International Journal of Basic and Applied Sciences, 14 (3) (2025) 116-124

SPC

International Journal of Basic and Applied Sciences

Website: www.sciencepubco.com/index.php/IJBAS
https://doi.org/10.14419/z470zd97
Research paper

A Hybrid Framework for Sarcasm Detection Using CB Technique

Ameya Parkar *, Rajni Bhalla

School of Computer Applications, Lovely Professional University, Phagwara, India
**Corresponding author E-mail: parkar.ameya@gmail.com*

Received: June 11, 2025, Accepted: July 11, 2025, Published: July 17, 2025

Abstract

In the last two decades, people have been posting about their choices, likes, dislikes, their views, and their opinions on websites. People tend to post text feedback, audio feedback, and video feedback on websites. Natural language processing techniques such as emotion detection, sentiment analysis, etc., can be used to judge the sentiments of the people. Businesses can use these techniques to judge their products and services and, in turn, decide the respective future actions to be taken based on the opinions. In sentiment analysis, the feedback can be judged as positive, negative, or neutral. Some people express their views sarcastically. By being sarcastic, the opinion is inverted in terms of polarity. Neural network techniques have been used recently for sarcasm detection. This study is focused on detecting sarcasm from text reviews using the CB technique of Convolution Neural Networks (CNN) and Bi-directional Encoder Representations from Transformers (BERT). This technique improves the performance compared to the individual deep learning techniques by having an F-score of 95%.

Keywords: *Convolution Neural Networks; Deep Learning; Natural Language Processing; Sarcasm Detection; Transformers.*

1. Introduction

Social media has been the trend for people across the globe. Around 63.7% of the world's population uses social media, according to Statista [1]. Many people connect using social media websites like X, Facebook, Instagram, etc. They use social media to discuss different topics like sports, politics, fiction, and so on. They express the reviews in chats, text reviews, videos, etc. Many people watch such videos and make up their minds whether they should watch the movie or not. People tend to trust the online posts shared by others on social media. All businesses nowadays have their website to get feedback about their products and services. The businesses make tie-ups with leading social media websites to gather information posted by people on the social media websites. Even a single review posted online influences people and tends to change their perspective on the product/service. For these reasons, it is vital that businesses take the input of the review posted and decide on the future course of action.

Opinion mining is a subcategory of Artificial Intelligence to extracts reviews from websites and other known places, and then the polarity of the review can be found out. In various domains of society, opinion mining is used to gather useful information. For finding the popularity of a movie, opinion mining is used, and the movie gets a certain rating. A new mobile phone is launched by a particular brand, and people express about the price of the phone, camera, video, speakers, etc.

Automation is difficult in tasks such as presence of sarcasm in text. Detection of sarcasm is another complex task in Natural language processing. People tend to post their opinions positively, despite the opinion being negative. This could trick any sentiment analysis model to rate the review as positive in spite of being negative. Due to this problem, the primary objective of this paper is to detect the presence of sarcasm in text reviews.

The objectives of this paper are as follows: We have proposed a hybrid technique of CB model (CNN + BERT) to detect sarcasm using the techniques of Convolution Neural Network and Bi-directional Encoder Representations from Transformers in text reviews. The model has been pre-trained with BERT word embeddings for the proposed model &Word2Vec for other techniques. The hybrid framework is tested on three publicly available datasets, which helps us in judging the performance of the model relative to other techniques used. The results of our model can be used by other researchers for their datasets as well as for judging real-time data.

The novelty of this research is the way the hybrid technique is used. The BERT technique was chosen to understand the context of the reviews and the dependencies between different words in the review. The CNN technique is chosen to understand the patterns in the data (like negations, exclamation marks, etc.), which are necessary to categorize the review as sarcastic. Hence, a combination of these techniques is used.

1.1. Related work

Researchers have used initially semantic features, syntactic features, pattern recognition, recognizing linguistic features, parts of speech tagging and others to detect sarcasm. Prasanna K. et al. [2] used Logistic Regression, encoder, Bidirectional LSTM and Bidirectional Gated Recurrent Unit techniques to detect sarcasm in two publicly available datasets. The model was detecting sarcasm based on linguistic features and nuances in the text. The models gave an accuracy score of 70% on the SARC dataset while accuracy score of 80.1% on the Headlines dataset. Goyal T. et al. [3] proposed methods to detect sarcasm & emotions for people having Alexithymia. Mobile ML models were used to analyze the data & real time feedback was taken. All the models were based on BERT algorithm and the dataset was on Twitter. An accuracy score of 83% was recorded for sarcasm detection. Suhartono D. et al. [4] gathered data from social media to make a customized dataset in Indonesian language on sarcastic texts. They studied and analyzed different ML models, language models & large language models. They concluded that pre-trained models had more performance scores than other models. Fitrianto R. et al. [5] highlighted the issues with sentiment analysis when sarcasm is involved in text reviews. They extracted tweets from website X in Indonesian language and used several variants of BERT transformer to detect sarcasm. Hassan M. et al. [6] used large language models based on transformers to detect sarcasm in Urdu tweets collected from Twitter. Self-annotation was done to classify the tweets as sarcastic & non sarcastic. BERT technique gave an accuracy of 79.54%. FastText technique was used for word embeddings. Wang Z. et al. [7] used Gated Recurrent Units to detect sarcasm in Chinese reviews with an f-score of 95%. Nirmala M. et al. [8] used decision tree technique with inbuilt libraries to detect sarcasm on tweets with a f-score of 80%. Dai J. et al. [9] used a combination of transformers, RNN and fuzzy logic to detect sarcasm. They stressed on finding the context in the reviews. Fuzzy logic was introduced to remove the ambiguity in the sarcastic reviews. Subsets of standard datasets were used for the study. The model gave an f-score of 89.54% on the Headlines dataset. Liu H. et al. [10] worked on multimodal sarcasm detection using a hierarchical fusion model. Attribute object matching & cross modal transformer were used to find relationships between the text and the objects in the image as well as for sentiment extraction. Li Y. et al. [11] worked on finding the relations between different modalities to detect sarcasm. Emotional information & context was captured by their model to improve the performance of detection of sarcasm. Their customized model gave an F-score of 75.2%. Waly R. et al. [12] used AraBERT classifier to detect sarcasm in Arabic reviews with an accuracy score of 87%. Tingre Y. et al. [13] used LSTM technique to detect sarcasm on the news headlines dataset with an accuracy score of 88%. Sreenivas P. et al. [14] used bidirectional LSTM to detect sarcasm in news headlines dataset with an accuracy score of 86%. Thakur H. et al. [15] used multi head attention LSTM to detect sarcasm on Reddit dataset with a f-score of 79%. Galal A. et al. [16] used transformers to detect sarcasm in Arabic tweets with a f-score of 68%.

1.2. Gaps in existing techniques

- Sarcasm expressed in polite way.
- Domain specific sarcasm.
- Culture specific sarcasm
- Short reviews
- Capturing deep context from reviews.

In this research, we have proposed the CB technique which captures the semantics and extracts deep contextual knowledge. Pattern recognition as well as contextual knowledge is captured by the CB technique to improve the detection of sarcasm in reviews. Also, due to pattern recognition and pre-training of the CB technique helps the model understand sarcastic clues in short reviews.

1.3. Datasets

The first dataset is from the Kaggle website & contains news headlines [17]. The news headlines dataset has 3 columns and 28619 records. The first column is the label stating “1 for sarcastic, 0 for non-sarcastic”. The second column is the news headlines column which contains the actual news headlines. The third column is the article link column which gives the link of the source of the headline. The dataset is balanced with a ratio of 50:50 for sarcastic headlines and non-sarcastic headlines. The second dataset [18] is on Reddit reviews & contains 80000 records. The first column is the label stating “1 for sarcastic, 0 for non-sarcastic”. The second column is the comment column which contains the reviews. The third column is the name of the reviewer. The fourth column is the topic of discussion of the review. The next few columns contain the date the review was written, the number of likes and the number of dislikes, etc. The dataset is slightly imbalanced with a ratio of 60:40 for sarcastic headlines and non-sarcastic headlines. The third dataset [19] is on tweets from Twitter & contains 8119 tweets. The first column is the tweet and the second column is the class of the tweet. The reason behind choosing these datasets were that they were publicly available as well as other researchers had done sarcasm detection tasks on these datasets. By choosing these datasets, a direct comparison could be done to judge the performance of our proposed model.

Section 2 is the Proposed Method. Section 3 is the techniques used in detail. Section 4 is the Results and Discussion section and Section 5 is the Conclusion and Future Scope section.

2. Proposed method

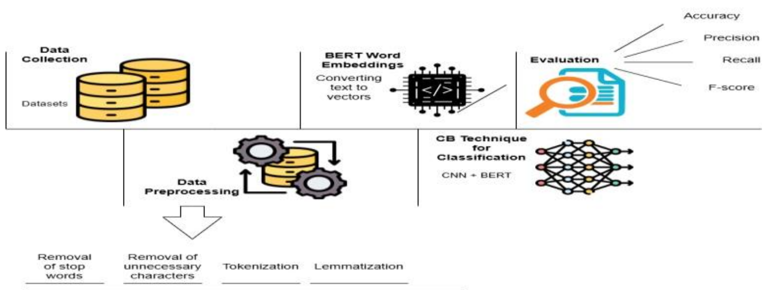

Fig. 1: Architecture Diagram Using CB Technique for Sarcasm Detection.

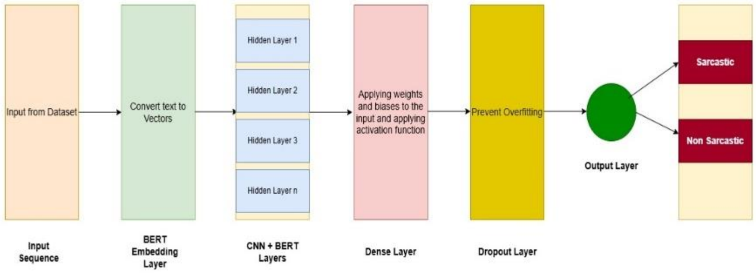

Fig. 2: Internal Architecture for Sarcasm Detection Using CB Technique.

The detailed framework is discussed in section 3.3.7. Figure 1 demonstrates the architecture used for Sarcasm detection in this research, and Figure 2 demonstrates the Internal Architecture used for Sarcasm Detection.

3. Techniques used in detail

3.1. Data processing

Both datasets in this study have a lot of noise and some irrelevant data related to this research study. Hence, pre-processing techniques had to be applied to both datasets to keep the relevant information & eliminate data that would not be useful. The pre-processing steps are as follows:

- Removal of columns that are not necessary for our research study. For the first dataset, we removed the "article_link" column. For the second dataset, we removed columns like the "name of the reviewer", "date of creation", "likes", "dislikes", etc.
- Tokenization to split every sentence's words into individual parts.
- Removal of stop words and hyperlinks.
- Lemmatization to convert each word to its original meaning.

3.2. Techniques to detect sarcasm

Research started initially on the detection of sarcasm by identifying sarcastic words, sarcastic opinions, understanding the semantics, patterns, syntax, etc. Recently, ML classifiers and DL classifiers have been extensively used for the sarcasm task. Some of the techniques are discussed below:

1) Machine learning

Alakrot A. et al. [20] worked on the Libyan Arabic dialects dataset for detecting sarcasm. Reviews in the Arabic language from Facebook were extracted. Lexical & syntactic features were extracted from the dataset. The SVM technique was used for training and predictions with an accuracy score of 79.15%. Pradhan J. et al. [21] extracted linguistic features from the news headlines dataset to detect sarcasm. They used ML classifiers, an ensemble model & a dense neural network to find sarcasm. The ensemble model gave an accuracy of 93% on the Headlines dataset. Chia Z. et al. [22] worked on finding irony & sarcasm in English & Chinese datasets. They used machine translation to detect sarcasm in texts. They used different translation settings and fine-tuned the model for sarcasm detection. They achieved an F-score of 79.50% on the online dataset of SemEval. Additionally, they created a new dataset for irony detection.

2) Neural networks

Thambi J. et al. [23] used a dataset of headline news to detect sarcasm. They made an ensemble model of ML algorithms and also used DL techniques. RNN technique gave the best performance score with an accuracy of 79%. Additionally, the LIME approach was used to understand the predictions of the model implemented. Grover V. et al. [24] worked on finding emojis in reviews and using the emojis to detect sarcasm in the reviews. They proposed a deep learning model and trained and tested it on two datasets with an F-score of 68.14%. Abuein Q. et al. [25] proposed a new model to detect sarcasm in Arabic language tweets. They made a customized dataset of Arabic tweets. A lot of pre-processing steps were used before feeding the data to the different DL models.

3) Tra``nsformers

Karkiner Z. et al. [26] tested different DL algorithms on the publicly available news headlines dataset. The BERT technique gave the highest accuracy score of 88% on the dataset. The RNN technique was the fastest to train and evaluate on the dataset. Wu B. et al. [27] worked on finding sarcasm in general and not specific to a particular domain. They used Low-Rank Adaptation for tuning large language models for finding sarcasm. This work could be carried out for different languages. They fine-tuned the BERT transformer technique to train and test English and Chinese datasets with an F-score of 90% for the English dataset & 94% for the Chinese dataset.

4) Other techniques

Puvvada S. et al. [28] used a combination of ML techniques to detect sarcasm from a Facebook dataset of sarcastic comments. The hierarchical clustering technique was used to make the cluster quality good. Emojis, texts, and sentiment were considered by the unsupervised technique. Lu Q. et al. [29] stressed that in multimodal sarcasm detection, it is necessary to include fact-sentiment incongruity. They used a graph-based network to select the best combinations of image & text to find the fact incongruity between them.

3.3. Approaches to detect Sarcasm in this study

Research started initially on the detection of sarcasm by identifying sarcastic words, sarcastic opinions, understanding the semantics, patterns, syntax, etc. Recently, DL classifiers have been extensively used for the sarcasm task. Some of the techniques are discussed below:

3.3.1. Recurrent neural networks (RNN)

RNN technique falls under deep learning, in which the output of the predecessor is given as input to the next step. It has a hidden layer that records information in a sequence. To reduce the complexity, the same parameters are used for each input. It differs from other deep learning techniques with the virtue that the same weights are carried across the entire system.

The equation for the current state is given as:

$$c_t = g(c_{t-1}, y_t) \tag{1}$$

Where c_t represents the current state, c_{t-1} represents the previous state, y_t represents the input state
The output is calculated as

$$o_t = w_o a_t \tag{2}$$

Where o_t represents output, w_o represents the output layer weight, a_t represents an activation function.

3.3.2. Long short-term memory (LSTM)

The LSTM technique was invented as an upgrade to the existing RNN technique. RNN suffers from the problem of learning dependencies from a long-distance point of view. In the LSTM technique, there is an additional cell given to store the dependencies from a long-distance point of view.
This additional cell is controlled by three gates:
a) Input gate: The input gate decides what kind of information has to be stored in the additional cell.
b) Forget gate: The forget gate keeps vital information in the additional cell and discards what is not vital.
The equation for the forget gate is given as

$$g_t = \delta\,(w_g\,(y_{t-1}, z_t) + c_g \tag{3}$$

Where w_g represents the weight matrix of the forget gate, (y_{t-1}, z_t) represents the joining of the previous hidden state & the current input c_g represents the bias, δ Is the sigmoid function.
c) Output gate: The output gate decides what kind of information is sent out from the additional cell.

3.3.3. Global vectors for word representation (GloVe)

GloVe is an unsupervised algorithm to find word embeddings of text. We built the model using a co-occurrence matrix of pairs of words. It was created so that the semantics between the different words in the dataset could be preserved. In this model, a lot of dense embeddings are available. Every word in the dataset can be expressed in around 100 dimensions in word embeddings. GloVe creates a word matrix for every word, and the relationship of that word with every other word in the dataset is expressed with a numerical decimal value. The optimization step is done after the matrix creation to create a dense vector for every word.

3.3.4. Bi-directional LSTM (Bi-LSTM)

The bi-directional LSTM technique is used in neural networks for a sequence model. It consists of 2 LSTM layers. Two unidirectional LSTMs are used in the Bidirectional LSTM technique. It can be considered as two different LSTM networks, with one going in the forward direction & the other in the backward direction. The LSTM going in the forward direction gets all the input in a sequential manner, while the LSTM in the backward direction gets it in a reverse way. Both the LSTMs give a probability vector as an output & the final output is the combination of both the probability vectors.
It is represented by the equation

$$v_n = v_n^f + v_n^b \tag{4}$$

Where v_n = outcome, v_n^f = forward outcome, v_n^b = backward outcome

3.3.5. Convolution neural networks (CNN)

CNN is a part of deep learning neural networks to make a computer understand, process text & images. They are specifically used in image interpretation & finding things in an image. Additionally, they can also be used for tasks such as sentiment analysis, sarcasm detection, etc.
A CNN typically has three layers:
Input layer: The input layer, as its name suggests, is used to input the text data. Depending on the task at hand, we decide on the number of neurons in this layer. Normally, the total number of features in the dataset dictates the number of neurons in this layer.
Inner layers/Hidden layers: The input from the text is given to the input layer and then given to the hidden layer. The number of layers & the number of neurons in each hidden layer are again dependent on the task to perform. At each of the hidden layers, weights are assigned to each neuron, and the neurons with higher weights are preferred over the lower weights. The calculation at each layer is found by doing multiplication of matrices by taking into account the output of the layer before that. Biases are taken into consideration, and adjustment is made accordingly. Activation function, like RELU, is used for understanding the patterns in the data.
Output layer: After the processing is done in the hidden layers, the outcome is passed to the output layer, which is the final layer. At this layer, the final output of the text is received, and the outcome is found by using a function like the sigmoid or SoftMax function.

3.3.6. Bi-directional encoder representations from transformers (BERT)

BERT is a transformer-based model used in Natural language processing to generate text and make predictions in text. The BERT architecture has an encoder and a decoder side. The words in a text review are fed one by one to the encoder part of the model. At the decoder level, the model predicts the outcome of the given text review. It scans the text in both directions, and the context is stored intermittently. It looks at all the words in the entire text review and makes predictions based on the words. Initially, the BERT model is fed all the reviews, and it is trained. BERT generates contextual embeddings that help in predicting the outcome of a certain review. For each word in the review, BERT makes multiple dimensions for it. A special layer exists in the model to make the predictions.

3.3.7. CB technique (CNN + BERT)

A hybrid model of CNN & BERT is introduced in this study for sarcasm detection. The necessary libraries for doing the task of sarcasm detection are imported, like numpy, pandas, transformers, keras, etc. After doing all the pre-processing techniques, all the remaining text was converted to tensors using BERT embeddings. In the Keras library, we have used the "Concatenate" layer for merging the outputs of two models into a single tensor. We have used. We used different combinations of dense layers, hidden layers, heads, and parameters. The combination of 12 dense layers, 78 hidden layers, 12 heads, and 110M parameters gave the best performance metrics amongst the different combinations used. A dropout layer was used to prevent overfitting. The sigmoid function was used as an activation function. Adam optimizer was used for convergence and to improve the training speed. Relu activation function was used to understand the complex patterns in the data. The dataset was split into a ratio of 80:20 for training and testing. For managing the loss, the binary cross-entropy function was used. For CNN, pad sequences were used to confine the length of each sentence to a pre-defined limit. A single, single-dimensional convolutional layer is used for processing sequences. A fully connected dense layer was used. Hyperparameter tuning was done on the CB technique to find the optimal combination of hyperparameters. For hyper-tuning, we initially focused on the learning rate. Concerning Table 1, different values of learning rate were tested. $1e^{-3}$ gave the best convergence rate. The batch size was the next parameter. We chose a batch size of 128 after trying the combinations mentioned in Table 1. The dropout rate was the next parameter. The dropout rate chosen was 0.2 after testing the combinations of different dropout rates.

Table 1: Hyperparameter Tuning for CB Technique

Learning Rate	Batch Size	Dropout Rate
$1e^{-1}$	16	0.2
$1e^{-2}$	32	0.3
$1e^{-3}$	64	0.5
$1e^{-4}$	128	0.6

4. Results and discussion

4.1. Evaluation metrics

The performances of the techniques used in this study were evaluated on different evaluation metrics. Evaluation metrics/Performance metrics are used to judge the performance of the model based on different criteria. The evaluation metrics are Accuracy, Precision, Recall & F-score. Accuracy focuses on the quantity of correctly predicted samples relative to the total number of samples. It applies to the positive class as well as the negative class. Accuracy can be used to judge a model as long as the dataset is evenly balanced. Precision focuses on the positive class of samples among all the samples that are classified as positive. It applies only to the positive class. Precision can be used to judge the correctly positive predictions in the dataset. Recall focuses on the correctly obtained positives about all the samples that have been predicted as positive. It is used when we want to see how many positive samples have been listed by the model. The F-score performance metric is useful when we want to judge the overall performance of the model. It is used as a combination of the harmonic mean of the recall score & the precision score. It gives an accurate picture of the performance of the model when the dataset is balanced as well as unbalanced.

4.2. Results

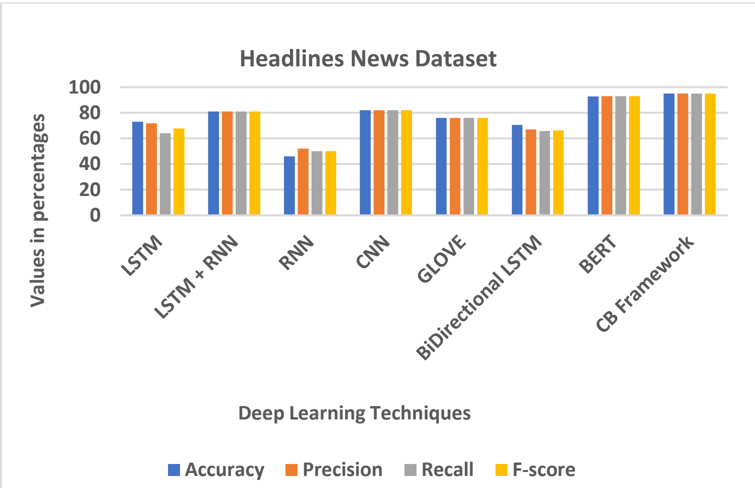

Fig. 3: Analysis of Deep Learning Techniques on Headline News Dataset for Sarcasm.

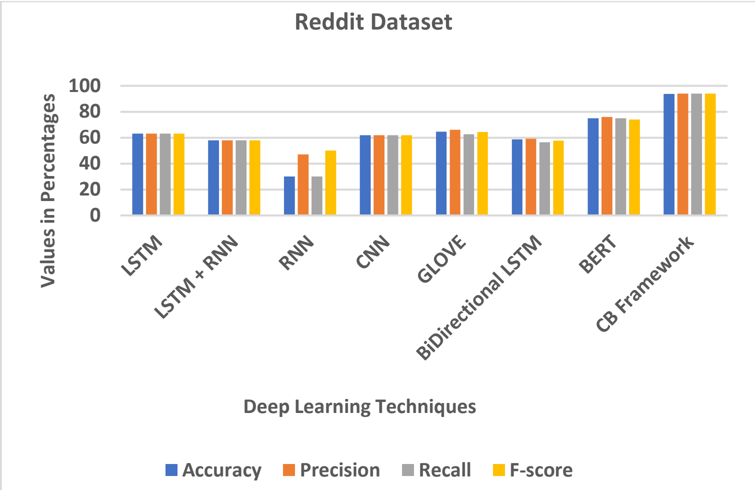

Fig. 4: Analysis of Deep Learning Techniques on Reddit Dataset for Sarcasm.

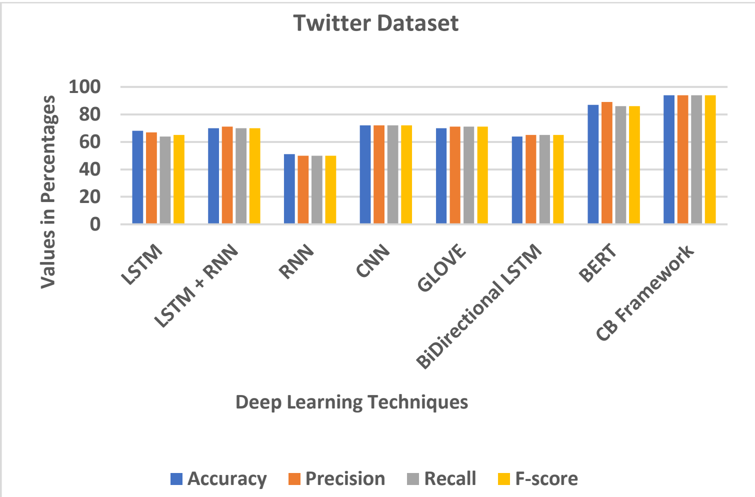

Fig. 5: Analysis of Deep Learning Techniques on Twitter Dataset for Sarcasm

In Figure 3, we have done a comparative analysis of deep learning techniques on the News Headlines dataset. The CB technique gives us a better performance score compared to the individual techniques, with an accuracy score of 95.10% and a F-score of 95%.
In Figure 4, we have done a comparative analysis of deep learning techniques on the Reddit dataset. The CB technique gives us a better performance score compared to the individual techniques, with an accuracy score of 93.81% and a F-score of 94%.
In Figure 5, we have done a comparative analysis of deep learning techniques on the Twitter dataset. The CB technique gives us a better performance score compared to the individual techniques, with an accuracy score of 94% and a F-score of 94%.

Table 2: Error Analysis of Techniques Used in This Research

Methodology	Number of Epochs	Time taken	Loss function
LSTM	30	2 hours	0.027
LSTM + RNN	30	5.5 hours	0.053
RNN	30	5.5 hours	0.071
CNN	100	1.5 hours	0.009
GLOVE	30	1 hour	0.037
Bi-directional LSTM	30	2 hours	0.405
BERT	100	5 hours	0.093
CB Technique	100	10 hours	0.208

Table 2 shows the error analysis of the techniques used. Factors like time taken to complete the task, number of epochs, and the value of the loss function are mentioned. In addition, we also conducted an ANOVA test to compare the BERT technique with the CB technique across multiple runs. The results gave the p-value of 0.0001, which indicates that there is an improvement in the CB technique compared to the BERT technique, and there is a performance gain.

Table 3: Performance Comparison of CB Technique with Existing Models on Headlines Dataset

Name of Paper & Year	Method	Accu-racy	Preci-sion	Re-call	F score
Harnessing Advanced Learning for Sarcasm Detection, 2024 [2]	Encoder	80.1	---	---	75.5
Sarcasm Detection in News Headlines with Deep Learning, 2024 [26]	BERT	88	---	---	---
Sarcasm Detection in News Headlines using ML and DL models, 2024 [23]	RNN	79	---	---	76
An Efficient Sarcasm Detection using Linguistic Features and Ensemble Machine Learning, 2024 [21]	Ensemble	93.75	---	---	---
Detect Sarcasm and Humor jointly by Neural Multi-Task Learning, 2024 [30]	RNN	---	79.34	79.8	79.61

Sarcasm Detection over Social Media Platforms Using Hybrid Ensemble Model with Fuzzy Logic, 2023 [31]	BERT + fuzzy logic	90.81	---	---	---
An Ensemble Model for Detecting Sarcasm on Social Media, 2022 [32]	Glove + LSTM	88.9	91.1	88.6	89.87
CB Technique		95.10	95	95	95

Table 4: Performance Comparison of CB Technique with Existing Models on Reddit Dataset

Name of Paper & Year	Method	Accuracy	Precision	Re-call	F score
IdSarcasm: Benchmarking & Evaluating Language Models for Indonesian Sarcasm Detection, 2024 [4]	Large Language Model	--	--	--	62.7
Sarcasm Detection over Social Media Platforms Using Hybrid Ensemble Model with Fuzzy Logic, 2024 [31]	BERT + fuzzy logic	85.38	--	--	--
Intermediate-Task Transfer Learning with BERT for Sarcasm Detection, 2022 [33]	BERT	--	--	--	77.53
Affection Enhanced Relational Graph Attention Network for Sarcasm Detection, 2022 [34]	Graph Attention Network	75.8	75.8	75.8	75.8
Detecting the target of sarcasm is hard: Really??, 2022 [35]	LSTM	71.5	--	--	--
Sarcasm detection using deep learning and ensemble learning, 2022 [36]	CNN + LSTM + GRU	81.13	--	--	--
CB Technique		93.81	94	94	94

Table 5: Performance Comparison of CB Technique with Existing Models on Twitter Dataset

Name of Paper & Year	Method	Accuracy	Precision	Re-call	F score
An attention approach to emoji-focused sarcasm detection, 2024 [24]	Bi-LSTM	--	--	--	52.3
An Emoticon-Based Novel Sarcasm Pattern Detection Strategy to Identify Sarcasm in Microblogging Social Networks, 2024 [8]	Decision Tree	--	--	--	81
BNS-Net: A Dual-Channel Sarcasm Detection Method Considering Behavior-Level and Sentence-Level Conflicts, 2024 [37]	BNS-Net	73.58	73.47	73.44	73.47
Detection of Sarcasm in Urdu Tweets Using Deep Learning and Transformer-Based Hybrid Approaches, 2024 [6]	BERT + BiLSTM	79.51	--	--	80.04
IdSarcasm: Benchmarking and Evaluating Language Models for Indonesian Sarcasm Detection, 2024 [4]	Pre-trained language models	--	--	--	76.92
Mobile Machine Learning Models for Emotion and Sarcasm Detection in Text: A Solution for Alexithymic Individuals, 2024 [3]	BERT	83	--	--	--
CB Technique		94	94	94	94

4.3. Discussion

Detecting the presence of sarcasm is a tough task from different perspectives. Some key aspects are: Ambiguity: In multimodal sarcasm detection, the tone of the person or the facial expression of the person gives an idea of the presence of sarcasm. However, for text data, there are no such clues. Context: Context in a sentence or multiple sentences is also necessary to understand the presence of sarcasm. Often, when contradiction is done for a particular review or a comment, it is an indication of the presence of sarcasm. Understanding the context of the review is important to judge the presence of sarcasm. Cross-Lingual text: Sarcasm is not only a challenge in the English language but across many popular languages across the world. Culture-wise wise different people will express sarcasm differently. Therefore, it is another challenge.

We have used deep learning techniques like RNN, LSTM, Bi-directional LSTM, etc., to detect sarcasm. These techniques underperform the proposed CB technique for the following reasons. The CB technique captures deep contextual information while the other techniques fail to do so. The CB technique captures the local as well as global patterns and semantics well compared to the other techniques. CB technique uses attention mechanisms to capture long sentence dependencies, while the other techniques fail to do it in an effective manner for detecting sarcasm.

In our research, we have focused on understanding sarcasm in text data by understanding the context of the data. By using the CB technique, we can understand the context of the review by filtering unnecessary words and focusing on the contextual terms in the review. Also, in the CB technique, linguistic features and context are taken into consideration, such that if any live data is given to the model for intentional manipulation, the model understands it and categorizes it accordingly. Also, the model handles adversarial robustness. i.e., It correctly classifies sarcasm written in capital letters or with excessive punctuation marks, etc.

In Table 3, a fine comparison is made of the CB technique with the models used by other researchers, and the CB technique gives a better performance on the Headlines dataset. In Table 4, a fine comparison is made of the CB technique with the models used by other researchers, and the CB technique gives a better performance on the Reddit dataset. In Table 5, a fine comparison is made of the CB technique with the models used by other researchers, and the CB technique gives a better performance on the Twitter dataset. It is to be noted that the performance is better on the Headline news dataset and the Twitter dataset, as the headline and the tweet contain the whole context of the topic. In the Reddit dataset, there are a lot of reviews with less contextual knowledge.

4.3.1. Limitations and ethical considerations

The context is difficult to capture in too short sentences. The CB technique is better here compared to other techniques due to the pre-training of BERT to capture sarcasm clues. Culture-specific sarcasm is difficult to capture by the CB technique. Quotations/Emphasis are detected by the CB technique due to the CNN detecting the patterns in the review. Also, the CB technique captures contrastive sentiments in the review easily due to the attention mechanism and pattern recognition.

5. Conclusion and future scope

Opinion mining is a vital task for most businesses/organizations to get an assessment of their products and services. Social media is a place where people all over the globe discuss and review different products. People tend to post reviews sometimes sarcastically. It is necessary to find such sarcastic reviews and invert their polarity to get a clear view of the reviews. For this reason, we have used DL techniques & a hybrid CB technique of CNN + BERT to find sarcasm in three publicly available datasets of News Headlines, Reddit, and Twitter. The hybrid framework gives better performance scores than the individual techniques, as well as compared to work done by other researchers. The novelty of this research is that we have done a fine comparison analysis between the existing techniques for sarcasm detection and proposed a new hybrid technique to detect sarcasm & find out the best technique amongst them.

Real-world applications where the CB technique model could be used. Social media Applications: A lot of people post content on social media. By using this model, the social media website could filter out sarcastic remarks that could be offensive. Customer Service: Chatbots are used as an initial step to resolve and help problems of different customers on different portals. The model could be used as part of the training for the chatbot. Advertising Campaigns: A lot of companies promote their new products by advertising. People comment about the products advertised. The model could be used to detect sarcastic reviews. Media and News Channels: Media and news channels post about a particular topic related to different fields, like politics, sports, etc. People put comments and express their views. The model could filter out the sarcastic comments written. Likewise, there are other application areas like Education, Sales, etc., where the model could be used. For future work, an evolutionary approach could be used to detect sarcasm as newer ways are found by people to express sarcasm. Also, support for sarcasm detection in languages used across the world.

References

[1] Statista Research Department, "Internet and social media users in the world 2024. Statista," 2024. [Online]. Available: https://www.statista.com/statistics/617136/digital-population-worldwide.

[2] P. K. M and K. Veeramani, "Harnessing Advanced Learning for Sarcasm Detection," *2024 Third International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)*, pp. 1–5, https://doi.org/10.1109/ICSTSN61422.2024.10671235.

[3] T. Goyal, D. H. Rajeshbai, N. Gopalkrishna, M. Tushar, and H. R. Mamatha, "Mobile Machine Learning Models for Emotion and Sarcasm Detection in Text: A Solution for Alexithymic Individuals," *2024 3rd International Conference for Innovation in Technology, INOCON 2024*, no. 2018, pp. 1–5, 2024, https://doi.org/10.1109/INOCON60754.2024.10511772.

[4] D. Suhartono, W. Wongso, and A. Tri Handoyo, "IdSarcasm: Benchmarking and Evaluating Language Models for Indonesian Sarcasm Detection," *IEEE Access*, vol. 12, no. June, pp. 87323–87332, 2024, https://doi.org/10.1109/ACCESS.2024.3416955.

[5] R. A. Fitrianto, A. S. Editya, M. M. H. Alamin, A. L. Pramana, and A. K. Alhaq, "Classification of Indonesian Sarcasm Tweets on X Platform Using Deep Learning," *Proceedings - International Conference on Informatics and Computational Sciences*, pp. 388–393, 2024, https://doi.org/10.1109/ICICoS62600.2024.10636904.

[6] M. E. Hassan, M. Hussain, I. Maab, U. Habib, M. A. Khan, and A. Masood, "Detection of Sarcasm in Urdu Tweets Using Deep Learning and Transformer Based Hybrid Approaches," *IEEE Access*, vol. 12, no. May, pp. 61542–61555, 2024, https://doi.org/10.1109/ACCESS.2024.3393856.

[7] Z. Wang, S. jie Hu, and W. dong Liu, "Product feature sentiment analysis based on GRU-CAP considering Chinese sarcasm recognition," *Expert Syst Appl*, vol. 241, no. February 2023, p. 122512, 2024, https://doi.org/10.1016/j.eswa.2023.122512.

[8] M. Nirmala, A. H. Gandomi, M. R. Babu, L. D. D. Babu, and R. Patan, "An Emoticon-Based Novel Sarcasm Pattern Detection Strategy to Identify Sarcasm in Microblogging Social Networks," *IEEE Trans Comput Soc Syst*, vol. 11, no. 4, pp. 5319–5326, 2024, https://doi.org/10.1109/TCSS.2023.3306908.

[9] J. Dai, "A BERT-Based with Fuzzy logic Sentimental Classifier for Sarcasm Detection," *2024 7th International Conference on Advanced Algorithms and Control Engineering, ICAACE 2024*, pp. 1275–1280, 2024, https://doi.org/10.1109/ICAACE61206.2024.10548550.

[10] H. Liu, R. Wei, G. Tu, J. Lin, C. Liu, and D. Jiang, "Sarcasm driven by sentiment: A sentiment-aware hierarchical fusion network for multimodal sarcasm detection," *Information Fusion*, vol. 108, no. March, p. 102353, 2024, https://doi.org/10.1016/j.inffus.2024.102353.

[11] Y. Li *et al.*, "An attention-based, context-aware multimodal fusion method for sarcasm detection using inter-modality inconsistency," *Knowl Based Syst*, vol. 287, no. November 2023, p. 111457, 2024, https://doi.org/10.1016/j.knosys.2024.111457.

[12] R. R. Waly, M. Saeed, A. Ashraf, and W. H. Gomaa, "Advancements in Arabic Tweets Sarcasm Detection & Sentiment Analysis with Fine-Tuned Models," *2024 Intelligent Methods, Systems, and Applications (IMSA)*, no. Subtask 1, pp. 292–298, 2024, https://doi.org/10.1109/IMSA61967.2024.10652758.

[13] Y. Tingre, R. Pingale, N. Choudhary, A. Kale, and N. Hajare, "Sarcasm Detection of Emojis Using Machine Learning Algorithm," *Proceedings of InC4 2024 - 2024 IEEE International Conference on Contemporary Computing and Communications*, vol. 1, pp. 1–4, 2024, https://doi.org/10.1109/InC460750.2024.10649041.

[14] B. Sreenivas Prasad, N. Akash Babu, T. Harithikeswar Reddy, A. Vishnu Vardhan Reddy, S. Vekkot, and P. C. Nair, "Sarcasm Detection in Newspaper Headlines," *2024 4th International Conference on Intelligent Technologies, CONIT 2024*, pp. 1–8, 2024, https://doi.org/10.1109/CONIT61985.2024.10627150.

[15] H. Thakur and J. Srivastava, "Sarcasm Detection with BiLSTM Multihead Attention," *2024 IEEE 9th International Conference for Convergence in Technology, I2CT 2024*, pp. 1–7, 2024, https://doi.org/10.1109/I2CT61223.2024.10543816.

[16] M. A. Galal, A. Hassan Yousef, H. H. Zayed, and W. Medhat, "Arabic sarcasm detection: An enhanced fine-tuned language model approach," *Ain Shams Engineering Journal*, vol. 15, no. 6, p. 102736, 2024, https://doi.org/10.1016/j.asej.2024.102736.

[17] Rishabh Misra, "News Headlines Dataset Kaggle," https://www.kaggle.com/datasets/rmisra/news-headlines-dataset-for-sarcasm-detection.

[18] Dan Ofer, "Reddit Dataset," https://www.kaggle.com/datasets/danofer/sarcasm.

[19] Nikhil John, "Twitter Dataset," https://www.kaggle.com/datasets/nikhiljohnk/tweets-with-sarcasm-and-irony/data.

[20] A. Alakrot, A. Dogman, and F. Ammer, "Sarcasm Detection in Libyan Arabic Dialects Using Natural Language Processing Techniques," *2024 IEEE 4th International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering, MI-STA 2024 - Proceeding*, no. May, pp. 761–767, 2024, https://doi.org/10.1109/MI-STA61267.2024.10599695.

[21] J. Pradhan, R. Verma, S. Kumar, and V. Sharma, "An Efficient Sarcasm Detection using Linguistic Features and Ensemble Machine Learning," *Procedia Comput Sci*, vol. 235, pp. 1058–1067, 2024, https://doi.org/10.1016/j.procs.2024.04.100.

[22] Z. L. Chia, M. Ptaszynski, M. Karpinska, J. Eronen, and F. Masui, "Initial exploration into sarcasm and irony through machine translation," *Natural Language Processing Journal*, vol. 9, no. September, p. 100106, 2024, https://doi.org/10.1016/j.nlp.2024.100106.

[23] J. Thambi, S. S. H. Samudrala, S. R. Vadluri, P. C. Nair, and M. Venugopalan, "Sarcasm Detection in News Headlines Using ML and DL Models," *Proceedings - 3rd International Conference on Advances in Computing, Communication and Applied Informatics, ACCAI 2024*, no. Ml, pp. 1–8, 2024, https://doi.org/10.1109/ACCAI61061.2024.10601793.

[24] V. Grover and H. Banati, "An attention approach to emoji focused sarcasm detection," *Heliyon*, vol. 10, no. 17, p. e36398, 2024, https://doi.org/10.1016/j.heliyon.2024.e36398.

[25] Q. Abuein, R. M. Al-Khatib, A. Migdady, M. S. Jawarneh, and A. Al-Khateeb, "ArSa-Tweets: A novel Arabic sarcasm detection system based on deep learning model," *Heliyon*, vol. 10, no. 17, p. e36892, 2024, https://doi.org/10.1016/j.heliyon.2024.e36892.

[26] Z. Karkiner and M. Sert, "Sarcasm Detection in News Headlines with Deep Learning," *32nd IEEE Conference on Signal Processing and Communications Applications, SIU 2024 - Proceedings*, pp. 1–4, 2024, https://doi.org/10.1109/SIU61531.2024.10600986.

[27] B. Wu, H. Tian, X. Liu, W. Hu, C. Yang, and S. Li, "Sarcasm Detection in Chinese and English Text with Fine-Tuned Large Language Models," *Proceedings - 2024 IEEE 10th Conference on Big Data Security on Cloud, BigDataSecurity 2024*, pp. 47–51, 2024, https://doi.org/10.1109/BigDataSecurity62737.2024.00016.

[28] S. Puvvada, Kusuma, M. V. Shaik, K. Galla, S. Venkatrama Phani Kumar, and K. Venkata Krishna Kishore, "A Novel Machine Learning Model for Sarcasm Detection on Facebook Comments," *Proceedings of the 3rd International Conference on Applied Artificial Intelligence and Computing, ICAAIC 2024*, pp. 546–552, 2024, https://doi.org/10.1109/ICAAIC60222.2024.10575012.

[29] Q. Lu, Y. Long, X. Sun, J. Feng, and H. Zhang, "Fact-sentiment incongruity combination network for multimodal sarcasm detection," *Information Fusion*, vol. 104, no. December 2023, p. 102203, 2024, https://doi.org/10.1016/j.inffus.2023.102203.

[30] Y. Diao, L. Yang, S. Li, Z. Hao, X. Fan, and H. Lin, "Detect Sarcasm and Humor Jointly by Neural Multi-Task Learning," *IEEE Access*, vol. 12, pp. 38071–38080, 2024, https://doi.org/10.1109/ACCESS.2024.3370858.

[31] D. K. Sharma, B. Singh, S. Agarwal, N. Pachauri, A. A. Alhussan, and H. A. Abdallah, "Sarcasm Detection over Social Media Platforms Using Hybrid Ensemble Model with Fuzzy Logic," *Electronics (Switzerland)*, vol. 12, no. 4, Feb. 2023, https://doi.org/10.3390/electronics12040937.

[32] D. K. Sharma, B. Singh, and A. Garg, "An Ensemble Model for detecting Sarcasm on Social Media," *Proceedings of the 2022 9th International Conference on Computing for Sustainable Global Development, INDIACom 2022*, pp. 743–748, 2022, https://doi.org/10.23919/INDIACom54597.2022.9763115.

[33] E. Savini and C. Caragea, "Intermediate-Task Transfer Learning with BERT for Sarcasm Detection," *Mathematics*, vol. 10, no. 5, 2022, https://doi.org/10.3390/math10050844.

[34] G. Li, F. Lin, W. Chen, and B. Liu, "Affection Enhanced Relational Graph Attention Network for Sarcasm Detection," *Applied Sciences (Switzerland)*, vol. 12, no. 7, 2022, https://doi.org/10.3390/app12073639.

[35] P. Parameswaran, A. Trotman, V. Liesaputra, and D. Eyers, "Detecting the target of sarcasm is hard: Really??," *Inf Process Manag*, vol. 58, no. 4, p. 102599, 2021, https://doi.org/10.1016/j.ipm.2021.102599.

[36] P. Goel, R. Jain, A. Nayyar, S. Singhal, and M. Srivastava, "Sarcasm detection using deep learning and ensemble learning," *Multimed Tools Appl*, 2022, https://doi.org/10.1007/s11042-022-12930-z.

[37] L. Zhou, X. Xu, and X. Wang, "BNS-Net: A Dual-Channel Sarcasm Detection Method Considering Behavior-Level and Sentence-Level Conflicts," *2024 International Joint Conference on Neural Networks (IJCNN)*, no. C, pp. 1–9, 2024, https://doi.org/10.1109/IJCNN60899.2024.10651549.