
*Asian Journal of Research in Computer Science*

***Volume 17, Issue 5, Page 212-221, 2024; Article no.AJRCOS.113437***
***ISSN: 2581-8260***

# Sarcasm Detection in Pidgin Tweets Using Machine Learning Techniques

**Khadijat T Ladoja [a*] and Ruth T Afape [a]**

[a] *Department of Computer Science, Faculty of Science, University of Ibadan, Nigeria.*

***Authors' contributions***

*This work was carried out in collaboration between both authors. Both authors read and approved the final manuscript.*

***Article Information***

DOI: 10.9734/AJRCOS/2024/v17i5450

**Open Peer Review History:**
This journal follows the Advanced Open Peer Review policy. Identity of the Reviewers, Editor(s) and additional Reviewers, peer review comments, different versions of the manuscript, comments of the editors, etc are available here: https://www.sdiarticle5.com/review-history/113437

***Original Research Article***

***Received: 09/01/2024***
***Accepted: 13/03/2024***
***Published: 18/03/2024***

## ABSTRACT

Detecting sarcasm in social media is of growing importance for applications such as monitoring, consumer feedback, and sentiment analysis. However, detecting sarcasm in Pidgin tweets poses unique challenges due to the blend of English and Pidgin languages, along with local cultural references. Existing models for sarcasm detection in English lack appropriate annotated data for Pidgin. This scarcity hinders the development of effective machine learning models. This research aims to address these challenges and create a model for accurate sarcasm detection in Pidgin tweets. Logistic Regression, XGBoost, Random Forest, and Vanilla Artificial Neural Network (ANN) classifiers were assessed, focusing on accuracy, precision, recall, and F1-score metrics on sarcasm data collected by curating and pre-processing a dataset of Nigerian Pidgin tweets. The XGBoost model demonstrated notable performance, attaining an accuracy of 85.78%, precision of 88.57%, recall of 94.44%, and F1-score of 91.41%. These outcomes underscored the model's prowess in discerning sarcastic and non-sarcastic expressions. By unfolding the intricacies of language in the Nigerian context, this research into sarcasm identification in Nigerian Pidgin text data introduced a comprehensive pipeline encompassing data curation, exploratory analysis, culturally tailored pre-processing, model training, evaluation, and prediction.

**Corresponding author: E-mail: ktladoja35@gmail.com;*

*Keywords: Sentiment analysis; machine learning; sarcasm; text data; pidgin; dataset; social media; linguistic landscape.*

## 1. INTRODUCTION

Sarcasm detection in social media has become increasingly important [1], driven by the prevalence of sarcastic language usage and its wide-ranging applications. This challenge is particularly pronounced in Pidgin tweets, where English and Pidgin languages intertwine [2], along with local cultural references, creating a unique linguistic landscape. These tweets exhibit complexity by blending English vocabulary with Pidgin grammar, resulting in non-standard syntax and vocabulary choices. Furthermore, informal language usage, abbreviations, and slang are prevalent in social media, including Pidgin tweets, which also introduce unique abbreviations and phonetic representations, adding to the complexity of sarcasm detection. The ability to accurately detect Sarcasm has numerous practical applications, including social media monitoring, consumer feedback analysis, and sentiment analysis [3]. Therefore, machine-learning techniques have emerged as practical tools for automatic sarcasm detection. Various languages across the globe have witnessed the development of machine learning models tailored for detecting sarcasm, enriching communication analysis across diverse linguistic landscapes. In English, researchers have pioneered advanced models utilizing sophisticated natural language processing techniques to discern sarcastic utterances amidst regular text [4]. Similarly, in Hindi, the intricate interplay of the language's extensive vocabulary and nuanced expressions has prompted investigations into sarcasm detection models, facilitating more accurate sentiment analysis in social media and textual communication [5].

Sarcasm in social media presents unique challenges due to the absence of non-verbal cues and reliance on textual content. The brevity and informality of social media platforms further complicate the detection and interpretation of Sarcasm. However, researchers have made significant strides in understanding and detecting Sarcasm in social media data. Introducing an interpretable Hybrid Neural Network structure, Misra and Arora [6] delve into the factors contributing to sentence sarcasm. Their quantitative experiments showcase a significant enhancement in classification accuracy by approximately 5% compared to a robust baseline. Specifically, the baseline achieved 84.88%, whereas the proposed method achieved 89.70%.

In their study, Kumar and Garg [7] aimed to classify Twitter data from the SemEval 2015 Task 11 benchmark and approximately 20k posts from Reddit into sarcastic or non-sarcastic categories using three predictive learning models. The first model employed conventional TF-IDF weighting with Multinomial Naïve Bayes, Gradient Boosting, and Random Forest classifiers, employing Ensemble Voting for output generation. The second model integrated semantic and pragmatic features along with the top-200 TF-IDF features, assessed using five baseline classifiers. The final model utilized deep learning, specifically LSTM and Bi-directional LSTM, with GloVe for semantic word embeddings. Empirical studies compared the three models for sarcasm classification using training and test set metrics, with the Bi-directional LSTM model achieving the highest accuracy rates of 86.32% and 82.91% for the Twitter and Reddit datasets, respectively.

In their work, Xiong et al. [8] proposed an innovative self-matching network equipped with a modified co-attention mechanism to tackle sentence incongruity. They also integrated a bi-directional LSTM encoder to capitalize on the compositional structure of sentences. The results of their experiments provide compelling evidence that the suggested model outperforms the majority of established baseline methods.

In their study, Potamias et al. [9] introduced a neural network approach that extends a pre-trained transformer-based network, enhancing it by incorporating a recurrent neural network. This model's effectiveness was evaluated on four datasets and compared against alternative methods. The results revealed that the proposed approach significantly outperformed all other methods and previously published studies, achieving accuracies of 82.00%, 79.00%, 91.00%, and 82.00% on the SemEval-2018 dataset, Reddit SARC2.0 Politics dataset, Sarcastic Rillof's dataset, and SemEval-2015 dataset, respectively.

In their investigation, Aggarwal et al. [10] delved into various deep learning architectures to detect sarcasm in tweets that blend Hindi and English languages. Their approach utilized bilingual word embeddings acquired from both FastText and Word2Vec methodologies. The deep learning models outperformed all existing methods, with the attention-based Bidirectional LSTMs achieving the highest accuracy at an impressive 78.49%.

In their work, Pawar and Bhingarkar [11] presented an approach for sarcasm detection utilizing patterns and leveraging Twitter data. They introduced four distinct sets of features enriched with specific sarcasm indicators. These feature sets underwent thorough examination and assessment for their influence on tweet classification using SVM, KNN, and random forest classifiers. The results indicated that the Random Forest classifier yielded the most favorable outcomes compared to SVM and KNN. In their research, Sundararajan and Palanisamy [3] introduced a feature selection method based on ensemble techniques to identify the optimal feature set for sarcasm detection in tweets. This selected feature set was then utilized to determine whether a tweet conveys sarcasm. Additionally, they developed a multi-rule-based approach to classify sarcasm into four different classes: polite sarcasm, rude sarcasm, raging sarcasm, and deadpan sarcasm. The ensemble feature selection algorithm for sarcasm detection demonstrated an overall accuracy of approximately 92.70%, while the multi-rule approach for identifying sarcasm class achieved accuracies of 95.98%, 96.20%, 99.79%, and 86.61% for each class, respectively.

In their study, Akula and Garibay [12] developed a model utilizing multi-head self-attention and gated recurrent units. They evaluated the efficiency and interpretability of this approach on various datasets. The model achieved impressive precision, recall, f1-score, and AUC of 97.90%, 99.60%, 98.70%, and 99.60%, respectively, on the Twitter dataset. Additionally, it attained accuracies of 81.0% and 80.0% on the Main and Political subsets of the Reddit SARC 2.0 dataset, respectively.

In their study, Goel et al. [13] utilized a combination of various neural techniques in an ensemble model to detect sarcasm on the internet. This ensemble model achieved an accuracy rate of approximately 96.00% for the News Headlines dataset and 73.00% for the Reddit dataset. Among the ensemble models proposed, the Weighted Average Ensemble exhibited the highest accuracy, reaching approximately 99.00% and 82.00% for the two datasets, respectively.

In their recent work, Sharma et al. [1] introduced a novel ensemble strategy integrating fuzzy evolutionary logic into its top layer. This fuzzy layer utilizes weights assigned to the probabilities generated by three models to effectively categorize objects. To validate this proposed model, experiments were conducted using various social media datasets, including the Headlines dataset, the "Self-Annotated Reddit Corpus" (SARC), and the Twitter app dataset. Impressively, this approach achieved accuracy rates of 90.81%, 85.38%, and 86.80%, respectively, demonstrating its effectiveness across different social media platforms.

In Pandey and Singh's study [14], they propose a novel model called BERT-LSTM, which combines BERT with LSTM. Using a pre-trained BERT model, they generate embeddings for a code-mixed dataset. These embeddings are then fed into an LSTM network with a single layer to classify sentences as sarcastic or non-sarcastic. Experimental findings demonstrate that the BERT-LSTM model surpasses other models on the code-mixed dataset, achieving a substantial improvement of up to 6% in F1-score for sarcasm detection.

In the work of Yue et al. [15], the Knowledge Fusion Network (KnowleNet) for sarcasm detection, combining prior knowledge and semantic similarity between images and text, was introduced. They employed novel word-level and sample-level cross-modal semantic similarity detection techniques, leveraging conceptual knowledge. Additionally, they utilized a contrastive learning approach with triplet loss to optimize sample features. Through experimentation, KnowleNet achieved state-of-the-art results, demonstrating superior performance in multimodal sarcasm detection with an accuracy rate of 88.87%.

Despite the availability of sarcasm detection models for English, the scarcity of annotated data for Pidgin sarcasm detection poses a significant hurdle. This scarcity impedes the training and evaluation of machine learning models tailored to this linguistic context. Traditional rule-based approaches to sarcasm detection struggle with the complexity and variability of language in Pidgin tweets. Thus, there is a pressing need to develop an accurate model for detecting sarcasm in Pidgin tweets to enhance sentiment analysis and social media understanding within the specific linguistic and cultural context of Pidgin speakers. This study addresses this gap by proposing a model designed for effective sarcasm detection in Pidgin tweets, focusing on the unique linguistic and cultural dynamics of this communication medium. Our objectives include developing a model that can navigate the linguistic complexity of Pidgin, adapt to informal language conventions, and successfully discern sarcasm in this distinctive context.

The remaining section of this paper is structured as follows: The technique used in this study is covered in Section 2, and the findings and results are covered in Section 3. This study wraps up in Section 4.

## 2. METHODOLOGY

In this study, we aimed to develop a model for sarcasm detection in pidgin tweets. Four machine learning models were developed on a sizable dataset of preprocessed pidgin tweets. The approach taken is illustrated in Fig. 1 below.

The framework comprises the following components:

i. Data Wrangling and Cleaning: This encompasses preparing raw data gathered from pidgin tweets for analysis. Tasks include removing duplicates, handling missing values, correcting errors, and standardizing formats to ensure dataset consistency and accuracy.
ii. Data Annotation: This entails labeling pidgin tweets as either sarcastic or non-sarcastic. Through manual or automated processes, each tweet receives a class label, facilitating supervised learning algorithms to identify patterns and make predictions.
iii. Feature Extraction: This involves selecting and converting pertinent information from pidgin tweets into numerical or categorical features suitable for machine learning models. Features may comprise linguistic cues, sentiment analysis scores, word embeddings, or other contextual data aiding in discerning sarcastic from non-sarcastic tweets.

Evaluation: This requires assessing the performance of sarcasm detection models using metrics like accuracy, precision, recall, F1-score, and area under the receiver operating characteristic curve (AUC-ROC). This phase gauges the models' effectiveness in correctly identifying sarcastic tweets while minimizing false positives and false negatives.

These stages are crucial for constructing robust sarcasm detection models for pidgin tweets, empowering researchers to scrutinize and comprehend the subtleties of sarcasm within this specific linguistic framework.

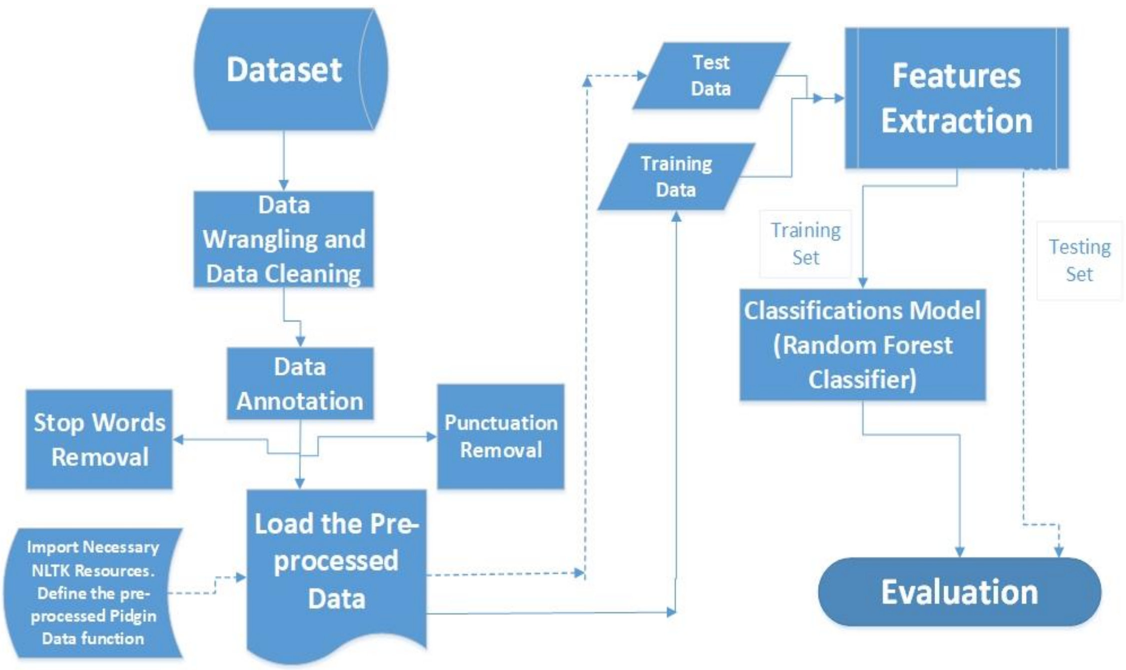


**Fig. 1. Proposed framework**

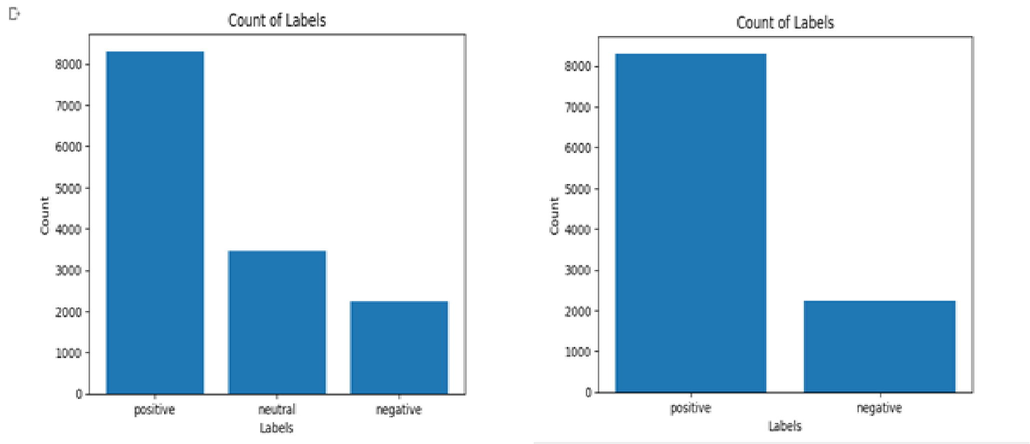


**Fig. 2. Image before and after neutral label removal**

| | A | B |
|---|---|---|
| 1 | Clean_Content | Label |
| 2 | we don realize hin importance after that city loss. maguire sabi defend. | 1 |
| 3 | don come give una free awoof ...awoof!!! o .. stay glued on top all of our social media platforms as informate dey roll in | 1 |
| 4 | people dey talk about iniesta, paul scholes, puyol, drogba, chioma and that agbero club chelsea. but i dey think about bruno fernandes and how e go improve ou | 1 |
| 5 | election latest | 1 |
| 6 | ht: 1 v leicester city 0. thank god for mason wey dey go mountain of fire. he use vex knack confam shot wey tear leicester city net. | 1 |
| 7 | sabi pikin pellegrini epp italo close shop yestaday as hin score for ï¸ | 1 |
| 8 | which full-back is better?  for luke shaw like for roberto carlos | 1 |
| 9 | we dey wish  hapi baidae as him turn 25 years old. | 1 |
| 10 | on dis day for 1927, tanda as official club for rome and we kolobi di city name, colours and symbol ï¸ so, dat mâ€¦ | 1 |
| 11 | we dey live dey torchlight how pipo and  dey prepare for governorship election for different states for nigeria, you fit join us for hia  make you sef chook mouth | 1 |
| 12 | 73' batshuayi don enta for fada abraham. | 1 |
| 13 | the red devils dey brim with confidence ahead of dem clash today. shey e go become their third victory in a row for ? get involved by predicting the outcome of t | 1 |
| 14 | niels hÃ¶gel: di german ex-nurse go serve life sentence afta im kill 85 patients | 0 |
| 15 | 55' jorginho don collect him mtn as usual. so far mount, rudiger and kante don collect mtn for first half. | 1 |
| 16 | frank lampard yarn say belle sweet am with the way the team play for second half but con say small small things na him make us we no dey win, jus see as we tal | 1 |
| 17 | last time wey sassuolo com olimpico nicolo zaniolo let all man know as e dey go! na 3 days remain to ... | 1 |
| 18 | atewa forest ðŸ‡¬ðŸ‡ don gain international recognition but for de wrong reason.  na oscar award-winning actor den environmental activist. | 0 |
| 19 | 90' batshuayi for score dia but he no score as valencia goalie gada the ball. | 1 |
| 20 | ed woodward: â€œna true say we no sign centre-back for 2018 summer as jose and the people wey dey recruit, bin get problem. but las las na me wey tell jose sa | 1 |
| 21 | oga frank lampard don win barclays premier league manager of the month (october) award. supa frank! | 1 |
| 22 | welkom back! | 1 |
| 23 | david de gea don save us! e for be 2-2. | 1 |
| 24 | i hail o ðŸ‡³ðŸ‡¬ two time heavy weight boxing champion of di world anthony joshua dey hail. im share dis foto im snap with some young children for naija . | 1 |
| 25 | nigeria d'tigress beat senegal to claim afrobasket 2019 champions | 1 |
| 26 | na only zakzaky shia followers get ban for ashura day procession | 0 |
| 27 | â€œyes na foul on de gea but wetin pogba dey do when dem cross that ball?.â€ | 1 |
| 28 | david silva!!!!!!! but de gea catch am. | 1 |
| 29 | important goal!! | 1 |

**Fig. 3. Screenshot of encoded label data**

The dataset used is from semantic enrichment of Nigerian pidgin English for contextual sentiment classification in the work of Oyewusi et al. [16]. There are 14,000 tweets in the data set.

To prepare the data for sarcasm detection, we performed various pre-processing steps:

**Removal of neutral labels:** As we focused on detecting Sarcasm, we filtered out rows with neutral sentiments. The Fig. 2 above shows distribution of labels before and after removal of neutral labels

**Encoding of labels:** We encoded the labels to facilitate training and evaluation of the sarcasm detection model. Sarcastic statements were labelled 1, while non-sarcastic statements were assigned labelled 0 as shown in Fig. 3 above.

i. **Text pre-processing:** To optimize the performance of the sarcasm detection model for Nigerian Pidgin text, we applied specific pre-processing steps:

- **Conversion to lowercase**: All text was converted to lowercase to ensure consistency.

- **Removal of punctuation and special characters**: Punctuation marks and special characters were removed from the text.
- **Tokenization**: The text was tokenized into individual words using the NLTK library.
- **Lemmatization**: Lemmatization was applied to convert words to their base forms, reducing inflectional forms to a joint base.

ii. **Feature Extraction**: The TfidfVectorizer was utilized to capture the importance of words in each document within the pre-processed data. TF-IDF values provided a nuanced representation, emphasizing words that are both prevalent within individual documents and distinctive across the entire corpus. This approach allowed the research to consider the significance of words in context while considering their broader distribution.

The CountVectorizer was employed to focus solely on the occurrence frequency of words within each document. By creating a matrix where rows represent documents and columns represent unique words, this method provided a straightforward representation of word counts. The resulting matrix encapsulated the document-specific distribution of words, offering a basic yet insightful view of the textual data. The CountVectorizer's simplicity and efficiency made it suitable for capturing word frequency patterns across the pre-processed dataset.

Both the TfidfVectorizer and CountVectorizer played pivotal roles in the feature extraction process. The TfidfVectorizer's ability to consider word importance in relation to the entire corpus added depth to the features, while the CountVectorizer's focus on word occurrences offered a more straightforward representation. These feature extraction techniques contributed to the research's ability to effectively prepare the pre-processed data for subsequent machine learning analyses, enhancing the project's capacity to uncover patterns and insights within the textual data.

iii. **Handling Data Imbalance using SMOTE:** Class imbalance can significantly impact model performance, particularly for binary classification tasks. To address this issue, the Synthetic Minority Over-sampling Technique (SMOTE) is employed. SMOTE generates artificial instances of the minority class by interpolating between existing instances, thus balancing the class distribution. The count of resampled labels, both 'negative' and 'positive', is visualized to demonstrate the improved class balance achieved through SMOTE. During the preprocessing, the data was found to be imbalanced, which could affect the model's result. After applying SMOTE, the dataset is then balanced as shown in Fig. 4.

iv. **Model Training and Evaluation:** The dataset was divided into training and testing sets. Four distinct machine learning algorithms: Logistic Regression, Random Forest Classifier, XGBoost and Vanilla ANN, are trained on the resampled dataset, each offering unique techniques and complexities to tackle the sarcasm detection task effectively. Metrics, including accuracy, precision, recall, and F1-score, were used to assess the model on the testing set after it had been trained using the training set.

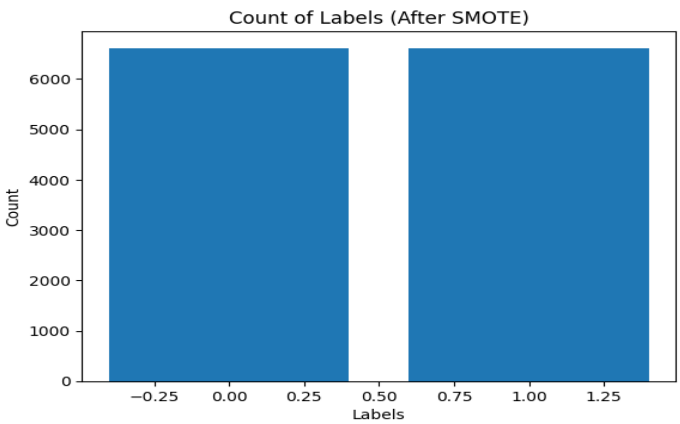


**Fig. 4. Image of balanced data after applying SMOTE**

## 3. RESULTS AND DISCUSSION

This section is dedicated to examining the deployment and assessment of the four classification models employed in this work. The results of the models are discussed below.

i. Fig. 5 below presents an evaluation of the Receiver Operating Characteristic (ROC) curves and Area under the Curve (AUC) scores for the four machine learning models. The ROC curves and AUC scores provide valuable insights into the models' ability to distinguish between positive and negative classes.

The XGBoost model exhibits the highest AUC value of 0.89, indicating that it can discriminate between positive and negative instances. This suggests that the XGBoost model's predictions are generally well-calibrated and can effectively rank cases based on their likelihood of belonging to the positive class.

i. Figs. 6, 7, 8 and 9 show confusion matrix for Logistics Regression, Random Forest, XGBoost and Vanilla Models respectively.
ii. The Table 1 shows the evaluation of accuracy, precision, recall and F1-Score of Logistics Regression, Random Forest, XGBoost and Vanilla Models. It analyses the performance of four machine learning models providing insights into their effectiveness for the given task.

The XGBoost model has the highest accuracy of 85.78%, demonstrating its ability to make accurate predictions. It also achieves a commendable recall of 94.44%, suggesting it effectively captures true positives. The F1-score of 91.41% indicates a strong balance between precision and recall, highlighting its overall performance. Logistic Regression and Random Forest models exhibit identical performance across all metrics, indicating that they might be simplistically modelling the data or potentially have similar underlying characteristics in their predictions. The Vanilla ANN achieved an accuracy of 83.18%, a precision of 92.38%, and a recall of 86.09%. While it does not surpass the top-performing XGBoost model, it demonstrates competitive results across the board.

Comparing these results to the results of Šandor and Babac [17] both aimed to develop models for detecting sarcasm using advanced techniques. In the study of Šandor and Babac [17], different classification models were employed, including logistic regression, ridge regression, support vector machine (SVM), and linear support vector classifier. These models were evaluated using performance metrics like accuracy, F1 score, recall, and precision, with cross-validation applied to training data. Among the traditional machine learning models, SVC showed promising performance across all metrics, not exceeding 71%. Deep learning models were also explored. Two neural network architectures, one with five layers and the other with six layers, were tested. Despite their simplicity, these models show competitive performance. Finally, a BERT-based model, fine-tuned for sarcasm detection, achieves the highest accuracy of 73.1%, outperforming the other models in recognizing sarcastic statements.

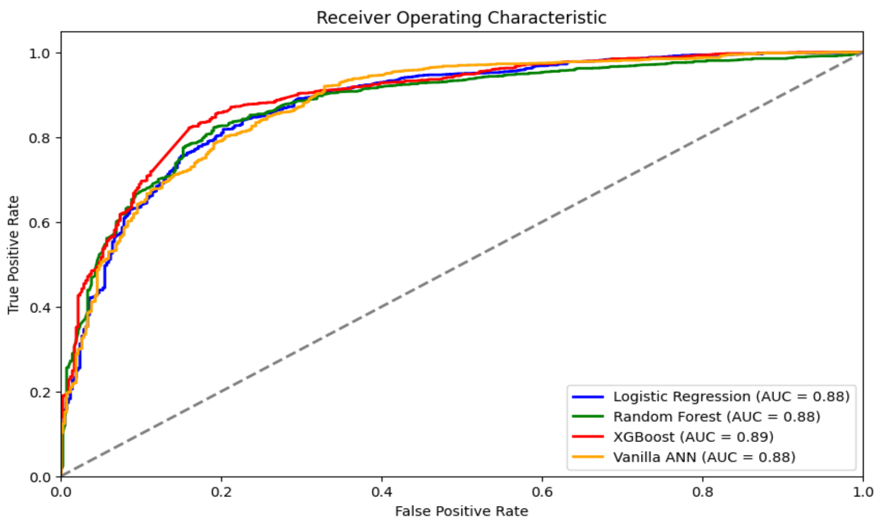


**Fig. 5. ROC Curve**

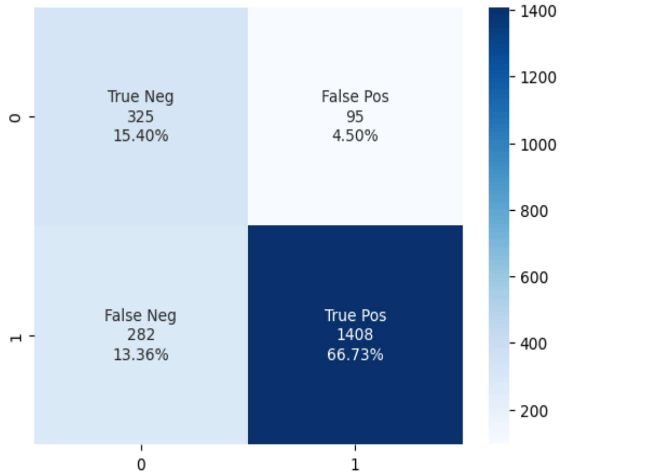


**Fig. 6. Logistic regression confusion matrix**

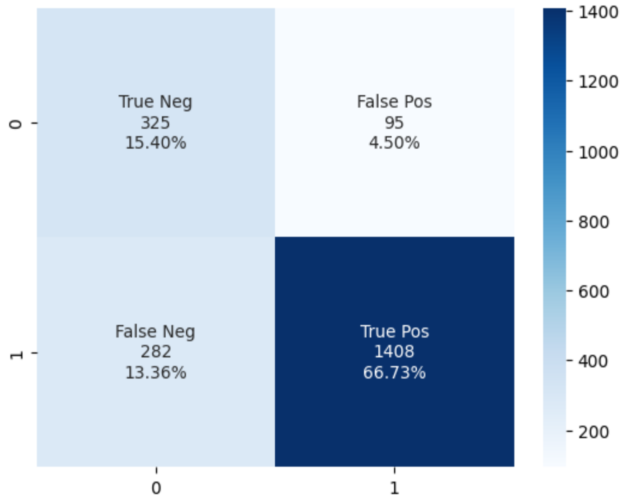


**Fig. 7. Random forest confusion matrix**

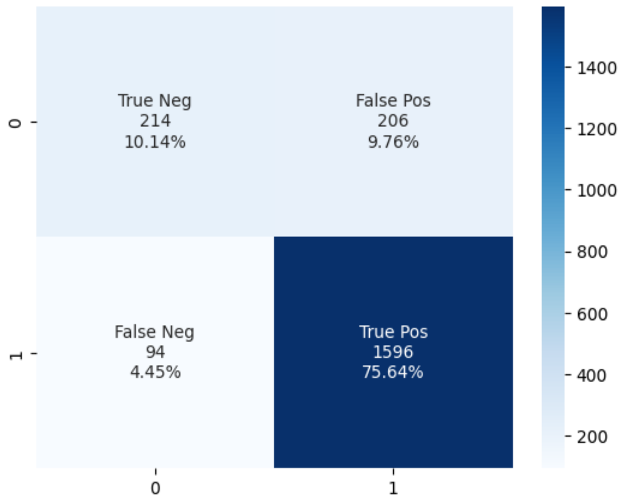


**Fig. 8. XGBoost confusion matrix**

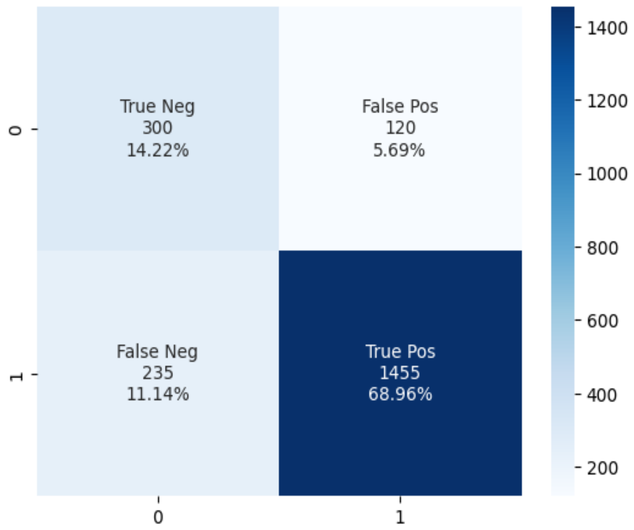


**Fig. 9. Vanilla ANN confusion matrix**

**Table 1. Classification Model Distributions**

| Model | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| **Logistic Regression** | 82.23 | 93.68 | 83.31 | 88.19 |
| **Random Forest** | 82.13 | 93.68 | 83.31 | 88.19 |
| **XGBoost** | 85.78 | 88.57 | 94.44 | 91.41 |
| **Vanilla ANN** | 83.18 | 92.38 | 86.09 | 89.13 |

Each study demonstrated strengths in its approach and provided valuable insights into the development of effective models for sarcasm detection, but our models achieved better results.

## 4. CONCLUSION AND FUTURE WORK

A valuable field of research enables us to comprehend better the complexities and subtleties of language in the Nigerian context: sarcasm identification in Nigerian Pidgin text data. This research gave a comprehensive pipeline for sarcasm detection in Nigerian Pidgin text data. This pipeline included data preparation, exploratory data analysis, and text preprocessing specific to Nigerian Pidgin, model training, evaluation, and prediction of new data. Four machine learning models, Logistic Regression, Random Forest, XGBoost, and Vanilla Artificial Neural Network (ANN), were evaluated. The XGBoost model shows the highest overall performance among the evaluated models, particularly excelling in accuracy, recall, and F1-score. The choice of deployment model would depend on specific trade-offs and priorities, such as the need for high accuracy or balanced precision and recall. Further model tuning and feature engineering could enhance the performance of these models. Although promising, there is still an opportunity for development and additional study. Deeper insights into the sentiment and linguistic diversity of Nigerian Pidgin writing can be gotten by improving the sarcasm recognition capabilities and opening up possibilities for applications in social media analysis, sentiment monitoring, and cultural studies.

## COMPETING INTERESTS

Authors have declared that no competing interests exist.

## REFERENCES

1. Sharma DK, Singh B, Agarwal S, Kim H, Sharma R. Sarcasm detection over social media platforms using hybrid auto-encoder-based model. Electronics (Switzerland). 2022;11(18).
2. Ghosh, Veale T. Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal. EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings. 2017;482–491.

3. Sundararajan K, Palanisamy A. Multi-rule based ensemble feature selection model for sarcasm type detection in Twitter. Computational Intelligence and Neuroscience; 2020.
4. Techentin C, Cann DR, Lupton M, Phung D. Sarcasm detection in native english and english as a second language speakers. Canadian Journal of Experimental Psychology. 2021;75(2):133–138.
5. Jain D, Kumar A, Garg G. Sarcasm detection in mash-up language using soft-attention based bi-directional LSTM and feature-rich CNN. Applied Soft Computing Journal. 2020;91.
6. Misra R, Arora P. Sarcasm detection using hybrid neural network; 2019. Available:https://doi.org/10.13140/RG.2.2.32427.39204
7. Kumar Akshi, Garg Geetanjali. Empirical study of shallow and deep learning models for sarcasm detection using context in benchmark datasets. Journal of Ambient Intelligence and Humanized Computing. 2019;14. DOI: 10.1007/s12652-019-01419-7
8. Xiong T, Zhang P, Zhu H, Yang Y. Sarcasm detection with self-matching networks and low-rank bilinear pooling. The Web Conference 2019 - Proceedings of the World Wide Web Conference, WWW. 2019;2115–2124.
9. Alexandros Potamias R, Siolas G, Stafylopatis A-G. A transformer-based approach to Irony and Sarcasm detection. ArXiv, arXiv. 2019;1911:10401.
10. Agrawal A, An A. Affective representations for sarcasm detection. 41st International ACM SIGIR conference on research and development in information retrieval, SIGIR. 2018;2018:1029–1032.
11. Pawar N, Bhingarkar S. Machine learning based sarcasm detection on twitter data. 2020;957–961.
12. Akula R, Garibay I. Interpretable multi-head self-attention architecture for sarcasm detection in social media. Entropy. 2021;23(4):394.
13. Goel P, Jain R, Nayyar A, Singhal S, Srivastava M. Sarcasm detection using deep learning and ensemble learning. Multimedia Tools and Applications. 2022; 81(30):43229–43252.
14. Pandey Rajnish, Singh Jyoti. BERT-LSTM model for sarcasm detection in code-mixed social media post. Journal of Intelligent Information Systems. 2022;60:1-20. DOI: 10.1007/s10844-022-00755-z
15. Yue Tan, Mao Rui, Wang Heng, Hu Zonghai, Cambria Erik. KnowleNet: Knowledge fusion network for multimodal sarcasm detection. Information Fusion. 2023;100:101921. DOI: 10.1016/j.inffus.2023.101921
16. Wuraola Oyewusi, Olubayo Adekanmbi and Olalekan Akinsande. Semantic enrichment of nigerian pidgin english for contextual sentiment classification; 2020.
17. Šandor D, Babac MB. Sarcasm detection in online comments using machine learning. Information Discovery and Delivery, (ahead-of-print); 2023.




*Peer-review history:*
*The peer review history for this paper can be accessed here:*
*https://www.sdiarticle5.com/review-history/113437*