Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 04 Issue 02
2023

Research Article
https://doi.org/10.56979/402/2023

Sarcasm Detection on Twitter using Deep Handcrafted Features

Hasnat Saleem[1*], Ahmad Naeem[1], Kamran Abid[1], and Naeem Aslam[1]
[1]Department of Computer Science, NFC-IET, Multan, 60000, Pakistan.
*Corresponding Author: Hasnat Saleem. Email: hasnatsaleem80@gmail.com.

Received: January 08, 2023 **Accepted:** February 20, 2023 **Published:** March 29, 2023.

Abstract: The recent advancement of social media has greatly impacted people's daily lives. People nowadays express their emotions through social media. Twitter is the most utilized social media platform for people to share information. As social media popularity is increasing a lot of information available on the internet becomes dubious and misleading. The sarcastic text is a true lie spread through social media and it is a statement that is different from the actual message. It is very challenging to recognize sarcasm from social media manually. Therefore, the detection of sarcasm is essential from social media using an advanced automated system based on deep learning methods. In this study, we have proposed a novel method for the detection of sarcasm. In this study, BoW, TF-IDF, and word embeddings are used to detect the prominent features from the text and a long-short memory (LSTM) network for identifying sarcastic remarks in a given corpus. The publicly available Twitter dataset is used in this study which is based on sarcasm, irony, and regular tweets. To evaluate the methods, we have used recall, precision, F1, and accuracy score as the evaluation parameters. The proposed model achieved a 99.01% accuracy for the detection of sarcasm on social media.

Keywords: Deep learning; Sarcasm detection; Social media; Feature engineering; Twitter.

1. Introduction

Sarcasm is an insidious social media phenomenon expressing biased and strongly felt opinions[1]. These kinds of opinions can be delivered through social media as social media provides a platform where people can openly express their feelings which might either have or not have a harmful impact on others. The early researches describe some of the benefits of sarcasm, such as the sarcastic text boosting creativity for problem-solving. The snarky text is a true lie embedded in the message statement. The meaning of a sarcastic text is different compared to the actual message. People use sarcastic text for criticism with indirectness on social media.

The sarcasm text associates exposing or constructing contradictions between intended meanings. Early research explored how sarcasm makes a statement sound more critical and easily misinterpreted. Sarcasm on social media can be spread due to numerous reasons[2]. People adopt sarcasm in social media texts based on their social behavior and daily life activities. The following are some of the primary causes of sarcasm in social media: Feeling insecure, Latent anger, and social awkwardness. Due to the complexity of automatic sarcasm detection, previous studies commonly used rule-based and statistical techniques. The classical approaches are only able to learn the text's importance. The advanced deep learning approach can be applied to understand the contextual information from the sarcasm text. Sarcasm is the study of extracting the emotion and sentiment related to any entities or discussion. The emotion of people about sarcasm expresses their negative or positive sentiments or opinions by using the relevant word in the text. For daily life activities, sarcastic opinions related to the products can be important for business organizations. Based on the extracted sentiment help to construct decision support systems. So, the timely detection of sarcasm from social media helps to overcome the stress and negativity spreading by expressing sarcasm.

There are numerous classical techniques to detect sarcasm from social media. The detection of sarcasm text is based on the five most common methods[3]follow as. The sarcasm in social media text is difficult to detect because of the absence of verbal tone. The verbal tone is mostly considered to convey sarcasm when speaking. Suppose a user on social media is being sarcastic in conservations. The user may add multiple letters to common words to indicate a sarcastic tone in the text. The user on social media might use words to indicate an elongated syllable. For example, if a user on social media posts a statement that other members do not believe, they may reply by posting "Right." However, stretching out the word sarcastically. We can say that the use of the word "Right" in the text is a form of sarcasm. The use of intense adjectives usually refers to hyperbolic language, which also indicates sarcasm in the text. Suppose a user on social media is enthusiastic about a subject that user is being hyperbolic, which is an indicator of sarcasm in text[20][27]. The more intense version of common words indicates hyperbole, which is sarcasm. The references to popular culture mentioned in the text can indicate sarcasm in the text on social media. We can check if a user is being sarcastic if the user uses any references in the post text. For example, the user puts his political view in the message by citing or referencing another source. The user may also ask a question that also indicates sarcasm. Sarcasm is commonly spread on social media by a user who is frustrated or angry. So, if reading a text post feels aggressive, it seems to be sarcastic. If users post heated arguments on social media, they spread sarcasm. The text messages words on social media can be analyzed by the meaning of words around them. The sarcasm can be detected by examining the context of a message. The potentially sarcastic portion of the message can be examined carefully, which may lead to sarcasm.

The contextual and textual features from Twitter tweet messages can be used for various text detection strategies such as emotion analysis, sarcasm detection, and many more[4]. The textual feature-based classification is utilized to categorize sarcasm-related social media posts and not sarcasm posts. In the textual dataset, many text characteristics can be used to determine the sarcasm in the text. The most commonly used textual feature forms are described below. The meaning or semantic aspect in the social media-related text is captured using its semantic features[5][26]. The text posts are then filtered by using semantic-based approaches to classify the text as sarcasm or not sarcasm. The meaningful patterns and insights from the textual data by utilizing the semantic features. The linguistic or lexical features represented the frequency of unique words extracted by using the BOW or TFIDF vectorization techniques to detect sarcasm from social media text. The punctuations, pronouns, verbs, and hashtags in the text related to sarcasm are the forms of linguistic features. The text recognition-based artificial intelligence techniques primarily used word count, dictionaries, and psycholinguistic features[6][12]. These features refer to the psycholinguistic features, which can be used to detect sarcasm from textual data.

To build an automated tool for detecting sarcasm from social media, we have used the Twitter platform[7]. In this research study, we used tweets with sarcasm to conduct our research experiments. Twitter is the most popular public social media tool, which was launched in 2006. According to a recent state, by using Twitter daily, 500 million tweets are uploaded with 100 million daily active Twitter users. Nowadays, people are so friendly to use Twitter. They are sharing their thoughts and ideas with a big audience. The events and news are commonly shared through Twitter. Twitter users can easily post a short message and tag the people they follow. Sarcasm detection is emerging as an important task for text classification and can be beneficial to many Text Mining applications, such as sentiment analysis, opinion mining, and marketing[1]. So, for sarcasm classification, data will be analyzed by the machine learning algorithm and neural networks after applying feature selection methods. Furthermore, these models will be evaluated by some evaluation parameters such as accuracy, precision, TNR TPR, recall, and f1-score[29].

Nowadays, deep learning techniques are highly involved in solving NLP-related tasks such as text classification, emotion analysis, and many more[8]. The deep learning-based models mimic human brain neural networks. So, deep learning-based neural networks can learn text over a long time for better performance. Deep learning neural networks utilize layers to process data. The deep learning models can able to learn long text sequences with high efficiency. There are a lot of challenges associated with sarcasm detection rightly. The sarcastic content on social media is written very politely, which is very difficult to detect[9]. People on social media use politeness to be sarcastic with highly formal words. So, the formation of advanced deep learning-based methods is tough, which can detect sarcasm online with high efficiency and less time consumption. There is another challenge the availability of a huge textual dataset. A big enough dataset is an issue, so applying the deep learning models well in learning about the sarcasm text.

The availability of huge sarcasm-related databases results in achieving high accuracy for sarcasm detection[17][20]. Our proposed study covers these challenges using advanced deep learning models and a tweet-based huge sarcasm-related database. The sarcasm must be handled on social media instantly as it might cause rumors, identity theft, compromising of confidentiality and authenticity, fake profiles, and likewise. The sarcasm on social media is a threat to societal stability. The sarcasm attracts greater attention and creates a deeper impact than other negative responses on social media. Since sarcasm is the opposite of what is said or written, therefore, it is difficult to reveal the nature of that comment. The manual detection of sarcasm text is very difficult and time-consuming. An artificial intelligence-based automated system must be built to detect sarcasm from social media[19]. The research aims to reduce and overcome anxiety through social media using an automated system through the investigation of numerous textual features and properties that can help in the detection of sarcasm from online content. Moreover, it automatically predicts and detects sarcasm without human intervention/efforts by utilizing deep learning and machine learning techniques[25]. Following are the key contribution of proposed method

1. In this study, BoW, TF-IDF, and word embeddings are used to detect the prominent features of the text.
2. The LSTM uses the above mention features to detect sarcasm, irony and normal tweets. To better detect sarcasm, our algorithm highlights key phrases and clauses in the text.
3. Using our data, we can replicate the findings of state-of-the-art models, allowing us to draw meaningful comparisons. As compared to competing models, our suggested model performs better.

The rest of the article is organized as follows.

Section 2 discusses the material and methods. Experimentation and results are discussed in section 3. Section 4 contains the conclusion.

2. Materials and Methods

The study workflow architecture for detecting sarcasm from social media text is analyzed in Figure 1. A textual dataset collected from Twitter social media platform is used for conducting our study experiments. The tweet's text is pre-processed to remove unwanted noise from it. Numerous textual feature engineering techniques such as BOW, TFIDF, and word embedding are applied. The pre-processed textual dataset is then split into two portions. One portion is a train set based on 80% of the dataset, and the other portion is a test set based on 20% of the dataset. The three-machine learning and two deep learning-based methods are applied to the dataset. The deep learning-based models LSTM and RNN are compared. The applied models are trained with a training set. The outperformed LSTM model was tested in real-time with a testing set. The outperformed model predictions are validated using different evaluation parameters. The accuracy, precision, recall, and f1 score are the model evaluation parameters used in this study.

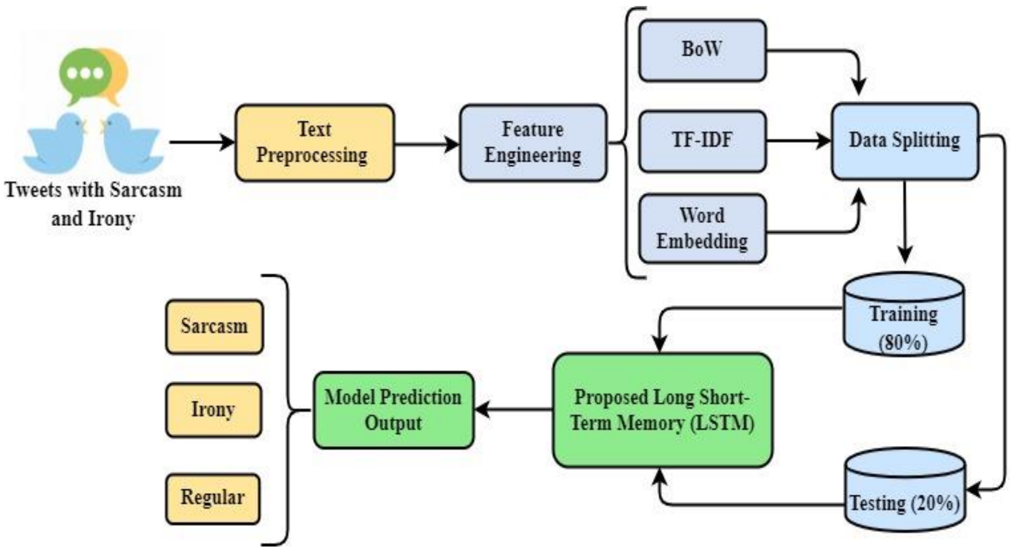

Figure 1. Study methodology analysis for sarcasm detection

2.1 Data Set

The textual dataset utilized in this research study is open-source. The dataset is publicly available online at the famous data repository Kaggle[16]. The dataset is formed by collecting tweets from Twitter social media platforms related to sarcasm and irony. The text dataset based on 60170 tweets is used and classified into one of the three target classes Regular, Sarcasm, and Irony. The dataset file contains two columns. The tweet column represents the textual message posted by a Twitter user, and the class column represents the target label categorized as Regular, Sarcasm, and Irony in Table 1.

Table 1. The dataset features descriptive analysis.

Feature	Distributions	Data type	Description
Tweet	60170	object	The text of the tweet
Class	60170	object	The respective class to which the tweet belongs

The below figure 2, contains 20681 for sarcasm 20894 for irony and 18595 tweets of regular comments.

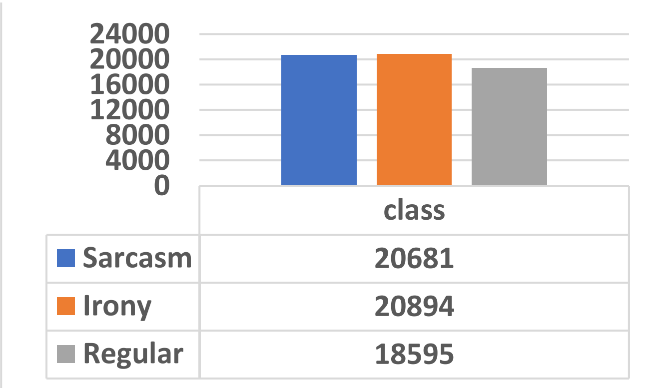

Figure 2. The target class-based data distribution analysis.

2.2 Text Pre-processing

The advanced text pre-processing techniques are applied to the sarcasm textual dataset to clean unwanted noise. The noise in the text causes the learning model barrier to detect sarcasm from social media text with high accuracy. Due to this reason, we have cleaned the dataset and made it pre-processed for applied machine and deep learning models. Initially, we dropped the null values from the dataset and converted the tweets column into string type. After pre-processing, the final dataset contains 60170 tweets. The followings are our stepwise text-cleaning strategies.

2.2.1 Lower case conversion

We have converted each tweet text into lowercase as it helps in the process of pre-processing. The technique removes the case-sensitivity of text which is beneficial for the text mining model for learning text sequences with similar patterns.

2.2.2 Stop words removal

We have cleaned our tweet text dataset by removing unnecessary stop words. Removal of stop words involves many advantages. The stop words are generally based on the pronouns and articles of languages. The stop words in the text are “the,” “are,” “a,” and many more. It reduces the text sequence size and training time of a learning model due to the valuable token involved[18]. The stop words removal enhances the performance of the applied machine and deep learning models.

2.2.3 Punctuation removal

The unwanted punctuations from the text of the tweet are removed during the pre-processing. They increase the overhead for a learning model during training. The punctions include question marks (?), commas (,), exclamation marks (!), colons (:), apostrophes ('), and many more[19]. These types of punctions are removed from each tweet text from our study dataset.

2.2.4 Stemming/lemmatize

We have applied the lemmatization/stemming techniques to our tweet textual dataset. The lemmatization normalizes the textual data. It converts the different inflected forms of words into the root form, which has the same meaning. This technique reduces the complexity of a learning model and results in achieving a high-performance score in text classification . For example, in tweet text, the word "caring" would be normalized as "care," which is the more convenient form of text for a learning model.

2.2.5 Tokenization

Finally, the pre-processed tweets text is converted into clean tokens, which it refers as tokenization. After removing all unwanted noise, the text is tokenized and converted into a sentence. The sentences are then converted into a clean token of words[8]. The words are then updated with tweet text related to the sarcasm.

2.3 Feature engineering

The numerous features of engineering techniques applied to the sarcasm related to the tweet textual data are analyzed in this section. The BOW, TFIDF, and word embedding-based textual data representations techniques are applied to build machine learning and deep learning models. The performance comparisons result by using these three feature engineering techniques are evaluated[13][24]. These used feature extraction techniques are described in this section.

2.3.1 Bag of Word

The Bag of Word (BOW) based feature engineering technique is applied to extract features from our textual dataset data[14][23]. The BOW feature extraction technique converts textual data into meaningful numerical data. The numerical data is based on the presence of the words in a particular sentence. The occurrence of a word in a sentence is represented by each encoded value in the numerical data. If a word is present in a sentence, the outcome is 1 else 0 in the representation feature map. The Bow has many disadvantages, such as no ordering of the words in the text, no information on the grammar of the sentences, and high computation resource.

2.3.2 Term Frequency-Inverse Document Frequency

The Term Frequency-Inverse Document Frequency (TFIDF) based feature engineering is conducted to transform the text of the tweet into a machine-learning feature vector[15]. The TFIDF is a numerical statistic method that is used to determine whether a word is to a document in a collection or corpus. The importance of each word is preserved by the TFIDF feature extraction technique compared to BOW[21]. The TF is a measure of how frequently a word appears in a text. The IDF determined the importance of each word in a text. The TFIDF has the disadvantage that it does not understand the context of a word in the text. The TFIDF feature representation map is calculated by the difference between TF and IDF. The formula of calculation TFIDF is represented as:

$$tf - idf = tf_{td} * \log(idf) \quad (1)$$

2.3.3 Word Embedding

As the BOW and TFIDF feature extraction techniques failed to understand the context of a word in the text, the word embedding vector concept is introduced. The word embedding feature engineering technique is applied to vectorize our tweet's textual dataset for deep learning techniques[16]. Each text in the tweet is converted and transformed into a sequence of integers. Every integer in the sequence is the index of a token in a dictionary. The sequence vector has the coefficient for each token that could be binary.

2.4 Dataset splitting

The tweets-based dataset used in this study is split into two portions with a splitting ratio of 80:20. The 80% set of our study textual dataset is used to train the applied machine learning and deep learning methods. To validate the performance of each applied model, we tested with 20% of the data. The data splitting in our study results in preventing the applied model's overfitting. The data splitting validates our study results from performance in real-time. The study models are now in a generalized form to detect the sarcasm from the social media text.

2.4.1 Long Short-Term Memory

Long Short-Term Memory (LSTM)[13]is an advanced neural network-based deep learning technique that is commonly used to solve classification problems. Generally, the LSTM model is used to learn the long sequence data, such as the long text comments data. The LSTM model learns and extracts patterns from long sequence datasets efficiently. The LSTM is an extended version of the RNN model to overcome some issues of the RNN model. The chain structure of the LSTM model is made up of neural networks and many memory blocks. The long-time sequence learning of the property of the LSTM model is achieved by using the three-gate named input, output, and forget gate. The flow of data patterns in the neural network cells is controlled by the gates in the LSTM model. Unlike a standard RNN unit, an LSTM unit keeps a memory cell Ct at time t. The output ht of an LSTM unit is determined by using the calculations below:

1. $f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f)$
2. $i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i)$
3. $C_t = tanh(W_c.[h_{t-1}, x_t] + b_c$
4. $o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o)$
5. $h_t = h_t * \tanh(c_t)$
6. $C_t = \sigma(f_t \times C_{t-1} + i_i \times C_t)$

2.5 Training and testing of applied models

All applied machine learning and deep learning techniques are trained and tested with the split dataset parts. When the training of each applied model is completed with 80% data portion, then each model is tested with 20% data and ready to detect the sarcasm from social media text.

2.6 Applied model's evaluations parameters

Each applied model is tested and evaluated based on some scientific evaluation's metric parametric. The four advanced metrics are used for the performance evaluation of applied models. The accuracy, precision, recall, and f1 scores are evaluation performance metrics used in this research study. The following are the basic notations of evaluations metrics parameters[16].

- True Positive (TP): The correct positive predictions of a target class are referred to as True Positive (TP).
- True Negative (TN): The correct negative predictions of a target class are referred to as True Negative (TN).
- False Positive (FP): The incorrect positive predictions of a target class are referred to as False Positives (FP).
- False Negative (FN): The incorrect negative predictions of a target class are referred to as False Negative (FN).

3. Experimentation and Results

The scientific results validations and results discussions are performed in this chapter. The results evaluations for each applied machine learning and deep learning model are comparatively analyzed. The results with different feature engineering techniques are described in detail.

3.1 Experimental setup

The scientific experimental preparations for conducting the research experiments are analyzed. The hardware system specifications are described in Table 2 with each system parameter. The applied machine and deep learning models are built using Sklearn API with version 1.0.2, Keras with version 2.9.0, and TensorFlow with version 2.9.2. The accuracy, precision, recall, and f1 score are our model evaluation parameters for sarcasm detection.

Table 2. The system parameters and specifications to conduct our study experiments.

System parameter	Specification
Vendor_id	Genuine Intel
Model	79
Model name	Intel(R) Xeon(R) CPU @ 2.20GHz
CPU MHz	2199.998
Cache size	56320 KB
Address sizes	46 bits physical, 48 bits virtual

3.1.1 Applied models

The hyperparameter tuning techniques are applied during the performance results evaluations of each applied method. To enhance the performance metrics scores, hyperparameter tuning is applied to each machine learning and deep learning model. The best-fit hyperparameters on which we achieved a high score for sarcasm detection are determined. A recursive process of training and testing is applied to find the best-fit hyperparameters. The applied machine learning and deep learning hypermeters are analyzed in Table 3.

Table 3. The hypermeters analysis of applied deep learning models.

Technique	Hyperparameters
LSTM	activation='softmax', loss = 'categorical_crossentropy', optimizer = 'adam', metrics='accuracy'
RNN	activation='softmax', loss = 'categorical_crossentropy', optimizer = 'adam', metrics='accuracy'

3.2 Results with deep learning

The performance results of applied neural network-based advanced deep learning methods are analyzed in this section. The text sequence learning-based deep learning methods are applied in comparison. The applied RNN model works using feedback loops in the recurrent layer, which maintain information in memory over time The accuracy, precision, recall, and f1 scores are calculated for each applied machine-learning model.

3.2.1 Results with Word Embeddings

The word embedding-based features are used to build the applied deep learning models, and the results are evaluated. Table 4 contains the results of each applied deep learning method for all performance metrics. The word embedding features are created by converting each tweet text into a sequence of integers which are the index of a token in a dictionary. The analysis shows that the deep learning model LSTM outperformed with a 99.01% accuracy score for sarcasm detection from text.

Table 4. The results of applied deep learning models with word embedding features.

Technique	Accuracy score (%)	Precision score (%)	Recall score (%)	F1-score (%)
LSTM	99.01	98.25	97.14	98.11
RNN	96.34	97.14	96.12	95.45

The vertical bar chart-based comparative performance analysis of applied deep learning methods is visualized in Figure 3. The shows that the LSTM model achieved a 99.01% score for accuracy, and 98.25% precision metric. The applied RNN model also achieved good scores, but less than the LSTM in comparison.

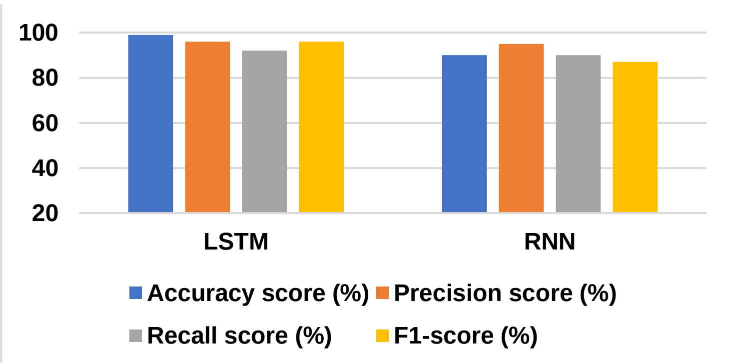

Figure 3. The performance comparison of applied deep learning methods with word embedding features

The time series-based performance comparative analysis of the applied deep learning method LSTM during training is visualized in Figure 4. The time series line graphs contain the results of the LSTM model for ten epochs of training. The results values are based on the training loss and train accuracy. This analysis concluded that the applied LSTM model achieved above 80% performance score during training and validation. The applied LSTM is in generalized form for detecting sarcasm from social media text.

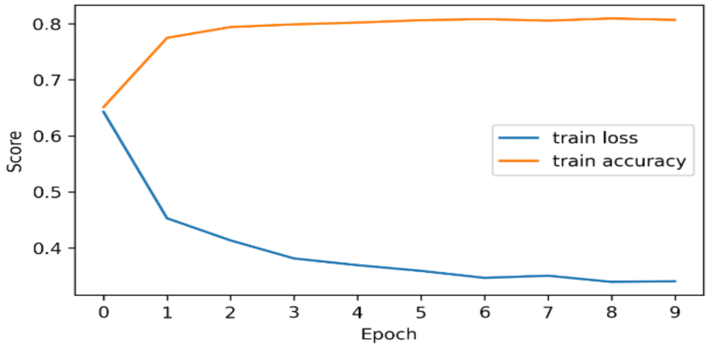

Figure 4. The time series-based performance comparison LSTM model during training.

The confusion matrix analysis of the applied deep learning-based LSTM method is analyzed in Fig 5. The confusion matrix's results of the applied model are extracted by using the word embedding features. The confusion matrix results summarize the performance of our applied deep learning-based LSTM technique using word embedding features in this analysis. The confusion matrix analysis is based on the three classes as the target. This analysis demonstrates that the applied LSTM model has minimum values of wrong predictions, which shows that they achieved a 99.01% performance accuracy score.

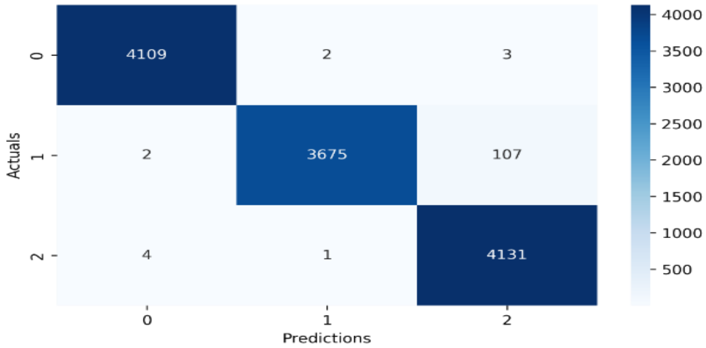

Figure 5. The confusion matrix applied deep learning-based LSTM technique.

The time series-based performance comparative analysis of the applied deep learning method RNN during training is visualized in Figure 6. The analysis line graphs are based on the results of the RNN model for ten epochs during the training. The training loss and train accuracy are the results of comparative values in the graph. This analysis concluded that the applied RNN model achieved above 70% performance score during training and validation. The applied RNN is in generalized form for detecting sarcasm from social media text.

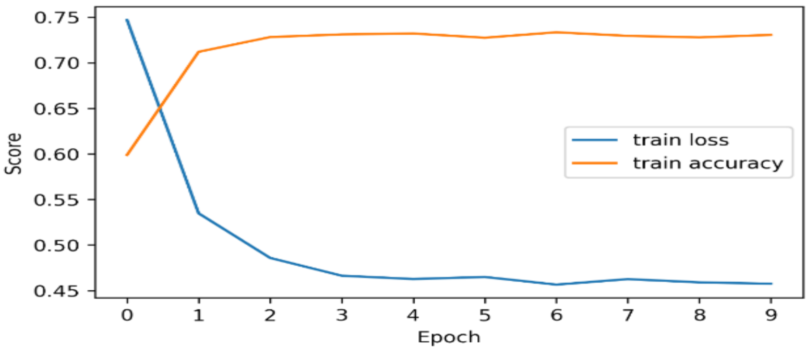

Figure 6. The time series-based performance comparison RNN model during training.

The confusion matrix analysis of the applied deep learning-based RNN method is analyzed in Fig 7. The word embedding features are used to extract the confusion matrix's results of the applied model. The confusion matrix results summarize the performance of the applied deep learning-based RNN technique using word embedding features in this analysis. The three classes as a target are utilized to build the confusion matrix analysis. In conclusion, the RNN model achieved an acceptable score for sarcasm detection in this study due to the minimum confusion matrix error rate in comparisons.

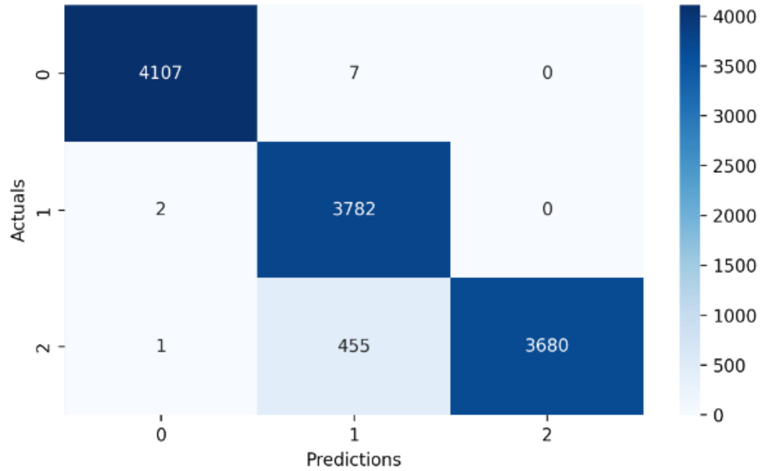

Figure 7. The confusion matrix applied deep learning-based RNN technique.

3.3 Result of the proposed LSTM technique

After the comparisons of the extensive results, we come to conclude that the deep learning-based LSTM model is outperformed with high-performance accuracy scores of 99.01%. The results of the proposed LSTM model for all performance metrics are analyzed in Table 5.

Table 5. The results of the proposed deep learning model with word embedding features.

Technique	Accuracy score (%)	Precision score (%)	Recall score (%)	F1-score (%)
LSTM	99.01	98.25	97.14	98.11

The bar chart-based performance metrics comparison analysis of the proposed LSTM model is visualized in Figure 8. The comparison analysis shows that the proposed approach achieved a 99.01% score for all evaluated performance metrics in this study.

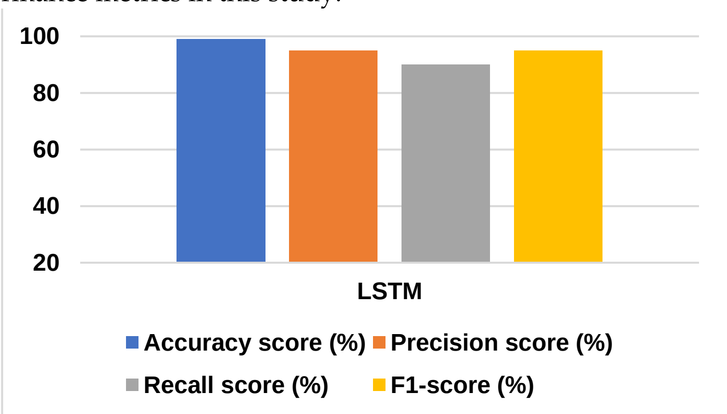

Figure 8. The performance comparison of the proposed deep learning method with word embedding features.

4. Conclusion

Sarcasm refers to the use of the communicating language that normally signifies the opposite to convey contempt. Sarcasm text is an insidious social media phenomenon expressing biased and strongly felt opinions. A sarcastic text statement's meaning is different from the actual message. On social media, people use sarcastic text for criticism with indirectness. Sarcasm is associated with exposing or constructing contradictions between intended meanings. The recognition of sarcasm from social media manually is a very challenging task. To detect and overcome sarcasm from social media text, we have proposed an advanced automated system based on a deep learning model in this study. The deep learning-based RNN and LSTM are the applied models to detect sarcasm from social media text. The Twitter dataset is split into a ratio of 80:20. The 80% portion of the dataset is used to train the applied machine learning and deep learning models. The applied methods are tested in real time with 20% of the data. Each applied method is fully hyperparameter tuned. The best-fit hypermeters of deep learning models are observed. The repeated process of training and testing is performed to find the best hypermeters. The hyperparameter tuning results in achieving high-performance accuracy scores for the detection of sarcasm from social media text.

In our current research study, we have applied deep learning techniques for detecting sarcasm. The transfer learning base techniques will be applied in the future. Our study used the Twitter tweet dataset for building the applied models. However, in the future, the textual dataset from different social media platforms will be used to build the models, such as Tinder, Facebook, Reddit, and many more. Sarcasm detection from different spoken language textual datasets will be performed, such as Urdu, Hindi, Spanish, and many more. Current research mostly focuses on textual datasets. However, in the future, images-based datasets and audio-based datasets will also be used to detect sarcasm. The visual-based dataset can be collected from different TV shows and YouTube channels. The facial expressions will be used to detect sarcasm from visual data.

Reference:

1. Eke CI, Norman AA, Liyana Shuib, Nweke HF.(2020). Sarcasm identification in textual data: systematic review, research challenges and open directions. Artif Intell Rev, 53, 4215-4258. doi:10.1007/S10462-019-09791-8/METRICS
2. Malave N, Dhage SN. (2020) Sarcasm detection on twitter: User behavior approach. Adv Intell Syst Comput,(pp:65-76). doi:10.1007/978-981-13-6095-4_5/COVER
3. Michelle G. 3 (2022) Ways to Detect Sarcasm in Writing - wikiHow.
4. Marmolejos L, AlOmar EA, Mkaouer MW, Newman C, Ouni A (2022). On the use of textual feature extraction techniques to support the automated detection of refactoring documentation. Innov Syst Softw Eng. 233-249. doi:10.1007/S11334-021-00388-5/METRICS
5. Yu Y, Cao H, Yan X, Wang T, Ge SS (2020). Defect identification of wind turbine blades based on defect semantic features with transfer feature extractor. Neurocomputing. 376, (pp.1-9). doi:10.1016/J.NEUCOM.2019.09.071
6. Mehta Y, Fatehi S, Kazameini A, Stachl C, Cambria E, Eetemadi S.(2020, November) Bottom-up and top-down: Predicting personality with psycholinguistic and language model features. Proc - IEEE Int Conf Data Mining, ICDM. (pp.1184-1189). doi:10.1109/ICDM50108.2020.00146
7. Yao F, Sun X, Yu H, Zhang W, Liang W, Fu K.(2021). Mimicking the Brain's Cognition of Sarcasm From Multidisciplines for Twitter Sarcasm Detection. IEEE Trans Neural Networks Learn Syst. doi:10.1109/TNNLS.2021.3093416
8. Almuzaini HA, Azmi AM.(2020) Impact of Stemming and Word Embedding on Deep Learning-Based Arabic Text Categorization. IEEE Access,8, 127913-127928. doi:10.1109/ACCESS.2020.3009217
9. Aboobaker J, Ilavarasan E.(2020). A Survey on Sarcasm detection and challenges. 2020 6th Int Conf Adv Comput Commun Syst (pp. 1234-1240). doi:10.1109/ICACCS48705.2020.9074163
10. NIKHIL JOHN. Tweets with Sarcasm and Irony | Kaggle.
11. Munková D, Munk M, Vozár M.(2013). Data Pre-processing Evaluation for Text Mining: Transaction/Sequence Model. Procedia Comput Sci.18,1198-1207. doi:10.1016/J.PROCS.2013.05.286
12. Al_Janabi S, Salman MA, Mohammed M.(2020). Pragmatic Text Mining Method to Find the Topics of Citation Network. Lect Notes Networks Syst.(pp.190-205). doi:10.1007/978-3-030-23672-4_15/COVER
13. Mittal V, Gangodkar D, Pant B.(2021) Deep Graph-Long Short-Term Memory: A Deep Learning Based Approach for Text Classification. Wirel Pers Commun. 119, 2287-2301. doi:10.1007/S11277-021-08331-4/METRICS
14. HaCohen-Kerner Y, Miller D, Yigal Y.(2020) The influence of preprocessing on text classification using a bag-of-words representation. PLoS One. e0232525. doi:10.1371/JOURNAL.PONE.0232525
15. Thakkar A, Chaudhari K.(2020) Predicting stock trend using an integrated term frequency–inverse document frequency-based feature weight matrix with neural networks. Appl Soft Comput, 96, 106684. doi:10.1016/J.ASOC.2020.106684
16. Srinivasan S, Ravi V, Alazab M, Ketha S, Al-Zoubi AM, Kotti Padannayil S.(2021) Spam Emails Detection Based on Distributed Word Embedding with Deep Learning. Stud Comput Intell, 161-189. doi:10.1007/978-3-030-57024-8_7/COVER
17. Onan A.(2019) Topic-Enriched Word Embeddings for Sarcasm Identi fi cation, 293-304. doi:10.1007/978-3-030-19807-7
18. Headline N, Detection S.(2019) News Headline Sarcasm Detection, (Itng):495-498.
19. Muaad AY, Davanagere HJ, Benifa JVB, et al.(2022) Artificial Intelligence-Based Approach for Misogyny and Sarcasm Detection from Arabic Texts..
20. Shrivastava M, Kumar S. (2021) Technology in Society Short communication A pragmatic and intelligent model for sarcasm detection in social media text. Technol Soc, 64, 101489. doi:10.1016/j.techsoc.2020.101489
21. Onan A. (2021) A Term Weighted Neural Language Model and Stacked Bidirectional LSTM Based Framework for Sarcasm Identification. doi:10.1109/ACCESS.2021.3049734
22. Li G, Lin F, Chen W, Liu B.(2022) applied sciences Affection Enhanced Relational Graph Attention Network for Sarcasm Detection.
23. Zhang Y, Liu Y, Li Q, Tiwari P, Member GS, Wang B. (2021) CFN : A Complex-Valued Fuzzy Network for Sarcasm Detection in Conversations, 29(12), 3696-3710.
24. Sharma DK, Singh B, Agarwal S, Kim H.(2022) Sarcasm Detection over Social Media Platforms Using Hybrid Auto-Encoder-Based Model.
25. Eke CI, Norman AA, Shuib L.(2021) Context-Based Feature Technique for Sarcasm Identification in Benchmark Datasets Using Deep Learning and BERT Model, 9, 48501-48518. doi:10.1109/ACCESS.2021.3068323
26. Akula R.(2021) Interpretable Multi-Head Self-Attention Architecture for.
27. Pandey R, Kumar A, Prakash J, Tripathi S.(2021) Hybrid attention-based Long Short-Term Memory network for sarcasm identification. Appl Soft Comput, 106, 107348. doi:10.1016/j.asoc.2021.107348
28. Lin Z, Ptaszynski M, Masui F, Leliwa G, Wroczynski M.(2021) Machine Learning and feature engineering-based study into sarcasm and irony classification with application to cyberbullying detection, 58(4), 102600.
29. Savini E, Caragea C. (2022) Intermediate-Task Transfer Learning with BERT for Sarcasm Detection, *Mathematics*, *10*(5), 844