TEST Engineering & Management

March - April 2020
ISSN: 0193-4120 Page No. 2049 - 2057

Sarcasm Detection on Tweets using Deep Learning

Prof. Rupali Amit Bagate
Dept. of Information Technology, Army Institute of Technology Pune,India rupali.bagate@gmail.com

Satyajit Singh
Dept. of Information Technology, Army Institute of Technology Pune,India
xinus404@gmail.com

Deepa Yadav
Dept. of Information Technology, Army Institute of Technology Pune,India
yadav.deepa1997@gmail.com

Swaraj Surendra Patel
Dept. of Information Technology, Army Institute of Technology Pune, India, swarajpatel001@gmail.com

Anshaj Upadhyay
Dept. of Information Technology, Army Institute of Technology Pune,India
anshajupadhyay1999@gmail.com

Article Info
Volume 83
Page Number: 2049 - 2057
Publication Issue:
March - April 2020

Article History
Article Received*: 24 July 2019*
Revised*: 12 September 2019*
Accepted*: 15 February 2020*
Publication*: 18 March 2020*

Abstract

Natural language processing is paving ways to finding meaning from various kind of themes of texts ranging from sentiment analysis to text based classification, often what happens in usual sentiment analysis is that the predicted sentiment might not be correct due to the presence of sarcasm in the text ,thus sarcasm detection is an important factor when producing insights from texts. Sarcasm is an important factor in today's world . Majority of the youth uses sarcasm in their texts to convey their feelings .The author has analyzed various methodologies used for the detection of sarcasm and their efficiency.

Keywords – *Sarcasm , Deep Learning , Tweets , Detection*

I. Introduction

Sarcasm in its own is a difficult thing to identify for humans also let alone machines. Sarcasm has the ability to flip the polarities of the predicted tweets. Author's work aims at identifying its presence in a given tweet accurately, for this the author has done research which is present in this paper. Sarcasm classification is a huge breakthrough in feedback mechanism improvement.

The classification of sarcasm can be done by using various artificial intelligence methods namely machine learning and deep learning, in this Work the author show how all those methods perform and evaluate different datasets for classification. Sarcasm is very common these days and is used in twitter a lot, so the author has focused on the tweets and used them to classify the tweets as sarcastic and non-sarcastic by identifying sarcasm in them. The relative work done in this field will also be discussed in the next section.

II. RELATED WORK

Various work that have been done in sarcasm classification using machine learning have been listed below –

March - April 2020
ISSN: 0193-4120 Page No. 2049 - 2057

Sarcasm Classification	Machine learning Approach				
	Supervised Learning				
	LM	*NB*	*SVM*	*T*	*E*
	LR/ME				*RF*
Gonzalez-Ibanez et al. [1]	✓		✓		
Bouazizi and Ohtsuki [2]	✓		✓		✓
Bammand and Smith [3]	✓				
Bouazizi and Ohtsuki [4]	✓	✓	✓		
Sulis et al. [5]	✓	✓	✓	✓	✓
Buschmeier et al. [6]	✓	✓	✓	✓	✓
Lunando &Purwarianti [7]	✓	✓	✓		
Ling and Klinger [8]	✓		✓	✓	
Weitzel et al. [9]	✓		✓		
Rajadesingan et al. [10]	✓		✓	✓	
Alcaide et al. [11]			✓		
Riloff et al.[12]			✓		
Reyes and Rosso [13]			✓	✓	
Reyes et al. [14]				✓	

TABLE I.

LR/ME: Logistic Regression/Maximum Entropy; Win: Winnow class.; LM: Linear Model; Banaue Bayes; SVM: Support Vector Machine;T: Tree-based; RF: Random Forest; E: Ensemble

III. DATA PREPROCESSING

To prepare the gathered data for use, it first has to be cleaned and processed . The main objective of data pre-processing is to reduce the vocabulary of terms used in the tweets. The data has to be processed before making it available for the classification phase. Transforming text into something that an algorithm can understand in order to work on it is a complicated process. There are three different parts:

a)Cleaning- It consists of removing those parts of the text which are less useful .It can be done by removing stop words , dealing with capitalization of characters and other details.

March - April 2020
ISSN: 0193-4120 Page No. 2049 - 2057

b)Normalization- It consists linguistic reduction of terms through Stemming or Lemmatization .

c)Analysis- It consists of statistical inquiry , manipulation and generalization of the dataset.

A. *Tokenization*

The first step is tokenization. Tokenization [15] is performed on tweets to break them down into perfect meaningful modules from a sentence. It is one of the main features of lexically analyzing the text. Here the aggregation of the sequence of characters takes place. The text is divided into several parts known as tokens which may contain either words or other elements. Generally separators such as white spaces are used for identifying individual elements but other separators such as different punctuation marks can also be used. A token could be a complete group of words or a whole paragraph, but most frequently words are used as tokens.

ArkTweetNLP is a library which provides tools for working with twitter messages. Twitter tweets may contain symbols like “@” mentions, “#” hashtags , emojis , retweets and abbreviations. ArkTweetNLP provides tools to identify such symbols and treats them as separate tokens.

Example- using Hashtag-Tokenizer to separate the hashtags in the word #Sarcasticirony into #Sarcastic and #irony.

B. *Stop Words Elimination*

Stop words[16] are generally the most common words in a language. These can be considered as features which have high frequency and are present in almost every text document. Stop words may include some common features such as pronouns like ‘he’, ‘she’ and conjunctions like ‘or’ , ‘and’ , ‘but’ etc. Such features either do not have much effect or add very less or no value on the categorization process. A list of stop words or commonly repeated features is maintained and a feature is removed whenever it matches any feature in the stop word list. In order to find the stop words the list of terms can be arranged by frequency. The terms which have high frequency can be picked according to their lack of semantics value. These terms should be removed from the text. Some words which occur rarely also need to be removed i.e. The words that occur only in very few documents.

C. *Stemming*

Stemming [16] is built upon the idea that words with the same stem are close in meaning. Stemming is the process of reducing or replacing inflected words to their stem or roots. Hence, transforming the text into a more accurate form that is easily analyzable . The basic idea behind stemming is grouping the words with common or close meanings together and thus try to enhance the efficiency of Natural Language Processing. For example the words : depressing , depressed , depress ,depresses , depression etc. are having a common stem here which is ”depress” accompanied by the suffixes as ”ed” , ”ing” , ”es” etc. To increase the efficiency stemming can be used , however care must be taken when applying stemming ,since it can increase bias.

For example: Words ‘experiment’ and ‘experience’ are words with different meaning but they have the same stem word i.e exper . It is an example of over stemming.

Sometimes words which have the same meaning such as ‘adhere’ and ‘adhesion’ may remain distinct after stemming. It is an example of under stemming.

WEKA (Waikato Environment for Knowledge Analysis) is a package which can be used to perform stemming operation. WEKA contains implementation of a Snowball Stemmer and Lovins Stemmer.

It was discovered that stemming reduced the accuracy of the sentiment analysis, therefore it is avoided in the final implementation of the sentiment analysis algorithm.

March - April 2020
ISSN: 0193-4120 Page No. 2049 - 2057

D. *Lemmatization*

Lemmatization[17] is the process of converting the words in a sentence to its dictionary form. It is an alternative approach for stemming. Stemming usually follows a heuristic approach that discards the ends of words with the objective of removing inflection whereas lemmatization usually aims at removing inflection by making use of a vocabulary and morphological analysis of words. The words left after removing inflectional ends are termed as lemma. There are a number of both commercial as well as open source plug-in components which can be used for linguistic processing for stemming or lemmatization. Lemmatization can get better results by determining the part of speech of the words and utilizing WordNet's lexical database of English. For example:

TABLE II.

Word	*After Stemming*
Studies	Studi
Studying	Study

TABLE III.

Word	*After Lemmatization*
Studies	Study
Studying	Study

E. *N-Grams*

N-grams[16] is the group of co-occurring n-items in a sequence, which can be speech or text, used largely for NLP purpose. According to this approach in an n-gram the adjacent n words are grouped together. The main objective of n-grams is to capture the language structure from the statistical point of view i.e to determine what letter or word is likely to come next of the given word. This method can decrease bias, but increases the statistical sparseness. The n-grams method improves the quality of text classification, however there is no way finding the optimum length of the n-gram i.e. the size of n-gram to use. Example: "You should definitely go there" is a sentence in the form of text. Unigram, bigram and trigram are shown in Fig 1.

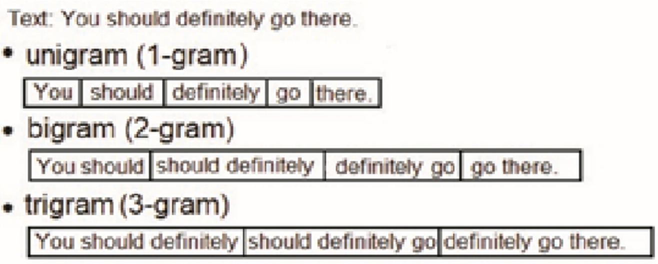


Fig. 1. N-grams example with 1,2,3 Gram

IV. CLASSIFICATION APPROACHES

A. *Machine Learning Approaches*

Machine Learning is mainly classified into two category – Supervised and Unsupervised learning.

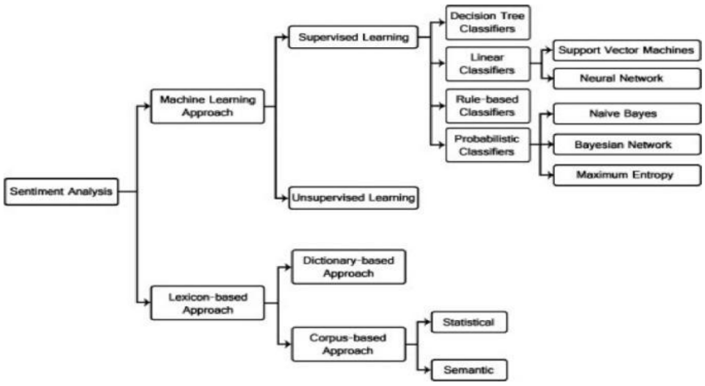


Fig. 2: Techniques used for sentiment analysis[18]

After the pre-processing of the dataset in above part , the classification of dataset using different Machine learning techniques have been carried out in this part. In supervised Learning the author mainly focused on –

1) *Logistic Regression:* It is used when target value in dataset is categorical. Before the Dataset is acted upon by a logistic function , it is fit into linear regression model . The selection of the type of logistic regression is based upon the number of different categorical value in dataset.

2) *Naïve Bayes:* It is a Probabilistic model that is being used in classification problems. To predict probability of features of the given dataset belonging to the particular class , it uses bayes theorem and naïve assumption.

The basic equation :-

March - April 2020
ISSN: 0193-4120 Page No. 2049 - 2057

P(B|A) = (P(A|B) * P(B)) / P(A)

3) Support Vector Machine: It is an algorithm in the field of machine learning used for both regression and classifications task. Commonly it is used for the task of classification. In this method hyperplanes are found which best differentiates two classes in which the data items are represented in n- dimensional space as a point and value of the particular co-ordinate represent the value of each feature.

B. *Deep Learning Approaches*

1) Convolutional Neural Network: Convolutional Neural Network is originally derives its idea from the same way the neural network is present in our brains. It consists of several layers which takes the data the input data and take their respective weights and sends them to further layers for processing .

The series of layers depending upon the requirement , are added in the form of neuron layers which process the inputs weights as is being associated , and then pass on to pooling , then to fully connected layer.

Later on the activation function are applied like Sigmoid ,tanh and ReLU[19].

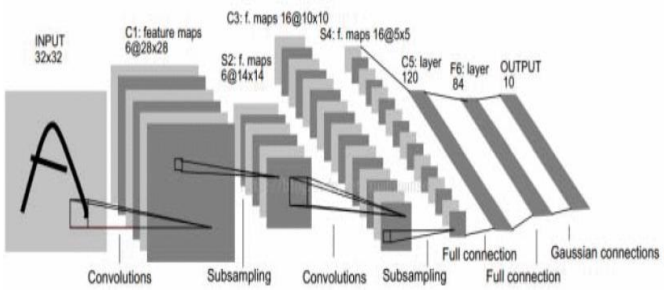


Fig. 3. The basic architecture proposed by Yen LeCunn [20]

Basically the Convolutional Neural Network is made of three main types of layers :

a) Convolutional Layer: One of the major and first steps in Convolutional Neural network is Convolutional Layer . Each input is being assigned weights , and in this layer only the first time features are being extracted from the input dataset so in our case , from all the images it extract the first predominant features. Feature maps are built in this layer as the features are being extracted from them and are processed .It learns the features by taking the input images in the form of small size matrices solving them , applying mathematical functions on them in which the small input matrix is multiplied with the filter matrix .

b) Pooling Layer: Pooling is one of the major parts of the cnn layers. It is used for reducing the feature maps and also it helps to increase the robustness of extracting features.It is basically of 3 types Max Pooling , Average Pooling and Sum pooling but most commonly used techniques are max and average pooling.The pooling layer is most commonly placed between the convolution layers.

c) Fully Connected Layer: Most commonly, Convolutional neural network is the classifier of layers which are fully-connected.They take into consideration all last layers neurons and connect them to each neuron of present layer.Fully connected layer do not have any spatialinformation. Since, softmax regression generates better outputs for probability distribution and hence used for classification.

2) Recurrent Neural Network: Recurrent Neural Network (RNN) is a from group of Neural Network(NN) in which the current step is input is taken from the output of the previous step. In conventional NN(Neural Networks) every output as well as input is independent .So RNN came into existence as there are some cases such as when for prediction of the next word in a sentence requires remembering of the previous words of the sentence.The one of the most precious feature of RNN is it’s Hidden state. It remembers the sequence information RNN have as “memory” which remembers all information about what has been calculated.

a) Long Short Term Memory Networks: It’s one of the types of RNN(Recurrent Neural

TEST Engineering & Management
March - April 2020
ISSN: 0193-4120 Page No. 2049 - 2057

Networks) that have a tendency to learn dependencies which are long-term. RNN(s) forms a reiteration chains of NN. RNN(s) have a reiteration of single tanh layer but LSTM have a very special module it has a four layer structure interacting in a very special way instead of a single tanh layer.

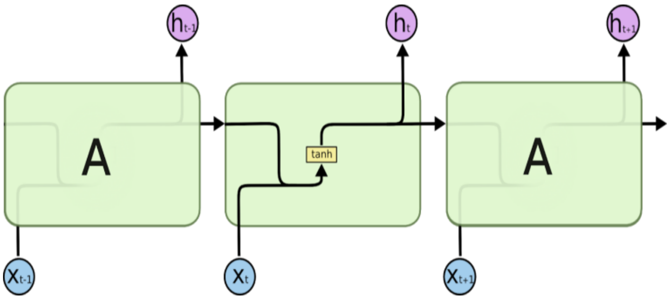


Fig.4. Example of LSTM structure with single layer

LSTMs also have this chain like structure, but the repeating module has a different structure. Instead of having a single neural network layer, there are four layers interacting in a very special way.

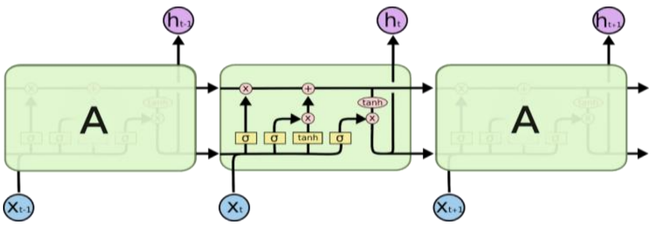


Fig. 5. Example of LSTM structure with single layer four interacting layers.

b) Gated Recurrent Unit: GRU is a special type of RNN(Recurrent Neural networks) shave a tendency to learn dependencies which are long-term. RNN(s) forms a reiteration chains of NN. RNN(s) have a reiteration of single tanh layer.

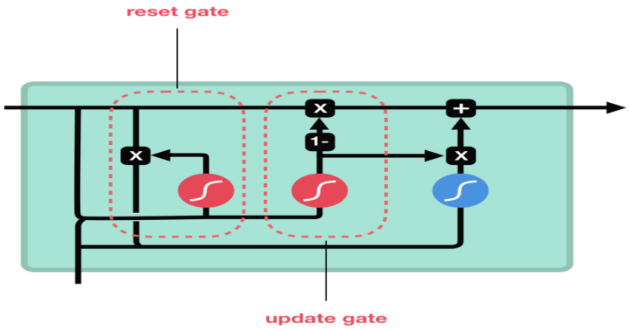


Fig.6.General architecture of GRU

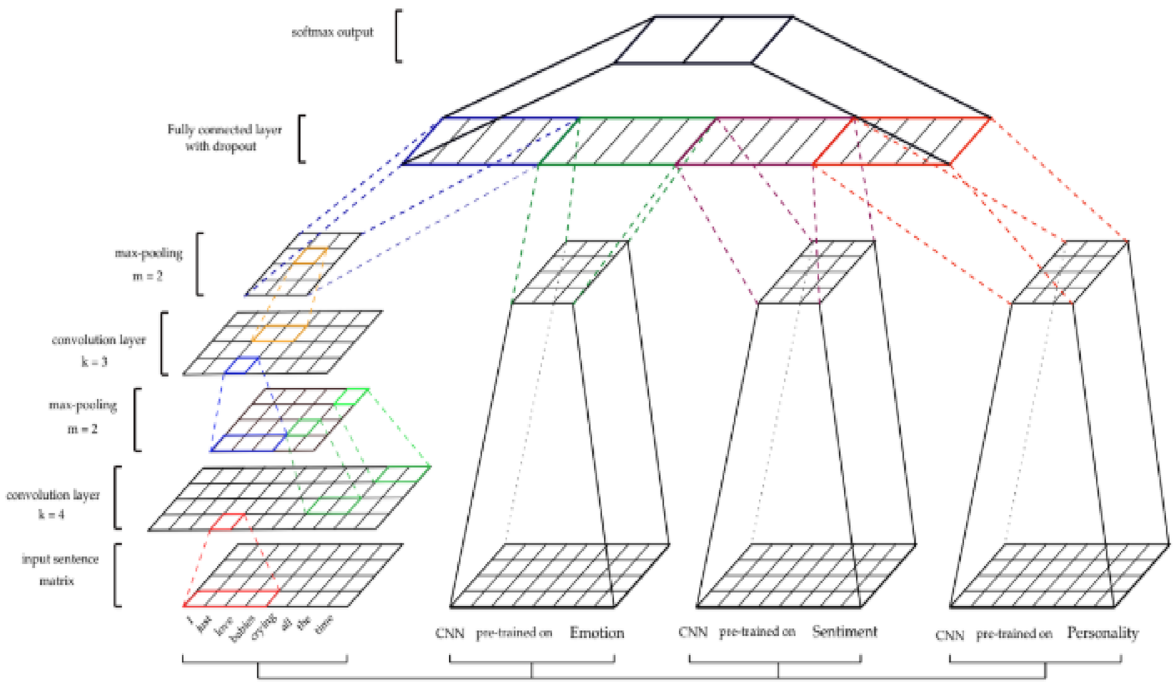
Update Gate

- This gate decides which information to keep and which to discard , similar to input and forget gate of LSTM.

Reset Gate

- This gate decides how much past information to forget.

3) Ensemble Methods: Soujanya Poria et al [21] in their work introduced a ensemble of CNNs to detect sarcasm , they used 4 models , one on Emotions , one on Sentiments , one of Personality and one on the text itself .Combining the fully connected layer of all four CNNs the final class (Sarcastic , Non-sarcastic) was predicted with up to an F-score of 90.70%



4) .

Fig.7. Ensemble method using various CNN models [21]

B	S	E	P	Dataset1		Dataset2		Dataset3	
				CNN	CNN-SVM	CNN	CNN-SVM	CNN	CNN-SVM
+				95.04	97.60	89.33	92.32	88.00	92.20
	+			-	87.00	-	86.50	-	73.50
		+		-	76.30	-	84.71	-	72.10
			+	-	75.00	-	77.90	-	74.41
	+	+	+	-	90.70	-	90.90	-	84.43
+	+			95.21	97.67	89.69	94.60	88.58	93.12
+		+		95.22	97.65	89.72	94.50	88.56	92.63
+			+	95.21	97.64	89.62	93.60	88.26	92.50
+	+	+	+	95.30	97.71	89.73	94.80	88.51	93.30

V. METHODS

Sarcasm detection is a complex task due to its ability to flip the polarity of a given sentence . Aniruddha Ghosh et al [22] in their work demonstrated that neural networks outperformed various other machine learning models in sarcasm detection . They studied the performance of various deep learning models like ANN, CNN,RNN, RecNN . They found LSTM implementation of RNN to be better and easier in sarcasm detection.

The author has studied various deep learning strategies , all of them showed that their results all depended on careful selection of hyper parameters .In the experiment with various models such as recursive SVM , ensemble of two CNNs ,LSTMs ,etc resulted in varied range of accuracy (evaluation metric : F-score , precision , recall).

TABLE IV.

Results on twitter dataset [22]

B - Baseline , S-Sentiments , E- Emotions , P-

March - April 2020
ISSN: 0193-4120 Page No. 2049 - 2057

Personality

VI. CONCLUSION

Detection of sarcasm accurately has a great potential in improving understanding of text processing, various works which have been studied have shown that their is a still a lot to research in this field . Deep learning approaches proved to be more efficient than their counterparts in classification of sarcasm in tweets. More improvements in the classification can be done by using various new methods which have been recently released such as extreme gradient boosting methods.

VII.REFERENCES

[1] R. Gonzalez-Ibanez, S. Muresan ve N. Wacholder, "Identifying sarcasm in Twitter: a closer look," in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Oregon, 2011.

[2] M. Bouazizi ve T. Ohtsuki, "A pattern-based approach for sarcasm detection on Twitter," IEEE Access, Volume 4, pp. 5477-5488, 2016.

[3] D. Bamman ve N. A. Smith, "Contextualized Sarcasm Detection on Twitter," in Ninth International AAAIConference on Web and Social Media, Oxford, 2015.

[4] M. Bouazizi ve T. Ohtsuki, "Opinion Mining in Twitter How to Make Use of Sarcasm to Enhance Sentiment Analysis," in Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining , Paris, 2015.

[5] E. Sulis, D. I. Hernandez Farias, P. Rosso, . G. Ruffo ve G. Ruffo, "Figurative Messages and Affect in Twitter: Differences Between #irony, #sarcasm and #not," Knowledge-Based Systems, Volume 108, p. 132–143, 2016.

[6] K. Buschmeier, P. Cimiano ve R. Klinger, "An Impact Analysis of Features in a Classification Approach to," in WASSA, Baltimore, 2014.

[7] E. Lunando ve A. Purwarianti, "Indonesian Social Media Sentiment Analysis With Sarcasm Detection," in 2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Kuta, 2013.

[8] J. Ling ve R. Klinger, "An Empirical, Quantitative Analysis of the Differences Between Sarcasm and Irony," in The Semantic Web, Crete, Springer, 2016, pp. 203- 216.

[9] L. Weitzel, R. C. Prati ve R. F. Aguiar, "The Comprehension of Figurative Language: What Is the Influence of Irony and Sarcasm on NLP Techniques?," in Sentiment Analysis and Ontology Engineering – An Environment of Computational Intelligence, Springer, 2016, pp. 49-74.\

[10] A. Rajadesingan, R. Zafarani ve H. Liu, "Sarcasm Detection on Twitter: A Behavioral Modeling Approach," in Proceedings of the Eighth ACM International Conference on Web Search and Data Mining , Shanghai, 2015.

[11] J. Alcaide, R. Justo ve . M. I. Torres, "Combining Statistical and Semantic Knowledge for Sarcasm Detection in Online Dialogues," in Pattern Recognition and Image Analysis, Yuri I. Zhuravlev, 2015, pp. 662

[12] E. Riloff, A. Qadir, P. Surve, L. De Silva, N. Gilbert ve R. Huang, "Sarcasm as contrast between a positive sentiment and negative situation," in *EMNLP 2013 - 2013 Conference on Empirical Methods in NaturalLanguage Processing, Proceedings of the Conference*, Seattle, 2013.

[13] A. Reyes ve P. Rosso , "Making objective decisions from subjective data: Detecting irony in customer reviews," Decision Support Systems , Volume 53 , no. 4, pp. 754- 760 , 2012 .

[14] A. Reyes, P. Rosso ve T. Veale, "A multidimensional approach for detecting irony in Twitter," Language Resources and Evaluation, Volume 47, no. 1, p. 239–268, 2013 .

[15] B.liu, Sentiment analysis and opinion mining SLHLT, vol. 5,no. 1, pp. 1-167,2012.

[16] Paras Dharwal ;Tanupriya Choudhary ;Rajat Mittal ;Praveen Kumar Automatic sarcasm detection using feature selection 23 Dec.2017

[17] Pu Han ; Si Shen ; Dongbo Wang ; Yanyun Liu The influence of word normalization in English document clustering 27 May 2012.

[18] Anandkumar D. Dave,, and Prof. Nikita P. Desai,. "A Comprehensive Study of Classification Techniques for Sarcasm Detection on Textual Data." International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) - 2016.

[19] T. Wang, D. Wu, A. Coates, A. Ng, "End-to-end text recognition with Convolutionalal neural networks", *International Conference on Pattern Recognition (ICPR)*, pp. 3304-3308, 2012.

March - April 2020
ISSN: 0193-4120 Page No. 2049 - 2057

[20] Y Lecun, L Bottou, Y Bengio et al., "Gradient-based learning applied to document recognition[J]", *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.

[21] Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Prateek Vij, "A Deeper Look into Sarcastic Tweets Using Deep Convolutional Neural Networks" Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers,pages 1601–1612, Osaka, Japan, December 11-17 2016.

[22] Aniruddha Ghosh, Tony Veale ,"Fracking Sarcasm using Neural Network" ,Proceedings of NAACL-HLT 2016, pages 161–169,San Diego, California, June 12-17, 2016.c 2016 Association for Computational Linguistics.

[23] Davidov Dmitry,Oren Tsur and AriRapport . "Semi-supervised recognition of sarcastic sentences in twitter and amazon." Proceedings of the Fourteenth Conference on Computational Natural Language Learning. Association for Computational Linguistics, 2010.

[24] González-Ibánez, Roberto, Smaranda Muresan, and Nina Wacholder. "Identifying sarcasm in Twitter: a closer look." Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2. Association for Computational Linguistics, 2011.

[25] Riloff, Ellen, et al. "Sarcasm as Contrast between a Positive Sentimentand Negative Situation." EMNLP. 2013.

[26] Rajadesingan, Ashwin, Reza Zafarani, and Huan Liu. "Sarcasm detection on Twitter: A behavioral modeling approach." Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. ACM, 2015.

[27] Lunando, Edwin, and Ayu Purwarianti. "Indonesian social media sentiment analysis with sarcasm detection." Advanced Computer Science and Information Systems (ICACSIS), 2013 International Conference on. IEEE, 2013.