Multi-Pipeline Approach for Sentiment Analysis of West-African Pidgin

Segun Aina, Omolara A. Ogungbe, Seun Ayeni, Aderonke R. Lawal, Oluwatoyin H. Odukoya, Joshua Etim

Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria

ABSTRACT

This paper presents a multi-pipeline approach to sentiment analysis, with the aim of improving both the accuracy and relevance of the results. Sentiment analysis of West African Pidgin has historically been fragmented, often involving the training of new models with pidgin data, and typically focusing on general sentiment polarity. This study seeks to address these gaps by adopting a holistic, multi-pipeline system for the comprehensive analysis of sentiment polarity in Pidgin text. A subject classifier was developed using the Logistic Regression algorithm to predict the relevance of a body of text to the specific subject matter. Data was collected from Twitter and processed into tokens, which were then used for training and evaluation. This enabled the model to handle a wide range of informal and context-specific words commonly found in Pidgin. For the sentiment analysis itself, a cross-lingual model, RoBERTa (XLM-R), was fine-tuned and expanded through transfer learning using the AfriBERTa model, developed by Ogueji et. al (2021). This fine-tuned model achieved an average F1-score of 74.5 over five runs, demonstrating its effectiveness in sentiment classification. The subject classifier also performed efficiently, achieving an accuracy of 0.81 in identifying relevant text. This multi-pipeline system demonstrates significant promise in enhancing sentiment analysis for Pidgin text, being the first to combine subject classification and cross-lingual sentiment analysis techniques. The results show that the proposed approach can be a valuable tool in natural language processing for underrepresented languages such as West African Pidgin.

KEYWORDS

Pidgin
Multi-Pipeline
West-African
Sentiment

1. INTRODUCTION

According to Pang and Lee (2008), sentiment analysis encompasses the computational examination of individuals' opinions, sentiments, attitudes, and emotions as conveyed through text. In the field of machine learning, sentiment analysis is classified under Natural Language Processing (NLP), and falls under the broader description of text mining which refers to the process of extracting useful information, knowledge, or patterns from large volumes of textual data using computational and statistical techniques (Feldman and Sanger, 2007; Berry and Kogan, 2010).

Across West Africa, over 75 million people speak the Pidgin language and represent a significant percentage of online discussions about products, companies and policies (Ihemere, 2006). African companies such as Jumia, Takealot and Konga has amassed large quantities of data in comments and reviews over the years which have a lot of potential for sentiment analysis. Voluminous research has been conducted using data from aforementioned companies to analyze the sentiment of customers with encouraging results such as the work of Olagunju *et. al* which resulted in an F1-score of 86.7%. However, the models used so far relied on the English language and could only infer sentiment from English words and phrases.

Target-dependent sentiment analysis has been a focal point in decision-driving sentiment analysis. It refers to the task of analyzing sentiment or opinion expressed towards a specific target or aspect within a given text (Jiang et. al., 2011). This holds significance owing to the fact that for automatic sentiment analysis to drive critical decisions, it must be targeted to the subject, not just to general polarity. A common approach to achieve target-dependent sentiment analysis is to use supervised learning algorithms such as Support Vector Machines (SVM), Naive Bayes, or Random Forests (Pang and Lee, 2008).

State-of-the-art approaches on English data make use of pre-trained language models (PLMs) and its variants (Muhammad *et al.*, 2022). For pidgin-text sentiment analysis, a transfer learning process takes place to introduce new data to the language models to leverage their superior performance. The XLM-R model was used to develop a pidgin analysis model, which conforms with the Hugging Face implementation. At the center of these approaches, and this study is the Transformer architecture. It is the dominant architecture for natural language processing (NLP), surpassing convolutional and recurrent neural networks for NLP tasks (Wolf *et al.*, 2020). The Transformer is based solely on attention mechanisms, to draw global dependencies between input and output (Vaswani *et al.*, 2017). The Transformers library (Wolf *et al.*, 2020) is dedicated to supporting the Transformer architecture and facilitating the distribution of pre-trained models. The AfriBERTa model is trained on the cross-lingual model XLM-R on pidgin data to develop a Transformer-architecture, sentiment analysis model leveraging on the high performance of the base model.

Text classification is a key technology for gaining insights from text and organizing that information (Shah *et al.*, 2020). Subject analysis, classification, and topic prediction are typically approached as supervised learning problems, which involve the utilization of classification algorithms. Logistic Regression is one such classifier commonly employed in these tasks. This algorithm provides a probabilistic estimation of class membership and can produce a binary outcome based on a predefined threshold.

This study aims to propose a novel approach for holistic sentiment analysis on the West African Pidgin language. The methodology employs a multi-pipeline approach where subject relevance and perceived sentiment is classified using finetuned models with the purpose of providing relevant metrics on the public perception of a given topic inclusive of local languages and slangs.

Corresponding Author: O. A. Ogungbe (*omolaraogungbe@gmail.com*)
Received 15 May 2024 | **Received in revised form** 23 September 2024 | **Accepted** 21 November 2024 | **Available online** 10 February 2025
The editor responsible for coordinating the review of this article and approving its publication was *I. P. Gambo*
P-ISSN: 1115-9782 **e-ISSN:** 2536-6807

2. RELATED WORKS

Limboi and Diosan (2022) proposed an unsupervised approach for Twitter sentiment analysis focused on the USA 2020 Presidential Election. Their methodology employed clustering algorithms to detect sentiment groups in unlabeled datasets, using the TF-IDF method for data representation. Internal and external validations, including comparisons with the VADER sentiment analysis tool, were conducted. The approach demonstrated comparable results to VADER in binary polarity evaluation.

However, classification methods remain the preferred choice in sentiment analysis due to their intuitive nature and the superior performance of more recent models. While Limboi and Diosan's (2022) unsupervised approach provides valuable insights into sentiment tendencies in political tweets, its reliance on clustering algorithms may limit its ability to capture nuanced sentiment shifts, especially in complex or context-dependent language such as Pidgin. Additionally, their approach focuses primarily on binary polarity, neglecting more fine-grained sentiment analysis.

The strengths of their work lie in its ability to handle large, unlabeled datasets without requiring prior labeled data, offering a potentially scalable solution for large-scale sentiment analysis. However, its weaknesses include the lack of detailed sentiment classification and reliance on simpler models like TF-IDF, which may not capture the full range of sentiment expressions in informal or multilingual contexts.

In contrast, my study adopts a more comprehensive, multi-pipeline approach that combines classification methods and cross-lingual models. By integrating the fine-tuned AfriBERTa model with a subject classifier, my approach addresses the limitations of binary polarity and can handle more complex languages such as West African Pidgin. Additionally, by incorporating transfer learning, this approach offers improved accuracy in sentiment analysis and a more nuanced understanding of sentiment across different contexts, overcoming the constraints of unsupervised clustering algorithms like those used by Limboi and Diosan.

Dang *et al.* (2020) conducted a comparative study on sentiment analysis using an unsupervised learning approach based on deep learning techniques. Their methodology involved preprocessing data with word embeddings and TF-IDF before applying Deep Neural Networks (DNNs), Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs). The models were evaluated comprehensively using five metrics: Accuracy, Precision, Recall, F-score, and AUC. The results showed that the CNN model achieved the best tradeoff between processing time and accuracy, reaching an accuracy rate of 80%. This study provides valuable insights into the effectiveness of deep learning models for sentiment analysis, with CNN demonstrating promising performance.

Their approach provides a clear understanding of model performance across different dimensions but relies on general sentiment datasets and lacks focus on more complex, multilingual, or informal language contexts. Additionally, while CNN performed well, the unsupervised approach does not explore the benefits of transfer learning or domain-specific fine-tuning, which could enhance performance, especially in niche language analysis. By incorporating the AfriBERTa model with transfer learning, my approach handles complex and underrepresented languages more effectively. Unlike Dang *et al.*'s unsupervised methods, my supervised classification approach offers better accuracy for nuanced sentiment analysis, especially in multilingual contexts.

Jiang *et al.* (2011) proposed a target-dependent Twitter sentiment classification methodology. They utilized syntactic parsing with a Maximum Spanning Tree dependency parser for subjectivity classification, followed by binary polarity classification for sentiment labeling. The authors introduced graph-based optimization, leveraging related tweets as context to improve accuracy. Their work is focused on Twitter sentiment analysis, specifically targeting sentiment classification based on context and primarily deals with short-form text.

A major contribution of this study is the introduction of graph-based optimization which improves sentiment classification by considering related tweets to provide context.

The target-dependent sentiment classifier achieved an accuracy of 66.0%, which increased to 68.3% with graph-based optimization. This indicates room for improvement. Moreover, the efficiency of real-time graph-based optimization for context retrieval as proposed by the authors may present computational challenges for large datasets. My work on this area builds on the inclusion of context awareness, but also enlists a cross-lingual, fine-tuned model with superior performance for low-resource languages like the West African Pidgin.

Devika *et al.* (2016) conducted a comparative study on different approaches in sentiment analysis. The methodology involved analyzing machine learning algorithms such as the K-Nearest Neighbor (KNN) Algorithm, and Naive Bayes algorithm, as well as rule-based and lexicon-based methods for opinion mining. The paper provided an overview of these algorithms, highlighting their strengths, weaknesses, and performance comparisions. It served as a reliable reference for understanding the arguments surrounding sentiment analysis approaches. Devika *et al.*, (2016) study offered valuable understanding into the diverse methods available in sentiment analysis, providing researchers and practitioners with a comprehensive understanding of their characteristics and applicability.

In this study, the Logistic Regression model implementation by Python's sklearn is adopted, and the NaijaSenti's AfriBERTa model, derived by training the XLM-Roberta (Cross-lingual Model - Roberta) model on an extensive Pidgin corpus. This approach in machine learning is known as transfer learning, and is very useful when there is a limited dataset on the target (Pidgin corpus). This enables us to leverage the performance of the extensive XLM-R model (developed and trained by Meta) on our dataset.

3. METHODOLOGY

3.1. System Description

This study focuses on the integration of machine learning pipelines to achieve the aim of obtaining relevant sentiment analysis for target topics with for text containing the West-African Pidgin language. The sequence diagram in Figure 1 illustrates the process. The system integrates a series of components including two machine learning pipelines and intermediate Natural Language Processing (NLP) features. Its features are exposed on web servers and user interfaces to aid visual interaction.

3.2. Data Collection and Preprocessing

Data was collected for the purpose of training, and during the operation of the system, for sentiment analysis processes. For training, the data used was from Kaggle datasets which was analyzed using Python's pandas and matplotlib libraries and preprocessed using Python's NLTK library. The data contained labelled tweets stating their relevance to the target topic (in this case, the 2023 APC political aspirant, Asiwaju Bola Ahmed Tinubu). 20% of the data was separated for training after cleaning the text in the preprocessing stage. The preprocessing operations include converting text to lowercase, removal of stop words, and lemmatization.

The Preprocessing pipeline made extensive use of Python's Natural Language Toolkit (NLTK) library Natural Language Processing to clean the data and handle imbalances in the dataset. The stopwords from each tweet in the dataset were removed using the corpus provided in the NLTK library. The stopwords include text that is considered irrelevant such as articles ("a", "an"), conjunctions, prepositions and common adverbs and adjectives. The process was carried out in code in

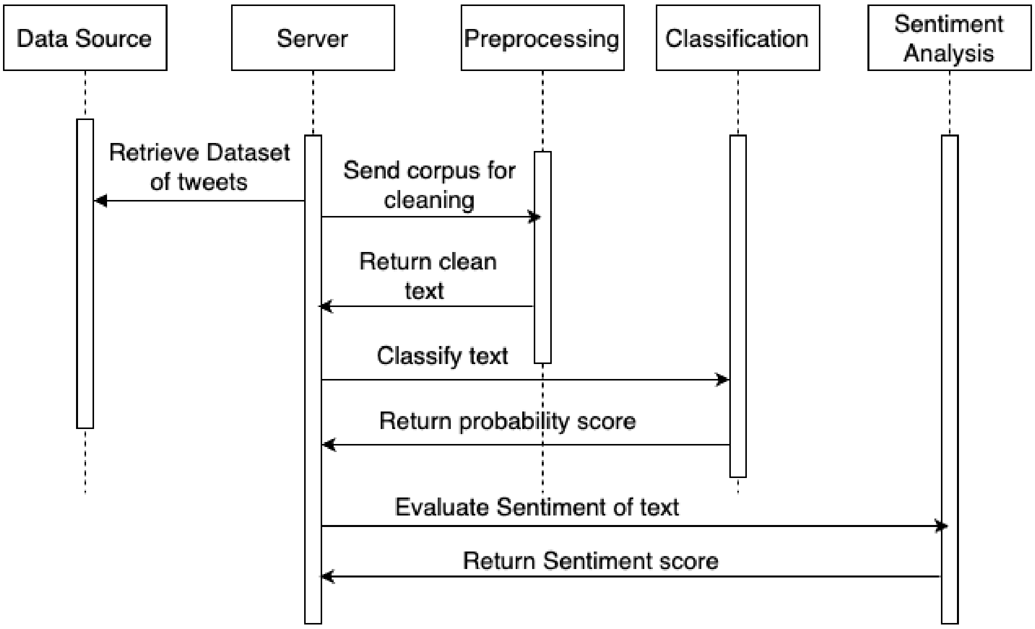

Figure 1: Sequence diagram depicting the multi-pipeline approach for sentiment analysis

several steps which include tokenization of the text, filtering and comparison against the library provided stopword list before being passed on for lemmatization. Lemmatization involves the normalization of text by reducing words to their base form or lemma, with the meaning of the word still preserved, for example "running" to "run", "better" to "good". In code, this was implemented using WordNetLemmatizer in the NLTK's stem library. These steps help in the reduction of dimensionality and improvement of model performance by focusing on more meaningful and significant words.

3.3. Classification model fitting and saving

The Logistic Regression model was used for the subject analysis pipeline. This model estimates the probability of class membership for a target, where a threshold is set, above or below which signifies whether the target is a member of the class or not. In this case, the model predicts whether the given tweet is relevant to the specific target (Tinubu) in a binary classification task. For the purpose of assigning weighted scores to sentiment values, the probability score is directly used instead of a binary classification result based on a threshold. The Logistic Regression model was developed by David Cox (1958) and can be represented by:

$$P(Y = 1 \vee X) = \frac{e^{a+bX}}{1+e^{a+bX}} \qquad [1]$$

where P is the probability of a 1 (or of Y being 1), X represents the input value and a and b are the model parameters. a is equal to P when X equals 0 while b serves for adjusting how quickly P changes with changing values of X.

The scikit-learn implementation of LogisticRegression was used with a solver algorithm of liblinear, which is adequate for small datasets, as opposed to sag and saga solvers which are more appropriate for larger datasets. Other hyperparameters are a C-index of 10, intercept_scaling of 1, tolerance of 1e-4, and a penalty of l2, which denotes the L2 penalty, also known as the Ridge regularization technique. The L2 penalty was applied by the model to prevent overfitting by adding the value of L2 to the model's cost function to result in more balanced coefficient values which improves the model's generalization performance. The L2 penalty can be represented by:

$$\lambda * \sum_{j=1}^{p}\left(\beta_j^2\right) \qquad [2]$$

where λ is the regularization parameter or **1/C**, p is the total number of coefficients in the model, and β_j is each coefficient in the model.

After splitting the collected data and training the model on the training set, the model was evaluated on the test set using the scikit-learn's classification report function, and various evaluation metrics such as precision, recall, F1-score, support, and confusion matrix were obtained to analyze the model.

Potential biases introduced by this model and its hyperparameters includes the assumption of a small dataset as expressed in the parameters, making it relatively unfit for large datasets, and its large inverse of regularization strength (C) of 10 (smaller values specify stronger regularization). This large value for C allows the model to be more sensitive to data points and features, leading to higher accuracy on the training set, but increases the risk of overfitting. A technique like cross-validation can be used to strike a balance in this case.

Finally, the model was saved using Python's joblib library. This saved model was used in a separate module as the subject analysis pipeline.

3.4. Sentiment Analysis pipeline

The Sentiment Analysis model used was the AfriBERTa large model, trained on the XLM-R (Cross-lingual Models Roberta) framework, which was developed by Alexis *et al.* (2020). The AfriBERTa model implements the aforementioned Transformer architecture, using the HuggingFace Transformers library. The HuggingFace Python library was used to retrieve the AfriBERTa model from the cloud repository, and the text input was tokenized and transformed by the model. The result was detached and converted to a numpy array, for which the softmax function (expressed in equation 2) was calculated.

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \qquad [3]$$

where z is the input vector and K is the number of classes in the multi-class classifier.

The result of the sentiment analysis model is expressed as a function of the probability of class membership. This provides a weighted-scoring model where the text is assigned relevance based on its closeness to the target topic.

3.5. Integration of pipelines

Python code was used to integrate the machine learning models and preprocessing features. This is due to its support for relevant libraries needed in this study. The Python code is exposed by a web server on port 5000 and is accessed by a single route, with a query parameter. The query parameter is used as a search term for the Twitter API accessed through Python's Tweepy API and authorization credentials from Twitter. The models are structured as Python modules and are called by the main Python code.

Golang programming language is used as the main backend entry point, due to its speed, low latency and support for advanced web server features such as concurrency. The Golang service is exposed on port 4000 and on a request from the client, it sends the request to the Python service.

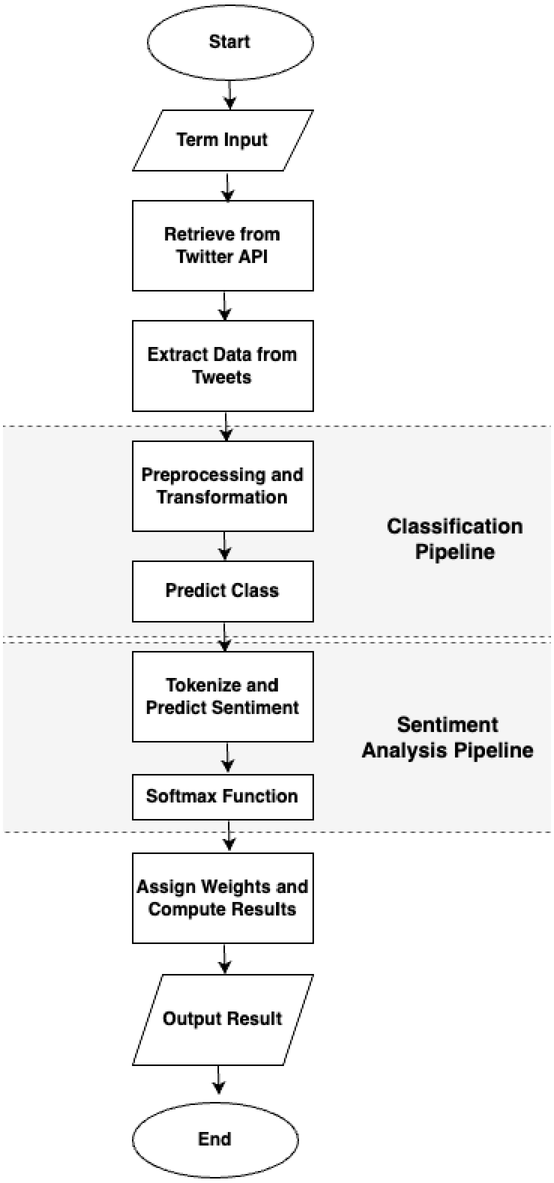
Start
Term Input
Retrieve from Twitter API
Extract Data from Tweets
Preprocessing and Transformation
Predict Class
Classification Pipeline
Tokenize and Predict Sentiment
Softmax Function
Sentiment Analysis Pipeline
Assign Weights and Compute Results
Output Result
End

Figure 2: Flowchart for multi-pipeline integration process

The Text classification model was saved using Python's joblib library. A module was written to access the saved model using the same library. That represented the classification pipeline. For the Sentiment analysis pipeline, a module was written to accept a text string and return sentiment scores based on the previously established algorithms. Integrative Python code was written to accept the search query, perform a search with the Twitter API and return a list of tweets, to preprocess these tweets and obtain relevant information, and pass them through the two pipelines, programmatically represented as Python modules. The integrated system is presented in Figure 2.

3.6. Interface and Visualization

General web technology was used to handle user interfaces using HTML (Hypertext Markup Language), CSS (Cascading Style Sheets), and Javascript. Javascript was used for the SPA (Single Page Applications) feature and was exposed on port 8080.

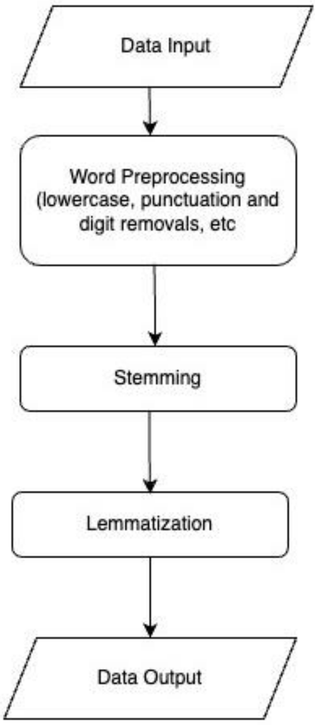
Data Input
Word Preprocessing (lowercase, punctuation and digit removals, etc
Stemming
Lemmatization
Data Output

Figure 3: Breakdown of data preprocessing stage

4. RESULTS AND DISCUSSION

Figure 3 and 4 presents findings on the dataset with the aim of determining subject relevance. The first purpose of classification is to discover if a given tweet covers the topic adequately. A common challenge to overcome is the spamming of hashtags for wider engagement by content creators. From the figures, the characteristics of subject relevant tweets can be extracted, improving the accuracy of relevant tweet classification. The subject of relevance used was "Tinubu" in this context, the 2023 political aspirant of the APC party in Nigeria.

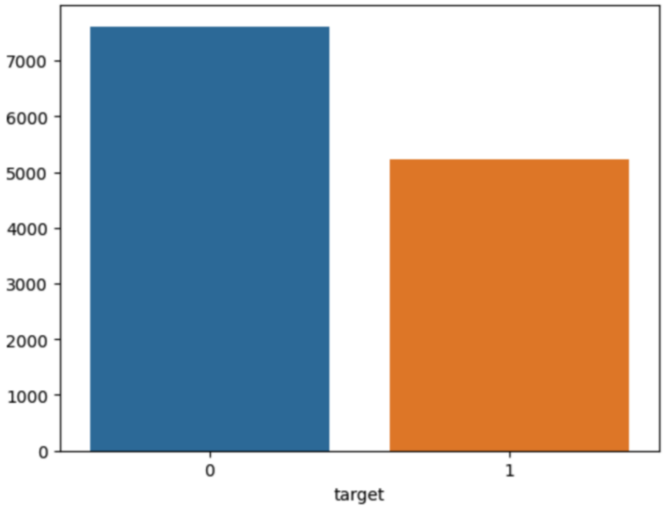

Figure 4: Labelled data ratio, relevant vs irrelevant tweets

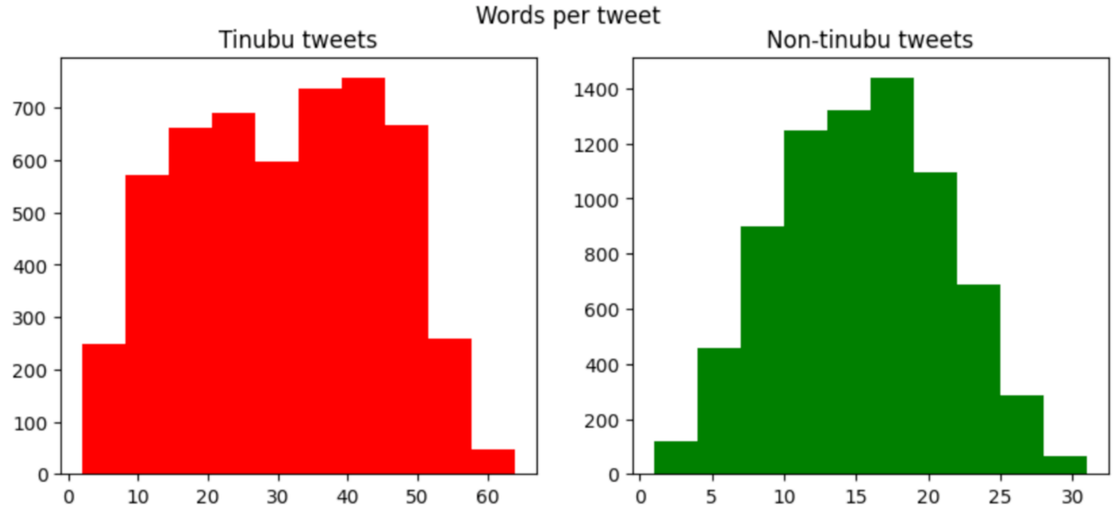

Figure 5: Analysis of word count for relevant vs non-relevant tweets

The dataset for testing was saved in a separate CSV (Comma Separated Values) file, and used to evaluate the Logistic Regression classification model. The value of C (inverse of regulariation strength) used was 10 to balance bias and variance, ensuring the model captures patterns in the data without overfitting. The solver (linear classifier) used was liblinear which uses a coordinate descent algorithm suitable for the small size of the dataset and the penalty was set to L2 (Ridge regularization) due to the non-sparse nature of the data.

For the AfriBerta model for sentiment analysis, the total size of the pretraining corpus was 0.94 GB. According to the original model authors (Ogueji *et al.*, 2021), the corpus contains approximately 5.45 million sentences and 108.8 million tokens taken from the British Broadcasting Corporation News website. Languages in in the dataset include Nigerian Pidgin, Igbo, Yoruba, Hausa, Somali, and other West African languages that contribute to the English Creole.

The three pretrained models were available for use (AfriBerta-small, AfriBerta-base and AfriBerta-large models) with varying complexities of 4, 8 and 10 layers respectively. Each model made use of 6 attention heads (to attend to multiple perspectives in the input data and effectively understand complex dependencies) and a maximum token length of 512 and were adjusted so that all models have approximately the same number of parameters (126 million, 111 million and 97 million for large, base and small, respectively, to facilitate a fair comparison and control complexity). The best performing model is the AfriBERTa large with an F1-score of 90.86 in Text Classification of text containing Yoruba compared to mBERT's 83.03 and XLM-R's 85.62. Higher F1-scores were also reported in other West African languages (Ogueji et. al, 2021). The datasets and benchmark experiments were adapted from the work of Muhammad et. al (2022).

Table 1 and Figure 6 show the result obtained from the classification report of the scikit-learn's model and the confusion matrix, respectively. The Sentiment Analysis model achieved an F1 score of 74.5 (average of 5 runs). In Table 1, the results of classification of both relevant and non-relevant tweets were shown. This describes the performance of the system in identifying tweets that were relevant or irrelevant to the topic of discussion (that is, binary classification of the topic of the tweet) through the implemented Logistic Regression algorithm.

For non-relevant tweets, the Logistic Regression model exhibited a precision of 0.82, implying that approximately 82% of the tweets predicted as non-relevant were indeed non-relevant. The recall, also known as sensitivity, attained a value of 0.84, indicating that the model correctly identified around 84% of the non-relevant tweets. The F1 score, a combined metric considering both precision and recall, yielded a value of 0.83. This score provides a balanced assessment of the model's performance, accounting for both the ability to correctly identify non-relevant tweets and the accuracy of such identifications. The support value, denoting the number of non-relevant tweets present in the evaluation dataset, amounted to 857 instances.

Table 1: Classification report

	Precision	Recall	F1 Score	Support
Non Relevant tweets	0.82	0.84	0.83	857
Relevant tweets	0.79	0.76	0.77	666
Accuracy			0.81	1523
Macro Average	0.80	0.80	0.80	1523
Weighted Average	0.81	0.81	0.81	1523

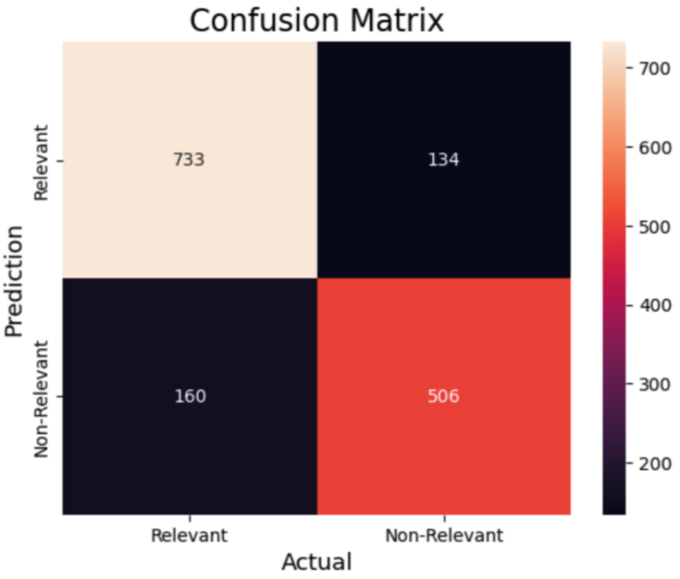

Figure 6: Confusion Matrix depicting the performance of the subject classification pipeline

For relevant tweets, the precision achieved was 0.79, recall, 0.76, and F1 score, 0.77. The support value amounted to 656 instances. The accuracy of the model was 0.81.

5. CONCLUSION AND RECOMMENDATION

This study explored sentiment analysis in Pidgin, a widely used West African language, aiming to develop an integrated pipeline approach for efficient sentiment analysis and classification of Pidgin text. Existing systems perform poorly on under-resourced languages and often require a large training corpus to yield satisfactory results. The solution proposed by this study achieved an accuracy of 81% in subject classification and efficiently leveraged a high-performance, sentiment analysis model for Pidgin.

The implemented AfriBERTa model demonstrates a significant improvement in self-supervised language models as it uses only a fraction of the dataset in mBERT and XLM-R and yet outperforms them in sentences containing West African language token. While the existence of English in Nigerian Pidgin could be an advantage on the part of larger models, a fractional increase in dataset size could easily make up for the bias.

It is recommended that the dataset used for training be expanded and more diversified to capture a wider range of linguistic expressions. A collaboration with native speakers to serve as human feedback can also help to improve the results of the system.

REFERENCES

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2), 1-135.

Feldman, R. and Sanger, J. (2007). The text mining handbook: Advanced approaches in analyzing unstructured data, 1-3. *Cambridge: Cambridge University Press*Ogueji K., Zhu Y., and Lin J. (2021). Small Data? No Problem! Exploring the Viability of Pretrained Multilingual Language Models for Low-resourced Languages. In Proceedings of the 1st Workshop on Multilingual Representation Learning, 116–126, *Punta Cana, Dominican Republic. Association for Computational Linguistics.*

Olagunju, Tolulope, Oyebode, Oladapo and Orji, Rita. (2020). Exploring Key Issues Affecting African Mobile eCommerce Applications Using Sentiment and Thematic Analysis. IEEE Access. 8. 114475-114486. 10.1109/ACCESS.2020.3000093.Berry, M. W. and Kogan, J. (2010). Text mining: Applications and theory, i-xiv. *John Wiley and Sons*.

Limboi, S. and Diosan, L. (2022). An unsupervised approach for Twitter Sentiment Analysis of USA 2020 Presidential Election 1-6.10.1109/INISTA55318.2022.9894264. Dang, N.C, Moreno-García, M.N. and De la Prieta, F. (2020). *Sentiment Analysis Based on Deep Learning: A Comparative Study. Electronics. 2020; 9(3):483*

Jiang L., Yu M., Zhou M., and Liu X. (2011). Target-dependent Twitter sentiment classification, 151-160. *In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*.

Muhammad S. H., Adelani D. I., Sebastian R., Ahmad I. S., Abdulmumin I., Bello B. S., Choudhury M., Emezue C. C, Abdullahi S. S, Aremu A., Jorge A. and Brazdil P. (2022). NaijaSenti: A Nigerian Twitter Sentiment Corpus for Multilingual Sentiment Analysis. In Proceedings of the Thirteenth Language Resources and Evaluation Conference, 590–602, Marseille, France. European Language Resources Association.

Devika, M.D., Sunutha, C., and Ganesh A. (2016). Sentiment Analysis: A Comparative Study on Different Approaches.

Alexis C., Kartikay K., Naman G., Vishrav C., Guillaume W., Francisco G., Edouard G., Myle O., Luke Z. and Veselin S. (2020). Unsupervised Cross-lingual Representation Learning at Scale. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8440–8451, Online. Association for Computational Linguistics.

Ihemere, Kelechukwu Uchechukwu. (2006). An Integrated Approach to the Study of Language Attitudes and Change in Nigeria: The Case of the Ikwerre of Port Harcourt City. In Olaoba F. Aransanyi & Michael A. Pemberton (eds.), Proceedings of the 36th Annual Conference on African Linguistics, 194–207.

Wolf T., Debut L., Sanh V., Chaumond J., Delangue C., Moi A., Cistac P., Rault T., Louf R., Funtowicz M., Davison J., Shleifer S., Patrick von Platen, Ma C., Jernite Y., Plu J., Xu C., Teven Le Scao and Gugger S. (2020). Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 38–45, Online. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., (2017). Attention is all you need. Advances in neural information processing systems, 30.

Shah, K., Patel, H., Sanghvi, D. and Shah M. (2020). A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for Text Classification. Augment Hum Res 5, 12Cox, D.R. (1958) The Regression Analysis of Binary Sequences. Journal of the Royal Statistical Society: Series B, 20, 215-242.

.