International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 2s
DOI: https://doi.org/10.17762/ijritcc.v10i2s.5942

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

Sarcasm Detection on Text for Political Domain— An Explainable Approach

Rupali Amit Bagate^{12}, Ramadass Suguna^{3}
^{13}Department of Computer Science & Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology Chennai, India;
^{2}Department of Information Technology, Army Institute of Technology, Dighi, Pune
vtd537@veltech.edu.in^{12}; drrsuguna@veltech.edu.in^{3}

Abstract: In the era of social media, a large volume of data is generated by applications such as the industrial internet of things, IoT, Facebook, Twitter, and individual usage. Artificial intelligence and big data tools plays an important role in devising mechanisms for handling this vast volume of data as per the required usage of data to form important information from this unstructured data. When the data is publicly available on the internet and social media, it is imperative to treat the data carefully to respect the sentiments of the individuals. In this paper, the authors have attempted to solve three problems for treating the data using AI and data science tools, weighted statistical methods, and explainability of sarcastic comments. The first objective of this research study is sarcasm detection, and the next objective is to apply it to a domain-specific political Reddit dataset. Moreover, the last is to predict sarcastic words using counterfactual explainability. The textare extracted from the self-annotated Reddit corpus dataset containing 533 million comments written in English language, where 1.3 million comments are sarcastic. The sarcasm detection based model uses a weighted average approach and deep learning models to extract information and provide the required output in terms of content classification. Identifying sarcasm from a sentence is very challenging when the sentence has content that flips the polarity of positive sentiment into negative sentiment. This cumbersome task can be achieved with artificial intelligenceand machine learningalgorithms that train the machine and assist in classifying the required content from the sentences to keep the social media posts acceptable to society. There should be a mechanism to determine the extent to which the model's prediction could be relied upon. Therefore, the explination of the prediction is essential. We studied the methods and developed a model for detecting sarcasm and explaining the prediction. Therefore, the sarcasm detection model with explainability assists in identifying the sarcasmfrom the reddit post and its sentiment score to classify given textcorrectly. The F1-score of 75.75% for sarcasm and 80% for the explainability model proves the robustness of the proposed model.

Keywords: Domain Tweets; Sarcasm; Social Media; Artificial Intelligence; Machine Learning; Deep Learning; Big Data; Explainable AI

I. Introduction

In today's web 2.0, people face many challenges because of technological advancements in different sectors, such as social media platforms, the stock market, the industrial internet of things, and e-commerce. These platforms generate a lot of data every second in decentralized world. This is leading technology from web 2.0 to web 3.0. Many technologies are being used, such as Artificial Intelligence(AI) research, Blockchain, and industrial internet of things(IIOT) sensor data applications. In this era of big data, people need to tackle the five v's— volume, velocity, variety, veracity, and value. Our research deals with all v's. such as dataset we are referring reddit platform where volume of data with greater velocity generated every second with variety and varacity of data present. As problem statement defines author tries to detect a sarcasm from text which is adding value for particular domain. Therefore we use various tools and technologies such as AI which is combination of ML and DL algorithms is helping us deal with volume, velocity, veracity, variety, and value.

The different application generates a variety of data in terms of processing capability. E.g., IIOT, data is generated from different sensors used in different applications. So, in IIOT challenging task is to deal with a variety of data and the volume present. Many applications, such as health care using internet of things(IOT) [1], also incorporate artificial intelligence and data science to provide efficient solutions and security to predict in advance to solve intradisciplinary problems. Whereas in applications such as natural language processing in big data where many social sites generate the bulk amount of data with variety and veracity. In such situations, Deep Learning (DL) and Machine Learning (ML) algorithms play an important role. The big data and data science approach is used in this research to work on data volume, variety, and veracity to deliver value to the end user. Therefore, AI plays a vital role in predicting the correct outcome for better value. In addition to this Explainable AI is playing an important role to prove end users how machine-predicted results are correct and justifiable. Explainable AI (XAI) is a perfect solution to justify the result generated in the era of machines.

Sarcasm detection is study of natural language processing where one has to study all aspects of natural language to identify the semantic of given sentence. The author in [2] proposed a model of sarcasm detection which is a combination of machine learning model, deep learning model, and BERT. The author worked on Twitter and Reddit conversation datasets with a combination of the above techniques. The methodology combined context response and response on both datasets using TfIdf and Doc2vec vectorizers. The author achieved an F1-score of 0.7222 on the Twitter dataset and 0.679 on the Reddit dataset. [3] Proposed a sentiment classification and quantification model where sarcasm detection derives features for improving sentiment analysis evaluation metrics. The author proposed an affect-cognition-sociolinguistics model to detect sarcasm, followed by an SVM classifier to classify sarcastic outcomes. As the study proves, sarcasm features contribute more to sentiment classification than word embedding vectors, NRC lexicon, n-gram words, and part of speech features. As seen in previous work, authors have worked on sentiment analysis and sarcasm detection using different ML and DL models along with many features.

The motivation behind choosing this idea as research topic is identifying implicit expression hidden behind the expressed emotions written in plain text. Such plain text looks very simple in nature, but intensions or emotions intended by user flips the polarity of sentence. Therefore sentiment analysis and sarcasm detection play an essential role in natural language processing. For example, if any comment is like “when something terrible happens: That is just what I needed today!Well, you must be thinking, what this is all about.” Do you believe this text is sarcastic? If so, why do you believe it is sarcastic? The words" terrible" and" needed" contribute to the text's sarcastic tone. So, that is all about proposed research trying to detect sarcasm with AI that can be explained with the help of counterfactual explanation. [4] have researched on similar lines. The author did a dialogue sarcasm detection applying XAI. Research used an ensemble approach and LIME and SHAP library to generate sarcasm explanations. Words identified to show the influence of it for sarcastic inclination. [5] have used a regional tweet dataset named Sarc-H. It is an Indian language dataset using word and emoji embedding for sarcasm detection. So as observed, many people use different languages for sarcasm detection to create value. We see now a day's increase in demand for the identification of sarcastic comments to create business value out of it. Businesses like e-commerce platforms and social media platforms sustain and be in completion. To achieve this, they must understand the market sentiments. Analyzing people's sentiments is a significant parameter now, affecting many sectors like the stock market and the economy. Analyzing sentiments is very challenging because of the sarcasm present in them. Thus, to solve such a challenge, sarcasm detection comes into the picture.

In today's world, there are a vast number of datasets available. Reddit plays a vital role in dataset contribution. Reddit political dataset was selected as the research topic. Reddit dataset is a platform where adults express their views with different emotions, such as violence, hate speech, and anger.

Identifying sarcasm from text without any support of tonal quality or image present is a very cumbersome task. Still, many sentences have explicit clues present to denote the presence of sarcasm, such as #sarcasm. This makes machine learning algorithms very easy to classify text or tweets. But often, such ML algorithm classification fails due to such type of training with clue present in it, which gives birth to this research work. The novelty of the proposed work is that author(s) have worked on texts where the input text present in the dataset is preprocessed for unique patterns. Every sentence has been processed to remove such clues. For example, text like—“ wow Monday blues !!!! #Sarcasm”. Here # sarcasm explicitly tells that the sentence is sarcastic, which is removed in preprocessing and stored in the training dataset. This makes the machine more resilient to tricky text during the training and testing phase. So, the author(s) did implicit sarcasm detection for the political domain dataset. As well as, the author(s) worked on domain-specific text (Political agenda texts) extracted from Reddit, which has rarely been researched by other researchers. In addition to this author is performing a task of sarcasm detection with additional features sentiments and emotions. Which is additional and best of the research work. On the contrary, the author(s) observed many sarcasm detection works have already been performed on general tweets and text concerning general topics by applying ML and DL algorithms. This kind of domain-specific work may contribute to gaining vital insights to increase the value of the business.

The paper is arranged into different sections. Section II contains previous work done in a similar area. It gives an idea of the work done in this field. Section III is about the dataset used for research work. Later, the methodology used for research work is shown in Section IV. Section V discusses the experimental setup and results. Lastly, Section VI concludes the work performed.

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

II. Literature Analysis

Strong emotions expressed with irony or mockery words are called sarcastic sentiments.The efficacy of polarity identification can be negatively impacted by sarcasm, which can change the polarity of a "seemingly positive" statement. Sarcasm detection is contributing to many fields such as NLP, marketing field, campaigning of political parties etc. Sarcasm detection is a very difficult task as compared to sentiment analysis, as it contains implicit sentiments. In sentiment analysis, most of the sentences are explicit in nature, whereas in sarcastic sentences, emotions are implicit in nature maximum times. Most of the time, while doing sarcasm detection, prior knowledge of historical comments or thoughts is very important, along with what intention the sentence has been written. Sarcasm detection has become very important nowadays for many domains. Nowadays, youth are expressing their thoughts and views on social media, which may affect the mentality/emotionality of other people. In a broader context, it may even affect the current trends. So, to analyze the current trends in advance from different social media platforms with correct predictions, sarcasm detection contributes a lot due to the fact that youth’s way of expression has become sarcastic in nature.

Systematic study always leads to good and innovative research. The aim of doing a systematic survey is to check what study has been done in the past in similar research areas. So, one can identify the gap between the past research and the upcoming problem statement. The literature survey is divided into two parts. Section 2.1 presents a survey about sarcasm detection techniques and methods, whereas Section 2.2 explains the literature survey of explainable AI in NLP.

2.1 Sarcasm Detection Survey

Parmar et al. [6] proposed an approach for sarcasm detection using Hadoop MapReduce programming. The author has used a hybrid approach combining lexical and hyperbole techniques to increase the different metrics parameters such as F1-score, precision, and accuracy. The Hadoop MapReduce architecture aims to improve the processing speed, as the author is working on real-time tweets, which needs a higher processing speed to work on a vast database. Cai et al. [7] implemented a multi-model system for sarcasm detection. Nowadays, people on Twitter post tweets along with images and videos. Therefore, the input of this model is a combination of text, image, and image attributes as a feature set for sarcasm detection. Usually, people have worked only on the text present on a tweet which is insufficient to grab a better accuracy. Therefore, hierarchical fusion multi-modal is proposed. Baziotis et al. [8] proposed a model which combines two different deep learning models called the ensemble approach. This model works on word and character levels without considering lexicons or handcrafted features. The model used a word2vec vectorizer followed by preprocessing and self-attention LSTM model for better results. Sarsam et al. [9] studied various machine learning algorithms and neural networks for sarcasm detection on Twitter. They also surveyed an extensive database search, categorized into two groups adaptive machine learning and modified machine learning algorithm. Also, a survey of commonly used machine learning techniques for sarcasm detection shows that SVM performs better on the Twitter dataset. Sometimes the authors have used SVM-CNN assembling along with handcrafted features such as part of speech, frequency, and lexical for better performance of sarcasm classification. Datasets and their description are explained in the next section. Razali et al. [10] have individually worked on handcrafted feature sets to detect sarcasm. The author has used CNN to extract the feature set further combined with a handcrafted one. Logistic regression has worked better than other machine learning algorithms while predicting sarcasm. Sarsam et al. [11] have used SVM and CNN with a combination of different lexical feature sets to improve prediction accuracy. The author has used part of speech, pragmatic features, and frequency of words as feature sets to identify the sarcasm in a better way. Table 1 summarizes more research work elaborating on different features and methodologies used by different authors with different datasets and methods.Sable et al. [36][37]have used SVM and CNN with a combination of various feature sets to improve text mining accuracy.

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

Table 1. Summary of Different Methods

Domain	Method	Features	Datasets
Sentiment and Social Media Analysis [12]	Logistic regression was used to find different classifiers. Dictionary has been used for defining the polarity of words and ML library scikit and support vector machine with linear kernel.	The difference between the star rating and the overall polarity of the review was found as a feature found for improving results.	The dataset by Filatova had 1,254 Amazon reviews containing 437 ironic and 817 non-ironic reviews.
Natural Language Processing [13]	Context incongruity has been captured using cousin similarity and used this along with a word embedding-based classifier to attain better results—used SVM perf to optimize the F score.	The semantic similarity and discordance between word embeddings have been found out for improving results.	The dataset contained 3629 englishquotes from GoodReads containing 759 quotes labelled as sarcastic.
Natural Language Processing [14]	A convolutional neural network has been used to classify sentiment-specific features, and SVM is used for final classification.	The shifting of sentiments in the text indicated the presence of sarcasm.	Three datasets have been used: A balanced dataset from Twitter with 50K sarcasticenglish tweets and 50K non-sarcastic tweets. Imbalanced dataset, containing 25K sarcastic and 75K non-sarcastic tweets, and Test dataset, obtained from Sarcasm Detector containing a total of 120K tweets with 20K sarcastic and 100K non-sarcastic tweets.
Natural Language Processing [15]	A neural network semantic model for detecting sarcasm using CNN with LSTM is followed by DNN. Semantic modelling was reviewed using SVM.	The author has used the Stanford constituency parser to parse the tweets, and sarcasm has been detected using CNN followed by DNN.	The dataset contained 39,000 englishtweets containing 21,000 non-sarcastic tweets, and 18,000 sarcastic tweets, two different publicly available sarcasm datasets were also used.
Natural Language Processing [16]	Binary logistic regression has been used with l2 regularization using tenfold cross-validation and splits among authors to avoid redundant tweets that result in tempered results. Extra features have been used based on extra-linguistic information such as details about the author and audience.	In addition to lexical clues and their corresponding sentiments as predictive features, features based on extra-linguistic information were used, such as details about the author and audience.	The dataset used is a Gardenhose sample of tweets from August 2013 to July 2014.

As studied in different surveys, many authors have referred to machine learning and deep learning approach with feature sets like handcrafted features, emojis, multi-model system, part of speech, lexical, and n-gram word combination. Therefore, our research tried to incorporate top things that can enhance sarcasm detection and improve accuracy. The following section discusses the literature survey on Explainable AI's role in sarcasm detection.

2.2 Explainable AI Survey

A framework that helps in the prediction of machine learning algorithm outcomes to enhance their performance and helps data scientist to recognize the behaviour of different ML and DL models. Explainable AI (XAI) helps the system to build trust among the data science community and users of AI products, which in turn makes believe in predictions of ML and DL outputs. As literature survey performed in the next paragraph explains how different authors used different methods during their research. To implement the concept of XAI, various types of techniques are available such as LIME, Shaply etc. which are discussed in the next paragraph.

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

Jacovi et al. [17] computed the scores reflecting the n-grams activating convolution filters in natural language processing (NLP). In this, he demonstrates the output that LIME will provide as the explanation for the choice of " cat " classification and a kind of heatmap that shows the contribution of pixels to the segmentation result. More formally, given that model f predicts $y = f(x)$ for input x, for some metric v, typically, a large magnitude of v(xi) indicates that component x_i is a significant reason for the output y. Bodria et al. [18] computed the explainability on Rotten Tomatoes Movie Review Dataset. Furthermore, they analyze how attention-based techniques can be exploited to extract meaningful sentiment scores with a lower computational cost than existing XAI methods. Treating the model as a black box, LIME perturbs the instance and trains a local linear classifier. The weights of the linear interpretable classifier create the heatmap on the word that contributes the most. Messalas et al. [19] worked on the MNIST dataset and tried Shapley values to provide accurate explanations, as they assigned each feature an essential value for a particular prediction. Fiok et al. [20] worked on the Twitter dataset to predict the volume of responses to tweets posted by a single Twitter account. Their study used Shapley Additive explanations (SHAP) to demonstrate limitations in the research on explainable AI methods involving Deep Learning Language Modeling in NLP, and the models achieved F1 scores of 0.655. Malolan et al. [21], in their paper, detected Deep-Fake videos and trained a Convolutional Neural Network architecture on a database of extracted faces from FaceForensics' Deep Fake Detection Dataset. Their model was able to solve the issue. However, the only thing left was the explainability, so the author used LIME and LRP Explainable AI techniques to provide crisp visualizations of the salient regions of the image focused on by the model. Moreover, by using LRP author plots visual heatmaps and highlights the salient features of the images. Yang et al. [22] found that the explanations using many current methods for generating textual-based explanations result in highly implausible explanations, damaging a user's trust in the system. To address these issues, in his paper, he proposes a novel methodology for producing plausible counterfactual explanations while exploring the regularization benefits of adversarial training on language models in the domain of FinTech. Furthermore, he found that this approach not only improves the model accuracy compared to the current state of the art and human performance but also generates counterfactual explanations that are significantly more reasonable based on human trials.

As observed from the above literature survey, very few researchers have worked in the field of NLP applying the concept of XAI. They tried different techniques for explainability by using Shapley, Lime, and LRP. We surveyed the above method and decided to go with the counterfactual explanation methodology for XAI. The following section describes datasets used for research purposes.

III. Dataset

Datasets processed for sentiment analysis can be classified into different types depending on how many words and sentences are present [23]. Text considered for sentiment analysis or sarcasm detection is classified as short text, long text, or dialogues. The proposed research focuses short text. So, users try to express their sentiments, views, and perception in limited words. Therefore, this can be challenging for a working model to identify the semantics of content from a sentiment perspective.

Many researchers like Poria et al. [14] and Tay et al. [24] have worked on labelled datasets with sarcasm in tweets. Working on datasets with sarcasm is easy for prediction in the testing phase. Riloff et al. [25] and Ghosh et al. [26] worked on a labelled dataset with #sarcasm in tweets. Dataset has described in two columns; column no one describes labels as zero or one, and column no two describes tweets expressed by different users. In any supervised learning approach, labels play a crucial role. Labels indicate whether an expressed tweet is sarcastic or non-sarcastic. Zero indicates the tweet is non-sarcastic, and one indicates the tweet is sarcastic. These tweets are divided into training and testing parts, and the model is learned. Here supervised learning approach plays a critical role in enhancing the result of classification. Our proposed research has selected a domain-specific political dataset, a supervised approach with no explicit hashtag for model training purposes.

We conducted an experiment on a Self-Annotated Reddit Corpus (SARC) [27] containing 533M total comments which are short text in nature extracted from tree like structre in the form of comment and parent comment, of which 1.34M were sarcastic. We took the subset of a dataset from Kaggle and selected only the political dataset. To extract a subset of subreddit author have written a small python code snippet by applying filter for political text. Table 2. Shows the total no of columns present in the refereddataset. The shape for our dataset is (39496, 10); author have dropped the non-relevant columns and selected only label, comment, and parent comment by constructinga new dataframe. The new shape of the dataframe is (39248, 3) which is out our final dataset for research, of which 39.32% are non-sarcastic, and 60.68% are sarcastic, which is classified as balanced. Therefore, extracted no of columns is

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

listed in table 3, which is used further for training and testing. Poria et al. [14] worked on balanced, imbalanced, and test datasets with the same model, and comparing the result; the balanced dataset produced a better F1 score than the imbalanced dataset.

Table 2. Columns description of the original dataset

Position	Comment Type
1	parent_comment_Happy
2	parent_comment_Angry
3	parent_comment_Surprise
4	parent_comment_Sad
5	parent_comment_Fear
6	comment_Happy
7	comment_Angry
8	comment_Surprise
9	comment_Sad
10	comment_Fear

Table 3. Newly created dataset columns

Comment Position	Comment Type
1	Label
2	Comment
3	Parent comment

IV. Methodology

The proposed research uses a Weighted Average Approach to detect sarcasm from a tweet. This ensemble approach lets several models provide an estimate in proportion to their trust or predicted performance. Figure 1 shows a proposed architecture of research. As shown in the Figure research domain, political sarcasm detection takes an input dataset from Reddit having dimension (39496, 10). Further, the dataset is processed, trained, and tested in our black box proposed model. In the final stage, the model predicts whether or not the given testing sentence is classified as sarcastic. We have used two techniques, one for detecting sarcasm using: The Weighted Average Approach. A weighted average ensemble is an approach that allows multiple models to provide a prediction in proportion to their trust or estimated performance. Moreover, the second part or technique explains detected sarcastic comments: Counterfactual Explanations.

The author's contribution towards work is that author tried to work on domain-specific sarcasm detection without hashtag sarcasm. Very less work has targeted domain-specific sarcasm identification problems without #sarcasm. Also, the model is trained so that text without a clue of #sarcasm are identified with given accuracy by the model.

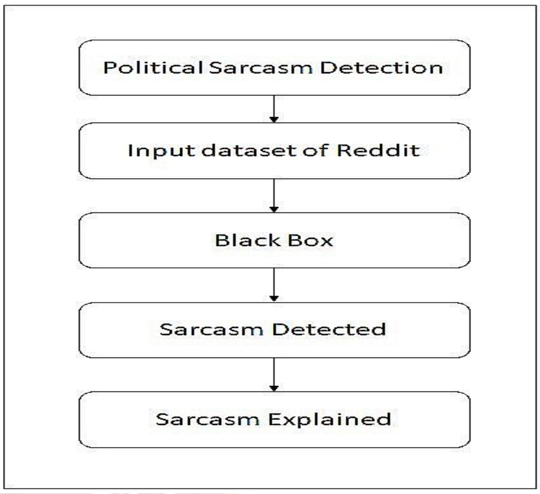

Figure 1. Sarcasm Detection with explainable AI

Here, the research work methodology is divided into three sections; section one talks about preprocessing datasets used by different researchers and explains why a specific dataset is selected for work. Section two explains different methods used for sarcasm detection and classification and how these methods are combined during the training and testing model. Section three describes explainable AI importance. Figure 1 shows the proposed architecture diagram of sarcasm detection with an explainable AI method.

Figure 1 describes the flow of the working model. In the beginning, input is given to the black box for political sarcasm detection. Inputs are reddit extracted text filtered for the political domain subject. Another Black box, a combination of various methodologies, processes this text for sarcasm detection. Once sarcasm is detected system tries to identify the combination of words that contributes to a sentence to be predicted as sarcastic. The other section describes every methodology in detail for better understanding.

4.1 Preprocessing

Data preprocessing is an essential step in any model development. We ran data preprocessing on a dataset having dimensions (39496, 3). Neatly prepared data always generates a good result after feeding to the model. As shown in Figure 2, data preprocessing combines steps such as dropping URL, tokenization, stop word removal, and lemmatization. Every step in the diagram plays a significant role in preparing a dataset for further processing.

As shown in Figure 2, every step plays a significant role in preparing a clean dataset for model training. A line of python code is written to clean the data as different models require. To provide a dataset as input to different models' data needs to be presented in vector format. Removing the

URLs, and irrelevant characters, converting all words to lowercase, and further tokenization of words is done. Tokenization will help the machine identify every individual word for stop word removal, stemming, lemmatization, and words having a length of less than two.

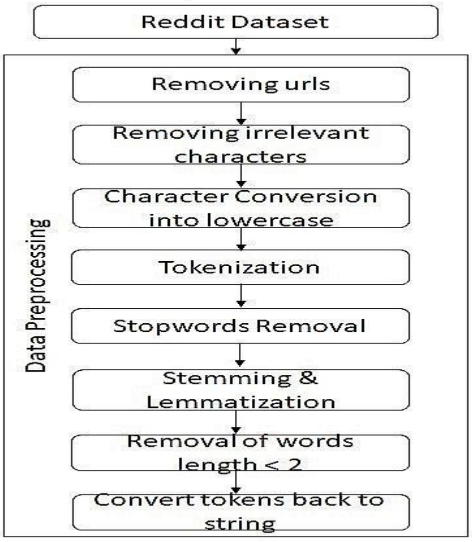
Reddit Dataset
Data Preprocessing
Removing urls
Removing irrelevant characters
Character Conversion into lowercase
Tokenization
Stopwords Removal
Stemming & Lemmatization
Removal of words length < 2
Convert tokens back to string

Figure 2. Dataset Preprocessing

a. *Removal of irrelevant character*

Removal of irrelevant characters like punctuations and numbers is also essential as there is no use for these characters, and they may insert noise in the model.

b. *Convert all characters into lowercase*

Since NLP (Natural language processing) is case sensitive, we convert the whole corpus into lowercase. Table 4 shows the converted output.

Table 4. Conversion of words to Lowercase

Raw	Lowercased
Canada canadA CANADA	canada

c. *Removing Stopwords*

Stopwords in any natural language are the most common words. Stopwords could not bring any value to the meaning. So, stopwords can be removed to analyze text data and create NLP models. Table 5 shows the procedure.

Table 5. Removing Stopwords

Sample text with stop words	Without stop words
Can listening be exhaustive?	listening, exhaustive
I like reading so i read	like, reading, read

d. *Stemming and Lemmatization*

Stemming and lemmatization establish the root shape of the terms inflected. The distinction is that stemmed does not contain actual words, while lemmatization translates the phrase into the actual dictionary word. So, the dataset cleaning requirement needs lemmatization. Figures 3 show the stemming and lemmatization process.

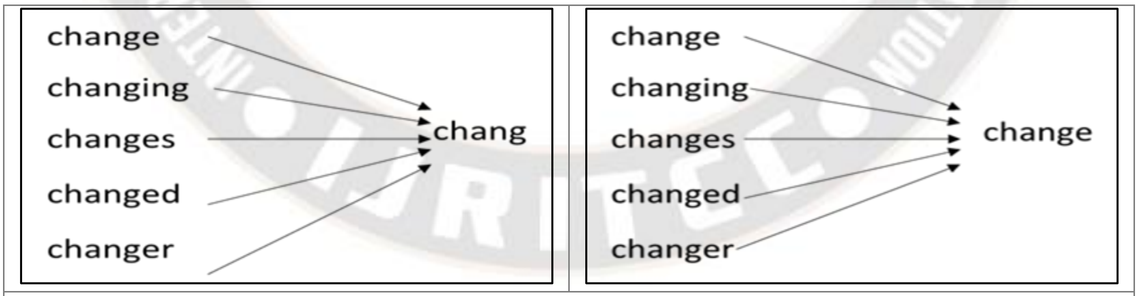

Figure 3. (a) Stemming (b) Lemmatization

The most significant benefit of lemmatization is that the sense of the word is taken into account in determining the meaning the consumer seeks. This mechanism enables noise to be reduced and consumer tasks accelerated. Therefore, in our code, lemmatization is used rather than stemming.

e. *Converting string from tokens*

After performing all the preprocessing techniques next step, the string is formed from a list of tokens. This preprocessed dataset precisely works with a model for improved results.

After the dataset is cleaned, it is provided to different machine learning and deep learning models. The following section describes the methodology.

4.2 Methods

Figure 4 shows a black box of the proposed

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

architecture. As described in architecture, the dataset consists of parent comments and preprocessed comments, as described in the earlier section. The output of data preprocessing is given to text2emotion and Vader sentiment library for feature extraction. Several emotions extracted are happy, angry, surprised, sad, and fearful from parent comments and comments. Along with this, negative, positive, and neutral sentiments are also extracted for comments and parent comments. These sets of features are fed to support the vector classifier. Parallel to this, data is also trained on a neural network. The nature and working of LSTM considers entire paragraph together for predication. So no need to provide parent comment explicitly. LSTM neural network extracts feautres automatically and train the network accordingly. But for SVC extracted features needs to be fed explicitly along with text formats that is comment and parent comment for sarcasm classification. After the model is trained on a neural network and SVC, the weighted average approach is used to improve the classification result. Table 6 shows the LSTM neural network architecture details used during traning.

Table 6. LSTM Model Architecture

Layer(type)	Output Shape	
		Param#
embedding(Embedding)	(None,385,50)	3468300
dropout(Dropout)	(None,385,50)	0
conv1d(Conv1D)	(None,381,64)	16064
maxpooling1d(MaxPooling1D)	(None,95,64)	0
lstm(LSTM)	(None,100)	66000
dense(Dense)	(None,1)	101

* Totalparams: 3,550,465
Trainableparams: 3,550,465
Non-trainable params: 0

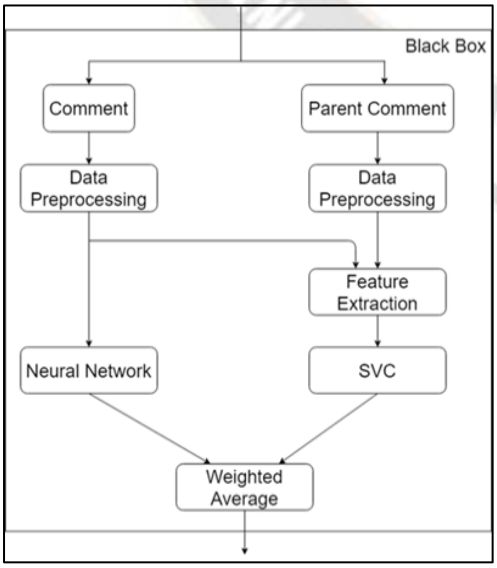

Figure 4. Black Box of Sarcasm Detection

Neural Network: In the working model, a neural network is embedded with different layers, SGD optimizer, metric and loss functions. Advanced neural network technique is used along with LSTM with 100 neurons and achieved testing F1-score of 74.95%. Table 7 shows a detailed sequential model architecture as in [28].

Support Vector Classifier: The author extracted emotion and sentiment as a feature from both comment and parent comment. Then trained extensively on different classifiers and achieved an F1-score of 74.56% on the Support Vector Classifier. We used pre-trained models text2emotion [29, 30] for feature extraction. This has given us features such as emotions (Happy, Angry, Surprise, Sad, Fear) and Polarity of Text (Negative, Neutral, Positive, Compound), respectively. After applying this to comment and parent comment total of 18 features are used for model training.

Table 7.NN Model Architecture [28]

Layer(type)	Output Shape	Param#
embedding(Embedding)	(None,106,50)	997750
dropout(Dropout)	(None,106,50)	0
conv1d(Conv1D)	(None,102,64)	16064
maxpooling1d(MaxPooling1D)	(None,25,64)	0
lstm(LSTM)	(None,100)	66000
dense(Dense)	(None,1)	101

* Totalparams: 1,079,915
Trainableparams: 1,079,915
Non-trainableparams:0

Fusion: –Each model generates a matrix that contains the probabilities of whether the text is sarcastic or not. Luintel[31], [32] calculates the weighted average of both outputs to get the improved final result for classification. Further classifying and identifying the words contributing to sarcasm are identified using XAI techniques.

4.3 Explainable AI

Many techniques are available to explain Black box models people have used; each has its pros and cons. We have chosen counterfactual explanations after doing an extensive survey. A counterfactual explanation explains causal situations. Thus, the prediction of individual instances is described by counterfactual explanation in machine learning and deep learning interpretability. For example, I will take one sample, such as, if I would not have gone into the rain, it is not why I can be wet. Here depends on the upcoming situation next event occurs. It is easy to implement and gives us all possible explanations. Humans are so clever as they can think counterfactually. Therefore, on top of that, humans can make up their minds and come to conclusions. To make the machine think as a human does, a counterfactual explanation is tried in our model as a part of

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 2s
DOI: https://doi.org/10.17762/ijritcc.v10i2s.5942
***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

the research.

V. Experimental Results

5.1 Sarcasm Detection

As shown in Figure 5, the proposed model is trained and tested with different machine learning and Deep learning models. The baseline model is trained with comment and parent comments from the Reddit dataset and applied to different machine learning and Deep Learning models such as Glove, CNN, and LSTM with dense layers. Figure 6a shows the F1-score of the baseline model, where Figure 6b shows the accuracy of the baseline model. Followed by Figure 7a and 7b show recall and precision, respectively. Figure 8 shows the baseline validation and training loss graph, as the observed loss is negligible, proving that the model's training is done accurately, which results in no overfitting of data. As a result of model training and testing, the F1-score of the baseline model is 70.19%. From all graphical notations, we conclude that training and testing measuring metrics generate a significant number, resulting in good training on data. Table 8 shows the confusion matrix of the baseline model. A true positive and true negative value is higher than a false positive and false negative. This shows the percentage of value prediction as true are actual true is more. Thus, it increases the value of precision and recall. Which automatically enhances the F1-score of the model.

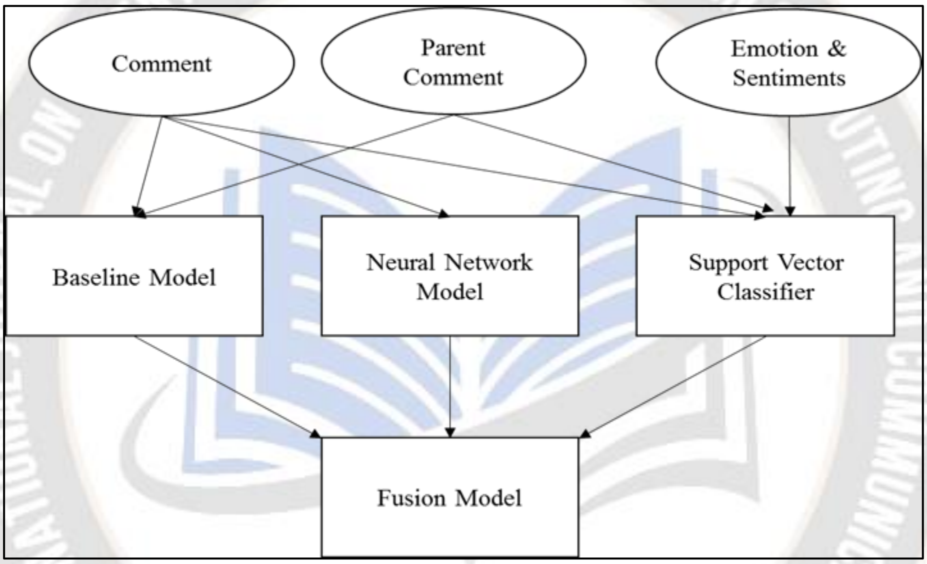

Figure 5. AI Federated Sarcasm Detection Flow

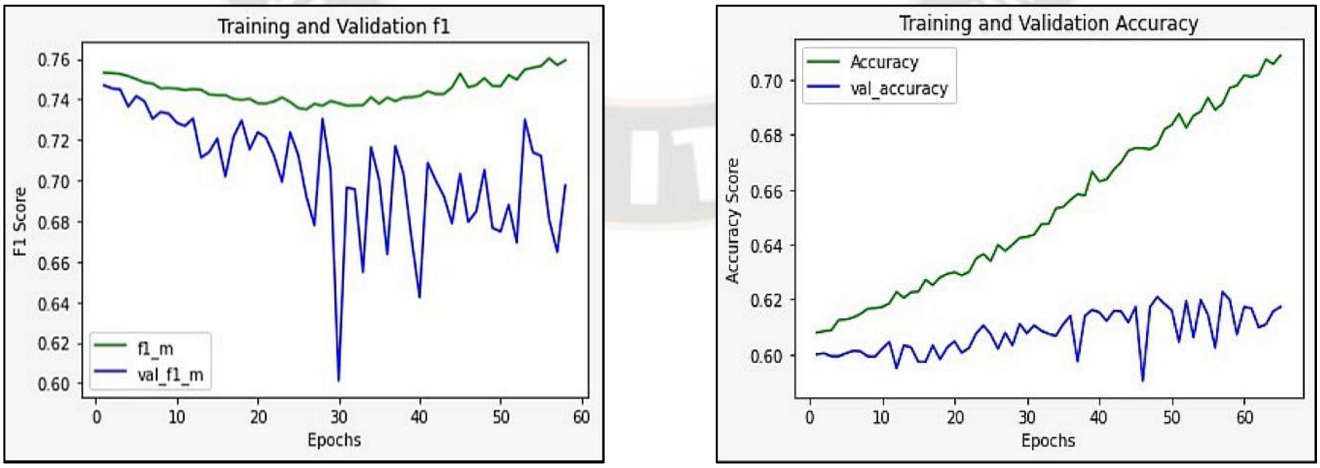

Figure 6. (a) Baseline F1 Model and (b) Baseline Accuracy Model

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

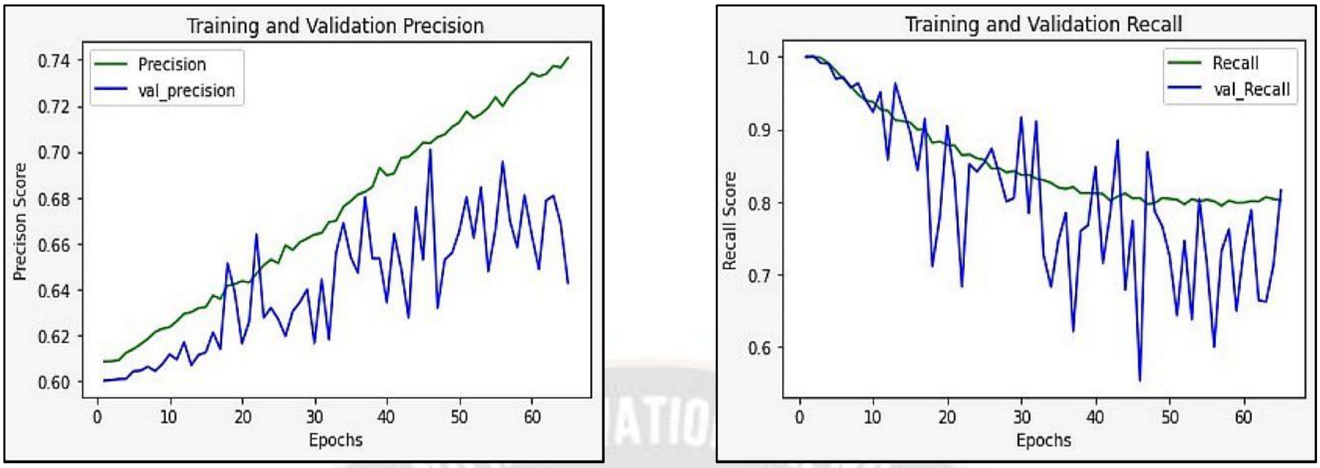

Figure 7. (a) Baseline Precision Model and (b) Baseline Recall Model

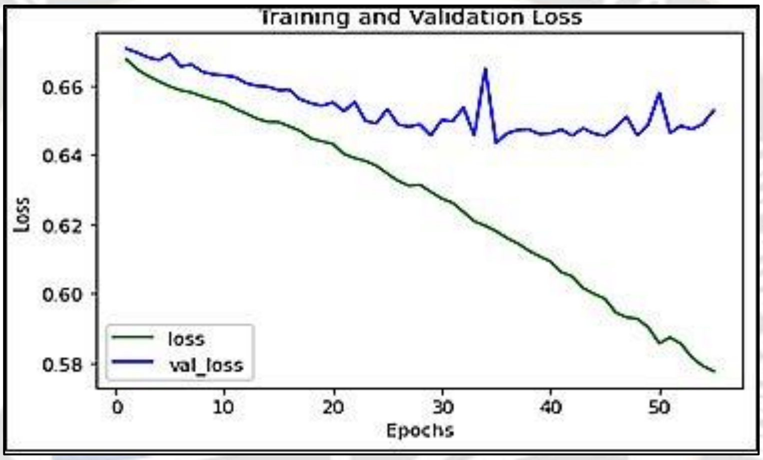

Figure 8. Baseline model Training and Validation Loss

Followed by next neural network model is trained for comment as a feature. The author trained the model for approximately 687 epochs by doing early stopping as a feature of python libraries. A Longet text is better processed by RNN. LSTM is carrying out text processing for better sarcasm detection. Figure 9ashows the LSTM F1-score graph. Figure 9b shows the LSTM accuracy graph. Figure 10a shows Precision Graph, followed by Figure 10b shows the LSTM recall graph. Figure 11 shows the overall training and validation loss for the LSTM model. In addition, if we observe the difference between validation and training loss which is meager, it results in good training of the model. Testing the F1-score of the NN model is 74.18 %. Similarly, the F1-score can be verified in Table 9. Table 9 shows the confusion matrix of the LSTM model. This shows that the value of TP and TN are greater than FN and FP. This results in good accuracy and all mearing metric.

Table 8. Confusion Matrix for Baseline model

		Predicted	
		Yes	**No**
Actual	**Yes**	1729	2901
	No	1468	5677

Then SVC is used with features set as emotions and sentiment of comment and parent comment. After training model testing F1-score came to 74.58%. Therefore, to improve the results, the weighted average approach is used as a fusion of all models resulting in testing the F1-score as 75.75%. The derived F1-score produces a much better harmonic mean of recall and precision for our proposed model. Table 8 shows a summary of results derived from different models used in research.

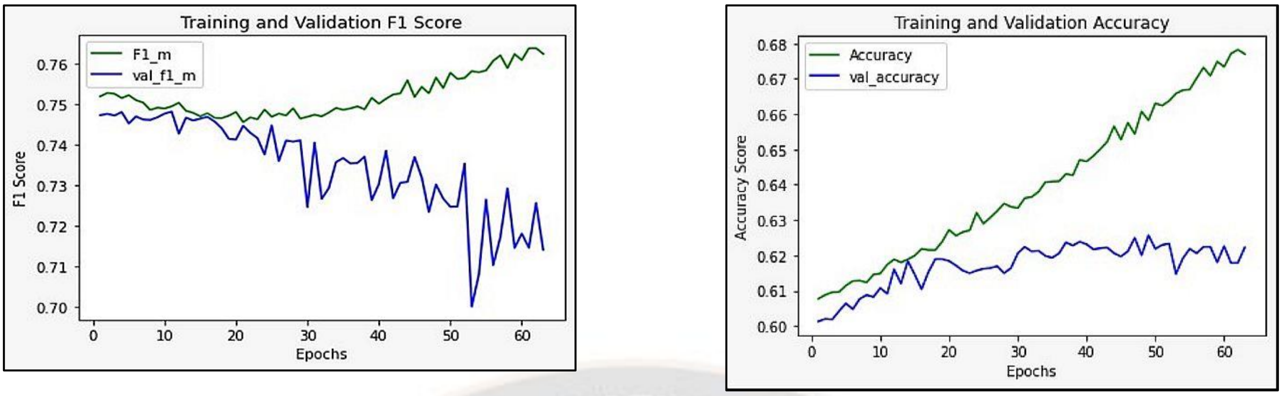

Figure 9. (a) LSTM F1-score Model and (b) LSTM Accuracy Model

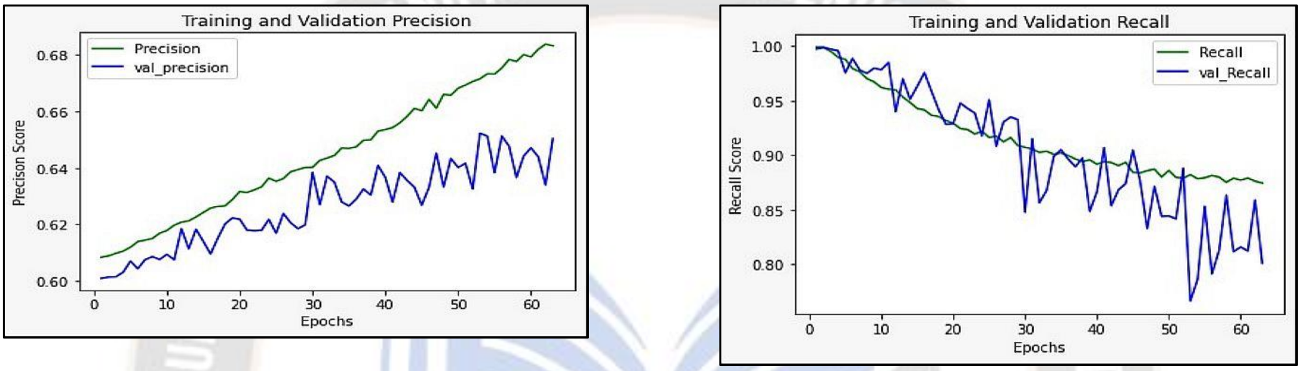

Figure 10. (a) LSTMPrecision Model and (b) LSTM Recall Model

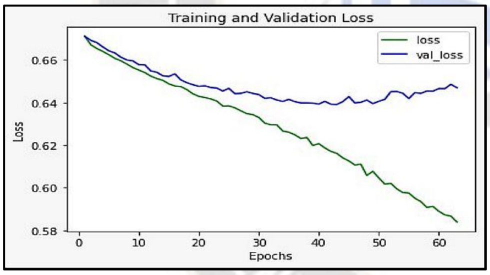

Figure 11. LSTM Model Training and Validation Loss

Table 9. Confusion Matrix for LSTM

		Predicted	
		Yes	**No**
Actual	**Yes**	1169	3461
	No	823	6322

Table 10. F1-score of Models

Models	**Layers**	**Features**	**Training F1-score**	**Testing F1-score**
Baseline	Glove, CNN, LSTM, Dense	Comment, Parent Comment	73.71%	70.19%
Neural Network	Glove, CNN, LSTM, Dense	Comment	74.98%	74.18%
Support Vector Classifier	SVC	Emotion & Sentiment (Comment and Parent Comment)	75.53%	74.56%
Fusion Result	Weighted Average	Combining all feature set	77.07%	75.75%

Figure 12 shows a visual representation of data presented in Table 10 for training and testing the F1-score of different implemented models. As shown in the graph fusion model generates a greater F1-score as compared to individual models. As shown in the graph, we can infer that training and testing F1-score are similar in the outcome. This infers

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

F1-score received as the outcome is far better after training and testing models for given tweets. Liebrecht et al. [33] and González-Ibánez, et al. [34] have performed sarcasm detection on tweets without hashtag sarcasm results into accuracy as 75% and 75.89%. Our proposed methodology with fusion result generates a better F1-score than compared work, focusing on domain-specific text.

5.2 Explainable AI

As a part of XAI, we performed some experiments with counterfactual explination. While implementing counterfactual explination author used multiset_permutations library in python. With the help of multiset_permutation library one can perform permutations and combination of different words to identify the required word which contributes more to have that sentence to be predicated as sarcastic. As a flow of work we a total of ten explanations were manually reviewed. It demonstrates that eight of the examples are excellent and can be believed. These sentences are inputed to counterfactual explination model. As a outcome of XAI the highlighted text is responsible for the text being detected as sarcastic. Below are a few examples of explanations for the prediction. Label 1 is predicated as sarcastic. Label 0 is predicted as non-sarcastic.

Prediction: 1
Explanations
Parent Comment: critic time get real realistic appraisal clinton v trump come trump favor maybe another candidate could have done better thats irrelevant clinton failed always fails answer question conduct secretary state critic expecting typical easy leading question hillary gotten whole campaign coupled bunch nazi question trump
Comment: "clearly watched two different show"
Prediction: 1
Explanations
Parent Comment:" woman vote republican dumb enough deserve lose right"
Comment: "care much woman willing determine everything isnt safe"

Here we have taken a bunch of 10 statements for testing out of which 8 are predicted correctly. Thus, the factual explanation gives an accuracy of 80% which is far better accuracy as compared to existing models.

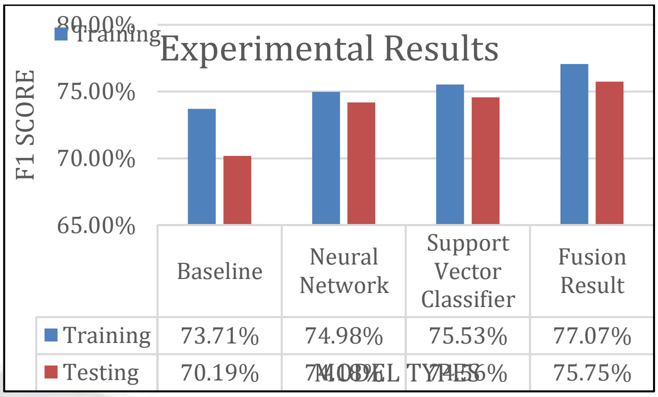

Figure 12. Experimental Results of Sarcasm Detection

VI. Conclusion

Sarcasm detection is a highly challenging part of sentiment classification. A text classified as positive can turn into a sarcastic sentence because of satirical emotion or words present in the text. To identify such satirical and ironic words from the sentence author used a combination of the baseline model, support vector machine, and Recurrent Neural Network. Different features provided to models are set of emotions, sentiment score of comment, and parent comment. After training and testing the model for several iterations, it gives a very good testing F1-score equal to 75.75% for sarcasm detection with the weighted average approach. As surveyed, many researchers have worked on generalized tweets and text. Despite working in the public domain, the author tried to work on domain-specific texts which can create value for any desired organization. The author also included counterfactual explanations for the prediction. We manually tested and were satisfied with 8 out of 10 explanations, indicating that our explanations provide results with 80% accuracy. This work adds novelty to the research article by targeting domain texts with XAI adding value to the business.

Limitations and Future Work

Limitations of current research work is as it works only on text. Working on text and identifying sarcasm is very cumbersome task. Therefore as a future scope of predicating sarcasm with higher accuracy text can be combined with combination of input datasets such audio, image and video of audience contributing their emotions on social media. Along with multiple types of data format researchers can refer to multilingual study as well. In [35] author have reffered a multilingual irony detection for French, English and Italian text. The aim of this paper is to implement a multilayerd architecture which works on implicit and explicit aspect paralally. Thus as a future work autor will try to combine this approach of multidaset format with

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

multilingual approach to target better prediction of sarcastic text.

Acknowledgement: Authors acknowledge the supportof their respective institutes.
Funding Statement: There is no funding to declare.
Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1]. Singh, P. D.; Kaur, R.; Singh, K. D.; Dhiman, G. A Novel Ensemble-Based Classifier for Detecting the COVID-19 Disease for Infected Patients. Inf. Syst. Front. 2021, 23 (6), 1385–1401. https://doi.org/10.1007/s10796-021-10132-w.

[2]. Kalaivani; Thenmozhi. Sarcasm Identification and Detection in Conversion Context Using BERT. In Proceedings of the Second Workshop on Figurative Language Processing; Association for Computational Linguistics: Stroudsburg, PA, USA, 2020.

[3]. Gupta, R. K.; Yang, Y. CrystalNest at SemEval-2017 Task 4: Using Sarcasm Detection for Enhancing Sentiment Classification and Quantification. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017); Association for Computational Linguistics: Stroudsburg, PA, USA, 2017.

[4]. Kumar, A.; Dikshit, S.; Albuquerque, V. H. C. Explainable Artificial Intelligence for Sarcasm Detection in Dialogues. Wirel. Commun. Mob. Comput. 2021, 2021, 1–13. https://doi.org/10.1155/2021/2939334.

[5]. Kumar, A.; Sangwan, S. R.; Singh, A. K.; Wadhwa, G. Hybrid Deep Learning Model for Sarcasm Detection in Indian Indigenous Language Using Word-Emoji Embeddings. Transactions on Asian and Low-Resource Language Information Processing; 2022.

[6]. Parmar, K.; Limbasiya, N.; Dhamecha, M. Feature Based Composite Approach for Sarcasm Detection Using MapReduce. In 2018 Second International Conference on Computing Methodologies and Communication (ICCMC); IEEE, 2018.

[7]. Cai, Y.; Cai, H.; Wan, X. Multi-Modal Sarcasm Detection in Twitter with Hierarchical Fusion Model. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics; 2019; pp 2506–2515.

[8]. Baziotis, C.; Nikolaos, A.; Papalampidi, P.; Kolovou, A.; Paraskevopoulos, G.; Ellinas, N.; Potamianos, A. NTUA-SLP at SemEval-2018 Task 3: Tracking Ironic Tweets Using Ensembles of Word and Character Level Attentive RNNs. In Proceedings of The 12th International Workshop on Semantic Evaluation; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018.

[9]. Sarsam, S. M.; Al-Samarraie, H.; Alzahrani, A. I.; Wright, B. Sarcasm Detection Using Machine Learning Algorithms in Twitter: A Systematic Review. Int. J. Mark. Res. 2020, 62 (5), 578–598. https://doi.org/10.1177/1470785320921779.

[10]. Razali, M. S.; Halin, A. A.; Ye, L.; Doraisamy, S.; Norowi, N. M. Sarcasm Detection Using Deep Learning with Contextual Features. IEEE Access 2021, 9, 68609–68618. https://doi.org/10.1109/access.2021.3076789.

[11]. Sarsam, S.M., Al-Samarraie, H., Alzahrani, A.I., Alnumay, W. and Smith, A.P., 2021. A lexicon-based approach to detecting suicide-related messages on Twitter. Biomedical Signal Processing and Control, 65, p.102355.

[12]. Buschmeier, K.; Cimiano, P.; Klinger, R. An Impact Analysis of Features in a Classification Approach to Irony Detection in Product Reviews. In Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis; Association for Computational Linguistics: Stroudsburg, PA, USA, 2014.

[13]. Joshi, A.; Tripathi, V.; Patel, K.; Bhattacharyya, P.; Carman, M. Are Word Embedding-Based Features Useful for Sarcasm Detection? In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Stroudsburg, PA, USA, 2016.

[14]. Poria, S.; Cambria, E.; Hazarika, D.; Vij, P. A Deeper Look into Sarcastic Tweets Using Deep Convolutional Neural Networks. arXiv [cs.CL], 2016.

[15]. Bamman, D.; Smith, N. Contextualized Sarcasm Detection on Twitter. Proceedings of the International AAAI Conference on Web and Social Media 2015, 9.

[16]. Suhaimin, M. S. M.; Hijazi, M. H. A.; Alfred, R.; Coenen, F. Natural Language Processing Based Features for Sarcasm Detection: An Investigation Using Bilingual Social Media Texts. In 2017 8th International Conference on Information Technology (ICIT); IEEE, 2017.

[17]. Jacovi, A.; Sar Shalom, O.; Goldberg, Y. Understanding Convolutional Neural Networks for Text Classification. In Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018.

[18]. Bodria, F.; Panisson, A.; Perotti, A.; Piaggesi, S. Explainability Methods for Natural Language Processing: Applications to Sentiment Analysis; 2020.

[19]. Messalas, A.; Kanellopoulos, Y.; Makris, C. Model-Agnostic Interpretability with Shapley Values. In 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA); IEEE, 2019.

[20]. Fiok, K.; Karwowski, W.; Gutierrez, E.; Ahram, T. Predicting the Volume of Response to Tweets Posted by a Single Twitter Account. Symmetry (Basel) 2020, 12 (6), 1054. https://doi.org/10.3390/sym12061054.

[21]. Malolan, B.; Parekh, A.; Kazi, F. Explainable Deep-Fake Detection Using Visual Interpretability Methods. In 2020 3rd International Conference on Information and Computer Technologies (ICICT); IEEE, 2020.

***Article Received**: 12 October 2022 **Revised**: 15 November 2022 **Accepted**: 20 December 2022*

[22]. Yang, L.; Kenny, E.; Ng, T. L. J.; Yang, Y.; Smyth, B.; Dong, R. Generating Plausible Counterfactual Explanations for Deep Transformers in Financial Text Classification. In Proceedings of the 28th International Conference on Computational Linguistics; International Committee on Computational Linguistics: Stroudsburg, PA, USA, 2020.

[23]. Bagate, R. A.; Suguna, R. Different Approaches in Sarcasm Detection: A Survey. In Intelligent Data Communication Technologies and Internet of Things; Springer International Publishing: Cham, 2020; pp 425–433.

[24]. Tay, Y.; Luu, A. T.; Hui, S. C.; Su, J. Reasoning with Sarcasm by Reading In-Between. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); Association for Computational Linguistics: Stroudsburg, PA, USA, 2018.

[25]. Riloff, E.; Qadir, A.; Surve, P.; De Silva, L.; Gilbert, N.; Huang, R. Sarcasm as Contrast between a Positive Sentiment and Negative Situation. In Proceedings of the 2013 conference on empirical methods in natural language processing; 2013; pp 704–714.

[26]. Ghosh, A.; Veale, T. Magnets for Sarcasm: Making Sarcasm Detection Timely, Contextual and Very Personal. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Stroudsburg, PA, USA, 2017.

[27]. Khodak, M.; Saunshi, N.; Vodrahalli, K. A Large Self-Annotated Corpus for Sarcasm. arXiv [cs.CL], 2017.

[28]. Bagate, R. A.; Suguna, R. Sarcasm Detection of Tweets without #sarcasm: Data Science Approach. Indones. j. electr. eng.comput. sci. 2021, 23 (2), 993. https://doi.org/10.11591/ijeecs.v23.i2.pp993-1001.

[29]. Dhar, S.; Bose, I. Emotions in Twitter Communication and Stock Prices of Firms: The Impact of Covid-19 Pandemic. Decision 2020, 47 (4), 385–399. https://doi.org/10.1007/s40622-020-00264-4.

[30]. Hutto, C.; Gilbert, E. Vader: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. Proceedings of the international AAAI conference on web and social media 2014, 8, 216–225.

[31]. Luintel, S.; Sah, R. K.; Lamichhane, B. R. A Hybrid Approach for Sarcasm Detection. Tech J 2019, 1 (1), 1–9. https://doi.org/10.3126/tj.v1i1.27581.

[32]. Bagate, R., & Suguna, R. (2022). Sarcasm Detection on Tweets: Ensemble Approach. International Journal of Next-Generation Computing, 13(3).

[33]. Liebrecht, C. C.; Kunneman, F. A.; Bosch, A. P. J. The Perfect Solution for Detecting Sarcasm in Tweets# Not; 2013.

[34]. González-Ibánez, R.; Muresan, S.; Wacholder, N. Identifying Sarcasm in Twitter: A Closer Look. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies; 2011; pp 581–586.

[35]. Karoui, Jihen, Farah Benamara, VéroniqueMoriceau, Viviana Patti, Cristina Bosco, and Nathalie Aussenac-Gilles. "Exploring the impact of pragmatic phenomena on irony detection in tweets: A multilingual corpus study." In 15th Conference of the European Chapter of the Association for Computational Linguistics, vol. 1, pp. 262-272. 2017.

[36]. Sable Nilesh Popat*, Y. P. Singh,” Efficient Research on the Relationship Standard Mining Calculations in Data Mining” in Journal of Advances in Science and Technology | Science & Technology, Vol. 14, Issue No. 2, September-2017, ISSN 2230-9659.

[37]. Sable Nilesh Popat*, Y. P. Singh,” Analysis and Study on the Classifier Based Data Mining Methods” in Journal of Advances in Science and Technology | Science & Technology, Vol. 14, Issue No. 2, September-2017, ISSN 2230-9659.