International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Design and Analyze the Machine Learning Based Sarcasm Prediction Model for Social MediaContext

[1]B Rajani, [2]Dr.Sameer Saxena, [3]Dr. B Suresh Kumar

Submitted: 29/01/2023 **Accepted:** 01/04/2023

Abstract: Sentiment Analysis examines the predominance of sarcastic language and its difficulties detecting sentiment in the text. The identification of sarcasm in the text is the focus of automatic sarcasm detection. Sarcasm recognition has been more popular in recent years and has extensive usein sentiment analysis. In this paper, we proposed sarcasm detection using machine learning. In the first phase, data has collected from social media sources such as Twitter and other synthetic datasets. The policy-based data filtration technique is used for data pre-processing and generating the normalized data vectors. The different feature extraction and selection approaches have been used for hybrid feature selection, such as TF-IDF, NLP features, Dependency features and lexicon- based features from the entire context. The various machine learning classification algorithms have been used to predict positive, negative and neutral sarcasm. The WEKA 3.7 machine learning framework has been utilized for classification. As a result, SVM produces higher classification accuracy of 95.60% over the conventional machine learning classifier.

Keywords: *Social media context, supervised machine learning, feature extraction and selection, sarcasm detection, sentiment classification.*

1. Introduction

What is sarcasm?

The next definition comes from the Oxford dictionary and it is as follows: The use of words that ordinarily indicates the opposite in order to make a mockery of someone or show scorn for them is known as sarcasm.

The fundamental objective is to identify sarcasm in each and every aural communication that relates to the victimised aspect of an event. Here, rather than detecting a positive or negative sentiment, the primary focus is on catching the sarcasm, which means to know whether or not a piece of text similar to a social media post is being sarcastic or not. Detection of a positive or negative sentiment is merely a by-product of this process [1]. This topic has been developed with hurdles, most notably due to the fact that sarcasm, unlike other human emotions, is very difficult to identify by a computer algorithmic programme. This has led to the formation of this discipline. Since that time, there has been an increase in the popularity of informal threads on social media, which has led to an increase in demand for sarcasm detection in conversations among academics working in the field of natural language processing (NLP).

Because it was required for social platforms to analyse sarcasm in their posts and tweets, the significance of sarcasm analysis became the primary focus of attention. Sarcasm analysis has become more important in recent years. The detection of sarcasm is one of the focused analytic fields available in NLP. Language researchers have given sarcasm a lot of attention since it's such an interesting linguistic development. In more recent times, people have come to regard this scientific sector highly. As social media and sentiment analysis become more prevalent, information processing experts have been more interested in developing automatic sarcasm detection methods.

Text mining is broken up into a few different categories, one of which is sentiment analysis. Sentimental analysis is a sort of text mining that employs context to extract and identify subjective data from a phrase or text. The core concept behind this is to derive the meaning of the text via the use of several machine learning techniques. In the field of NLP, knowing people's subjective views may often be gleaned via the use of sentiment analysis. However, the findings of the research are also influenced by bias if people utilise sarcasm in the remarks that they make [2] [3]. Being able to recognise sarcasm is a necessary skill for accurately interpreting other people's motives and objectives in their actions.

Identifying sarcasm may be a challenging endeavour

1Research ScholarAmity University, Jaipur
rajanib@adc.edu.in
2Associate Professor Amity University, Jaipur
ssaxena1@jpr.amity.edu
3Associate Professor Sanjay Ghodawat University, Kolhapur
sureshkumarbillakurthi@gmail.com

since it is mostly determined by the surrounding circumstances, the information that came before it, and the manner in which the line was uttered or written. Being able to recognise sarcasm is a vital skill to have in order to correctly interpret other people's motives and intentions. The use of sarcasm detection in natural language processing applications may be beneficial to a number of different areas of interest, including market research, opinion mining, and data classification. However, detecting sarcasm is also a very difficult assignment, since it is mostly dependent on the surrounding context, one's past knowledge, and the manner in which the line was uttered or written. This makes it one of the more challenging aspects of the work. Sarcasm detection may be a specialised subfield within the realm of natural language processing (NLP) [2], a subset of sentiment analysis in which the primary emphasis is placed on sarcasm rather than the identification of any one emotion across the board. Therefore, the objective of this field is to determine whether or not a certain text contains sarcasm. Distinguishing between people's perspectives about a product, politics, service, or people delivers a tonne of benefits to businesses. This is the primary purpose of this article and its sarcasm detection technology.

It is critical to have the opportunity to differentiate between objective and subjective information. It contributes to the generation of organised knowledge, which is a component of the overall body of essential information for decision-support systems as well as individual decision-making. This project's goal is to analyse the emotion of the text, whether it be comments or reviews, with the advice of learning algorithms. This will help resolve the ambiguity of the meaning and increase the quality classification technique of a massive amount of user textual data available in social media. Increased precision in the identification of sarcasm. When compared to approaches that are rule- based or lexicon-based, approaches that are based on computer science have a better level of accuracy and more consistently deliver answers that are relevant because they take into account a greater number of additional elements. This may be the case since ML systems are able to take into consideration a far larger number of data points, as well as the most minute specifics of behaviour patterns associated with a particular account. Reduced amount of human labour needed for further verification. ML-driven systems filter out around 99.90% of conventional patterns, leaving just 0.1 percent of occurrences that need to be validated by subject matter experts. Lesser extent of the fallacy: A component's polarity establishes the direction of the communicated sentiment, which might be positive, negative, or neutral, depending on whether the user is thinking positively, negatively, or neutrally about the thing being considered. Capacity for recognising novel patterns and adjusting easily to shifting conditions. Unlike rule-based systems, machine learning algorithms adapt to the constantly changing economic and environmental variables. This allows them to outperform rule-based systems. They make it possible for analysts to recognise new suspicious patterns and develop new rules to prevent new kinds of sarcasm from being used.

Identifying sarcasm may be a challenging endeavour since it is mostly determined by the surrounding circumstances, the information that came before it, and the manner in which the line was uttered or written. Being able to recognise sarcasm is a vital skill to have in order to correctly interpret other people's motives and intentions. The use of context reasoning in natural language processing applications may be beneficial to a number of different areas of interest, including marketing strategy, collaborative filtering, and the classification of material. In order to identify sarcasm, this project will use NLP in conjunction with the most effective course of action to implement Hybrid Methodology of Classifiers [3]. The accuracy of sentiment analysis is one of the primaries focuses of this project, along with the development of a predictive model that will guarantee that the total forecast will be accurate. This project employs machine learning to analyse the emotion of text, opinions, and reviews. This contributes to the resolving of ambiguity in context and improves the overall sentiment analysis of a huge quantity of user textual data collected from social media.

Our study intends to test many different machine learning methodologies, such as support vector machines, artificial neural networks, and naive bayes, as well as lexical-based approaches, such as sentiment classification. Overall accuracy, preciseness, and recall are really the three metrics of assessment that will be used in this scenario. The following is a summary of the most important contributions that this research has made.

- To extract hybrid feature extraction and selection methods from large text and train themodule accordingly.
- To develop a binary as well machine learning classification algorithm to evaluate theperformance of various feature section methods.
- Finally predict the result as positive or negative with sarcasm or non-sarcasm using variousmachine learning classifies.

The rest of the paper described in Section 2 provides a summary of recent research, and section 3 describes proposed work with system architecture; section 4 discusses the algorithm design used for prosed methodology, while results and discussion have elaborated in section 5. Finally, conclusion and future work has demonstrated in section 6.

2. Literature Survey

Sentiment classification from text data is a critical problem in ABSA. As shown in Figure 1, context identification strategies are classified as frequency-based methods, NLP approach, Deep Learning approach, syntax-based methods, Unsupervised Machine Learning methods, Supervised Machine Learning methods, and hybrid methods.

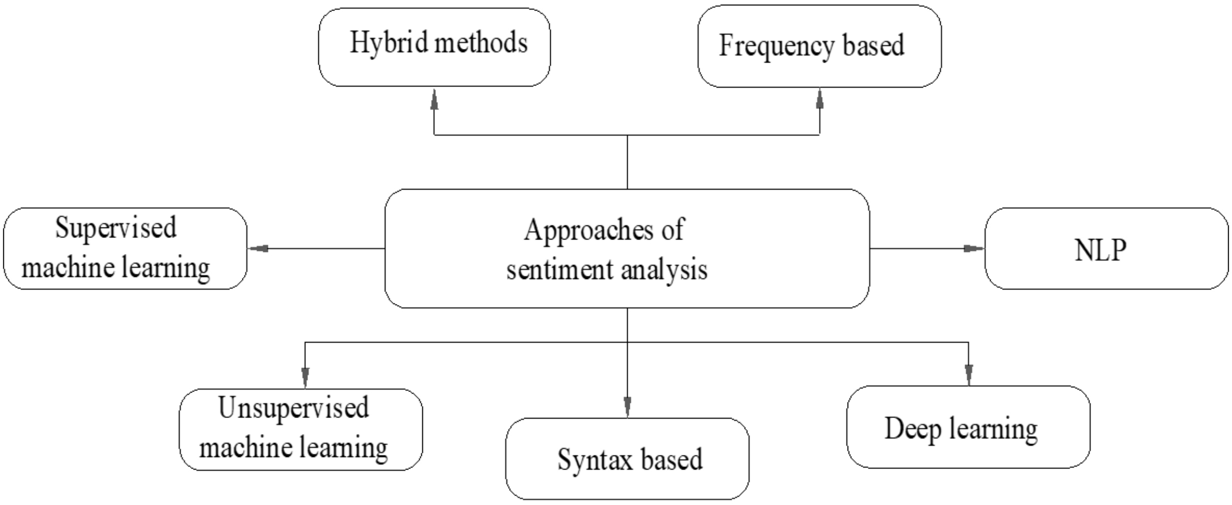

Figure 1: numerous techniques of sentiment detection and classificationFrequency Based approach

When classifying texts, factors such as frequently occurring words are taken into consideration [1]. These little word clusters are favoured over the complete dictionary since they are easier to understand. After the first processing, the attributes may then be selected according to their frequency. In some methodologies, the nouns, adjectives, and adverbs that make up a sentence are taken as features. Therefore, the characteristics that have a frequency of occurrence count that is higher than a certain threshold are selected.

NLP Approach

Within the realm of natural language processing, parallel advancements have happened in the area of pre-trained language models [2]. These breakthroughs include Embedding's from Language Model and Bidirectional Encoder Representation from Transformers (BERT). These language models have undergone pre-training on enormous quantities of unannotated text, and the application of these models has proven enhanced performance with far less labelled data and significantly quicker training. BERT is designed to take into account the context of a word on both the left and right sides at the same time. It is used to enhance performance in a range of NLP tasks such as ABSA.

Deep learning Approach

DL is an effective method for determining the feelings that are associated with the aspects. Finding the connections between different terms in a review is closely connected to the ABSA method. When considering the connections between words in a sequence, it is possible for certain DL systems, such as Recurrent Neural Networks (RNN), long short-term memory (LSTM), and Bi- LSTM, to do so effectively [3].

Syntax Based approach

The properties of syntactical links may be found using these techniques. The FB techniques search for aspect frequencies, while the SB methods search for features of syntactical interactions [4]. This is the primary distinction between the two types of approaches. The problem of aspect abstraction as well as the extension of the emotion lexicon may be solved using the method of twofold propagation. This system has the ability to uncover additional phrases that indicate a certain attitude by making use of the detected qualities. A series of criteria that are based on linguistic connections were developed in order to find the words that describe the emotion. This strategy needs just a limited corpus in order to accomplish what it sets out to do.

Supervised Machine learning approach

For the purposes of testing and validating models, machine learning (ML) methodologies make use of labelled data [5]. Every algorithm, before it can be utilised for SML, has to go through the training process.

In order to train these algorithms, a significant quantity of example data is necessary. Because machine learning techniques are feature-based, features are the major impetus behind supervised method development.

Unsupervised Machine learning approach

It is also known as cluster analysis, which is a type of machine learning that is used to discover previously unnoticed patterns in a dataset without the use of pre-labelled data [6]. This type of machine learning is used to find previously unnoticed patterns in a dataset using the method described above. The latent Dirichlet allocation is the technique that is used most often in UML methods for the purpose of aspect detection. In the sense that it makes use of a latent layer to encode word and text semantic connections, it is indistinguishable from the PLSA. Both LDA and LSA are capable of being used for the same purpose; the key difference between the two lies in the fact that LDA makes use of the Dirichlet prior to the subject delivery, whilst LSA makes use of a uniform topic delivery. During the process of modelling topics and documents, LDA makes use of amethod called bag-of-words to establish a direct connection between the names of subjects and the qualities or entities associated with those names.

Muhammad Zubair Asghar et. al. [7] proposed a system Aspect-based opinion mining framework using heuristic patterns. The work proposed an integrated framework comprising of an extended set of heuristic patterns generated using POS tags for aspect extraction, a hybrid sentiment classification module with the additional support of intensifiers and negations, and a summary generator. The system obtained classification results with improved precision (0.85) when compared to the alternative methods available. This method is quite generalized and it can classify aspect-based opinions in multiple domains.

Kim Schouten et. al. [8] proposed a system Aspect Category Detection for Sentiment Analysis for supervised as well as unsupervised learning. In this work, the first method presented is an unsupervised method that applies association rule mining on co-occurrence frequency data obtainedfrom a corpus to find aspect categories. The second, supervised, method uses a rather straightforward co-occurrence method where the co-occurrence frequency between annotated aspect categories and both lemmas and dependencies is used to calculate conditional probabilities. If the maximum conditional probability is higher than the associated, trained, threshold, the corresponding aspect category is assigned to that sentence. The accuracy of the system is around 83% for a supervised method.

3. Proposed System Design

Sarcasm identification is a fascinating issue in sentiment classification. Because of the ambiguity and complexity of nature. Detecting sarcasm more effectively from the dataset is a time- consuming job. This difficulty is one of the reasons why academics are interested in sarcasm detection research. In our study strategy, data is gathered and a detailed depiction of a Learning Model is provided utilising different machine learning approaches for deciding on-line sarcastic identification, with the advantage of achieving robust and dependable outcomes. This section we demonstrate detail description of sarcasm detection using NLP and machine learning techniques.

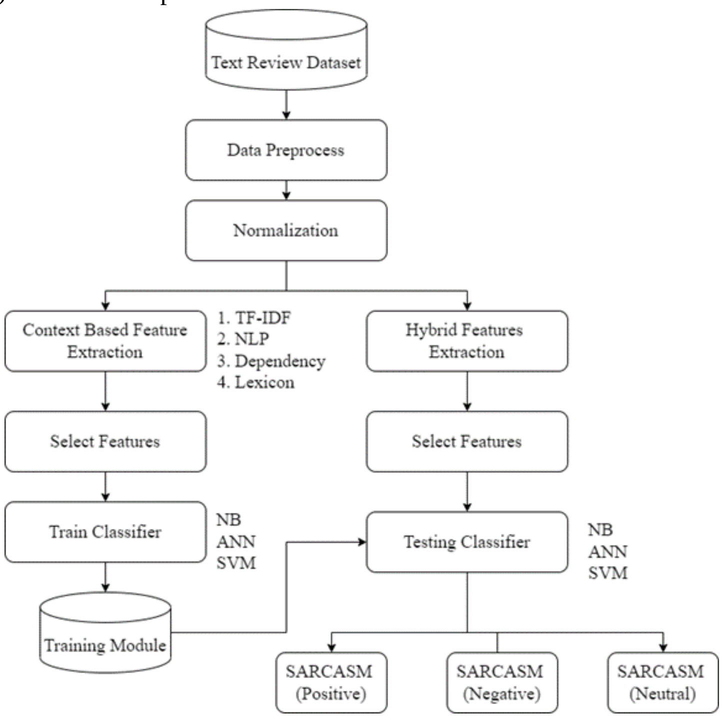

Figure 1: Proposed system architecture for sarcasm detection on social media text

The above figure 2 illustrates the proposed system architecture with overall execution, and it first deals with data pre-processing which contains sentence detection, tokenization, stopword removal etc. The standard stopword dictionary has been used, which is available on https://gist.github.com/larsyencken/1440509. Once the stopword removal has been done, it needs to execute feature extraction.

Porter Stemming and Lemmatization are two crucial feature extraction processes of pre-processing modules throughout appropriate feature extraction [2, 3]. The stemming method renovates all the affected words in the text into a root form called stem words. e. g., ‘nationalization,’ ‘nationalize,’ are each converted into the stem ‘national.’ Stemming gives a faster performance in applications where accuracy is not a significant issue. Lemma features include its base term and a converted term called stemming forms. For example, the terms “finishes”, “finished and “finishing” has “finish” as their lemma. Lemmatization words collection collectively different inflected structures of the word into a single one. Porter stemming eliminates such word inflexions only while required, and; Lemmatization generate substitute words with their respective base form. For example, the words “saying” and “says” are reduced to “say” in a steaming process, but lemmatization generates it to “said” and “say,” respectively; therefore, lemmas features are considered as further accurate than stemming.

The lemmas feature considers a complete feature extraction module, and it uses for important feature selection from the available n feature sets. The fundamental objective behind selecting respective features for the entire lemmas set is to eliminate unnecessary feature sets and remove the noise from the background knowledge dataset. The system proposed a hybrid weight calculation and correlation coefficient method to extract the best quality using document frequency. Given threshold value also contribute the removing the best parts only from existing terms. The proposed method used below algorithms to select the best quality using weighted matrix and correlation coefficient weight.

Support Vector Machine (SVM) :

A homogeneous partitioning of all data points is the outcome of using a support vector machine (SVM) to generate a maximum margin boundary. On each side of the margin, there is a sample of data that has been labelled as a support vector, and there is at least one support vector for each category of data. These support vectors are a reflection of the margin's borders, and they may be used to build the hyperplane that bisects the margin, as indicated in the equation below, as well as to build the hyperplane that bisects the margin in the equation below.

$$\vec{w}\,.\,\vec{x}\; + b\; = 0$$
$$y = mx\; + b$$

The hyperplane equation and the line equation are represented by the aforementioned equations. It is the job of a support vector machine (SVM) to locate weights for all of the sample data x in such a way that the data points are partitioned in accordance with a decision rule. The support vector machine (SVM) will execute a transformation into a new space if the sample points cannot be separated linearly. The use of a kernel function is what makes this possible. It is not necessary to have prior knowledge of the transformation into the new space in order for the kernel function to do the calculation of the dot product of two vectors in the new space. In text classification issues, SVMs performed much better than other classification approaches.

Naïve Bayes :

As can be seen in the equation below, one of the fundamental presumptions of the Bayes theorem is that the characteristics that make up a particular tuple are conditionally independent;

$$P(T|C_i) = \prod_{j=1}^{K} P\left(t_j|C_i\right)$$

Here, $P(t_j|C_i)$ is the number of tuples of class C_i and value t_j divided by the number of tuples of class C_i.

ALGORITHM DESIGN

To implement this work, we design a new modified deep learning-based convolutional neural network classifier called SVM. This algorithm is divided into two phases such as training and resting. The training module generates the rules for the entire module, while the testing phase validates disease detection and classification tests.

After that, randomly select a training sample (xit, yit) from the entire training set, where it 1,..., m is the target of the selected training sample at the tth iteration. First, assign a zero vector to the weight value W1, and then randomly select a training sample from the whole training set. The primary function of the objective is (1)

Secondly, calculate the gradient according to Formula (1), and then the distance can be expressedby

$$\nabla_t = \lambda W_t - \alpha_l$$

where $\alpha_t = \begin{cases} 1, & \text{if } y_{i_t}\langle W_t, x_{i_t}\rangle < 1 \\ 0, & \text{Otherwise} \end{cases}$.

The updated formula of matrix W is as follows.

(2)

where ηt = 1 (λt) . Then an updated weight matrix W based on Formulas (2) can be obtained by

$$W_{t+1} = \left(1 - \frac{1}{t}\right) W_t + y_{i_t} x_{i_t} \quad (3)$$

In practice, Formula (3) is used to find minima or maxima by iteration.

The implementation of proposed model for training and testing phase are described in detail inbelow section

Proposed SVM Classification Algorithm

Input: Weighted coefficient correlation matrix for each selected term M, Test dataset test_data

Output: class label prediction for each instance

Step 1: for each t in M

Step 2 : read all index M [i…n] values

Step 3 : if ($M[i] \geq 0.0$)

Newlable Convert label 1 for respective tMN[] Newlable[i……n]

End for

Step 4 : generate updated binary matrix MN[]

Step 5 : input and preprocess test_data

$$Lemmas[] = \sum_{k=0}^{n} input[k \dots . n].\{Stopword, Lemmitization\}$$

Step 6: for each (s in $Lemmas$)T[] s.split(tokens)

For each (term t in T)

$$f(x) = t \parallel \sum_{n=1}^{m}(MN[n].values) \text{ if(exist)}$$

Step 7 : calculate mean for each aspect category based on $f(x)$

Step 8 : calculate belief for all categories

Step 9 : Return highest belief category as aspect for test instance.End for

End for

4. Results and Discussions

The prosed implementation has been done Java 1.7 including NetBeans, with weka 3.7 machine learning framework. Large text twitter dataset has been used for evaluation with proposed ML classifiers. Almost 1200 instances are available with this dataset including 900 for training and 300 for testing of each. In the below section we describe a proposed system accuracy and efficiencywith various input parameters

According to Figure 3 we have evaluated our proposed system with two existing classification algorithms such as ANN and NB. The machine learning associated SVM provides 85.6% accuracy while ANN and NB maxhine learning-based classification algorithm provides 87.1% and 88.3% accuracy respectively. The proposed SVM produces 95.6% average accuracy on similar dataset. Therefore, SVM brings higher accuracy than other state-of-art algorithms

Table 1: Classification accuracy for sarcasm detection with various features extractionmethods with different machine learning classifiers

Methods / classifier	**ANN**	**NB**	**SVM**
TF-IDF	84.6	87.2	93.1
Dependency	83.2	88.0	93.8
Bigram	82.4	89.2	94.3
Lexicon	85.6	86.5	95.3
NLP	87.1	88.3	95.6

Table 2 and figure 3 describe the classification accuracy for sarcasm detection from social media text. The five feature extraction techniques have been demonstrated with some unique features, and three different machine learning classification algorithms have been shown for the performance evaluation. The SVM produces higher classification accuracy over other classification algorithms.

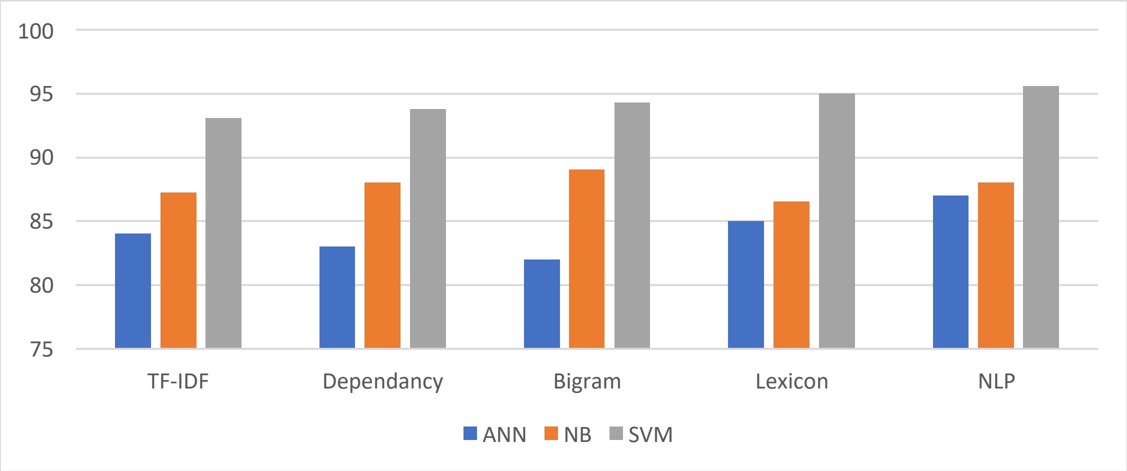

Fig. 3: Classification accuracy for sarcasm detection with various features extractionmethods with different machine learning classifiers

The presented methodologies cover the identification and prediction of sarcasm using machine learning. The first discusses a technique to feature selection using a hybrid feature selection technique. This indicates that eliminating duplication among chosen criteria improves classification performance. Then in selection phase selects attributes for task category prediction based on dependency linkages. In terms of classification performance, this technique reveals that features obtained using specific grammatical rules beat features retrieved using all rules. The hybrid feature set combines multivariate filter-based lemma characteristics with selected grammatical rule-based features. The use of a hybrid feature set improves the performance of real-world class and sentimentrecognition tasks.

5. Conclusions

This research describes sarcasm in tweets and other textual information using mixed feature extraction and machine learning approaches. We detailed these previous efforts using NLP bases, statistical techniques, and different learning methodologies. In summary, much research has been conducted on context reasoning; in particular, previous systems employed several characteristics to train respective sarcasm classifiers. Sarcasm is often detected by inconsistency in language. Still, a system may need to go beyond the content of the text to identify ironies, such as author data and tweet history. Such signals have aided in the improvement of sarcasm recognition systems' performance. In this module, we extract various features from input text data and evaluate them with multiple machine learning classifiers such as ANN, NB and SVM. The SVM produces higher accuracy of 95.60% on different cross-validations. In comparatives analysis, our model reduces higher precision over the other state-of-art systems. We also discussed current advancements in sarcasm detection, where researchers experimented with deep learning architectures and shown improved performance compared to statistical baseline techniques.

References

[1] Menaria HK, Nagar P, Patel M. Tweet sentiment classification by semantic and frequency base features using hybrid classifier. InFirst International Conference on Sustainable Technologies for Computational Intelligence 2020 (pp. 107-123). Springer, Singapore.

[2] Basiri ME, Abdar M, Cifci MA, Nemati S, Acharya UR. A novel method for sentiment classification of drug reviews using fusion of deep and machine learning techniques. Knowledge- Based Systems. 2020 Jun 21;198:105949.

[3] Jelodar H, Wang Y, Orji R, Huang S. Deep sentiment classification and topic discovery on novel coronavirus or COVID-19 online discussions: NLP using LSTM recurrent neural network approach. IEEE Journal of Biomedical and Health Informatics. 2020 Jun 9;24(10):2733-42.

[4] Bai X, Liu P, Zhang Y. Investigating typed syntactic dependencies for targeted sentiment classification using graph attention neural network. IEEE/ACM Transactions on Audio, Speech, and Language Processing. 2020 Dec 2;29:503-14.

[5] Pathak AR, Agarwal B, Pandey M, Rautaray S. Application of deep learning approaches for sentiment analysis. InDeep learning-based approaches for sentiment analysis 2020 (pp. 1-31). Springer, Singapore.

[6] Vashishtha S, Susan S. Fuzzy interpretation of word polarity scores for unsupervised sentiment analysis. In2020 11th international conference on computing, communication and networking technologies (ICCCNT) 2020 Jul 1 (pp. 1-6). IEEE.

[7] Asghar MZ, Khan A, Zahra SR, Ahmad S, Kundi FM. Aspect-based opinion mining frameworkusing heuristic patterns. Cluster Computing. 2017:1-9.

[8] Schouten K, Van Der Weijde O, Frasincar F, Dekker R. Supervised and unsupervised aspect category detection for sentiment analysis with co-occurrence data. IEEE transactions on cybernetics. 2017 Apr 14;48(4):1263-75.