

## Article Information

**Submitted:** December 08, 2023

**Approved:** January 11, 2024

**Published:** January 12, 2024

**How to cite this article:** Faseeh M, Jamil H. A Capsule Neural Network (CNN) based Hybrid Approach for Identifying Sarcasm in Reddit Dataset. *IgMin Res.* Jan 12, 2024; 2(1): 013-017. IgMin ID: igmin137; DOI: 10.61927/igmin137; Available at: [www.igminresearch.com/articles/pdf/igmin137.pdf](http://www.igminresearch.com/articles/pdf/igmin137.pdf)

**Copyright license:** © 2024 Faseeh M, et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Keywords:** Sarcasm; CNN; LSTM; Sentiment; Word2vec; TF-IDF



## Mini Review



# A Capsule Neural Network (CNN) based Hybrid Approach for Identifying Sarcasm in Reddit Dataset

**Muhammad Faseeh and Harun Jamil\***

Department of Electronic Engineering, Jeju National University, Jeju-si, Jeju-do, Republic of Korea, 63243

\*Correspondence: Harun Jamil, Department of Electronic Engineering, Jeju National University, Jeju-si, Jeju-do 63243, Republic of Korea, Email: [harunjamil@hotmail.com](mailto:harunjamil@hotmail.com)

## Abstract

Sarcasm, a standard social media message, delivers the opposite meaning through irony or teasing. Unfortunately, identifying sarcasm in written text is difficult in natural language processing. The work aims to create an effective sarcasm detection model for social media text data, with possible applications in sentiment analysis, social media analytics, and online reputation management. A hybrid Deep learning strategy is used to construct an effective sarcasm detection model for written content on social media networks. The design emphasizes feature extraction, selection, and neural network application. Limited research exists on detecting sarcasm in human speech compared to emotion recognition. The study recommends using Word2Vec or TF-IDF for feature extraction to address memory and temporal constraints. Use feature selection techniques like PCA or LDA to enhance model performance by selecting relevant features. A Capsule Neural Network (CNN) and Long Short-Term Memory (LSTM) collect contextual information and sequential dependencies in textual material. We evaluate Reddit datasets with labelled sarcasm data using metrics like Accuracy. Our hybrid method gets 95.60% accuracy on Reddit.

## Introduction

Sarcasm involves saying something contrary to one's true meaning, often with a mocking tone and context-dependent cues. It can be challenging to detect in written English. For instance, a statement like "We have nice weather" on a rainy day may appear positive but can be sarcastic due to underlying frustration. Social media, like Twitter, has become famous for expressing thoughts on various topics. Many companies are keen on gauging public sentiment across diverse topics such as movies, songs, politics, and product reviews. In text analysis, it can be challenging to discern the true meaning of a sentence that may appear innocuous but is satirical or biting, like a review of a coffin being "very comfortable for sleeping." Automated methods, such as sarcasm identification, have emerged to tackle this issue, with studies like Shubham Rendhalkar, et al. [1] focusing on sentiment analysis. Sentiment analysis helps organizations understand people's opinions, including politics, services, and individuals. Sarcasm, a standard tool for societal commentary or expressing dissatisfaction, can also be employed for humour, wit, and cleverness.

Sarcasm is a nuanced and context-dependent form of communication. It relies on correctly interpreting cues such as tone, context, and shared cultural knowledge. Sarcasm's purpose is to

convey contradiction, which can lead to unclear communication if misunderstood. Sarcasm can be transmitted through tone of speech, facial expressions, and nonverbal cues, as well as in written forms like emails, texts, and social media posts. In recent times, sarcasm has gained popularity in digital communication due to its efficiency in delivering humour or criticism. However, written sarcasm is challenging to detect and can result in miscommunication and misunderstandings. Therefore, sarcasm is a distinct communication method that requires careful interpretation. Applications of sarcasm detection in various domains: Social Media Monitoring, Customer Service, Mental Health, Political Discourse, etc.

Most sarcasm detection systems today leverage NLP techniques and machine learning models. These techniques usually use sentiment analysis, language patterns, and contextual information to identify sarcastic sentences. However, there are issues with these methods that keep them from being precise. Many current methods effectively identify overt sarcasm but struggle with more complex or context-dependent cases. The complexity and context-dependence of sarcasm make it challenging for existing approaches to produce consistently correct findings.

The suggested method uses various machine-learning approaches to set itself apart from current methods. The suggested solution uses

cutting-edge deep learning models to embrace the dynamic nature of sarcasm, in contrast to many existing approaches that rely on predefined linguistic features. This methodology uses the power of Word2vec and TF-IDF with the Capsule CNN method to achieve the desired results.

## Literature review

Most sarcasm detection algorithms worldwide concentrate on separating utterances or classifying textual data. A limited amount of research is dedicated to altering textual data components to enhance classification accuracy significantly. Various Deep learning techniques are present to detect sarcasm detection in text [2-5]. To improve the accuracy of sarcasm categorization, Mehndiratta, et al. [6] Deep learning models such as CNN and LSTM were utilized, in addition to hybrid approaches such as CNN-LSTM and LSTM-CNN. They used the GloVe and FastText word embedding methods to analyze the SARC Reddit dataset, focusing on the deployment of balanced data. Qiao Nie L, et al. [7] use a Mutual-enhanced incongruity learning network for multi-modal sarcasm detection, one of the best novel techniques for multi-model sarcasm detection.

Usman Naseem, et al. [8] conducted research that studied sentiments expressed on social media platforms and identified irony and sarcasm. Their T-DICE intelligent contextual embedding strategy used transformer-based deep algorithms to deal with noise relevant to the context. Additionally, they used attention-based Bidirectional Long Short-Term Memory to determine sentiment in postings displayed across several platforms. A different group of researchers, Ghosh, et al. [9], used two other models to identify instances of sarcasm.

This research aims to investigate a deep neural network architecture that combines a recurrent neural network (RNN) with a convolutional neural network (CNN) for word-based deep learning inside a particular task embedding. Three different deep learning models, namely RNN, CNN, and Attentive RNN, are investigated in this study. These findings prove that Attentive RNN performs extraordinarily well when applied to datasets derived from Twitter. This correlates with the research by Zhang, et al. [10], which involves evaluating several neural network algorithms to identify sarcasm in tweets. In addition, the study's results investigated the effects of continuous automatic and discrete manual characteristics. More specifically, a bi-directional gated recurrent neural network is deployed to gather information about grammatical and semantic aspects. Researchers in this field have used local tweet data and neural networks to extract contextual clues from Twitter history automatically. Their findings reveal that adding neural features enhances sarcasm detection accuracy, with different error distributions compared to discrete manual approaches. They've also used sophisticated false positive detection techniques [11].

Tay, et al. [12] investigated sarcasm study using LSTM-based intra-attention. They created a system for detecting multi-modal sarcasm in tweets by analyzing text and image data from Twitter. The work proposes a multi-modal hierarchical fusion model that combines information from text, images, and image attributes across

three modalities. They also constructed a multi-modal sarcasm detection dataset based on Twitter. They discovered their approach, and the three modalities were valuable and beneficial, as seen by dataset evaluation findings.

AAFAB is an Adversarial and Auxiliary Features-Aware Model (AAFAB) proposed by A. Kumar and Narapareddy [13]. This model captures semantics by capturing sentence meaning with contextual word embeddings from BERT and integrating it with high-quality manually generated auxiliary features for sarcasm detection. MHABiLSTM (Multi-head Attention Bidirectional LSTM) is their proposed model for identifying sarcasm in the SARC Reddit Dataset. Experiment results indicate that combining AAFAB with supplemental features and adversarial training improves performance.

## Methodology

The dataset used for this task is the Self-Annotated Reddit Corpus (SARC), which contains 1.3 million sarcastic comments from Reddit. The dataset was created by harvesting Reddit comments with the sarcasm tag. Redditors frequently use this tag to indicate that their comment is intended to be regarded in jest and should not be taken seriously, and it is generally a reliable indicator of sarcastic comment content. Data is available in balanced and imbalanced (i.e. accurate distribution) forms. (True ratio is roughly 1:100). We used an 80/20 ratio for training and testing.

The methodology used in Figure 1 in this work entails several procedures to preprocess the text data and build an effective sarcasm detection model. Preprocessing the Reddit dataset is essential in improving data quality before applying more complex algorithms like Word2Vec and Tf-IDF. Tokenization is the first stage in breaking down text into discrete pieces called tokens, which are often words or sentences. Tokenization is required in the case of the Reddit dataset to convert the raw text into a format that can be easily analyzed. This technique contributes to comprehending the text's structure, allowing token-level stages such as stop word removal and stemming to be applied. Stop words are common words with no significant meaning and are frequently deleted to focus on the more informative keywords. Examples include the words "the," "and," and "is." During preprocessing, removing stop words reduces noise in the dataset, allowing the model to focus on the relevant material and enhance efficiency during subsequent analysis. Lowercasing is the process of turning all text to lowercase. This step guarantees that the text data is consistent, as machine learning models may treat uppercase and lowercase versions of the same word as separate entities. Lowercasing solves this problem by ensuring uniformity in word representation and increasing the model's overall performance. Stemming is reducing words to their root or basic form by removing suffixes. For example, "running" is shortened to "run." This technique is especially important when different spellings of the same term have the same meaning. By treating related words as the same item, stemming reduces the complexity of the dataset and improves the model's capacity to generalize. Lemmatization is a more complicated strategy than stemming in that it entails reducing words to their base or root form while considering context and meaning. Instead

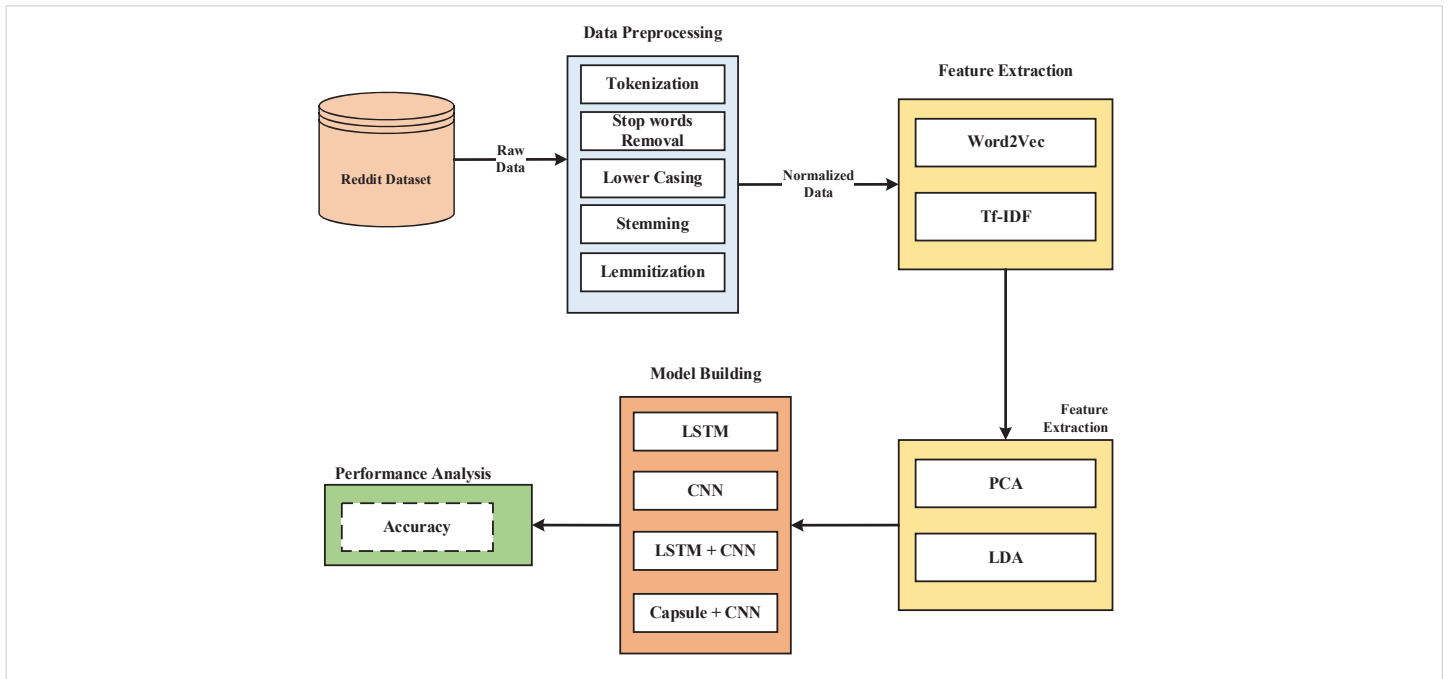


Figure 1: Proposed Capsule Neural Network (CNN) based Hybrid Approach.

of stemming, Lemmatization assures that the generated words are legitimate and meaningful. This stage adds to better semantic understanding, essential in applications where the original words must be preserved.

After completing these preparation processes, the Reddit dataset suits more complex approaches such as Word2Vec and Tf-IDF. Word2Vec collects semantic associations between words and generates vector representations that reflect their context. Tf-IDF (Term Frequency-Inverse Document Frequency) applies weights to words based on their frequency in a document and across the entire dataset, allowing essential phrases to be identified while reducing the influence of common words. These strategies, when combined, form a robust foundation for deriving valuable insights from the Reddit dataset.

Following the initial preprocessing phases, using Principal Component Analysis (PCA) and Latent Dirichlet Allocation (LDA) improves the feature representation and gives more profound insights into the Reddit dataset. PCA is a dimensionality reduction approach that converts high-dimensional data into a lower-dimensional space while maintaining the most critical information. After tokenization, top word removal, lowercasing, stemming, and lemmatization, the Reddit dataset may still contain many features (words). PCA aids in dimensionality reduction by identifying the primary components or directions in the data that capture the most variance. This not only expedites subsequent analysis but also helps to visualize the material in a more manageable space. Using PCA may remove redundant information, and the model's computational efficiency can be enhanced without surrendering too much information. LDA is a probabilistic model used for document categorization and topic modelling. LDA can be used to detect hidden topics inside documents in the context of the Reddit dataset. Each document (post or comment)

is a bag of words after the initial preparation procedures. LDA then identifies subjects in the documents likely to generate the observed words. This aids in comprehending the underlying themes present in the dataset and allows for document categorization based on these themes. LDA provides a probabilistic framework for assigning themes to documents and words to topics, making it a valuable tool for exploring the Reddit dataset's latent structures.

By including PCA and LDA after the first preprocessing stages, we optimize the dataset for efficient modelling and obtain insights into the underlying structures and topics of the Reddit content. These strategies aid in extracting relevant patterns and allow for a more thorough understanding of the data included in the dataset. PCA's decreased dimensionality and LDA's identified topics are valuable inputs for machine learning tasks.

In sarcasm detection, my research employed various deep learning architectures, including LSTM, CNN, LSTM + CNN, and Capsule with CNN. Through rigorous experimentation and analysis, it became evident that the Capsule with CNN model outperformed the others in accuracy and effectiveness. Combining the strengths of capsule and convolutional neural networks, this hybrid approach demonstrated superior capabilities in capturing intricate patterns and nuanced contextual information, which are essential for discerning sarcastic nuances in text. The robust performance of the Capsule with CNN model underscores its potential as a powerful tool for sarcasm detection tasks, showcasing the significance of innovative hybrid architectures in natural language processing applications. The comparative evaluation of LSTM, CNN, LSTM + CNN, and Capsule with CNN highlights the advancements in deep learning methodologies and emphasizes the importance of choosing the exemplary architecture for specific tasks, ultimately contributing to the advancement of sentiment analysis and natural language understanding.

## Experimental results

We use accuracy as an evaluation matrix for sarcasm prediction using the Reddit dataset. Accuracy is a popular statistic for evaluating classification jobs whose goal is to anticipate the class or category of instances correctly. Accuracy can be used to evaluate the performance of your selected machine learning models after feature extraction and dimensionality reduction in the context of your preprocessing and subsequent analysis of the Reddit dataset. Accuracy is achieved through equation 1.

$$Accuracy = \frac{(TP + TN)}{(TP + FP + TN + FN)} \tag{1}$$

Various combinations of neural network models have been evaluated for accuracy until the maximum results are achieved, as shown in Table 1 and Figure 3. The first model, a combination of Capsule Networks and Convolutional Neural Networks (CNN), demonstrates a high accuracy of 0.956. This suggests an effective synergy between the Capsule Network’s ability to capture hierarchical relationships and CNN’s prowess in feature extraction. Next, a model employing Long Short-Term Memory (LSTM) networks alone achieves an accuracy of 0.876. This indicates LSTM’s competence in handling sequential data, although it’s slightly less effective than the Capsule + CNN model. The standalone CNN model closely rivals the top performer with an accuracy of 0.955, underscoring

**Table 1:** Accuracy comparison of Multiple techniques.

Techniques	Accuracy
LSTM	0.876
LSTM+CNN	0.894
CNN	0.955
Capsule + CNN	0.956

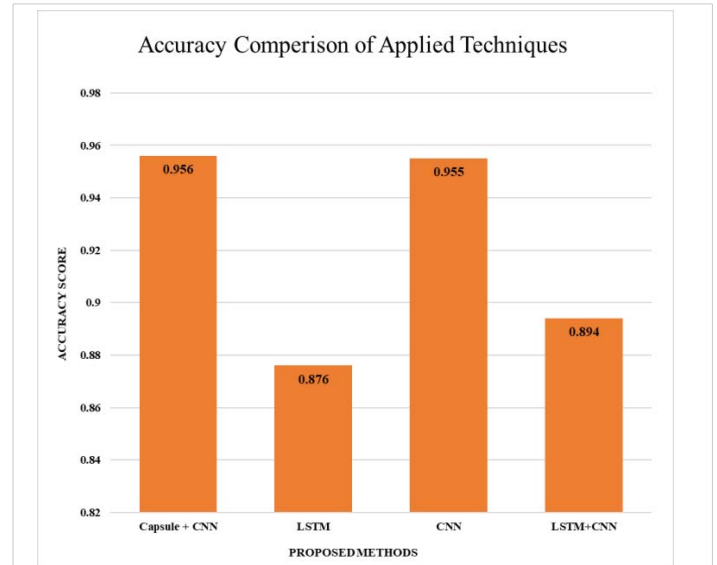


Figure 3: Accuracy Comparison of Applied Techniques.

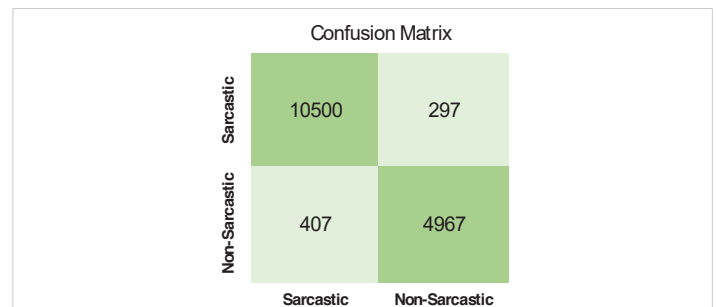


Figure 4: Confusion matrix of Capsule + CNN technique with best results.

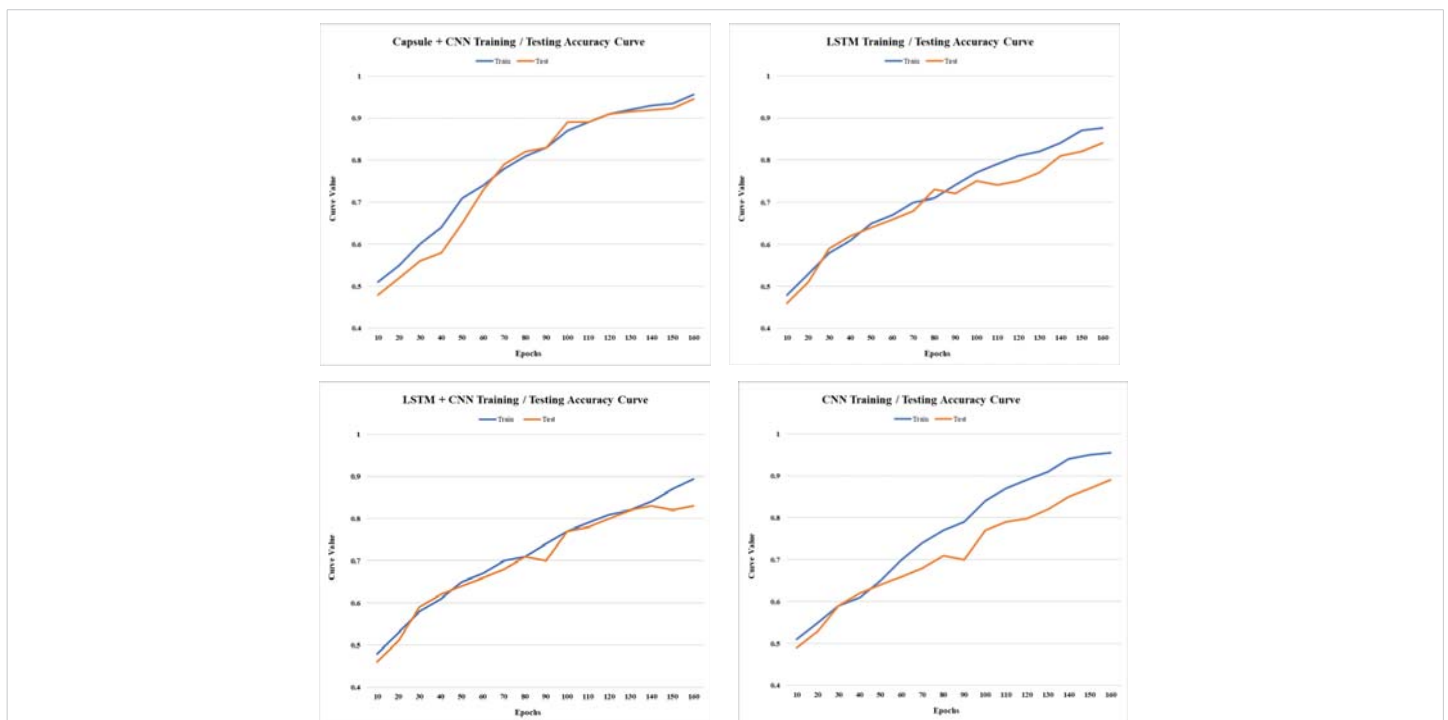


Figure 2: The comparison of the Proposed Capsule Neural Network (CNN) based Hybrid Approach with other Approaches in Terms of Training and Testing Loss.

CNN's robustness in pattern recognition tasks. Lastly, the hybrid LSTM+CNN model, which combines the sequential data processing strength of LSTM with the spatial data processing ability of CNN, achieves an accuracy of 0.894. This blend, while effective, seems to be slightly less efficient than the standalone CNN or the Capsule + CNN models in this specific dataset (Figure 2-4).

## Conclusion

The hybrid strategy of merging Capsule Neural Network (CNN) and LSTM models has provided numerous benefits for text data processing in NLP. The Capsule Neural Network (CNN) supports hierarchical feature representation, collecting local and global text input aspects. The power of Word2vec and Tf-IDF, along with PCA and LDA, shows tremendous results with an accuracy of 0.956. Combining Features extracted from PCA and LDA, Capsule Neural networks can capture complex patterns and representations in text data, leading to increased performance in various NLP tasks such as text classification, sentiment analysis, named entity recognition, etc. The experimental results and literature study indicate that the hybrid strategy can outperform individual models or other traditional approaches in terms of performance. Depending on the dataset and task, it has demonstrated promising outcomes regarding accuracy. The hybrid technique has also proven robustness to varying-length inputs, adaptability to diverse NLP tasks, and, in some situations, the possibility for state-of-the-art performance.

In the future, our goal will go beyond the successful online deployment of the sarcasm detection algorithm, as we aim to develop and improve its performance constantly. The inclusion of real-time learning processes is one possible option for future research. We hope to secure the system's ability to identify sarcasm in the ever-changing landscape of internet communication by integrating adaptive algorithms that can dynamically update the model based on growing language trends and nuances in online conversation. Furthermore, user feedback and participation will be critical in the model's progress, as we intend to construct a feedback loop that allows for continual improvement based on user input. Furthermore, we intend to expand the model's applicability to multiple social media sites, adjusting its architecture to each platform's specific characteristics to give a seamless and successful sarcasm detection experience. Finally, we intend to stay on the cutting edge of advances in natural language processing and machine learning, allowing our

sarcasm detection model to meet current needs and evolve to address emerging challenges in the dynamic realm of online communication.

## References

1. Shubham R, Chandankhede C. Sarcasm detection of online comments using emotion detection. 2018 International conference on inventive research in computing applications (ICIRCA). IEEE, 2018.
2. Vitman O, Kostiuk Y, Sidorov G, Gelbukh A. Sarcasm detection framework using context, emotion, and sentiment features. *Expert Systems with Applications*. 2023; 234: 121068.
3. Šandor D, Babac MB. Sarcasm detection in online comments using machine learning. *Information Discovery and Delivery*, (ahead-of-print). 2023.
4. Pandey R, Singh JP. BERT-LSTM model for sarcasm detection in code-mixed social media posts. *Journal of Intelligent Information Systems*. 2023; 60(1): 235-254.
5. Tan YY, Chow CO, Kanesan J, Chuah JH, Lim Y. Sentiment Analysis and Sarcasm Detection using Deep Multi-Task Learning. *Wirel Pers Commun*. 2023;129(3):2213-2237. doi: 10.1007/s11277-023-10235-4. Epub 2023 Mar 4. PMID: 36987507; PMCID: PMC9985100.
6. Pulkit M, Soni D. Identification of sarcasm using word embeddings and hyperparameters tuning. *Journal of Discrete Mathematical Sciences and Cryptography*. 2019; 22.4: 465-489.
7. Qiao Y, Jing L, Song X, Chen X, Zhu L, Nie L. Mutual-enhanced incongruity learning network for multi-modal sarcasm detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 2023; 37: 9507-9515.
8. Usman N. Towards improved deep contextual embedding for the identification of irony and sarcasm. 2020 International joint conference on neural networks (IJCNN). IEEE. 2020.
9. Aniruddha G, Veale T. Fracking sarcasm using neural network. *Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment, and social media analysis*. 2016.
10. Zhang M, Zhang Y, Fu G. Tweet sarcasm detection using deep neural network. In *Matsumoto Y., Prasad R. (Eds.), Proceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers*. 2016; 2449-2460.
11. Alexandros PR, Siolas G, Stafylopatis AG. A transformer-based approach to irony and sarcasm detection. *Neural Computing and Applications*. 2020; 32: 17309-17320.
12. Yi T. Reasoning with sarcasm by reading in-between. *arXiv preprint arXiv:1805.02856* (2018).
13. Avinash K. Adversarial and auxiliary features-aware bert for sarcasm detection. *Proceedings of the 3rd ACM India Joint International Conference on Data Science Management of Data (8th ACM IKDD CODS 26th COMAD)*. 2021.