



Automatic Sarcasm Detection on Cross-Platform Social Media Datasets: A GLoVe and Bi-LSTM Based Approach

Saima Farhan

(Department of Computer Science, Lahore College for Women University, Lahore, Pakistan
 <https://orcid.org/0000-0002-6610-9290>, saima.farhan@lcwu.edu.pk)

Rubiya Shoukat

(Department of Computer Science, Lahore College for Women University, Lahore, Pakistan
 <https://orcid.org/0009-0008-9400-7938>, rubiashoukat1@gmail.com)

Aqsa Aslam

(Department of Computer Science, Lahore College for Women University, Lahore, Pakistan
 <https://orcid.org/0000-0002-3447-8278>, aqsa.aslam28@gmail.com)

Abstract: Sarcastic remarks on social media platforms have become commonplace, with people expressing their bad feelings in a quite positive manner or in a mocking way. This contradictory nature of sarcasm makes its detection a very challenging task. Many researchers have provided their solutions to perform automatic sarcasm detection from a single domain dataset. Most of them have considered only the content of the text and has ignored the context of the text. Understanding that the context of a text is the most important factor in determining either it is sarcastic or not. This study aims to detect sarcastic remarks from multi-domain dataset by using Bi-LSTM model employed with pre-trained GloVe word embeddings because the GloVe embeddings and Bi-LSTM both are good at capturing contextual information from the provided data. The dataset is generated by the concatenation of three publicly available datasets, the gosh tweets, the news headlines dataset, and the sarcasm corpus v2 dataset. GloVe embeddings will extract contextual and semantic features from the text while the Bi-LSTM model will get trained and tested on those features. The proposed model has achieved 86.35% of accuracy with 88% of Recall, Precision, and F1-score. Different experiments have been done to test the model's reliability. This study's findings indicate that the proposed model yields state-of-the-art or comparable results. The proposed study aids in improving the performance of sentiment analysis. It will also help individuals, and different organizations to identify accurate sentiments of people about individuals or products of an organization.

Keywords: NLP, Sentiment Analysis, Sarcasm, Deep Learning, Bi-LSTM, GloVe embeddings

Categories: I.2, I.2.7, I.7

DOI: 10.3897/jucs.104790

1 Introduction

Social media has become an attractive everyday activity, and it is giving the most up-to-date information about a person or organization [Yue et al., 2019]. Finding out what people think (their feelings) about something, whether it's a product, a new policy, a political issue, a service, or a person, can be very useful for any organization [Gang Wang a, b et al., 2014] [Vyas and Uma, 2018]. To analyze public attitudes or views,

Sentiment Analysis (SA), which comes under the umbrella of Natural Language Processing (NLP), has gained enormous popularity in recent years.

SA examines the feelings, emotions, or opinions present in a piece of text [Medhat et al., 2014][Chen et al., 2011]. SA helps organizations figure out how people feel about politics, market information [Yung-Ming Li, 2013], product evaluations, and other things so they can keep their people happy [Onan, 2019]. SA classifies statements as positive, negative, or neutral and when it comes to complex statements like sarcastic statements, SA performs misclassifications. The prevalence of figurative language on social media sites makes SA challenging [Savini et al., 2019]. “Sarcasm” is a type of figurative language which refers to the representation of negative emotions positively [del Pilar Salas-Zárate et al., 2020].

Among different types of sentiments, most people find sarcasm an attractive way of expressing their sentiments. Research on Twitter sentiments [Peng et al., 2019] reveals sarcastic remarks are larger during a service outage than on typical days, and negative sarcastic tweets elicit more social media reactions than literal negative tweets. The statistics of this analysis are provided in Figure 1.

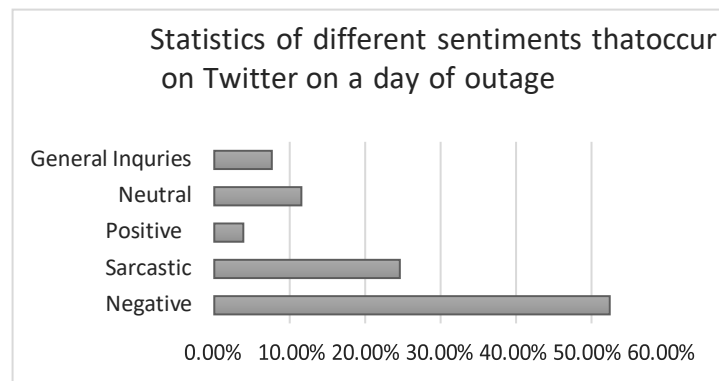


Figure 1: Statistics of different sentiments on Twitter on the day of the outage

Identifying sarcasm has always been difficult [Chaudhari and Chandankhede, 2017]. In recent years, in the domain of Artificial Intelligence (AI), sarcasm detection has gained popularity. According to research [Maynard and Greenwood, 2014], sarcasm detection improves SA performance by 50%. Therefore, a variety of approaches have been given to construct a system for automatic sarcasm identification, including Machine Language (ML) [Reyes et al., 2013][Swami et al., 2018], and Deep Learning (DL) models [Verma et al., 2021][Poria et al., 2016]. A number of researchers have considered different sets of features based on the author’s linguistic style [Mukherjee and Bala, 2017] to identify sarcasm. Likewise, to determine the presence of sarcasm in tweets with help of ML classifiers Naïve Bayes (NB), Decision Tree (DT), Support Vector Machine (SVM), Random Forest (RF), and Adaboost [Bharti et al., 2017] are used. Conventional ML systems employ handcrafted features that involve feature engineering and domain expertise [Lecun et al., 2015] but it may lead to biasness. So, to reduce the effort for handcrafted features, DL methods are introduced that allow automatic feature extraction. It has been observed that DL algorithms outperform traditional ML

algorithms [Abdi et al., 2019]. Sarcasm identification from a multi-domain dataset has not been investigated enough and needs the researcher's attention.

This study aims to detect sarcasm in social media platforms using the multi-domain dataset. In the proposed study, a DL based method is used to determine sarcasm by using a bidirectional Long Short Term Memory (Bi-LSTM) model with a pre-trained global vector (GloVe) word embeddings [Pennington et al., 2014]. Bi-LSTM is best suited for the proposed approach as it is ideal for time series data [Karim et al., 2017] and can memorize extensive data sequences [Siami-Namini et al., 2019]. GloVe embedding works effectively with large corpora [Bhoir et al., 2017]. Three benchmark datasets have been used the Gosh Tweet dataset [Ghosh and Veale, 2017], the News Headlines dataset by Rishabh Misra [Misra and Arora, 2019], and the Sarcasm Corpus V2 dataset [Oraby et al., 2016]. Initially the data is preprocessed to remove noise. GloVe embedding collects contextual cues from data and delivers it to Bi-LSTM, which learns on supplied data and predicts sarcasm. The proposed research concludes that the recommended study can help organizations to identify people's opinions about their products or services and also the correct classification of sarcastic statements can boost SA's performance by making it possible for SA to be able to understand complex and multifaceted statements.

Rest of the paper is organized as follows: Section 2 deals with a literature survey of sarcasm detection, Section 3 deals with material and methods used in this approach, Section 4 provides experimental strategy, results and discussions and Section 5 presents the conclusion and future work.

2 Related Work

Sarcasm detection has grabbed the interest of numerous researchers who come up with a variety of cutting-edge methods. Different researchers have used ML methods, and DL methods or have focused on feature engineering methods to detect sarcasm. This section provides a brief review of existing methods for sarcasm detection. Earlier, researchers focused on feature engineering approaches to detect sarcasm including, statistical and semantic features [Alcaide et al., 2015], N-grams [Rosso, 2012], lexical [Liu et al., 2014], contextual features [Wang et al., 2015], etc. Nagwanshi and Madhavan [Nagwanshi and Veni Madhavan, 2014] used supervised learning to identify sarcasm by extracting lexical, syntactic, semantic, pragmatic, politeness-rating, and attitude-flipping information. They believe these features indicate sarcasm. Justo, Corcoran, et al. [Justo et al., 2014] suggested a technique for detecting sarcasm using a collection of characteristics. The study produced several sets of features by mixing two characteristics from various available features including mechanical turk cues, statistical cues, linguistic, semantic, length, and concept/polarity information. The semantic feature set has been found to be more valuable in sarcasm.

Bouazizi and Ohtsuki [Bouazizi and Otsuki, 2016] studied pattern-based sarcasm recognition in tweets. This technique uses sentiment-related, punctuation-related, syntactic-semantic, and pattern-related characteristics. These features can detect sarcasm, but they need manual effort in feature extraction. A multi-feature fusion framework has been conducted to detect sarcasm by Eke, Norman, et al [Eke et al., 2021b]. This approach uses lexical characteristics to detect sarcastic inclinations of an individual. The lexical sarcastic leaning feature is then combined with eight additional

features i.e, emotion, pragmatic, hashtag, discourse marker, semantic, emoticon, microblog length, and syntactic features. Several ML models have been designed to detect sarcasm. Paving a way to effective sarcasm detection approaches using the ML model, an approach known as Intelligent ML-based Sarcasm Detection and Classification (IMLB-SDC) [Vinoth and Prabhavathy, 2022] has been developed that extracts features from the data with the help of the Term Frequency-Inverse Document Frequency TF-IDF method. The IMLB-SDC used SVM, to perform classification.

A strategy has been used on tweets in which the researcher relied not only on what was said but also on who wrote it such as Mukherjee and Bala [Mukherjee and Bala, 2017]. They examined authorial style and extracted function words and n-grams for categorization using Fuzzy clustering and NB algorithms. Focusing on sarcastic statements present in the Twitter dataset, Govindan et al [Govindan and Balakrishnan, 2022] have used SVM, RF, and RF with Bagging. During an analysis of ML techniques, it was found that most research employed SVM since it delivers promising results in text categorization.

DL approaches have also been studied that can detect sarcastic statements automatically. In an approach [Razali et al., 2021] to identify sarcasm, CNN features are mixed with the handcrafted feature set. To determine sarcasm from given data, different techniques of DL have been applied such as Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN) and GRU [Goel et al., 2022]. A hybrid approach has been introduced by Jamil et al [Jamil et al., 2021] in which a CNN, has been used for feature extraction and LSTM for training and testing on those features. The study obtained remarkable results on a multi-domain dataset used for sarcasm detection. Using neural networks and a DL model, Onan and Toçoğlu [Onan and Tocoglu, 2021] employed a method to detect sarcasm online. The suggested research used word embedding models using trigrams to recognize sarcastic phrases. A three-layer Bi-LSTM model has been designed to recognize sarcastic statements. A transformer-based strategy has been presented with a DL model, LMTweets [Ahuja and Sharma, 2022] that extracts features from the dataset, and then the characteristics are fed to CNN for classification.

A method proposed by [Bhardwaj and Prusty, 2022] pre-processes the text by using BERT, and then feeds the pre-processed text into a hybrid DL model for the purposes of training and classification where the CNN and LSTM are both components of this hybrid model. A hybrid attention-based method [Pandey et al., 2021] using LSTM and 16 linguistic features was employed to recognize sarcasm. Each statement has two types of features: one drawn from the linguistic feature set and the other from DL. These traits are combined and classified into sarcastic and non-sarcastic comments. Researchers have also focused on an individual expression tendency in social media to detect sarcasm [Du et al., 2022]. A dual-channel CNN model has been used to investigate the text's emotive context. For the detection of sarcasm on the News Headlines [Misra and Arora, 2019] dataset CNN-LSTM, approach has been presented. The model has an embedding layer that converts words into their vector spaces and then vectors are passed to the CNN layer. After the convolutional layer, a Bi-LSTM has been employed for predicting sarcasm.

3 Material and Methods

The methodology of the proposed research work consists of Data acquisition, Data preprocessing, Data splitting, Feature extraction, and classification. The overall architecture of the proposed model is depicted in *Figure 2*.

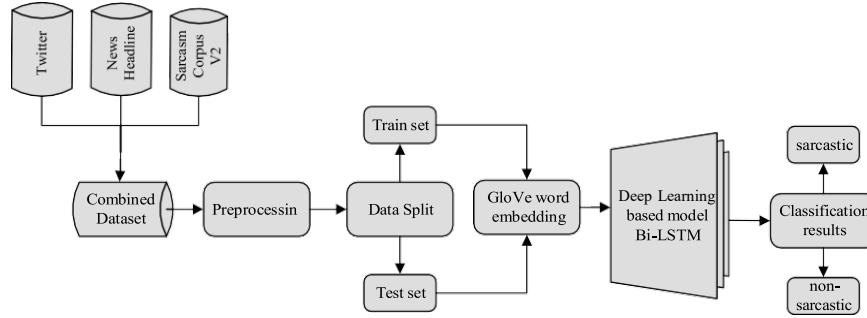


Figure 2: An overview of the proposed model

Datasets

The datasets that have been used in proposed research are *Twitter (Gosh and Veale)* [Ghosh and Veale, 2017]; *The News Headlines dataset* [Misra and Arora, 2019] which is based on datasets from two news websites, *The Onion*¹ which publishes sarcastic version of current events, and *Huffpost*² which contains real (non-sarcastic) headlines; and *Sarcasm Corpus V2* [Oraby et al., 2016] dataset, a subset of an Internet Argument Corpus (IAC) dataset that contains sarcastic posts. Sarcasm Corpus V2 includes three types of sarcasm *General (GEN)*, *Rhetorical Questions (RQ)*, and *Hyperbole (HYP)*.

These datasets are publicly available and contains English textual data. In each dataset, there are two labels 0,1 where 0 denotes non-sarcastic statement and 1 denotes sarcastic statement. To create a huge, complex, and multi-domain dataset, these three datasets are concatenated. The detail of each dataset utilized in this investigation is provided in Table 1.

	Twitter (Gosh and Veale)	News Headlines dataset	Sarcasm Corpus V2dataset			Combined Dataset (multi- domain)
			GEN	RQ	HYP	
Sarcastic	18488 (46%)	13634 (48%)	3260 (50%)	851 (50%)	582 (50%)	36632 (47%)
Non- Sarcastic	21292 (54%)	14985 (52%)	3260 (50%)	851 (50%)	582 (50%)	40873 (53%)
Total	39780	28619	6520	1702	1164	77,505

Table 1: Statistical information of datasets

¹ <https://www.theonion.com/>

² <https://www.huffpost.com/>

Data Preprocessing

The major challenge that NLP encounters is the presence of noise in the data. The following preprocessing steps have been applied for noise reduction as shown in *Figure 3*.

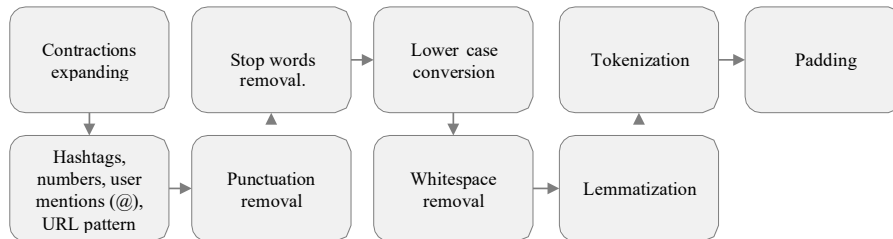


Figure 3: Data preprocessing steps

All the contractions present in texts are expended to their full forms using python package *contractions*, as the machine cannot understand abbreviated terms. The data that does not add information to the model is removed such as hashtag, URL pattern, usermentioned @, numbers, punctuation marks, whitespace, and stop words.

To make the data consistent, the text is converted into lowercase. A text normalization technique, called lemmatization, has been applied using python library *spaCy* that converts different forms of a word to their root form, known as *lemma*. Data tokenization is performed which splits a sentence or words into smaller fragments or *tokens*. These tokens are very effective in discovering patterns. Padding is applied to the data by using *pad_sequences* function from Keras library to make the textual data same in shape and size.

Data Splitting

The dataset has been divided into a ratio of 80:20 for training and testing of the model, where 80 percent of the data is utilized for training and 20 percent is used for testing respectively.

GloVe Word Embedding

GloVe embedding has been applied to capture insightful features from the dataset. GloVe stands for global vectors and is one of the famous words embedding techniques which is known for its ability to capture the semantic and contextual features from the given text. To get the vector representation of the words, GloVe [Pennington et al., 2014] embedding has been used. The vector space in GloVe is used to determine similarity and dissimilarity between the words. Words having similar semantics are bunch together in same place whereas words with different meanings are mapped at farther apart within the vector space. The distance between words in vector space relates to their semantic similarity.

GloVe's benefit over other word embedding approaches is that it captures the global word-word co-occurrence matrix (global statistics or global context) in addition to the local context information (local statistics) of the words. Another reason for choosing

pre-trained GloVe embedding is that it has good performance and possess a quicker training speed because it has been pre-trained on huge datasets, and has reduced the overhead of computation.

Proposed DL Model

In the proposed framework, a sequential modeling method has been used which allows layer by layer model construction. The structural design of the proposed classification model is presented in *Figure 4*.

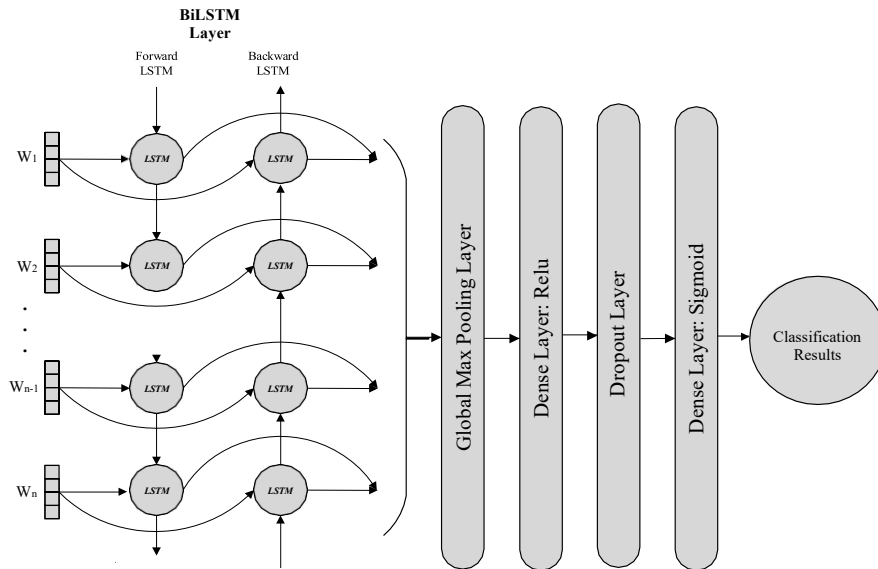


Figure 4: Structure of proposed DL model

3.1.1 Embedding Layer

The proposed model starts with an input layer known as embedding layer, which takes a series of tokens as input and passes it to the Bi-LSTM layer. This embedding layer uses embedding as weights obtained from the GloVe word embedding.

3.1.2 Bi-LSTM Layer

The Bi-LSTM model has been used that originates from the concatenation of two LSTMs i.e. forward LSTM and backward LSTM. Bi-LSTM can capture contextual information in both directions, and it can also memorize long input sequences as it has additional memory. Each of the forward and backward LSTMs of Bi-LSTM has three gates of their own, input gate i_t , output gate o_t , forget gate f_t , and a memory cell c_t that can be calculated using equation 1, equation 2, equation 3, and equation 4 respectively. The candidate for memory cell c_t at timestamp (t) and the output of hidden state h_t can be obtained using equation 5 and equation 6 respectively.

$$i_t = \sigma (W_i x_i + U_i h_{t-1}) \quad (1)$$

$$o_t = \sigma (W_o x_i + U_o h_{t-1}) \quad (2)$$

$$f_t = \sigma (W_f x_i + U_f h_{t-1}) \quad (3)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c} \quad (4)$$

$$c_t = \tanh (W_c x_i + U_c h_{t-1}) \quad (5)$$

$$h_t = o_t \odot \tanh c_t \quad (6)$$

In the above equations, the variable x_i denotes the word vector of current input, U and W are the weight factors, h_t indicates the vector of hidden state and \tanh is the activation function. The equation for Bi-LSTM is given in equation 7.

$$h_{bilstm} = \text{concat} [\vec{h} \overleftarrow{h}] \quad (7)$$

Here, h_{bilstm} represents Bi-LSTM and \vec{h} indicates forward LSTM and \overleftarrow{h} denotes backward LSTM. Bi-LSTM learns the context of word vectors by sending them to 64×64 neurons in each layer in forward and backward directions.

3.1.3 Global Max Pooling Layer

A global max pooling 1D layer has been applied that reduces processing cost by calculating the maximum value in a matrix for each input vector. It can learn invariable characteristics and reduce overfitting.

3.1.4 Dense Layer with ReLU Activation Function

The dense layer in the proposed research uses rectified linear (ReLU) activation function $F(x)$. Each dense layer neuron is directly connected to every preceding layer neuron. The dense layer's neurons are activated by ReLU to feed the dropout layer. ReLU determines whether to activate a neuron and also helps to prevent exponential computation power. Equation 8 defines ReLU formally.

$$F(x) = \max (0, x) \quad (8)$$

This function returns a 0 when it receives negative input but for any positive value x , it returns that value back. Therefore, the output of this function ranges from 0 to infinity.

3.1.5 Dropout Layer

The purpose of dropout layer is to reduce overfitting. The dropout layer works by randomly losing neurons and their connections, which forces the model to increase all the neurons' generalization capacity. In the proposed architecture, the retention probability is set to 0.1, meaning neurons have a 90% chance of being active and a 10% chance of being dropped.

3.1.6 Dense Layer with Sigmoid Activation Function

This is the output layer of the proposed model. Here the sigmoid activation function has been used since it is optimal for binary classification.

In this study a loss function *binary_crossentropy* has been used which compares the predicted class with actual output and calculates loss. “Adam” function has been used as an optimizer.

4 Results and Discussion

This section presents the details of all experiments done along with their results. The results of this study are discussed and compared with existing approaches.

Experiments

In this study, several experiments have been conducted to analyze the Bi-LSTM’s ability to identify sarcasm from complex, large, and multi-domain data. Experiments diversified data complexity from single to multi-domain. During literature review, it has been observed that the prior studies have suggested their approaches using single dataset or even a combination of two datasets. No prior research has been found which has used a combination of three datasets altogether. The proposed study has been compared with existing research and to make this comparison fair, experiments have been conducted with a single dataset, with two datasets combined, and then using all three datasets altogether. The results show that the proposed model has outperformed the established practices that are using a single or even two datasets combined.

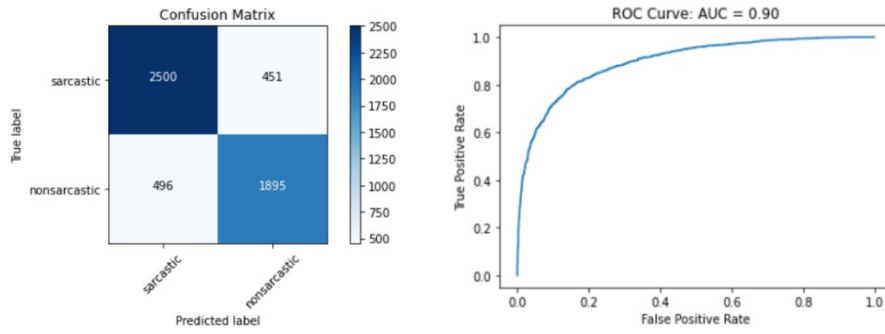
4.1.1 Experiments with single domain dataset

A total of 3 experiments are conducted using a single domain dataset. The datasets used are News Headlines, Gosh Tweets, and Sarcasm Corpus V2 with "sarcastic" and "non-sarcastic" entries. The findings of these experiments are presented in Table 2. The confusion matrices and AUC-ROC of these experiments are provided in Figure 5,

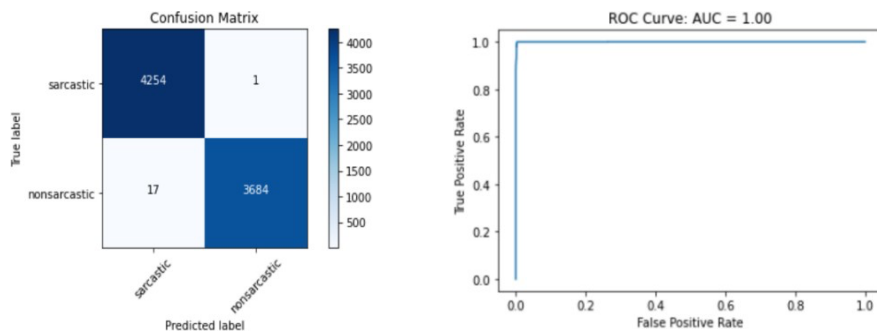
The findings show that the model has yielded higher predictive performance on Twitter dataset by achieving 99.96% prediction accuracy as compared to other datasets. The comparison of the proposed model's performance with several known techniques using single domain dataset is shown in Table 3 where it is evident that the proposed model outperforms the others, with its scores in bold and the highest existing score is italic.

Dataset	Accuracy	Class	Recall	Precision	F1-score
News Headlines	82.27%	Sarcasm	0.81	0.79	0.80
		Non-sarcasm	0.83	0.85	0.84
		WeightedAvg	0.82	0.82	0.82
Twitter (Gosh and Veale)	99.96%	Sarcasm	1.00	1.00	1.00
		Non-sarcasm	1.00	1.00	1.00
		WeightedAvg	1.00	1.00	1.00
Sarcasm Corpus V2	71.86%	Sarcasm	0.71	0.71	0.71
		Non-sarcasm	0.73	0.73	0.73
		WeightedAvg	0.72	0.72	0.72

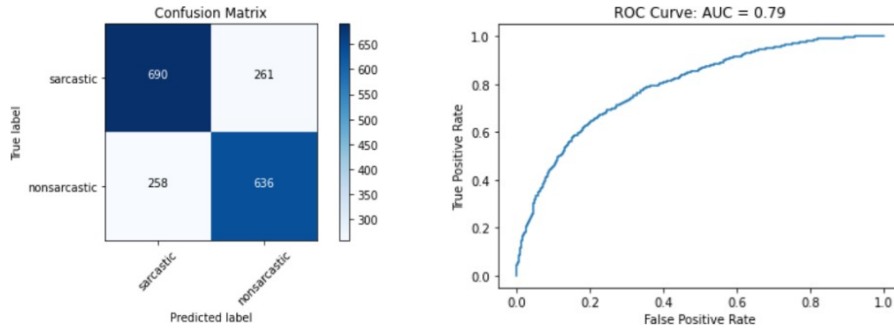
Table 2: Experimental results of single domain datasets



(a) Confusion matrix and AUC-ROC for News Headlines



(b) Confusion matrix and AUC-ROC for Twitter (Gosh and Veale)



(c) Confusion matrix and AUC-ROC for Sarcasm Corpus V2

Figure 5: Confusion matrix and AUC-ROC with single domain dataset. The above diagrams a,b,c represents confusion matrix and AUC-ROC of each experiment with single domain dataset.

Reference	Model	Accuracy	Recall	Precision	F1-score
[Xiong et al., 2019]	Self-matching (SMSD) Bi-LSTM	80.09%	72.56%	76.39%	74.42%
[Du et al., 2022]	CNN + Bi-LSTM	73%	--	--	76%
[Shrivastava and Kumar, 2021]	BERT	70.6%	72.5%	68.7%	70.5%
[Mandal and Mahto, 2019]	CNN-LSTM	86.16%	--	--	--
[Kumar and Garg, 2023]	TF-IDF, LSTM/Bi-LSTM	86.32%	84.39%	81.61%	85.25%
[Akula, 2021]	Multi-head Attention + Gated Recurrent Unit	88.60%	81.8%	80.9%	81.2%
[Eke et al., 2021a]	Bi-LSTM, BERT	98.21%	98.0%	98.50%	98.50%
Proposed Model	GloVe + Bi-LSTM	99.96%	100%	100%	100%

Table 3: Comparison of the proposed approach with existing approaches using single domain dataset.

4.1.2 Experiments with multi-domain datasets (two datasets)

Since the proposed study evaluate the Bi-LSTM's performance on a multi-domain dataset, the next 3 experiments are conducted using a multi-domain dataset formed by concatenating two of the three datasets i.e. "Twitter (Gosh and Veale) + News Headlines", "Sarcasm Corpus V2 + Twitter (Gosh and Veale)" and "News Headlines + Sarcasm Corpus V2" that contain both sarcastic and non-sarcastic entries. The accuracy, recall, precision and F1-score of these experiments are shown in Table 4.

It can be observed from Table 4, that the proposed model has gained higher predictive accuracy of 92.19% on multi-domain datasets formed by the concatenation of the "Sarcasm Corpus V2 + Twitter (Gosh and Veale)" dataset as compared to others. "Twitter (Gosh and Veale) + News Headlines" achieved an accuracy of 91.04% whereas a lowest accuracy of 78.55% is observed with "News Headlines + Sarcasm Corpus V2" combined dataset.

The proposed approach is compared to the existing technique that likewise uses a multi-domain dataset formed by the concatenation of two datasets and the results are presented in Table 5. The findings of the comparison show that the proposed approach outperforms the already existing one.

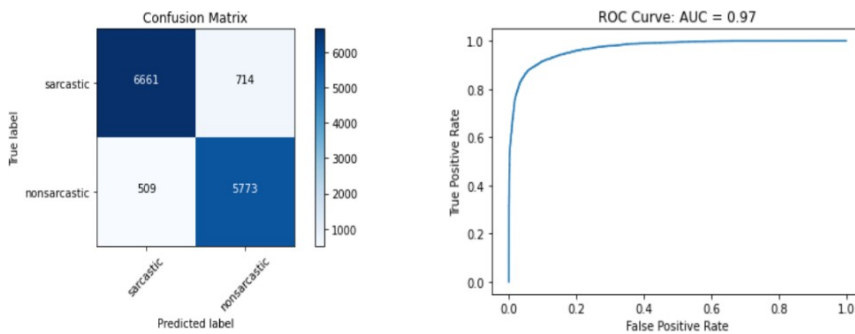
Confusion matrix and AUC-ROC curve for all three experiments are presented in Figure 6 respectively.

Dataset	Accuracy	Class	Recall	Precision	F1-score
Twitter (Gosh and Veale) + News Headlines	91.04%	Sarcasm	0.89	0.92	0.90
		Non-sarcasm	0.93	0.90	0.92
		WeightedAvg	0.91	0.91	0.91
Sarcasm Corpus V2+Twitter (Gosh andVeale)	92.19%	Sarcasm	0.91	0.92	0.92
		Non-sarcasm	0.93	0.92	0.93
		WeightedAvg	0.92	0.92	0.92
News Headlines +Sarcasm Corpus V2	78.55%	Sarcasm	0.73	0.80	0.77
		Non-sarcasm	0.83	0.77	0.80
		WeightedAvg	0.79	0.79	0.78

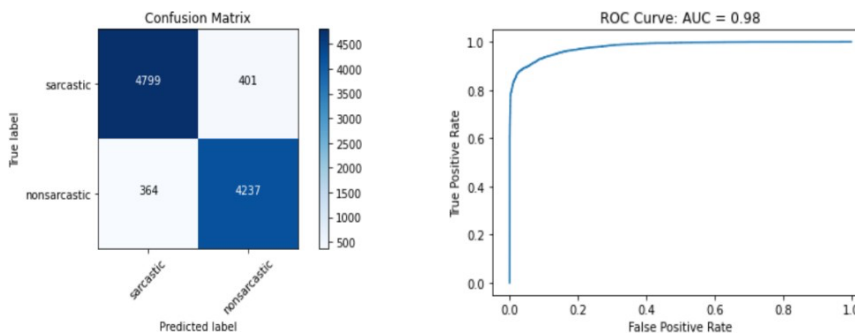
Table 4: Experimental results of multi-domain datasets (two datasets)

Reference	Datasets	Model	Accuracy	Recall	Precision	F1-score
[Jamil et al., 2021]	Tweets + News Headlines	CNN-LSTM	91.6%	91%	91%	91%
Proposed approach	Gosh Tweets + Sarcasm Corpus V2	GloVe + Bi-LSTM	92.19%	92%	92%	92%

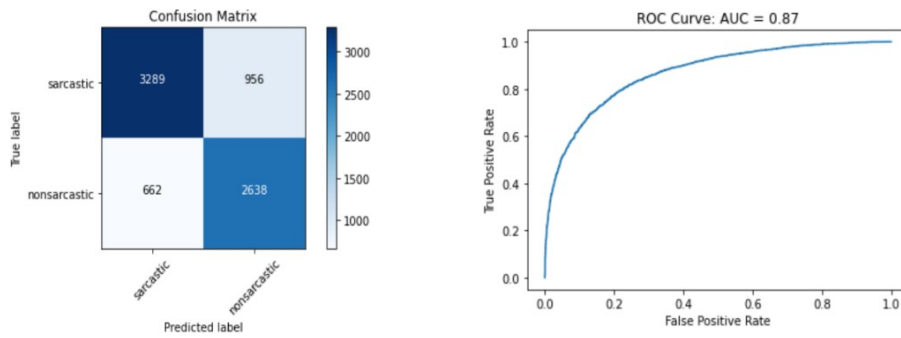
Table 5: Comparison of the proposed approach with existing approach using a multi-domain dataset (two datasets)



(a) Confusion matrix and AUC-ROC for “Twitter (Gosh and Veale) + newsheadlines”



(b) Confusion matrix and AUC-ROC for “Sarcasm corpus v2 + Twitter (Gosh and Veale)”



(c) Confusion matrix and AUC-ROC for “News headlines + Sarcasm corpus v2”

Figure 6: Confusion matrix and AUC-ROC for multi domain datasets (two datasets)

4.1.3 Experiment with the multi-domain dataset (three datasets)

In the last experiment, instead of concatenating two datasets to build a multi-domain dataset, this experiment concatenates three datasets from three distinct domains i.e. News Headlines [Oraby et al., 2016], Sarcasm Corpus V2 [Alcaide et al., 2015] and Gosh Tweets [Misra and Arora, 2019]). The results obtained from this experiment are presented in Table 6 which indicates that the proposed approach has gained tremendous success by obtaining 95.40% of accuracy. Contributing as a measure of performance, Figure 7 provides confusion matrix and AUC-ROC. The results of this experiment are taken into consideration as novelty of the proposed approach.

Dataset	Accuracy	Class	Recall	Precision	F1-score
Twitter (Gosh and Veale) + News Headlines + Sarcasm Corpus V2	86.35%	Sarcasm	0.87	0.85	0.86
		Non-sarcasm	0.86	0.88	0.87
		Weighted Avg	0.88	0.88	0.88

Table 6: Experimental results of the multi-domain dataset (three datasets)

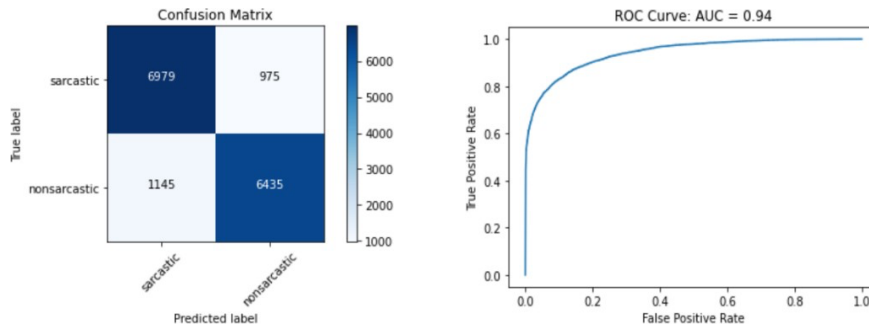


Figure 7: Confusion matrix and AUC-ROC for Multi domain dataset (three datasets)

The strategy used to develop this approach for sarcasm detection with a multi-domain dataset considerably improved the performance of the proposed approach. The usage of GloVe word embedding with the Bi-LSTM model has a remarkable contribution to the proposed approach in detecting sarcasm. The findings show that the proposed approach has outperformed on single dataset and even when the two different datasets have been combined. Even when more complexity has been increased by concatenating three different datasets from three different domains where each domain has its own linguistic characteristics and patterns, the model still attains a remarkable performance by achieving an accuracy of 86.35% along with 88% Recall, Precision, and F1-score on such a heterogeneous dataset. It indicates proposed model is generalized model which can identify sarcasm from a broader domain. The capability of the proposed model to detect sarcasm from such multi-domain dataset, is a testimony of its robustness and adaptability.

5 Conclusion

Due to social media's broad use, researchers have been interested in identifying sarcasm in recent years. Most of the research is based on sarcasm detection from a single domain dataset, and such techniques yield comparable stability but few studies have identified sarcasm in a multi-domain dataset. Sarcasm detection in itself is a very challenging task and its detection from multi-domain datasets makes it more challenging as the size, and complexity of data is increased with such data. In this study, a DL model, Bi-LSTM, with GloVe word embeddings has been presented to detect sarcasm from a multi-domain dataset. Three datasets from three different domains are used in this study including the Gosh Tweets dataset, News Headlines dataset, and Sarcasm Corpus V2 dataset that are further concatenated to form a multi-domain dataset. With the cross-validation technique namely, the "Hold-out-method", data has been split into an 80:20 ratio where 80% of data is used in training and 20% is used for testing. The proposed method has used GloVe word embeddings to get contextual and semantics features from the text as these values provide more information to the model. The model has achieved 86.35% accuracy and 88% Recall, Precision, and f1-score with a multi-domain dataset formed by concatenating three datasets.

Different experiments are also performed to testify the authenticity of the proposed model while varying the complexity of data from a single domain to a multi-domain. The model outperformed previous techniques in each trial, achieving 99.96% accuracy with a single domain dataset and 92.19% accuracy with a multi-domain dataset formed by the concatenation of two datasets. Even with a multi-domain dataset produced by concatenating three datasets, the model's prediction accuracy is 86.35%. The results indicate that the proposed GloVe+Bi-LSTM model can yield promising results with the multi-domain datasets while outperforming the results of existing state-of-the-art approaches that are based on a single domain dataset.

In the future, an attempt can be made to see if the proposed model can be used to spot sarcasm in other languages such as Roman Urdu language as Roman Urdu is being widely used in Asia. Moreover, a number of new datasets can be added to make the model more generalized with even more accuracy, and different types of sarcasm (Irony, Passive aggression, Insult, Humor, Satire, etc) can be considered for detection purposes as a future goal.

References

- [Abdi et al., 2019] Abdi et al.: 'Deep learning-based sentiment classification of evaluative text based on Multi-feature fusion'; *Information Processing & Management*, 55, 4 (2019). Retrieved from <https://www.sciencedirect.com/science/article/abs/pii/S0306457318307416>
- [Ahuja and Sharma, 2022] Ahuja, R., Sharma, S. C.: 'Transformer-Based Word Embedding With CNN Model to Detect Sarcasm and Irony'; *Arabian Journal for Science and Engineering*, 47, 8 (2022), 9379–9392. <https://doi.org/10.1007/s13369-021-06193-3>
- [Akula, 2021] Akula, R.: 'Interpretable Multi-Head Self-Attention Architecture for'; *Entropy*, 23, 394 (2021), 1–14.
- [Alcaide et al., 2015] Alcaide et al.: 'Combining Statistical and Semantic Knowledge for Sarcasm Detection in Online Dialogues'; *Pattern Recognition and Image Analysis (2015)*. Retrieved from https://link.springer.com/chapter/10.1007/978-3-319-19390-8_74#citeas
- [Bhardwaj and Prusty, 2022] Bhardwaj, S., Prusty, M. R.: 'BERT Pre-processed Deep Learning Model for Sarcasm Detection'; *National Academy Science Letters*, 45, 2 (2022), 203–208. <https://doi.org/10.1007/s40009-022-01108-8>
- [Bharti et al., 2017] Bharti, S. K., Naidu, R., Babu, K. S.: 'Hyperbolic Feature-based Sarcasm Detection in Tweets: A Machine Learning Approach'; 2017 14th IEEE India Council International Conference, INDICON 2017 (2017). <https://doi.org/10.1109/INDICON.2017.8487712>
- [Bhoir et al., 2017] Bhoir, S., Ghorpade, T., Mane, V.: 'Comparative analysis of different word embedding models'; *International Conference on Advances in Computing, Communication and Control 2017, ICAC3 2017, 2018-Janua (2017)*, 1–4. <https://doi.org/10.1109/ICAC3.2017.8318770>
- [Bouazizi and Otsuki, 2016] Bouazizi, M., Otsuki, T.: 'A Pattern-Based Approach for Sarcasm Detection on Twitter'; *IEEE Access*, 4 (2016), 5477–5488. <https://doi.org/10.1109/ACCESS.2016.2594194>

- [Chaudhari and Chandankhede, 2017] Chaudhari, P., Chandankhede, C.: ‘Literature survey of sarcasm detection’; Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking, WiSPNET 2017, 2018-Janua (2017), 2041–2046. <https://doi.org/10.1109/WiSPNET.2017.8300120>
- [Chen et al., 2011] Chen, L., Liu, C., Chiu, H.: ‘A neural network based approach for sentiment classification in the blogosphere’; Journal of Informetrics, 5, 2 (2011), 313–322. Retrieved from <http://dx.doi.org/10.1016/j.joi.2011.01.003>
- [del Pilar Salas-Zárate et al., 2020] del Pilar Salas-Zárate, M., Alor-Hernández, G., Sánchez-Cervantes, J. L., Paredes-Valverde, M. A., García-Alcaraz, J. L., Valencia-García, R.: ‘Review of English literature on figurative language applied to social networks’; Knowledge and Information Systems, 62, 6 (2020), 2105–2137. <https://doi.org/10.1007/s10115-019-01425-3>
- [Du et al., 2022] Du, Y., Li, T., Pathan, M. S., Teklehaimanot, H. K., Yang, Z.: ‘An Effective Sarcasm Detection Approach Based on Sentimental Context and Individual Expression Habits’; Cognitive Computation, 14, 1 (2022), 78–90. <https://doi.org/10.1007/s12559-021-09832-x>
- [Eke et al., 2021a] Eke, C. I., Norman, A. A., Shuib, L.: ‘Context-Based Feature Technique for Sarcasm Identification in Benchmark Datasets Using Deep Learning and BERT Model’; IEEE Access, 9 (2021a), 48501–48518. <https://doi.org/10.1109/ACCESS.2021.3068323>
- [Eke et al., 2021b] Eke, C. I., Norman, A. A., Shuib, L.: ‘Multi-feature fusion framework for sarcasm identification on twitter data: A machine learning based approach’; PLoS ONE (Vol. 16) (2021b). <https://doi.org/10.1371/journal.pone.0252918>
- [Gang Wang a, b et al., 2014] Gang Wang a, b, C., *, Jianshan Sun c, D., C, J. M., E, K. X., Jibao Gu d: ‘Sentiment classification: The contribution of ensemble learning’; Decision Support Systems, 57, 77–93 (2014). Retrieved from <https://www.sciencedirect.com/science/article/abs/pii/S0167923613001978>
- [Ghosh and Veale, 2017] Ghosh, A., Veale, T.: ‘Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal’; EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings, , 2002 (2017), 482–491. <https://doi.org/10.18653/v1/d17-1050>
- [Goel et al., 2022] Goel, P., Jain, R., Nayyar, A., Singhal, S., Srivastava, M.: ‘Sarcasm detection using deep learning and ensemble learning’; Multimedia Tools and Applications, 81, 30 (2022), 43229–43252. <https://doi.org/10.1007/s11042-022-12930-z>
- [Govindan and Balakrishnan, 2022] Govindan, V., Balakrishnan, V.: ‘A machine learning approach in analysing the effect of hyperboles using negative sentiment tweets for sarcasm detection’; Journal of King Saud University - Computer and Information Sciences, 34, 8 (2022), 5110–5120. <https://doi.org/10.1016/j.jksuci.2022.01.008>
- [Jamil et al., 2021] Jamil, R., Ashraf, I., Rustam, F., Saad, E., Mehmood, A., Choi, G. S.: ‘Detecting sarcasm in multi-domain datasets using convolutional neural networks and long short term memory network model’; PeerJ Computer Science, 7 (2021), 1–24. <https://doi.org/10.7717/peerj-cs.645>
- [Justo et al., 2014] Justo et al.: ‘Extracting relevant knowledge for the detection of sarcasm and nastiness in the social web’; (2014).
- [Karim et al., 2017] Karim, F., Majumdar, S., Darabi, H., Chen, S.: ‘LSTM Fully Convolutional Networks for Time Series Classification’; IEEE Access, 6 (2017), 1662–1669. <https://doi.org/10.1109/ACCESS.2017.2779939>

- [Kumar and Garg, 2023] Kumar, A., Garg, G.: ‘Empirical study of shallow and deep learning models for sarcasm detection using context in benchmark datasets’; *Journal of Ambient Intelligence and Humanized Computing*, 14, 5 (2023), 5327–5342. <https://doi.org/10.1007/s12652-019-01419-7>
- [Lecun et al., 2015] Lecun, Y., Bengio, Y., Hinton, G.: ‘Deep learning’; *Nature*, 521, 7553 (2015), 436–444. <https://doi.org/10.1038/nature14539>
- [Liu et al., 2014] Liu, P., Chen, W., Ou, G., Wang, T., Yang, D., Lei, K.: ‘Sarcasm detection in social media based on imbalanced classification’; *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8485 LNCS (2014), 459–471. https://doi.org/10.1007/978-3-319-08010-9_49
- [Mandal and Mahto, 2019] Mandal and Mahto: ‘Deep CNN-LSTM with word embeddings for news headlines sarcasm detection’; (2019).
- [Maynard and Greenwood, 2014] Maynard, D., Greenwood, M. A.: ‘Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis’; *Proceedings of the 9th International Conference on Language Resources and Evaluation, LREC 2014* (2014), 4238–4243.
- [Medhat et al., 2014] Medhat, W., Hassan, A., Korashy, H.: ‘Sentiment analysis algorithms and applications: A survey’; *Ain Shams Engineering Journal*, 5, 4 (2014), 1093–1113. <https://doi.org/10.1016/j.asej.2014.04.011>
- [Misra and Arora, 2019] Misra, R., Arora, P.: ‘Sarcasm Detection using Hybrid Neural Network’; *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)* (Vol. 1). Association for Computing Machinery (2019). <https://doi.org/10.13140/RG.2.2.32427.39204>
- [Mukherjee and Bala, 2017] Mukherjee, S., Bala, P. K.: ‘Sarcasm detection in microblogs using Naïve Bayes and fuzzy clustering’; *Technology in Society*, 48 (2017), 19–27. <https://doi.org/10.1016/j.techsoc.2016.10.003>
- [Nagwanshi and Veni Madhavan, 2014] Nagwanshi, P., Veni Madhavan, C. E.: ‘Sarcasm detection using sentiment and semantic features’; *KDIR 2014 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (2014), 418–424. <https://doi.org/10.5220/0005153504180424>
- [Onan, 2019] Onan, A.: ‘Two-Stage Topic Extraction Model for Bibliometric Data Analysis Based on Word Embeddings and Clustering’; *IEEE Access*, 7 (2019), 145614–145633. <https://doi.org/10.1109/ACCESS.2019.2945911>
- [Onan and Tocoglu, 2021] Onan, A., Tocoglu, M. A.: ‘A Term Weighted Neural Language Model and Stacked Bidirectional LSTM Based Framework for Sarcasm Identification’; *IEEE Access*, 9 (2021), 7701–7722. <https://doi.org/10.1109/ACCESS.2021.3049734>
- [Oraby et al., 2016] Oraby, S., Harrison, V., Reed, L., Hernandez, E., Riloff, E., Walker, M.: ‘Creating and Characterizing a Diverse Corpus of Sarcasm in Dialogue’; *SIGDIAL 2016 - 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, Proceedings of the Conference* (2016), 31–41. <https://doi.org/10.18653/v1/w16-3604>
- [Pandey et al., 2021] Pandey, R., Kumar, A., Singh, J. P., Tripathi, S.: ‘Hybrid attention-based Long Short-Term Memory network for sarcasm identification’; *Applied Soft Computing*, 106 (2021), 107348. <https://doi.org/10.1016/j.asoc.2021.107348>

- [Peng et al., 2019] Peng, W., Adikari, A., Alahakoon, D., Gero, J.: ‘Discovering the influence of sarcasm in social media responses’; *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9, 6 (2019), 1–6. <https://doi.org/10.1002/widm.1331>
- [Pennington et al., 2014] Pennington et al.: ‘Glove: Global vectors for word representation’; In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2014).
- [Poria et al., 2016] Poria, S., Cambria, E., Hazarika, D., Vij, P.: ‘A deeper look into sarcastic tweets using deep convolutional neural networks’; *COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers* (2016), 1601–1612.
- [Razali et al., 2021] Razali, M. S., Halin, A. A., Ye, L., Doraisamy, S., Norowi, N. M.: ‘Sarcasm Detection Using Deep Learning with Contextual Features’; *IEEE Access*, 9 (2021), 68609–68618. <https://doi.org/10.1109/ACCESS.2021.3076789>
- [Reyes et al., 2013] Reyes, A., Rosso, P., Veale, T.: ‘A multidimensional approach for detecting irony in Twitter’; *Language Resources and Evaluation*, 47, 1 (2013), 239–268. <https://doi.org/10.1007/s10579-012-9196-x>
- [Rosso, 2012] Rosso, R. and: ‘Making objective decisions from subjective data: Detecting irony in customer reviews’; *Support Systems* (2012).
- [Savini et al., 2019] Savini, E., Caragea, C., Erdem, A., Elena, K., Baralis, M., Di Torino, P.: ‘An Exploration of Sarcasm Detection Using Deep Learning’; (2019). Retrieved from <https://webthesis.biblio.polito.it/12440/1/tesi.pdf>
- [Shrivastava and Kumar, 2021] Shrivastava and Kumar: ‘A pragmatic and intelligent model for sarcasm detection in social media text’; (2021).
- [Siami-Namini et al., 2019] Siami-Namini, S., Tavakoli, N., Namin, A. S.: ‘The Performance of LSTM and BiLSTM in Forecasting Time Series’; *Proceedings - 2019 IEEE International Conference on Big Data, Big Data 2019* (2019), 3285–3292. <https://doi.org/10.1109/BigData47090.2019.9005997>
- [Swami et al., 2018] Swami, S., Khandelwal, A., Singh, V., Akhtar, S. S., Shrivastava, M.: ‘A Corpus of English-Hindi Code-Mixed Tweets for Sarcasm Detection’; (2018), 1–9. Retrieved from <http://arxiv.org/abs/1805.11869>
- [Verma et al., 2021] Verma, P., Shukla, N., Shukla, A. P.: ‘Techniques of Sarcasm Detection: A Review’; *2021 International Conference on Advance Computing and Innovative Technologies in Engineering, ICACITE 2021*, 7 (2021), 968–972. <https://doi.org/10.1109/ICACITE51222.2021.9404585>
- [Vinoth and Prabhavathy, 2022] Vinoth, D., Prabhavathy, P.: ‘An intelligent machine learning-based sarcasm detection and classification model on social networks’; *Journal of Supercomputing*, 78, 8 (2022), 10575–10594. <https://doi.org/10.1007/s11227-022-04312-x>
- [Vyas and Uma, 2018] Vyas, V., Uma, V.: ‘Approaches to Sentiment Analysis on Product Reviews’; *MI* (2018), 15–30. <https://doi.org/10.4018/978-1-5225-4999-4.ch002>
- [Wang et al., 2015] Wang, Z., Wu, Z., Wang, R., Ren, Y.: ‘Twitter Sarcasm Detection Exploiting a’; (2015), 332–336. <https://doi.org/10.1007/978-3-319-26190-4>
- [Xiong et al., 2019] Xiong, T., Zhang, P., Zhu, H., Yang, Y.: ‘Sarcasm detection with self-matching networks and low-rank bilinear pooling’; *The Web Conference 2019 - Proceedings of the World Wide Web Conference, WWW 2019* (2019), 2115–2124. <https://doi.org/10.1145/3308558.3313735>

[Yue et al., 2019] Yue, L., Chen, W., Li, X., Zuo, W., Yin, M.: 'A survey of sentiment analysis in social media'; *Knowledge and Information Systems*, 60, 2 (2019), 617–663. <https://doi.org/10.1007/s10115-018-1236-4>

[Yung-Ming Li, 2013] Yung-Ming Li, T.-Y. L.: 'Deriving market intelligence from microblogs'; *Decision Support Systems*, 55, 206–217 (2013). Retrieved from <https://www.sciencedirect.com/science/article/abs/pii/S0167923613000511>