874	0.7763	0.6931
2500	0.804	0.7523	0.7652
3000	0.7966	0.7486	0.7042
3500	0.7726	0.7319	0.7208
4000	0.7966	0.7319	0.7079
4500	0.7652	0.7597	0.7245
5283	0.7652	0.7134	0.7227



Accuracy chart SGD (unigram vs bigram vs tri-gram) + Stopword

Figure 4: SGD Accuracy with Stopwords

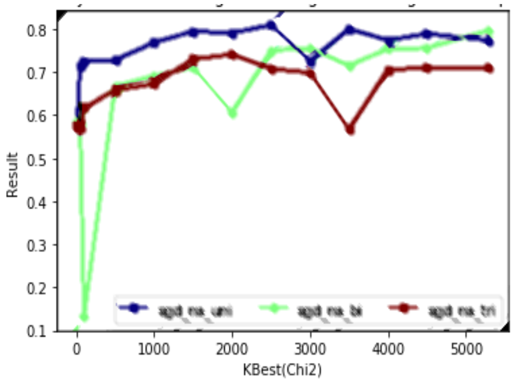


Accuracy chart NB (unigram vs bigram vs tri-gram) – stop word

Figure 3: Naïve Bayes Accuracy without stop word

Table 4: SGD Accuracy without Stopwords

SGD Classifier (without stop word)			
KBest	Unigram	Bi-Gram	Tri-Gram
10	0.5767	0.5841	0.5693
50	0.7153	0.5841	0.5693
100	0.7245	0.133	0.6173
500	0.7282	0.6672	0.658
1000	0.7689	0.6913	0.6728
1500	0.7948	0.7116	0.7301
2000	0.7911	0.6062	0.7412
2500	0.8096	0.7504	0.7079
3000	0.7264	0.756	0.6987
3500	0.8003	0.7153	0.5674
4000	0.7744	0.7541	0.706
4500	0.7892	0.756	0.7097
5283	0.7744	0.7966	0.7097



Accuracy chart SGD (unigram vs bigram vs tri-gram) Stopword

Figure 5: SGD Accuracy without Stopwords

Sentiment analysis on tweets was presented using Naïve Bayes and Stochastic Gradient Decent Classifiers models in which the effect of stopwords preprocessing were determined. The analysis of important features in classifying positive/negative sentiments were done with Chi-square as a feature extraction to dimensionality of the features. As the number of tweets increases, the accuracy of the N-grams (unigram, bigram and trigram) increases. Using Naive Bayes with stopwords and/or without stopwords are not affected, this implies that Naive Bayes classifier is not really affected with occurrences of stopwords when classifying the sentiments analysis. While, Stochastic Gradient Decent (SGD) classifier on tweets between unigram, bi-gram and tri-gram without stopwords are higher compared to those with stopwords. It shows that removing stopwords increase the accuracy of the classifier.

5.0 Conclusion and Future works.

The accuracy result of unigram for Naive Bayes both with stop word and without stop word is always the highest out of all the three (unigram, bi-gram & tri-gram) results with bi-gram sometimes overtaking unigram result in some of the feature reduction number used.Trigram provides us with the least results thereby; trigram is not suitable for sentiment analysis. In the future work, Neural Network models such as ANN, RNN using word embeddings (Document Vectors) can be used to classify Twitter sentiments.

References

[1] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R. (2011). Sentiment analysis of twitter data. In: Proceedings of the Workshop on Languages in Social Media, Association for Computational Linguistics, 30–38.

[2] Zainuddin, N., Selamat, A. (2014). Sentiment analysis using support vector machine. In: 2014 International Conference on Computer, Communications, and Control Technology (I4CT), IEEE, 333–337.

[3] Thelwall, M., Buckley, K., Paltoglou, G. (2011). Sentiment in twitter events. J. Am. Soc. Inform. Sci. Technol. 62(2), 406–418

[4] Ding, X., Liu, B., Yu, P.S. (2008). A holistic lexicon-based approach to opinion mining. In: Proceedings of the 2008 International Conference on Web Search and Data Mining, ACM. 231–240.

[5] Melville, P., Gryc, W., Lawrence, R.D. (2009). Sentiment analysis of blogs by combining lexical knowledge with text classification. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM. 1275–1284.

[6] Conover M. D., Ferrara E, Menczer F, Flammini A (2013) The Digital Evolution of Occupy Wall Street. PLoS ONE 8(5): e64679. doi:10.1371/ journal.pone.0064679. 1-5.

[7] Liu B., Blash E., Chen Y, G., Shen D. (2013). Scalable Sentiment Classification for Big Data Analysis Using Naïve Bayes Classifier, Journal of International Conference on Big Data (IEEE)., 99-104

[8] Liu B.(2012). Sentiment Analysis and Opinion Mining,", A Review Article on Synthesis Lectures on Human Language Technologies, 5(1): 1-167.

[9] Zhang, L., Jia, Y., Zhou, B., and Han, Y. (2012). Microblogging sentiment analysis using emotional vector. In Cloud and Green Computing (CGC), 2012 Second International Conference on IEEE. 430–433.

[10] Asiaee T, A., Tepper, M., Banerjee, A., and Sapiro, G. (2012). If you are happy and you know it... tweet. In Proceedings of the 21st ACM international conference on Information and knowledge management, ACM 1602–1606.

[11] Hu, X., Tang, J., Gao, H., and Liu, H. (2013a). Unsupervised sentiment analysis with emotional signals. In Proceedings of the 22nd International Conference on World Wide Web,Steering Committee, 607–618.

[12] Martınez-C´amara, E., Montejo-R´aez, A., Martın-Valdivia, M., and Urena-L´opez, L. (2013). Sinai: Machine learning and emotion of the crowd for sentiment analysis in microblogs. In Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013), Atlanta, Georgia, USA.

[13] Fusilier, D.H., Montes-y Gomez, M., Rosso, P., Cabrera, R.G. (2015). Detecting positive and negative deceptive opinions using pu-learning. Inf. Process. Manage. 51(4):433–443.

[14] Yetisgen-Yildiz M. and Pratt W. (2005). The effect of feature representation on MEDLINE document classification, AMIA Annual Symposium Proceedings, 849-853.

[15] Turney, P., and Pantel, P. (2010). From Frequency to Meaning: Vector Space Models of Semantics, Journal of Artificial IntelligenceResearch 37: 141-188.

[16] Beil, F. Ester, M. Xu, X. (2002). Frequent term-based text clustering, Proc. of Int'l Conf. on knowledge Discovery and Data Mining, KDD' 02,. 436–442.

[17] Hutto, C. J. Gilbert,E. E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eighth International Conference on Welogs and Social Media (ICWSM-14) Ann Arbor, MI, June.

[18] Pang B. and Lee L. (2005).Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. Proc. 43rd Annu. Meet. Assoc. Comput. Linguist., 3(1): 115–124.

[19] Abbasi A., France S., Zhang Z., and Chen H. (2011). Selecting attributes for sentiment classification using feature relation networks, IEEE Trans. Knowl. Data Eng., 23(3): 447–462.

[20] Kim S. and Hovy E. (2006). Extracting Opinions , Opinion Holders and Topics Expressed in Online News Media Text, in ACL Workshop on Sentiment and Subjectivity in Text, 1–8.

[21] Tan L. K. W., Na J. C., Theng Y. L., and Chang K. (2011). Sentence-level sentiment polarity classification using a linguistic approach," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 7008 LNCS, 77–87.

[22] Di Fabbrizio G., Aker A., and Gaizauskas R. (2011). STARLET: Multi-document Summarization of Service and Product Reviews with Balanced Rating Distributions in 11th IEEE International Conference on Data Mining Workshops, 67–74.

[23] Wang, Y., and Wang X.J., (2005). A New Approach to feature selection in Text Classification", Proceedings of 4th International Conference on Machine Learning and Cybernetics, IEEE, 6: 3814-3819.

[24] Montanes, E., Ferandez, J., Diaz, I., Combarro, E.F and Ranilla, J. (2003). Measures of Rule Quality for Feature Selection in Text Categorization", 5th international Symposium on Intelligent data analysis , Germeny-2003, Springer-Verlag 2003, Vol2810, 589-598.

[25] Haddi, E., Liu, X., Shi, Y. (2013). The role of text pre-processing in sentiment analysis. Procedia Comput. Sci. 17, 26–32

[26] Uysal, A.K., Gunal, S. (2014). The impact of preprocessing on text classification. Information Processing Management. 50(1): 104–112

[27] Baccianella, S., Esuli,A. Sebastiani, F. (2010). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10), European, 2200-2204

[28] Saha, S. Yadav, J. Ranjan, P. (2017). Proposed Approach for Sarcasm Detection in Twitter. Indian Journal of Science Technology, 10(25): 1-8

[29] Shapire, R.. 2017. Machine Learning Algorithms for Classification. Princeton Computer Science. Department of Computer Science at Princeton Universitywww.cs.princeton.edu/~shapire, 1-50

[30] Pang, B., Lee, L., Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of the ACL-02 conference on Empirical methods in natural language processing, 10: 79–86.

[31] Tripathy, A., Agrawal, A., Rath, S.K. (2016). Classification of sentiment reviews using n-gram machine learning approach. Expert Syst. Appl. **57**: 117–126.

[32] Dave, K., Lawrence, S., Pennock, D.M. (2003). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In Proceedings of the 12th international conference on World Wide Web, 519–528.

[33] Khairnar, J., Kinikar, M. (2013). Machine learning algorithms for opinion mining and sentiment classification. International Journal of Scientific and Research Publications, 3(6), 1–6.

[34] Po-Wei L., Bi-Ru D.(2013). Opinion Mining on Social MediaData", IEEE 14th International Conference on Mobile Data Management,Milan, ISBN: 978-1-494673-6068-5, http://doi.ieeecomputersociety.org/10.1109/MDM, 91-96.

[36] Saif, H., He, Y., Alani, H. (2012)Alleviating data sparsity for twitter sentiment analysis. In: CEUR Workshop Proceedings (CEUR-WS. org).