Original Research Article

Sarcasm Detection in Pidgin Tweets

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.

Keywords: sentiment analysis, machine learning, sarcasm, text data, pidgin

1. INTRODUCTION

Sarcasm detection in social media has become increasingly important (Sharma et al., 2022), 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 (Ghosh and Veale, 2016), 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(Sundararajan and Palanisamy, 2020). 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 (Techentin et al., 2021). 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 (Jain et al., 2020).

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.Misra and Arora (2019) introduced an interpretable Hybrid Neural Network structure that offers a deeper understanding of the factors contributing to sentence sarcasm. The

quantitative experiments demonstrated that the proposed model enhances classification accuracy by approximately 5% compared to a robust baseline with the baseline having 84.88% and the proposed method 89.70%.

Xiong et al. (2019) introduced an innovative self-matching network that incorporates a modified co-attention mechanism to address sentence incongruity. Additionally, a bi-directional LSTM encoder was integrated to leverage the compositional structure of sentences. The outcomes of these experiments offer strong proof that the suggested model outperformed the majority of established baseline methods.
Potamias et al. (2019) introduced a neural network approach that extends a pre-trained transformer-based network, further strengthening it by incorporating a recurrent neural network. The efficiency of this model was evaluated on four datasets and compared to other methods. The outcomes indicated that the proposed approach surpassed all other methods and previously published studies by a significant margin, achieving accuracy of 82.00%, 79.00%, 91.00%, and 82.00% on SemEval-2018 dataset, Reddit SARC2.0 Politics dataset, Sarcastic Rillof's dataset, and SemEval-2015 dataset respectively.
Aggarwal et al (2020) explored different deep learning architectures for detecting sarcasm in tweets that mix Hindi and English languages. The approach leveraged bilingual word embeddings obtained from both FastText and Word2Vec methodologies. The deep learning models surpassed the performance of all existing methods. Among these models, the attention-based Bidirectional LSTMs achieved the highest accuracy, reaching an impressive 78.49%.
Pawar and Bhingarkar (2020) introduced an approach based on patterns, utilizing Twitter data for sarcasm detection. Four distinct sets of features, rich in specific sarcasm indicators, were presented. These feature sets are thoroughly examined and assessed for their impact on tweet classification using SVM, KNN and random forest classifiers. The Random Forest classifier achieved the best outcomes compared to SVM and KNN.Sundararajan and Palanisamy (2020) introduced a feature selection method based on ensemble techniques to discern the best feature set for detecting sarcasm in tweets. This selected feature set was then utilized to determine whether a tweet conveys sarcasm. Subsequently, a multi-rule-based approach was developed to classify sarcasm into four different class: 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.
Akula and Garibay (2021) created a model by utilizing multi-head self-attention and gated recurrent units. The approach's efficiency and interpretability are evaluated on different datasets. The model achieved precision, recall, f1-score and AUC of 97.90%, 99.60%, 98.70% and 99.60% respectively on twitter dataset, and 81.0% and 80.0% accuracy on Main and Political subsets of Reddit SARC 2.0 dataset, respectively.
Goel et al. (2022) employed various neural techniques combined in an ensemble model to detect sarcasm on the internet. It 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 demonstrates the highest accuracy, reaching around 99.00% and 82.00% for the two datasets, respectively.
Sharma et al (2022) introduced an innovative ensemble strategy that incorporates fuzzy evolutionary logic in its top layer. The fuzzy layer utilizes weights assigned to the probabilities produced by the three models to categorize objects effectively. To validate this proposed model, experiments were conducted using several 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.

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 figure 1 below.

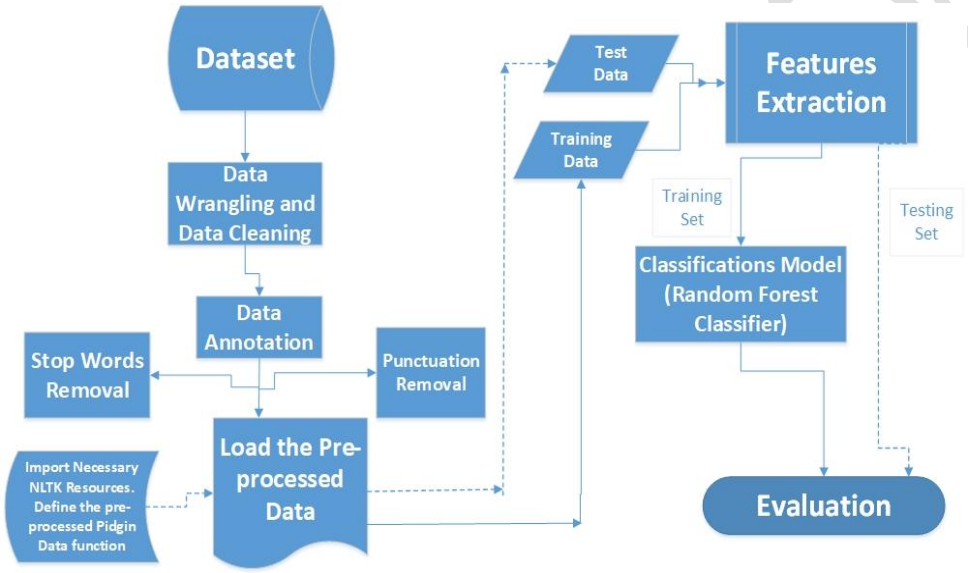

Fig. 1. System framework

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

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

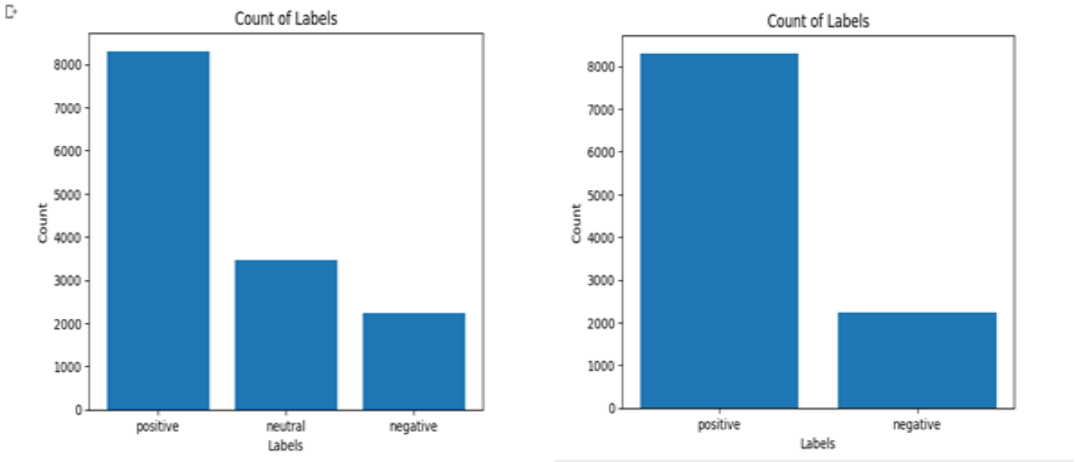

Fig. 2. Image before and after neutral label removal

ii. **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 figure 3 below.

	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 ī¸	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 ī¸ 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 tak	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€œeyes 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

iii. 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.

iv. 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.

v. 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.

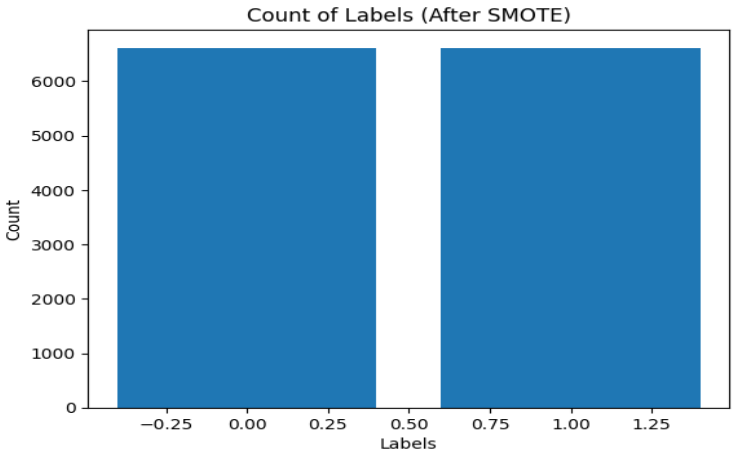

Fig. 4. Image of balanced data after applying SMOTE

vi. 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.

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. Figure 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.

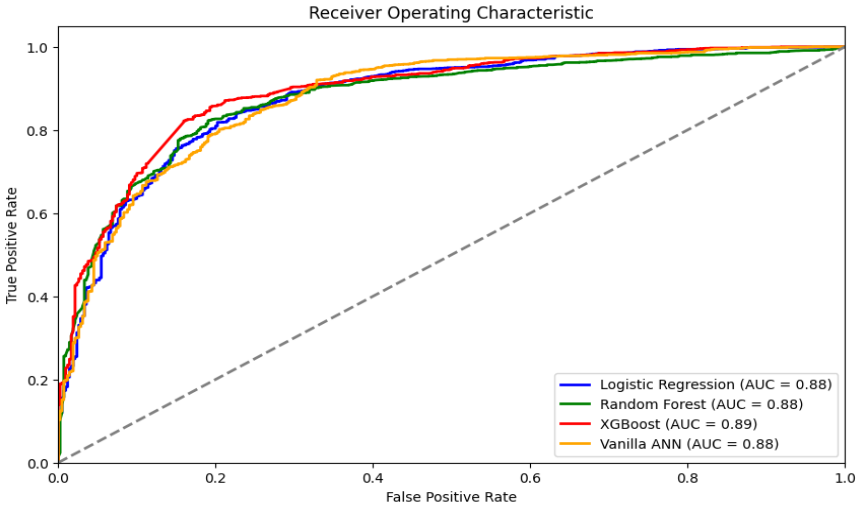

Figure 5: ROC Curve

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.

ii. Figure 6, 7, 8 and 9 show confusion matrix for Logistics Regression, Random Forest, XGBoost and Vanilla Models respectively.

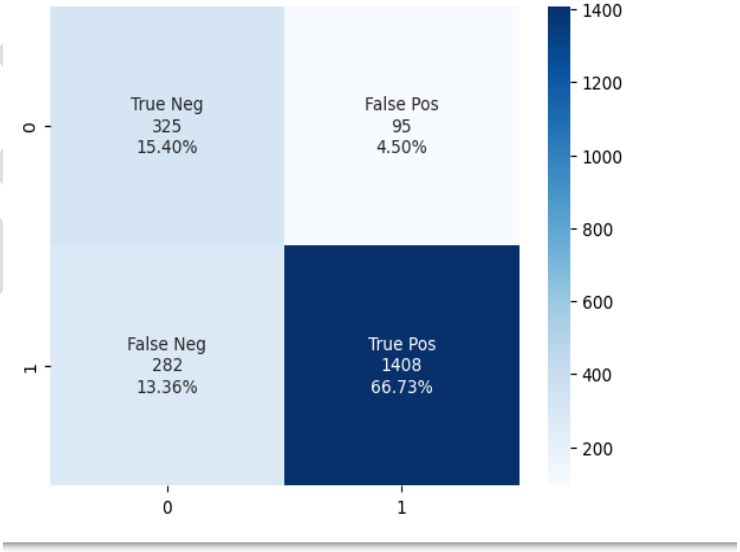

Fig. 6. Logistic Regression confusion matrix

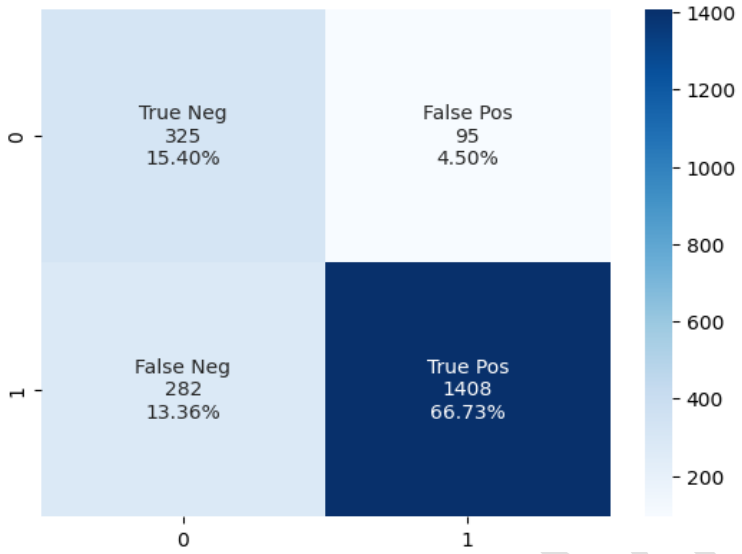

Fig. 7. Random forest confusion matrix

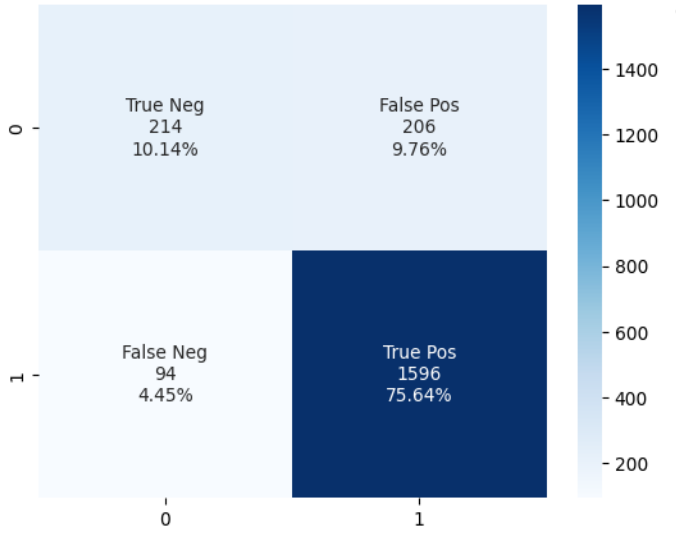

Fig. 8.XGBoost confusion matrix

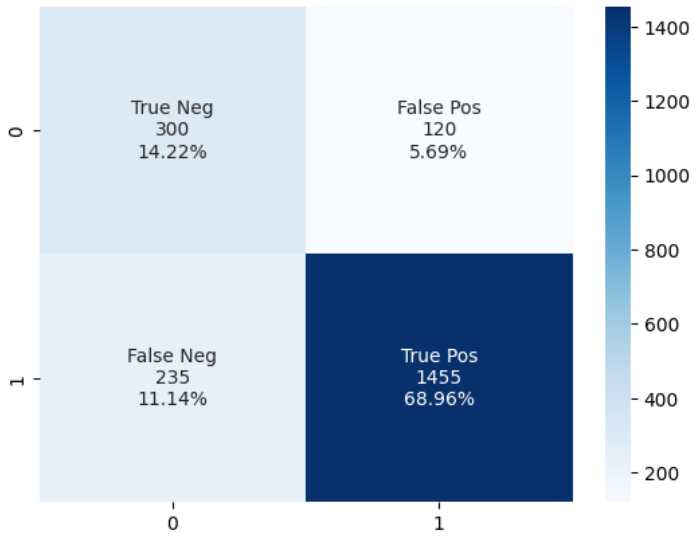

Fig. 9. Vanilla ANN confusion matrix

iii. The table 1 below 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.

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

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.

4. CONCLUSION

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.

REFERENCES

Agrawal, A., and An, A. 2018. Affective representations for sarcasm detection. 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2018, 1029–1032.

Akula, R., and Garibay, I. 2021. Interpretable Multi-Head Self-Attention Architecture for Sarcasm Detection in Social Media. Entropy 2021, Vol. 23, Page 394, 23(4), 394.

Alexandros Potamias, R., Siolas, G., and Stafylopatis, A.-G. 2019. A Transformer-based approach to Irony and Sarcasm detection. ArXiv, arXiv:1911.10401.

Ghosh, and Veale, T. 2017. Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal. EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings, 482–491.

Goel, P., Jain, R., Nayyar, A., Singhal, S., and Srivastava, M. 2022. Sarcasm detection using deep learning and ensemble learning. Multimedia Tools and Applications, 81(30), 43229–43252.

Jain, D., Kumar, A., and Garg, G. 2020. Sarcasm detection in mash-up language using soft-attention based bi-directional LSTM and feature-rich CNN. Applied Soft Computing Journal, 91.

Misra, R., and Arora, P. 2019. Sarcasm Detection using Hybrid Neural Network. https://doi.org/10.13140/RG.2.2.32427.39204

Pawar, N., and Bhingarkar, S. 2020. Machine Learning based Sarcasm Detection on Twitter Data. 957–961.

Sharma, D. K., Singh, B., Agarwal, S., Kim, H., and Sharma, R. 2022. Sarcasm Detection over Social Media Platforms Using Hybrid Auto-Encoder-Based Model. Electronics (Switzerland), 11(18).

Sundararajan, K., and Palanisamy, A. 2020. Multi-rule based ensemble feature selection model for sarcasm type detection in Twitter. Computational Intelligence and Neuroscience, 2020.

Techentin, C., Cann, D. R., Lupton, M., and Phung, D. 2021. Sarcasm detection in native English and English as a second language speakers. Canadian Journal of Experimental Psychology, 75(2), 133–138.

Xiong, T., Zhang, P., Zhu, H., and Yang, Y. 2019. 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.