Sarcasm Detection from Dravidian Language Text

C.Jerin Mahibha[1,*], Gersome Shimi[2] and Durairaj Thenmozhi[3]

[1]Meenakshi Sundararajan Engineering College, Chennai
[2]Madras Christian College,Chennai, India
[3]Sri Sivasubramaniya Nadar College of Engineering, Chennai

Abstract

Sarcasm is a form of verbal irony where someone says the opposite of what they really mean, often to mock or convey contempt. It's typically used to emphasize a point or highlight absurdity. Sarcasm is used to express criticism, mockery, or disdain in a more indirect or humorous way. Sarcasm detection can be tricky because it often relies on tone, context, and sometimes even body language, which aren't always present in text. There is an increasing demand for sarcasm and sentiment detection on social media texts which are largely code-mixed for Dravidian languages. Code-mixing is a prevalent phenomenon in a multilingual community and the code-mixed texts are sometimes written in non-native scripts. Systems trained on monolingual data fail on code-mixed data due to the complexity of code-switching at different linguistic levels in the text. We participated in the shared task at DravidianCodeMix@FIRE-2024 and have proposed a model that identifies the sarcasm and sentiment polarity of the code-mixed social media comments and posts in Tamil-English and Malayalam-English using the data set shared for the task. We contributed a Language Agnostic Embedding based Multi Layer Perceptron (MLP) classifier to process the classification of text as Sarcastic and Non sarcastic. The macro F1 score obtained by the proposed model for the language Tamil is 0.68 and 0.70 for Malayalam.

1. Introduction

Sarcasm is a way to express one's thoughts differently, which would not hurt them directly. The information may be prompted in a humorous way that does not harm the stakeholders directly but make them think and interpret the content. Sarcasm often relies on tone and context, which cannot be easily understood online. The absence of verbal cues like voice intonation or facial expressions can lead to misinterpretation, turning what was meant as humor into something offensive or harmful[1]. The shares task DravidianCodeMix@FIRE-2024 [2] aims in identifying the texts which has sarcasm associated with it. Even though many algorithms prevail in classifying text, it may struggle to distinguish between sarcasm and genuine harm or abuse, which is difficult. so some advocacy of scrutiny or content moderation need to be performed which is challenging in determining the intent or context of the text, and excessive filtering can risk suppressing free expression [3]. Identifying sarcasm in code-mixed languages presents a pivotal challenge within the domain of natural language processing, given the widespread use of multilingual and multicultural communication on social media platforms [4], [5].

Sarcasm detection relies on various features, such as Lexical, Syntactic and Semantic. Lexical feature analyze specific words and phrases commonly associated with sarcasm. Syntactic Features examine grammatical structures and parts of speech for unusual patterns. Semantic features interpret contextual meaning and sentiment to identify contradictions or implied sarcasm. Effective detection often combines these features, particularly when using machine learning techniques [6]. Recently, BERT (Bidirectional Encoder Representations from Transformers) has become a cutting-edge tool for a range of natural language processing tasks, including detecting sarcasm. Its ability to understand context and nuances in text makes it highly effective for identifying sarcastic remarks [7].

Most of the existing classification systems use machine learning models on a labelled dataset and have been successful in detecting and eradicating negativity. However, to enhance free expression

Forum for Information Retrieval Evaluation, December 12-15, 2024, India.

[*]C.Jerin Mahibha
[*]Corresponding author.
[†]These authors contributed equally.

jerinmahibha@msec.edu.in (C.Jerin Mahibha); gshimi2022@gmail.com (G. Shimi); theni_d@ssn.edu.in (D. Thenmozhi)

through social media, instead of eliminating ostensibly unpleasant words, positivity in the comments could be recognize and encouraged [8]. Research activities on low resource languages like Dravidian languages could be enhanced by the process of data augmentation like Pseudolabeling [9].

The paper is organized with Section 2 and 3 discussing on related works and datasets, Section 4 and Section 5 discuss on system description and results, Section 6 and 7 contributes the error analysis and conclusion.

2. Related Works

Transformer models like BERT, RoBERTa, and DeBERTa had been used for the process of sarcasm classification Jang and Frassinelli [10] which had focused on cross-data comparisons. Madhumitha M et al. had also used different transformer models for detecting sarcasm from Tamil text [11]. An automated system had been designed to detect sarcasm and classify its various types, to extract valuable insights from user reviews by leveraging Natural Language Processing (NLP) and Deep Learning (DL) algorithms. The data included 55,000 end-user comments collected from seven software applications on the Play Store. A unique sarcasm coding framework had been developed by Fatima et al. [12] through a detailed analysis of the reviews, identifying common sarcastic expressions such as Irony, Humor, Flattery, Self-Deprecation, and Passive Aggression. The system compared the results of different DL models by fine-tuning the parameters of the classifiers. The feedback and comments had been pre-processed and balanced for more accurate analysis and classification.

A multitask learning framework had been introduced by Tan et al. [13], that leverages a deep neural network to model that relates sarcasm detection and sentiment analysis, improving overall performance. The method had surpassed current approaches, achieving a 3% improvement with an F1-score of 94%. Krishnan et al. [14] had used a methodology with ELMo embedding based Convolutional Neural Network model and TF-IDF based Gaussian Naive Bayes classifier using the dataset provided by SemEval-2022 to discern and classify different types of irony within textual content.

A context-based technique had been developed for sarcasm detection using three models, Bi-LSTM, BERT and Feature Fusion by Eke et al. [15]. Bi-LSTM Model utilized embedding-based representation with Bidirectional Long Short-Term Memory (Bi-LSTM) and Global Vector (GloVe) embeddings for context learning. BERT based model employed a Transformer architecture with pre-trained Bidirectional Encoder Representations from Transformers (BERT) for context understanding. Feature Fusion Model, combined BERT features, sentiment-related features, syntactic features, and GloVe embeddings with conventional machine learning methods. The techniques were tested on two Twitter datasets and achieved high precision rates of 98.5% and 98.0%, in sarcasm identification.

A method to detect sarcasm from Telugu and Tamil tweets had been proposed using a dataset from Twitter [16]. The approach included Machine Learning models like Decision Tree, Naive Bayes, Logistic Regression, Support Vector Machine, and Random Forest—and Deep Learning techniques like Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. The results demonstrated high performance, with accuracy rates of 95.68% for Telugu tweets and 95.37% for Tamil tweets.

3. Data set

The datasets considering languages Tamil and Malayalam that are used to implement sarcasm detection were the training, evaluation, and test datasets that were provided by the organizers of the shared task. Each instance of the training dataset had a label specifying whether the text was sarcastic or non-sarcastic.

The data distribution of the training and development dataset for Tamil is shown in Table 1, and for Malayalam, it is shown in Table 2. In the case of Tamil, the training dataset comprised 29570 instances, with 7830 falling under the sarcastic category and 21740 under the non-sarcastic category. Similarly, the development dataset for Tamil contained 6336 instances, including 1706 in the sarcastic category and the

Table 1
Data Distribution for Tamil

Category	Training Dataset	Evaluation Dataset
Sarcastic	7830	1706
Non-Sarcastic	21740	4630

Table 2
Data Distribution for Malayalam

Category	Training Dataset	Evaluation Dataset
Sarcastic	2499	521
Non-Sarcastic	10689	2305

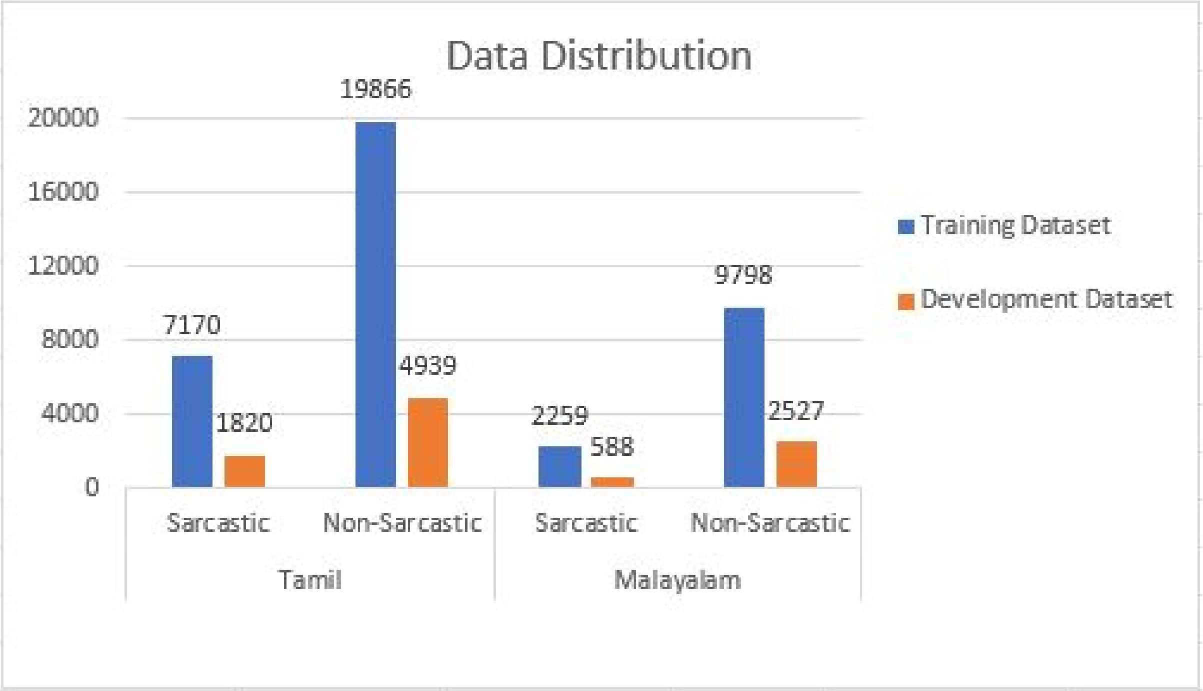

Figure 1: Data distribution

rest in the non-sarcastic category, highlighting the data's imbalance. The test dataset for Tamil consisted of 6338 instances, which were used for evaluating the model's predictions. For Malayalam, the training dataset consisted of 13188 instances, with 2499 categorized as sarcastic and 10689 as non-sarcastic. The development dataset for Malayalam had 2826 instances, including 521 in the sarcastic category and the remaining in the non-sarcastic category, indicating a similar data imbalance. Lastly, the test dataset for Malayalam encompassed 2826 instances, utilized for assessing the model's predictive performance.

4. System Description

The proposed system uses a MLP classifier with Language Agnostic embeddings for the task of binary classification of Tamil and Malayalam text into Sarcastic and Non-Sarcastic category. The proposed architecture of the system is shown in Figure 2. The initial phase of the process involved gathering the three datasets provided by the task organizers: the training dataset, development dataset, and testing dataset. The training dataset was preprocessed, which resulted in a clean and structured data. This prepared dataset was used to train the model and was evaluated using the development dataset.

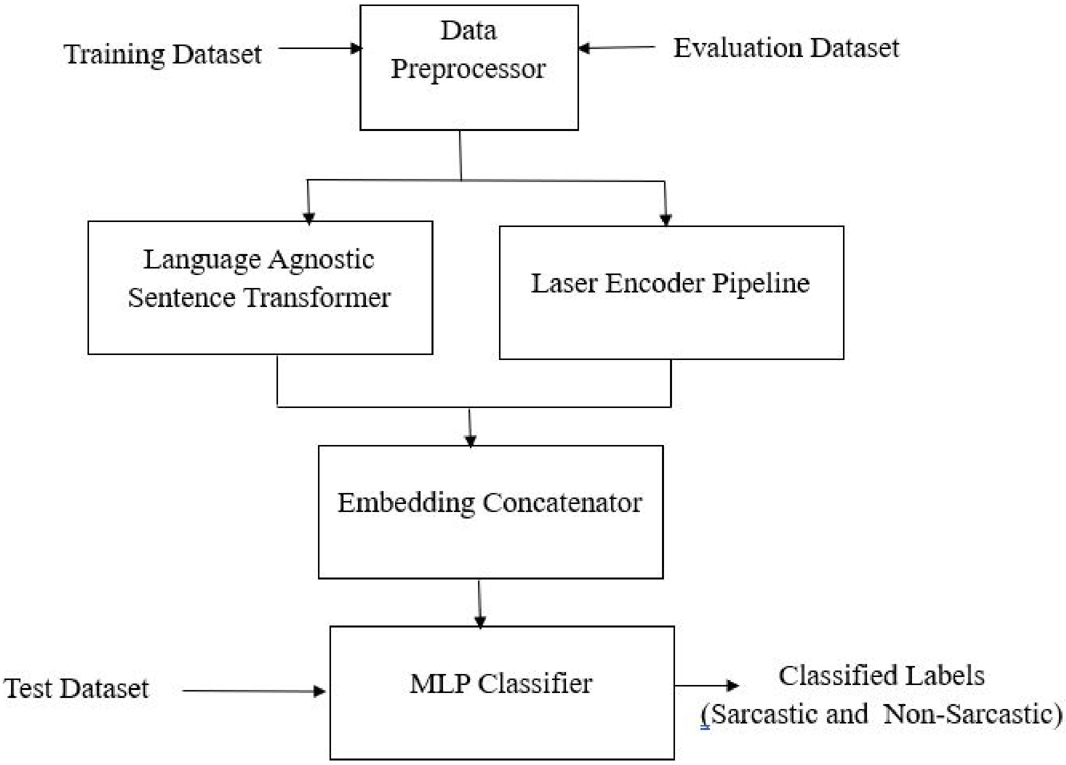

Figure 2: System Architecture

4.1. Preprocessing

The dataset was cleaned by the process of preprocessing. Preprocessing is the technique of removing unimportant information from texts, which are not used during the classification process. It is performed by removing stop words, symbols, and special characters in addition to that root words are extracted using stemmer and lemmatization algorithms before the dataset is fed to the model.

4.2. MLP classifier with Language Agnostic Embeddings

The proposed system used a MLP classifier for which custom generated embedding was provided as input. Language agnostic sentence transformer was used to generate text embeddings. As the Language agnostic sentence transformer is multilingual in nature and support both Tamil and Malayalam languages, the same model was used to generate the embeddings for all the given texts. Similarly Laser encoder pipeline was used to generate LASER embeddings for all the texts. Both these embeddings were concatenated to generate a final set of embeddings using which the MLP classifier was trained. The hyper parameters associated with the MLP classifier are: random state was set as 42, the maximum iteration was set as 300, relu activation function was used, the parameter alpha was set as 0.05, learning rate as adaptive and solver as adam.

The model was then used to make predictions for the testing dataset. The entire process, from data preprocessing to model building and evaluation, is illustrated in the proposed architecture. This architecture underscores the significance of data preprocessing, model training, and fine-tuning in enhancing the accuracy of our sarcasm detection system. During testing, the selected model generated contextual embeddings and predicted labels for the text in the dataset, playing a crucial role in identifying sarcasm. The proposed model when evaluated using the evaluation dataset, it provided a Macro F1 Score 0.68 and 0.7 for Tamil and Malayalam respectively.

Table 3
Performance score

Model	F1-Score
Tamil	0.68
Malayalam	0.70

5. Result

The metrics that was used for the evaluation of the task was the macro-F1 score. The F1 score is an overall measure of a model's accuracy that merges precision and recall. An extreme F1 score means that the classification has happened, accompanied by a reduced number of false positives and low false negatives. Within a classification report, the initial focus is on assessing a binary classification model for two distinct categories, typically labeled as 0 and 1. For class 0, precision provides insights into how accurately our model predicts instances as class 0, while recall sheds light on how effectively it identifies actual instances belonging to class 0. Conversely, concerning class 1, precision aids in evaluating the accuracy of our predictions for this specific category, while recall delves into the model's ability to pinpoint genuine class 1 instances. These fundamental metrics play a pivotal role in comprehending the model's effectiveness for each class within a binary classification scenario. The performance metrics, particularly the F1 score of development dataset, for both Tamil and Malayalam are presented in Table 3 The evaluation of the tasks primarily relied on the macro-F1 score achieved by our proposed model. It is worth emphasizing that our proposed model yielded remarkable results, securing a macro F1 score of 0.68 for Tamil and 0.70 for Malayalam, forming the basis for task evaluation.

The proposed system culminated in securing the 7th position in the Tamil language category and the 6th position in the Malayalam language category on the leaderboard. These rankings underscore the effectiveness of our model in the Dravidian CodeMix@FIRE-2024 shared task, highlighting its prowess in sarcasm detection within code-mixed Tamil-English and Malayalam-English social media comments and posts.

6. Error Analysis

The F1 score obtained for the task using the proposed Language Agnostic model shows that more false positive and false negative classifications have occurred. One reason for this could be considered as the data imbalance nature of the dataset. The confusion matrix of the proposed system considering the languages Tamil and Malayalam is represented in figures 3 and **??**, respectively. Considering the number of instances for the class labeled non-sarcastic is higher, and the F1 score, precision, and recall associated with this class are high when compared to the class, sarcastic. This represents that the number of misclassifications increases when the number of instances for training is lower, which is associated with data imbalance. Data augmentation could be considered to improve the model's performance. Examples of Tamil texts that are misclassified due to the above reasons are shown in Table 4. Considering the first and the second text of the table, it has the specific sarcasm marker ”kola gaandla”, ”mass” and are classified as sarcastic instead of the correct label of non-sarcastic. The third and fourth text of the table has the sarcasm marker ”veriyan ” and ”vasul manan....” which has been misclassified as Non sarcastic. All these example texts show that sarcasm markers and sarcasm play a major role in the process of classification and identifying whether the text is associated with sarcasm. The binary classification of the task is implemented by generating a concatenated language agnostic embedding for training and testing the dataset. The task was implemented considering the languages Tamil and Malayalam.

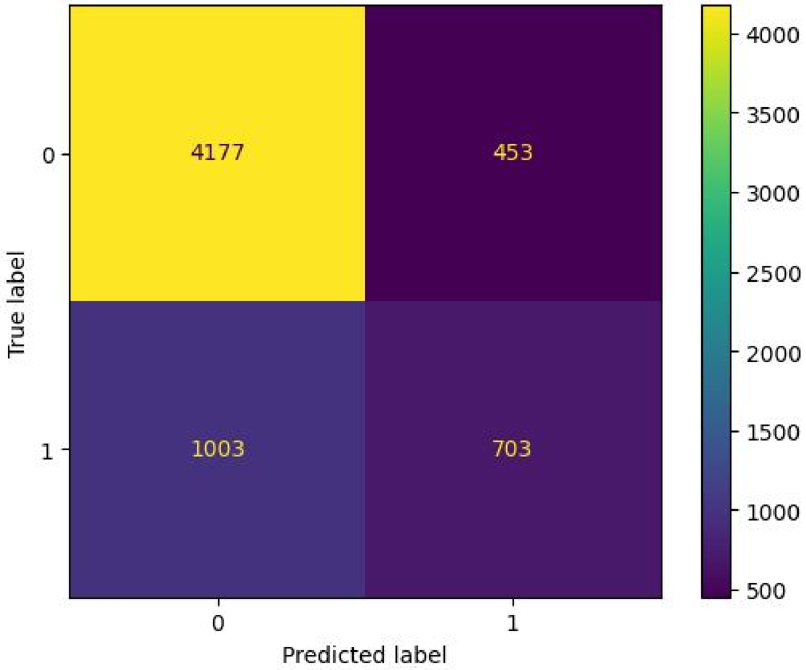

Figure 3: Classification Report - Tamil

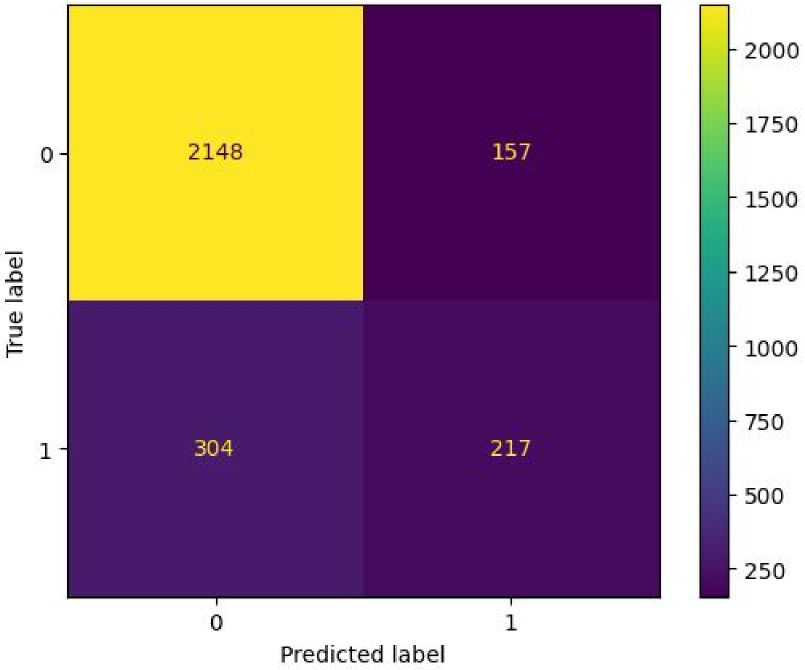

Figure 4: Classification Report - Malayalam

7. Conclusion

The process of detecting sarcasm from Dravidian language texts has gained considerable importance due to its interconnectedness with various application domains. Recognizing this significance, FIRE-2024 introduced a shared task that focused on sarcasm detection within Dravidian languages, classifying text into either sarcastic or non-sarcastic categories. The exploration of this task, conducted under the banner of Dravidian CodeMix@FIRE-2024, underscores the value of harnessing advanced natural

Table 4
Error Analysis

S.No.	Text	Actual Label	Predicted Label
01.	1st day la 3 shows kku ticket eduthuttu padam release aagaama kola gaandla irukkaen.. ..	Non-Sarcastic	Sarcastic
02.	Jayam ravi mass panna porean da	Non-Sarcastic	Sarcastic
03.	I am thalapathi veriyan but Thala trailar super	Sarcastic	Non-Sarcastic
04.	yanga thalapathi tha da vasul manan.....#manda	Sarcastic	Non-Sarcastic

language processing techniques for the analysis of sarcastic texts. The proposed system, we employed a language agnostic embedding with MLP classifier, to tackle the challenge of detecting signs of sarcasm within social media text. This approach allowed to categorize social media content into sarcastic and non-sarcastic category.

The outcomes derived from the proposed model exhibit effectiveness in capturing subtle linguistic cues and contextual information indicative of sarcasm. The opportunities for further enhancement can involve the incorporation of more extensive contextual information, which could enhance the accuracy of sarcasm detection. Additionally, the adoption of hybrid approaches, leveraging the strengths of various deep learning models, may further bolster the efficiency and precision of detecting sarcasm within the text data.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] D. Bharti, B. Vachha, R. Pradhan, K. Babu, S. Jena, Sarcastic sentiment detection in tweets streamed in real time: A big data approach, Digital Communications and Networks 2 (2016). doi:10.1016/j.dcan.2016.06.002.

[2] B. R. Chakravarthi, S. N, B. B, N. K, T. Durairaj, R. Ponnusamy, P. K. Kumaresan, K. K. Ponnusamy, C. Rajkumar, Overview of sarcasm identification of dravidian languages in dravidiancodemix@fire-2024, in: Forum of Information Retrieval and Evaluation FIRE - 2024, DAIICT , Gandhinagar, 2024.

[3] D. K. Sharma, B. Singh, S. Agarwal, N. Pachauri, A. A. Alhussan, H. A. Abdallah, Sarcasm detection over social media platforms using hybrid ensemble model with fuzzy logic, Electronics 12 (2023). URL: https://www.mdpi.com/2079-9292/12/4/937. doi:10.3390/electronics12040937.

[4] N. Sripriya, T. Durairaj, K. Nandhini, B. Bharathi, K. K. Ponnusamy, C. Rajkumar, P. K. Kumaresan, R. Ponnusamy, C. Subalalitha, B. R. Chakravarthi, Findings of shared task on sarcasm identification in code-mixed dravidian languages, FIRE 2023 16 (2023) 22.

[5] B. R. Chakravarthi, N. Sripriya, B. Bharathi, K. Nandhini, S. Chinnaudayar Navaneethakrishnan, T. Durairaj, R. Ponnusamy, P. K. Kumaresan, K. K. Ponnusamy, C. Rajkumar, Overview of the shared task on sarcasm identification of Dravidian languages (Malayalam and Tamil) in DravidianCodeMix, in: Forum of Information Retrieval and Evaluation FIRE - 2023, 2023.

[6] B. Rajani, S. Saxena, B. S. Kumar, Detection of sarcasm in tweets using hybrid machine learning method, Journal of Autonomous Intelligence 7 (2024). URL: https://jai.front-sci.com/index.php/jai/article/view/800. doi:10.32629/jai.v7i4.800.

[7] J. D. M.-W. C. Kenton, L. K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of naacL-HLT, volume 1, 2019, p. 2.

[8] B. R. Chakravarthi, Hope speech detection in youtube comments, Social Network Analysis and Mining 12 (2022) 75.

[9] B. R. Chakravarthi, A. Hande, R. Ponnusamy, P. K. Kumaresan, R. Priyadharshini, How can we detect homophobia and transphobia? experiments in a multilingual code-mixed setting for social media governance, International Journal of Information Management Data Insights 2 (2022) 100119.

[10] H. Jang, D. Frassinelli, Generalizable sarcasm detection is just around the corner, of course!, 2024. URL: https://arxiv.org/abs/2404.06357. arXiv:2404.06357.

[11] M. M, K. Akshatra M, T. J, C. Mahibha, D. Thenmozhi, Sarcasm detection in dravidian languages using transformer models (2023).

[12] E. Fatima, H. Kanwal, J. A. Khan, N. D. Khan, An exploratory and automated study of sarcasm detection and classification in app stores using fine-tuned deep learning classifiers, Automated Software Engineering 31 (2024). URL: https://link.springer.com/10.1007/s10515-024-00468-3. doi:10.1007/s10515-024-00468-3.

[13] Y. Y. Tan, C. O. Chow, J. Kanesan, J. H. Chuah, Y. Lim, Sentiment analysis and sarcasm detection using deep multi-task learning, Wireless Personal Communications 129 (2023) 2213 – 2237. URL: https://api.semanticscholar.org/CorpusID:257355920.

[14] D. Krishnan, J. M. C, T. Durairaj, GetSmartMSEC at SemEval-2022 task 6: Sarcasm detection using contextual word embedding with Gaussian model for irony type identification, in: Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), Association for Computational Linguistics, Seattle, United States, 2022, pp. 827–833. URL: https://aclanthology.org/2022.semeval-1.114. doi:10.18653/v1/2022.semeval-1.114.

[15] C. I. Eke, A. A. Norman, L. Shuib, Context-based feature technique for sarcasm identification in benchmark datasets using deep learning and bert model, IEEE Access 9 (2021) 48501–48518. doi:10.1109/ACCESS.2021.3068323.

[16] R. P. Kumar, G. Bharathi Mohan, Y. Kakarla, J. S. L, K. Gnapika Sindhu, T. V. Sai Surya Chaitanya, B. Ganesh, N. H. Krishna, Sarcasm detection in telugu and tamil: An exploration of machine learning and deep neural networks, in: 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2023, pp. 1–7. doi:10.1109/ICCCNT56998.2023.10306775.