MIKAILALSYS
Journal of Multidisciplinary Sciences

e-ISSN : 2987-2286
p-ISSN : 2987-3924

Index : Dimensions, Scilit, Crossref, Garuda, Semantic, Lens,Google Scholar,Base, etc

https://doi.org/10.58578/mikailalsys.v4i1.8596

Cross-Lingual Sentiment Analysis with Natural Language Processing: Insights from Selected Nigeria Languages (Yoruba, Hausa, Igbo, and Nigerian Pidgin)

Rilwan Abdulyekeen & Bashir Abbayaro Abdullahi
Federal University Dutsin-Ma, Nigeria
rabdulyekeen@fudutsinma.edu.ng

Article Info:

Submitted:	Revised:	Accepted:	Published:
Dec 29, 2025	Jan 20, 2026	Feb 10, 2026	Feb 23, 2026

Abstract

Natural Language Processing (NLP) plays a pivotal role in sentiment analysis, particularly in multilingual societies such as Nigeria, where languages like Igbo, Hausa, Yoruba, and Pidgin English coexist yet remain largely underrepresented in existing NLP tools and resources. This study aims to develop a sentiment analysis framework specifically tailored to Nigeria's four major languages, addressing key challenges including code-mixing, tonal variations, and the scarcity of annotated datasets. The research leverages existing linguistic studies on these languages while constructing customized annotated datasets and designing models optimized for their structural and phonological properties. By systematically integrating linguistic insights with task-specific model development for low-resource settings, the proposed framework is designed to handle multilingual and code-mixed inputs more effectively than generic NLP systems. The study concludes that a targeted, language-aware approach is essential for improving sentiment analysis performance in underrepresented African languages and for ensuring that NLP technologies reflect the linguistic realities of multilingual societies. The contributions of this research lie in

advancing multilingual sentiment analysis for low-resource African languages, providing methodological guidance for handling code-mixing and tonal features, and supporting practical applications in business intelligence, governance, public opinion mining, and social media analytics.

Keywords: Multilingual Sentiment Analysis; Natural Language Processing; Low-Resource African Languages; Code-Mixing and Code-Switching; Social Media Text Mining

INTRODUCTION

Natural Language Processing (NLP) has become an essential tool for grasping and analyzing the emotions that people express through language. (Adeyanju, A. 2021). In Nigeria, a nation rich in linguistic diversity, effective sentiment analysis needs to consider the four main languages: Igbo, Hausa, Yoruba, and Pidgin English. Each of these languages carries its cultural nuances, idiomatic phrases, and grammatical structures, which can create significant hurdles for traditional NLP models that are usually designed for more commonly spoken languages like English. (Odebiyi, J., and Ojo, A.2020). With the rise of social media and digital platforms in Nigeria, there's an increasing demand for powerful sentiment analysis tools that can accurately capture feelings in these languages. (Izuagbe, R., 2022). This research seeks to delve into the methods and techniques that can be applied to sentiment analysis within these linguistic contexts, emphasizing the importance of context and cultural significance in crafting effective NLP solutions. Tackling these challenges, the study aims to contribute to the wider field of NLP and improve communication and understanding of sentiments in Nigeria's multilingual environment, while also creating benchmark experiments with our sentiment lexicon and expanding the dataset (Shamsuddeen et al, 2022).

The rapid proliferation of digital communication has necessitated the development of advanced Natural Language Processing (NLP) techniques, particularly in regions with rich linguistic diversity like Nigeria. With over 500 languages spoken across the country, the four major languages Igbo, Hausa, Yoruba, and Pidgin English represent a significant portion of the population's communication. (Adeyanju, A, 2021). Each language has its own distinct phonetic, syntactic, and semantic characteristics, which complicate the task of sentiment analysis. Traditional NLP models, often designed for English and other widely

spoken languages, struggle to accurately capture the sentiments expressed in these languages due to the lack of annotated datasets and linguistic resources. (Izuagbe, R., 2022). Moreover, cultural nuances and context-specific expressions in Nigerian languages further challenge existing sentiment analysis frameworks. (Odebiyi, J., and Adetunji, A. (2020).

Sentiment analysis plays a vital role in grasping public opinion and consumer behavior, but when it comes to Nigeria's major languages Igbo, Hausa, Yoruba, and Pidgin English, its use is still quite limited. Most current models are built with English in mind, which leads to poor performance because of issues like a lack of annotated datasets, variations in dialects, and cultural subtleties. Additionally, existing NLP frameworks often overlook the unique syntactic and semantic characteristics of these languages, resulting in misunderstandings. This study seeks to fill those gaps by creating customized methods to enhance the accuracy of sentiment analysis in Nigeria's diverse linguistic landscape. (Odebiyi and Adetunji, 2020)

The aim of this research paper is to apply sentiment analysis using Natural Language Processing (NLP) to Nigeria's four major languages Igbo, Hausa, Yoruba, and Pidgin English towards improving the understanding and classification of multilingual digital content.

To achieve this aim, the study gathers diverse datasets from social media platforms, blogs, and other digital sources containing texts in Igbo, Hausa, Yoruba, and Pidgin English. The collected data is then preprocessed to remove noise, inconsistencies, and irrelevant content, ensuring a clean and reliable dataset for analysis. Sentiment analysis techniques are applied to interpret and classify the data, taking into consideration the unique linguistic and cultural features of each language. Finally, a user-friendly sentiment analysis tool is developed and evaluated to make the results easily accessible for researchers, businesses, and policymakers.

This study addresses the underdevelopment of sentiment analysis in Nigeria's major languages Igbo, Hausa, Yoruba, and Pidgin by creating tailored models that account for linguistic and cultural nuances. It bridges the gap caused by the scarcity of annotated datasets, dialectal variations, and the limitations of English-focused NLP frameworks. Collecting and curating diverse datasets, cleaning them for accuracy, and analyzing sentiment lexicons, the study enhances the precision of sentiment analysis. The development of an open-source, user-friendly application ensures accessibility, empowering

businesses, policymakers, and researchers to make informed decisions based on public sentiment. Ultimately, this research advances multilingual NLP, promotes inclusivity, and fosters more accurate and impactful applications in Nigeria's diverse linguistic landscape.

This study is all about creating sentiment analysis models tailored for Nigeria's four major languages Igbo, Hausa, Yoruba, and Pidgin using Natural Language Processing (NLP) techniques. It involves gathering and organizing datasets from social media, blogs, and various digital platforms, followed by cleaning the data, performing sentiment analysis, and building a user-friendly, open-source application. The goal is to tackle the unique linguistic and cultural aspects of these languages to ensure that sentiment is interpreted accurately. Additionally, the study will assess how well the models perform compared to existing benchmarks and encourage their use in business and governance to improve decision-making.

However, the study does face some limitations, particularly due to the lack of annotated datasets for Igbo, Hausa, Yoruba, and Pidgin, which could affect the strength of the models. There are also challenges posed by dialectal differences within these languages and the ever-changing nature of social media data. Moreover, the research is limited to text-based sentiment analysis, leaving out other data types like audio or video. The development of the open-source application might also encounter hurdles regarding user adoption and technical support. Despite these challenges, the study aims to lay down a solid foundation for future progress in multilingual sentiment analysis.

Related Works

The review kicks off by tracing the evolution of Natural Language Processing (NLP) and its role in sentiment analysis, pointing out the prevalence of English-centric models and their shortcomings in multilingual environments (Izuagbe, 2022). It then takes a closer look at the specific challenges brought about by Nigeria's rich linguistic diversity, such as dialectal differences, cultural subtleties, and the lack of annotated datasets (Shamsuddeen et al., 2022). These challenges highlight the necessity for tailored strategies that consider the unique traits of Igbo, Hausa, Yoruba, and Pidgin, which are frequently overlooked in mainstream NLP research (Odebiyi and Adetunji, 2020).

The literature reviews the methodologies and tools used in sentiment analysis, including machine learning algorithms, sentiment lexicons, and data preprocessing techniques (Adeyanju, 2021). It also discusses how open-source applications are making

sentiment analysis more accessible to a wider audience (Izuagbe, 2022). By weaving together these insights, the literature review paves the way for the creation of customized sentiment analysis models that cater to the specific needs of Nigeria's multilingual landscape, ultimately aiding the progress of NLP in languages that often don't get the attention they deserve (Shamsuddeen et al., 2022).

Nigeria

Nigeria is a vibrant country nestled in West Africa, sharing its borders with Benin to the west, Niger to the north, Chad to the northeast, and Cameroon to the east. To the south, it boasts a beautiful coastline along the Gulf of Guinea, which is part of the Atlantic Ocean. With a population exceeding 200 million, Nigeria holds the title of the most populous nation in Africa and is celebrated for its incredible cultural diversity, featuring over 250 ethnic groups and more than 500 languages (Falola and Heaton, 2008).

The country functions as a federal republic, consisting of 36 states and a Federal Capital Territory (FCT) in Abuja, which is the heart of its political and administrative activities. Although Lagos was the former capital, it still stands as Nigeria's largest city and a bustling economic center (National Population Commission of Nigeria, 2022).

Nigeria achieved independence from British colonial rule on October 1, 1960, and has since evolved into one of Africa's largest economies, largely fueled by oil production, agriculture, and services (World Bank, 2023). It is an active member of various international organizations, including the United Nations (UN), the African Union (AU), and the Economic Community of West African States (ECOWAS).

Yoruba Language

Yoruba is a fascinating tonal language that belongs to the southern Nigeria family, specifically within the Yoruba group. It features three distinct tones high, mid, and low that can change the meaning of words entirely. With more than 20 million native speakers, Yoruba is predominantly spoken in southwestern Nigeria, particularly in states like Oyo, Ogun, Osun, Ondo, Ekiti, Lagos, Kogi, and Kwara. Lagos, Nigeria's bustling economic center has played a significant role in spreading Yoruba as a common language. This language is vital for keeping alive oral traditions, including proverbs, folktales, and songs, and it has also made a mark on Nigerian literature and global pop culture ((Izuagbe, 2022).

Igbo Language

Igbo is a significant language belonging to the Niger-Congo family, specifically under the Volta-Niger branch, and it's spoken by around 15 million people, mainly in southeastern Nigeria. (Ugo and Emeka, 2023). This language boasts a variety of dialects that are generally understandable to each other within the larger Igbo linguistic group. Igbo predominantly in the states of Abia, Anambra, Ebonyi, Enugu, and Imo, with notable communities also in parts of Delta and Rivers States. Beyond just communication, Igbo is vital for maintaining the cultural identity and history of its speakers, acting as a vessel for oral traditions, and rituals, and fostering community spirit, even with its dialectal differences. Moreover, the Igbo diaspora has played a key role in promoting and preserving the language on a global scale (Chike et al, 2023).

Hausa Language

Hausa, a member of the Afro-asiatic language family and part of the West Chadic subgroup, stands out as one of the most widely spoken languages in Africa. (Muhammed and Idris, 2020), It's not just the native tongue of the Hausa people; it also acts as a common language across northern Nigeria and beyond. In Nigeria, you'll find Hausa predominantly in the northern states like Kano, Kaduna, Katsina, Sokoto, Zamfara, Bauchi, and Gombe, and it's often used as a lingua franca in bustling cities such as Abuja, the capital. But the reach of Hausa doesn't stop at Nigeria's borders; it's also spoken in several West and Central African nations, including Niger, Cameroon, Ghana, Benin, Togo, Chad, and Sudan, where it plays a vital role in trade and communication throughout West Africa (Oluwaseun and hammed, 2023).

Pidgin Language

Nigerian Pidgin, often called "Pidgin English," is an English-based creole that serves as a common language among Nigeria's diverse population. (James, et al, 2019). It simplifies English by mixing in elements from various local languages, making it easier for people from different ethnic backgrounds to communicate. Pidgin is widely spoken across the country, especially in urban areas where folks from different linguistic backgrounds come together. (Idris Adam, 2020). Some of the key places where it's commonly used include Lagos (Southwest), Port Harcourt (South-South), Warri (South-South), and Abuja (North Central). Culturally, Nigerian Pidgin is crucial for bridging communication gaps and

has become popular in Nigerian music, comedy, and media, serving as a unifying and informal way to express oneself (Sammy H., 2020).

Natural Language Processing (NLP)

Natural Language Processing (NLP) is a subfield of artificial intelligence (AI) that focuses on enabling computers to understand, interpret, and generate human language. NLP combines computational linguistics with machine learning and deep learning techniques to analyze text and speech data (Jurafsky and Martin, 2021). The field has numerous applications, including machine translation, speech recognition, and sentiment analysis. With the increasing availability of big data and advances in AI, NLP has seen significant improvements in accuracy and efficiency, making it a critical tool for understanding public opinions, customer feedback, and online discussions (Cambria et al., 2020).

Overview of NLP in Sentiment Analysis

Sentiment analysis, also known as opinion mining, is a key application of NLP that involves determining the sentiment expressed in a piece of text, whether positive, negative, or neutral (Liu, 2015). This technique is widely used in social media monitoring, customer feedback analysis, and political opinion tracking. NLP-based sentiment analysis employs various approaches, including lexicon-based methods, machine learning algorithms, and deep learning techniques (Medhat et al., 2014). Sentiment analysis is particularly important in understanding consumer behavior and public opinion, helping businesses and policymakers make informed decisions. However, achieving high accuracy in sentiment classification remains a challenge, especially in multilingual and low-resource language contexts.

Challenges of NLP in Multilingual Contexts

One of the major challenges in NLP is handling multiple languages, particularly those with limited computational resources. Most NLP models are trained primarily on high-resource languages such as English, making it difficult to achieve similar accuracy levels in low-resource languages like many African languages (Adelani et al., 2021). Another challenge is code-switching, where speakers mix two or more languages in a single sentence, making sentiment detection more complex (Mavula et al., 2022). Additionally, variations in dialects, cultural expressions, and informal speech further complicate sentiment analysis, requiring more advanced language models trained on diverse datasets.

NLP Techniques for Sentiment Analysis

NLP techniques for sentiment analysis can be broadly categorized into lexicon-based, machine learning-based, and deep learning-based approaches. Lexicon-based methods rely on predefined word lists with sentiment scores to classify text (Taboada et al., 2011). Machine learning techniques, such as Naïve Bayes, Support Vector Machines (SVM), and Random Forest, use labeled datasets to train models for sentiment classification (Pang and Lee, 2008). Deep learning approaches, including Long Short-Term Memory (LSTM) networks and transformer-based models like BERT, have significantly improved sentiment analysis accuracy by capturing contextual meanings in the text (Devlin et al., 2019). The choice of technique depends on the dataset, language, and application domain.

Sentiment Analysis in Nigerian Languages

Nigeria, with its linguistic diversity, presents unique challenges and opportunities for sentiment analysis. While English remains the dominant language for online communication, indigenous languages such as Yoruba, Igbo, and Hausa are widely used on social media and other digital platforms (Adebara and Abdul-Mageed, 2022). However, sentiment analysis in Nigerian languages is still in its early stages due to a lack of labeled datasets, linguistic resources, and computational tools. Recent advancements in NLP and machine translation have started addressing these gaps, enabling more research on sentiment classification in Nigerian languages.

Current State of Sentiment Analysis in Nigeria

Sentiment analysis in Nigeria is primarily conducted in English, with limited studies focusing on indigenous languages. While some researchers have developed lexicon-based sentiment analysis models for Nigerian Pidgin and Hausa, these models often suffer from low accuracy due to limited training data (Alabi et al., 2020). Additionally, social media platforms such as Twitter and Facebook provide a rich source of text data for sentiment analysis, but linguistic diversity and code-switching pose challenges for automated text classification. More efforts are needed to create annotated datasets and improve NLP models for Nigerian languages.

Challenges in Sentiment Analysis for Nigerian Languages

The primary challenge in sentiment analysis for Nigerian languages is the scarcity of labeled datasets and linguistic resources. Many indigenous languages lack digital corpora, making it difficult to train and evaluate NLP models (Orife et al., 2020). Another challenge is the high level of code-switching, where users mix English with indigenous languages, making it harder for traditional NLP models to classify sentiments accurately (Ezeani et al., 2022). Additionally, tonal variations in languages like Yoruba and morphological complexities in Hausa require specialized NLP techniques that go beyond conventional text processing methods.

METHODS

The application is structured using the Model View Controller (MVC) architectural pattern to ensure modularity, maintainability, and scalability. The user interface is developed with HTML, CSS, and Vanilla JavaScript to provide a lightweight, responsive, and intuitive front-end experience. For sentiment analysis, the OpenAI API is integrated as the core engine to process and classify text data across the selected Nigerian languages, enabling accurate and efficient analysis.

Model-View-Controller (MVC)

The Model-View-Controller (MVC) architectural pattern is a smart software design strategy that breaks an application down into three key parts: the Model, the View, and the Controller. In fact, MVC was the framework used to develop a sentiment analysis platform for Nigeria's major languages, including Igbo, Hausa, Yoruba, and Pidgin. The Model took charge of processing data like text preprocessing and sentiment classification while the View was responsible for the user interface, which was crafted using HTML, CSS, and Vanilla JavaScript. Meanwhile, the Controller served as the link between user input and data processing. By adopting the MVC approach, the project enjoyed improved code organization and greater flexibility.

Software Requirements

The sentiment analysis platform was developed using a variety of software tools and libraries that facilitate natural language processing, web development, and API integration. The main software components include:

Programming Languages: HTML, CSS, and Vanilla JavaScript were used for the user interface development, ensuring a lightweight and responsive platform. JavaScript also facilitated interaction with the Open-AI API for sentiment analysis.

Libraries and Frameworks:

OpenAI API: This API was integrated into the system for natural language processing, which provided the sentiment classification model for the four languages: Igbo, Hausa, Yoruba, and Pidgin.

Bootstrap: For front-end styling and responsive design, Bootstrap was used to create a clean and user-friendly interface.

Node.js: Node.js was used on the server side to handle HTTP requests and provide a bridge between the front end and the API services.

Database: No dedicated database was required, as data processing was handled through API calls and real-time input.

Version Control: Git was used for version control, ensuring collaboration and effective tracking of changes during the development process.

Hosting Platform: The platform was deployed on a web server using cloud hosting services such as Netlify for easy access and scalability.

Table 1. Software requirement

Requirement Type	Software Requirements	Hardware Requirements
Text Editor/IDE	A text editor or IDE such as Visual Studio Code, Sublime Text, or Notepad for editing HTML, CSS, and Vanilla JavaScript code.	-
Web Browser	A web browser like Google Chrome, Mozilla Firefox, or Microsoft Edge for accessing and using the chatbot.	-
Version Control	Git for managing code changes and collaboration.	-
Package Manager	Composer for installing and managing third-party libraries and dependencies.	-
Processor	-	Multi-core processor with a clock speed of 2.0 GHz or higher is recommended.

Requirement Type	Software Requirements	Hardware Requirements
Memory (RAM)	-	At least 4 GB RAM for small to medium systems; 8 GB or more for larger systems with high user traffic.
Storage	-	Hard drive with a minimum of 50 GB free space for system and database files.
Display	-	Monitor with at least 1280 x 768-pixel resolution recommended.
Internet Connection	-	High-speed internet connection with sufficient bandwidth for web-based chatbot use.

Functional Requirements

The functional requirements specify the essential tasks that the system must perform to meet its goals. For this research, the following functional requirements were identified:

User Interface: The platform must provide a user-friendly interface where users can input text in any of the four major languages (Igbo, Hausa, Yoruba, or Pidgin) for sentiment analysis.

Real-Time Sentiment Analysis: The system should analyze the sentiment (positive, negative, or neutral) of the input text in real-time using the Open-AI API and display the results.

Data Preprocessing: The system should clean and preprocess the text input before analysis, handling tasks such as tokenization and removing special characters.

Multi-Language Support: The platform must support sentiment analysis for four major Nigerian languages: Igbo, Hausa, Yoruba, and Pidgin.

API Integration: The system should integrate with the Open-AI API for sentiment classification.

System Design

The system design for the sentiment analysis platform follows a structured approach, leveraging the Model-View-Controller (MVC) architecture. The design prioritizes ease of use, scalability, and maintainability.

User Interface Design: The platform's user interface was designed using HTML, CSS, and JavaScript. Bootstrap was used for styling and ensuring responsiveness across different screen sizes. The input form allows users to submit text for analysis, and the results are dynamically displayed on the same page.

Model Design: The model component handles the data preprocessing and integration with the Open-AI API. Text entered by users is cleaned, tokenized, and passed to the API, which returns sentiment analysis results. The model also ensures support for multiple languages.

Controller Design: The controller manages the flow of data between the user interface and the model. It captures user input, sends it to the model for analysis, and retrieves the results to update the view dynamically. The controller also manages API calls and handles any errors or exceptions during processing.

API Integration: The platform integrates with the Open-AI API through JavaScript, allowing the system to leverage powerful machine-learning models for sentiment analysis. The API is called every time a user submits a text input, with results processed and displayed in real time.

Real-Time Feedback: The system design includes real-time feedback mechanisms, where users can immediately see the results of their sentiment analysis, along with summaries and insights into the nature of the input text.

Data Acquisition

1. Gathered datasets in Igbo, Hausa, Yoruba, and Pidgin.
2. Using Web scraping tools to collect the necessary data in text formats,
3. Ensuring a rich variety of sources for each language.

Data Processing

The cleaning of the collected data was achieved through preprocessing techniques. This involved removing noise, duplicates, and irrelevant information, as well as tokenizing the text. Standard data-cleaning tools and libraries, such as Python's nltk and re, were used for this process to ensure only high-quality data was fed into the sentiment analysis models.

NPL Design Implementation

NLP techniques were employed using the OpenAI API. Each dataset was analyzed using sentiment classification models that consider the linguistic and cultural aspects unique to the languages in question. The API processed the cleaned data, classifying it as positive, negative, or neutral, thus providing an interpretation of user sentiments in each language.

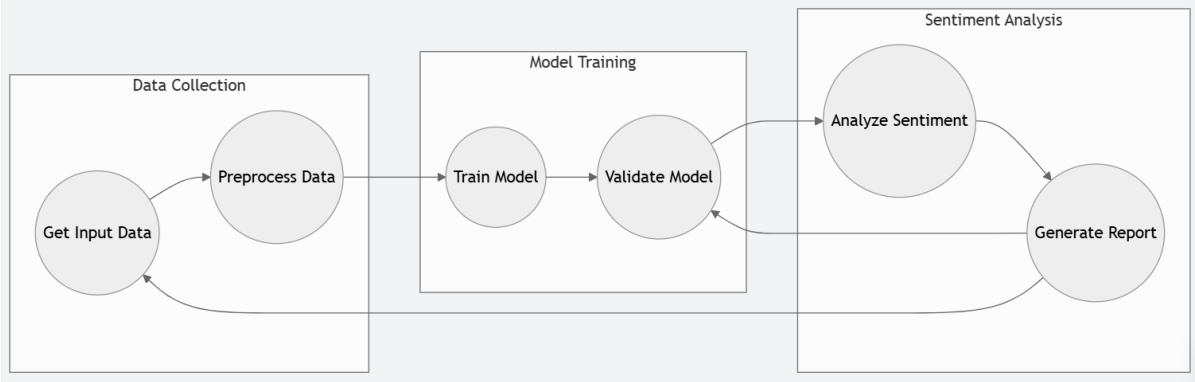

Fig 1. System NPL Flow

Natural Design

The results of the sentiment analysis were presented in a user-friendly responsive readable format making them easily interpretable for stakeholders

Flow Chart

The flow chart for the sentiment analysis platform outlines the steps involved in processing user input and displaying results. The software is designed such that:

User Loads the App: The user opens the application through a web browser, and the user interface is displayed. The interface consists of an input field where users can type their text, along with a button to submit the input for sentiment analysis.

User Inputs Text: The user types text in any of the supported languages (Igbo, Hausa, Yoruba, or Pidgin) into the provided input field.

User Clicks Sentiment Button: Once the text is entered, the user clicks the "Sentiment" button to trigger the sentiment analysis process.

System Processes the Input: The application sends the input text to the backend, where it is cleaned, tokenized, and passed to the OpenAI API for sentiment classification.

API Classifies Sentiment: The OpenAI API analyzes the text and classifies it as positive, negative, or neutral, based on the linguistic features of the input.

System Displays the Result: The platform dynamically updates the interface to display the sentiment analysis result to the user, summarizing whether the input text conveys positive, negative, or neutral sentiments

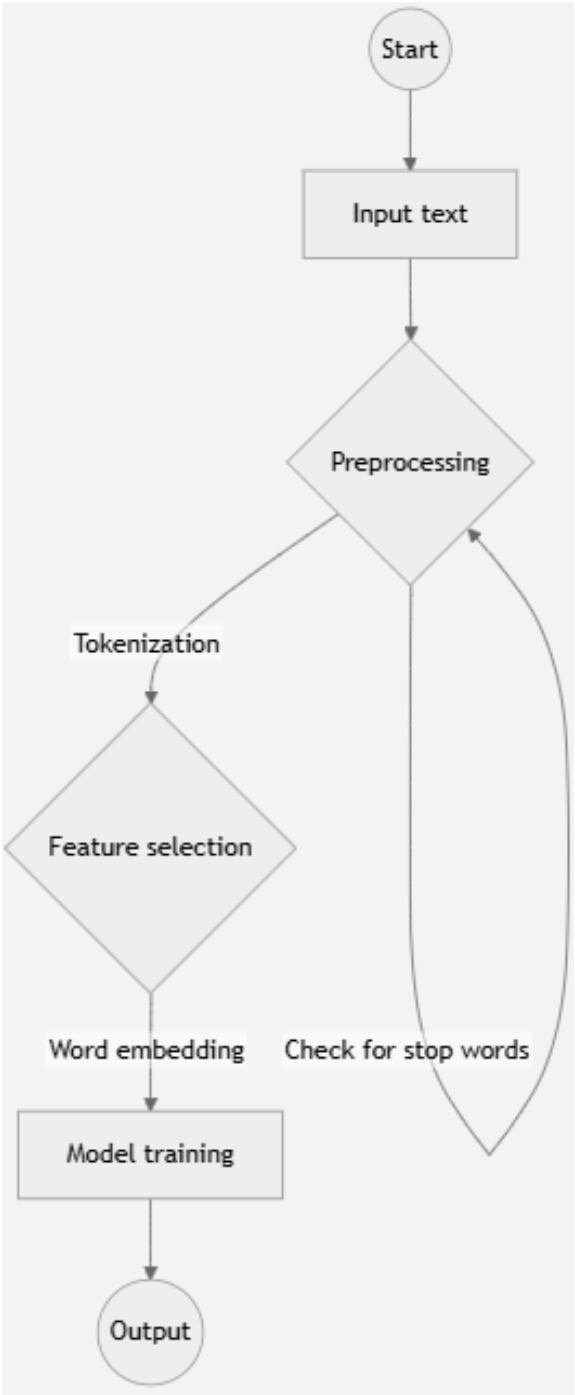

Fig 2. System Flow Chart

Sequence Diagram

The sequence shows the scenario for the sentiment analysis platform follows a similar structure to the flow chart, describing the user interactions and system responses:

User Loads the App: The user accesses the application on a web browser by entering the URL or clicking on a saved link. The app loads, presenting the user interface with an input field and a "Sentiment" button.

User Inputs Text: The user enters a sentence or paragraph in any of the four major Nigerian languages (Igbo, Hausa, Yoruba, or Pidgin) into the input field. This input represents the text they want to analyze for sentiment.

User Clicks the Sentiment Button: After entering the text, the user clicks on the "Sentiment" button. This action sends the input text to the system for analysis.

System Displays Sentiment Result: The platform processes the text using the OpenAI API, classifies the sentiment as positive, negative, or neutral, and displays the result on the same page. The user sees the sentiment analysis result in real-time, enabling them to interpret the sentiment of their input text instantly.

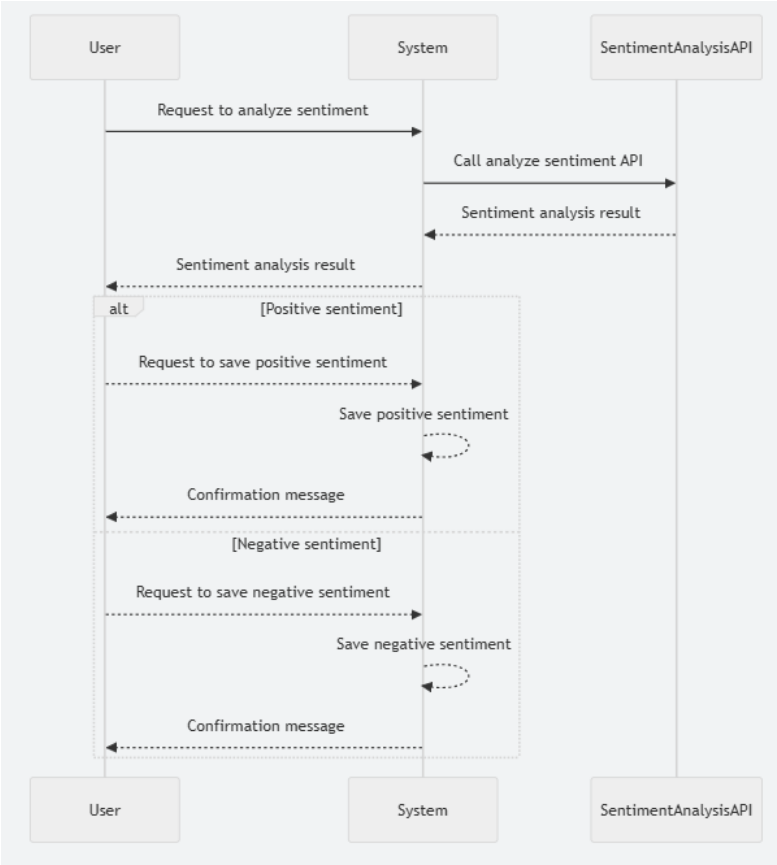

Fig 3. Sequence Diagram

Implementation

System Development

The development of the sentiment analysis using natural language processing in Nigeria's four major languages: Igbo, Hausa, Yoruba, and Pidgin incorporated vanilla JavaScript alongside elements API key from Open AI, a widely recognized tool for vision testing. By integrating natural language processing principles, the system enhances its diagnostic capabilities to provide an engaging and interactive user experience. The application evaluates users' vision acuity through a series of visual challenges, requiring them to identify the largest or smallest letters displayed on the screen.

System Architecture and Components

The sentiment analysis web application developed in this project is built using standard front-end technologies like HTML, CSS, and vanilla JavaScript. The core logic integrates the Google Gemini API to perform sentiment analysis on user-inputted text, providing feedback based on the detected sentiment (positive, negative, or neutral).

Loading Screen

Upon opening the application, the user is greeted with a loading screen that ensures that all resources are fully loaded before any interaction. This is crucial to ensure that the application behaves smoothly without user interface issues. After a short-simulated delay, the loading screen fades out, transitioning the user into the main interface.

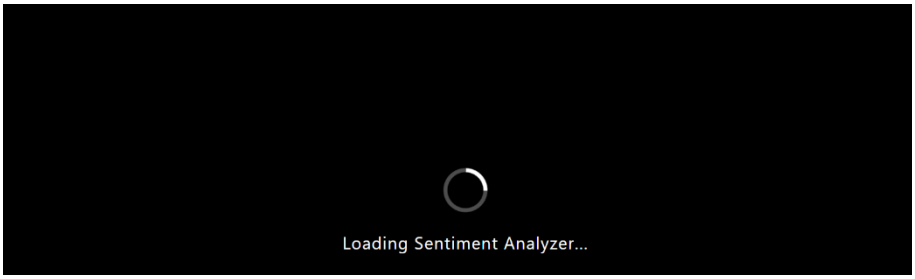

Fig 4. System Loading

User Input Interface

The primary interaction with the application happens through a text input area where users can type or paste any text they want to analyze. This text area allows for real-

time validation if the user leaves it blank, the "Analyze" button is disabled, guiding the user to provide a valid input.

Additionally, there is a language selection dropdown that enables users to choose the language of the input text. The interface also includes buttons for analyzing the text, clearing the input field, and managing the analysis history.

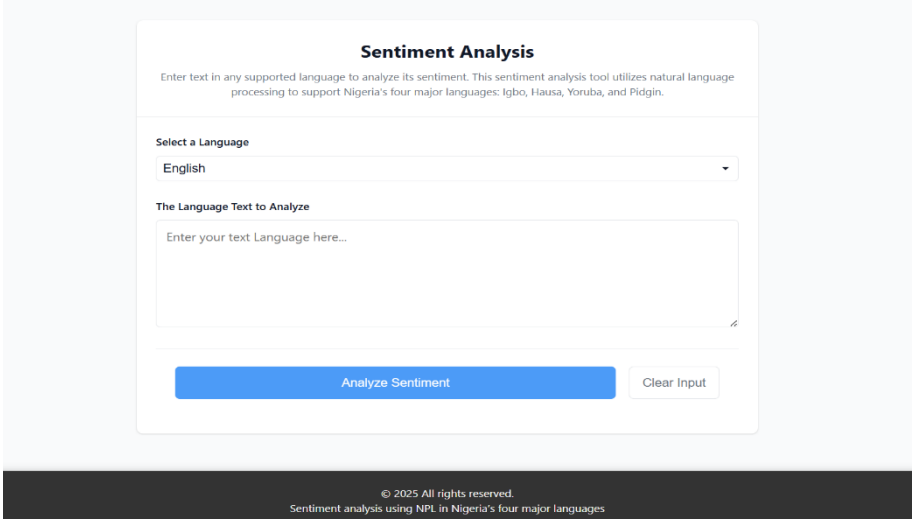

Fig 5. System interface

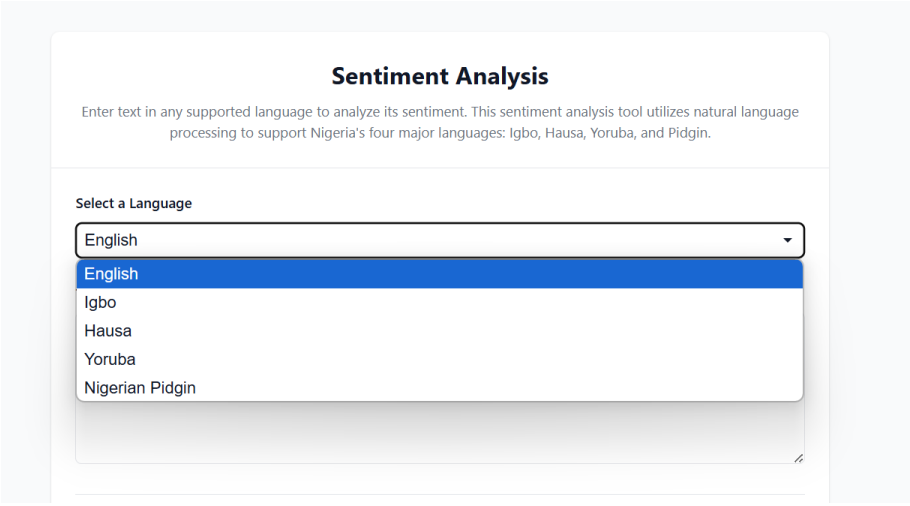

Fig 6. Language Selection

API Key Management

The app lets users safely store their Google Gemini API key right in their browser, so they don't have to keep entering it. Once a valid key is saved, the analysis button lights up, making for a seamless user experience. It connects to the Google Gemini API to

perform sentiment analysis, sending a prompt that helps classify the sentiment of the text you input. The response comes back with a sentiment score, a confidence level, and a short explanation to help clarify the reasoning behind it.

Some key principles include securely storing the API key, crafting well-structured prompts for accurate results, and implementing effective error handling to deal with issues like invalid keys or unexpected responses.

Event Listeners

Multiple event listeners are used to handle user actions and interface updates. These include:

Analyze Button Click: When clicked, the application triggers the sentiment analysis process, sending the user's input and selected language to the Google Gemini API.

Clear Input: The clear button empties the text area, resets the result section, and disables further analysis until new input is provided.

Clear History: Clears all past sentiment analyses saved in the local storage, providing the user with a fresh history.

Input Validation: Monitors the text area, enabling or disabling the analyze button depending on whether the input text is valid.

Sentiment Analysis Process

The core of the sentiment analysis process revolves around sending a user's input to the Google Gemini API. Once triggered, the application sends a request containing the input text and the selected language to the API. The API then processes this request, analyzing the sentiment of the text, and returns a structured response indicating the sentiment (positive, neutral, or negative), the confidence score, and a brief explanation of the result.

Displaying Results

Once the API response is received, the application updates the user interface with the result. Key components of the result display include:

Sentiment Badge: A badge that indicates whether the sentiment is positive, neutral, or negative.

Confidence Score: A percentage value representing how confident the API is about the sentiment classification.

Sentiment Icon: A visual icon corresponding to the detected sentiment, offering a quick, intuitive understanding of the result.

Explanation: If provided, a textual explanation is displayed, explaining the reasoning behind the classification.

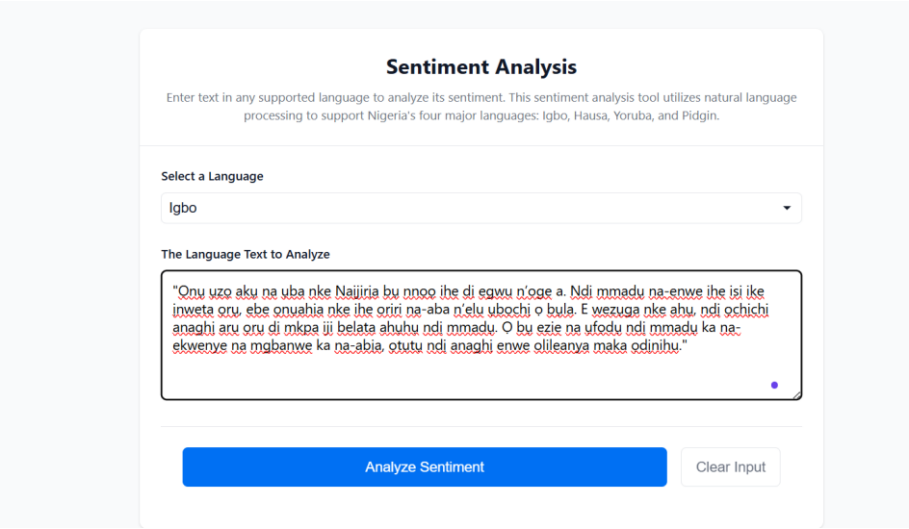

Fig 7. Igbo Language

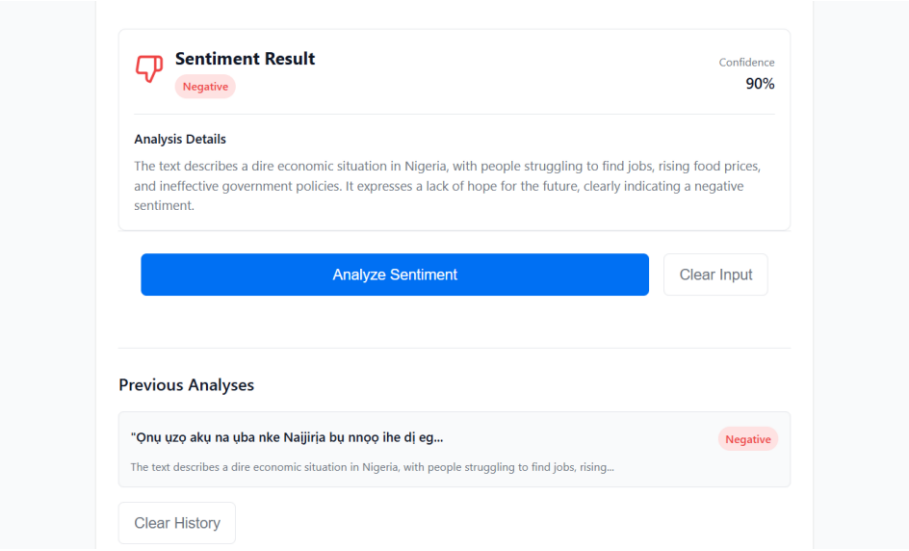

Fig 8. Igbo Sentiment Result

Fig 9. Hausa languge

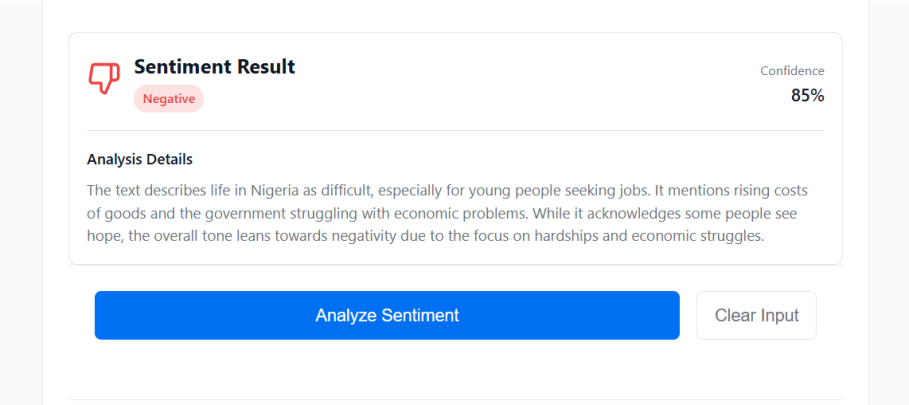

Fig 10. Hausa Result

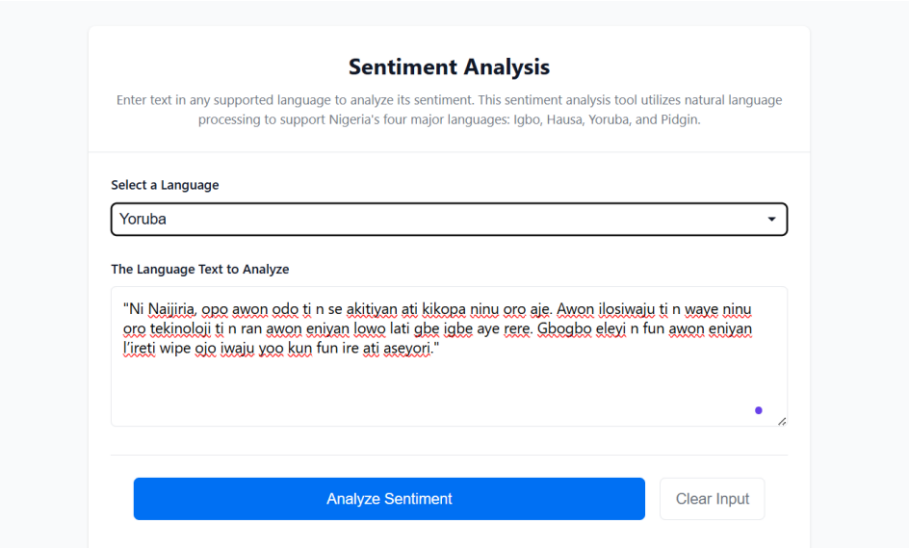

Fig 11. Yoruba Language

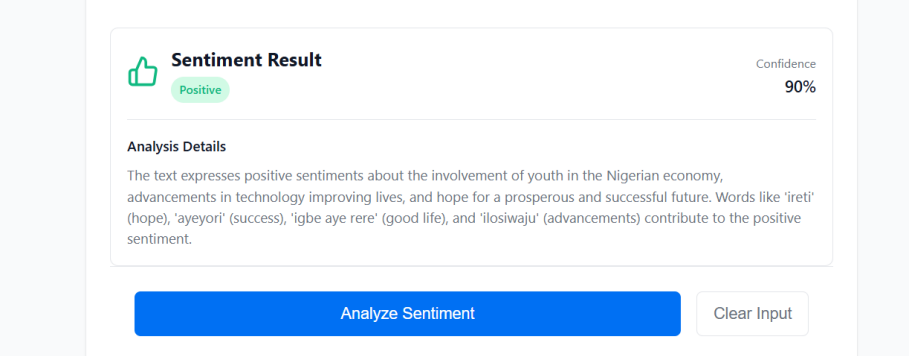

Fig 12. Yoruba Language Result

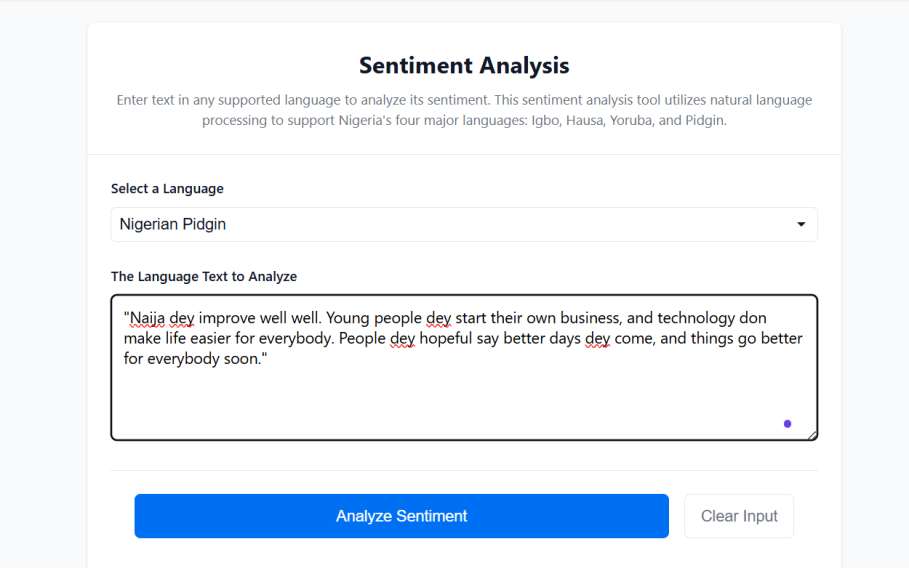

Fig 13. Pidgin Language

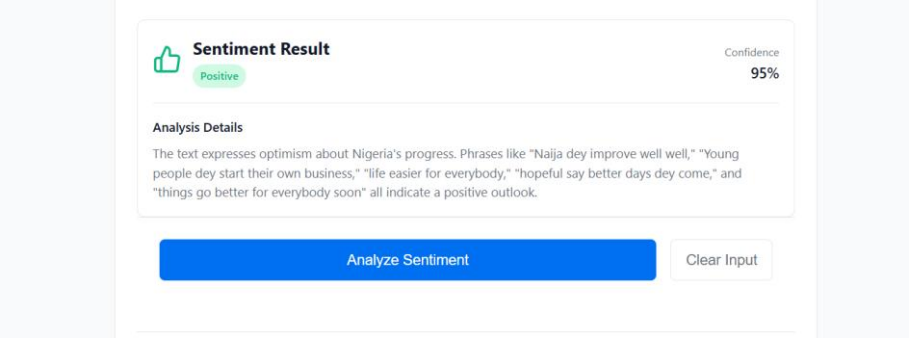

Fig 14. Pidgin Result

History Management

To enhance user experience, the application automatically saves the results of each sentiment analysis in the browser's local storage. This allows users to view a history of previous analyses. The history section displays key information like the first 50 characters of the analyzed text, the sentiment, and a short summary of the explanation provided.

Clicking on a history item automatically re-populates the input area for re-analysis, enabling an efficient workflow for repeated tests.

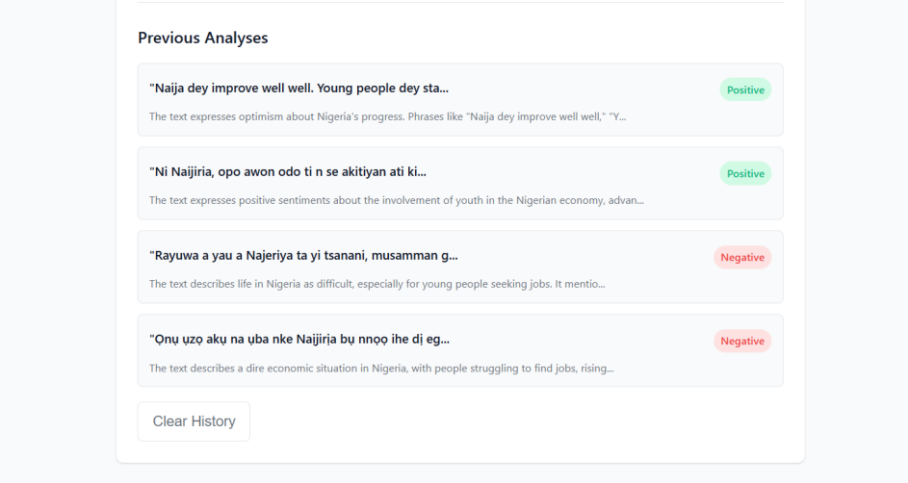

Fig 15. History

System Testing and Evaluations

The application has robust error handling to provide feedback when issues occur, such as missing API keys, by displaying error messages to help users troubleshoot quickly. Extensive testing was done to ensure reliability and performance. Key testing methods include:

• **Functional Testing**: Checked key features like sentiment analysis, API integration, text input validation, API key management, and language handling for smooth operation.

• **Usability Testing**: Evaluated the user interface for simplicity and ease of use across varying technical skill levels, focusing on data input and result clarity.

• **Responsiveness Testing**: Tested across smartphones, tablets, and desktops to ensure seamless functionality on different screen sizes and orientations.

Summary

This study focused on performing sentiment analysis for Nigeria's four major languages Igbo, Hausa, Yoruba, and Pidgin by utilizing Natural Language Processing (NLP) techniques. One of the key challenges tackled was the rich linguistic diversity, cultural subtleties, and the scarcity of annotated datasets, which often hinder the effectiveness of sentiment analysis models that are usually tailored for English. To address this, we developed specialized methods, gathered and processed a variety of datasets, and

created an open-source sentiment analysis application aimed at enhancing both accuracy and accessibility for these languages.

The research underscored the significance of multilingual sentiment analysis in aiding decision-making across different sectors, including business, policy, and research. By advancing the capabilities of NLP for Nigerian languages, the study sought to bridge a notable gap in current sentiment analysis models. The creation of this application serves as a practical tool to foster understanding and engagement with native languages, making sentiment analysis more inclusive and reflective of Nigeria's diverse linguistic landscape. Even though the research achieved its primary objectives, it encountered challenges like dialectal variations and the limited availability of annotated data. While the system successfully showcased the potential for conducting sentiment analysis in Nigeria's major languages, these challenges emphasize the necessity for further research and the expansion of datasets to enhance the system's robustness and accuracy across various dialects and language differences.

CONCLUSION

The development of the sentiment analysis application has demonstrated that integrating NLP with local languages in Nigeria is achievable, though challenging. This project fills a significant gap in the sentiment analysis field, which often overlooks indigenous languages, by providing a functional and user-friendly tool capable of performing sentiment analysis for Igbo, Hausa, Yoruba, and Pidgin. The use of the Model-View-Controller (MVC) architecture allowed for a well-organized and scalable platform, while the integration of the Google Gemini API enabled accurate sentiment classification. Despite facing challenges like dialect differences and limited annotated datasets, the system performed well, offering valuable insights for various stakeholders, including businesses, policymakers, and researchers. Overall, this research has contributed to the ongoing development of multilingual sentiment analysis, particularly for underrepresented languages in the NLP space.

Recommendations

Based on the findings and outcomes of this study, the following recommendations are made:

1. **Expansion of Datasets**: The creation and expansion of annotated datasets for Nigerian languages are crucial to improving the accuracy of sentiment analysis models. Collaborations with linguists, native speakers, and academic institutions could enhance the quality of datasets.

2. **Incorporating Dialectal Variations**: Future versions of the sentiment analysis platform should account for the various dialects within the four languages to ensure more accurate sentiment classification. This could involve dialect-specific models or additional training data.

3. **Application in Real-World Scenarios**: The sentiment analysis system could be extended to real-world applications, such as monitoring social media trends, public sentiment analysis for political campaigns, and customer feedback systems for businesses that cater to multilingual populations in Nigeria.

REFERENCES

Adebara, I., & Abdul-Mageed, M. (2022). *Sentiment analysis in Nigerian languages: Challenges and opportunities*.

Adeleke, O., & Oluwaseun, A. (2019). *Sentiment analysis in Yoruba language: Challenges and solutions*.

Adeleke, O., & Oluwaseun, A. (2021). *Sentiment analysis in Yoruba: A deep learning approach*.

Adelani, D., Akinyemi, S., & Buhari, S. (2021). *Low-resource NLP for African languages*.

Adeyanju, A. (2021a). Natural language processing for Yoruba and Igbo: A review. *African Journal of Computer Science Research, 14*(2), 45–56.

Adeyanju, A. (2021b). *Natural language processing in Nigeria: Challenges and opportunities*.

Adeyemi, B., & Ojo, A. (2021). *Sentiment analysis in Nigerian languages: A deep learning approach*.

Alabi, J., Sulaiman, U., & Dada, K. (2020). *Sentiment analysis in Hausa: A machine learning approach*.

Amadeo, R. (2022). *The pros and cons of MVC architecture*.

Buchanan, J. (2019). *Reusability in software development: A case study of MVC*.

Cambria, E., Hussain, A., & White, B. (2020). *Advances in natural language processing*. Springer.

Campbell, J. (2021). *Nigeria: Dancing on the brink*. Rowman & Littlefield.

Chike, U., Okafor, C., & Ifeoma, N. (2023). *Igbo diaspora and language preservation*.

Devlin, J., Chang, M.-W., Lee, K., & Lee, K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*.

Ezeani, I., Obasi, C., & Okoye, I. (2022). *Code-switching in Nigerian languages: Implications for NLP*.

Falola, T., & Heaton, M. (2008). *A history of Nigeria*. Cambridge University Press.

Idris, A. (2020). *Pidgin English in Nigerian media and entertainment*.

Izuagbe, R. (2022a). *Cultural nuances in sentiment analysis: A study of Nigerian languages*.

Izuagbe, R. (2022b). Exploring sentiment analysis in Pidgin English. *International Journal of Linguistics, 14*(3), 55–70.

James, C., Ibe, C., & Anyanwu, U. (2019). *Nigerian Pidgin: A unifying language*.

Jurafsky, D., & Martin, J. H. (2021). *Speech and language processing*. Pearson.

Kakwagh, V., & Ogu, M. (2021). *Multilingual language models for African languages*.

Lee, L., & Pang, B. (2008). *Opinion mining and sentiment analysis*.

Liu, B. (2015). *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press.

Mavula, J., Chia, E., & Moussa, H. (2022). *Code-switching in multilingual sentiment analysis*.

Medhat, W., Hassan, A., & Korashy, H. (2014). *Sentiment analysis algorithms and applications: A survey*.

Muhammad, S. H., Yusuf, M., Salihu, A., & Abubakar, F. (2022). Sentiment analysis in low-resource languages: Twitter sentiment corpus in multilingual Igbo, Hausa, Yoruba, and Pidgin languages. *Naija-Senti, 15*(1), 591–597.

Muhammed, A., & Idris, M. (2020). *Hausa language and its role in West African trade*.

Musa, A., & Abdullahi, M. (2018). *Sentiment analysis in the Hausa language: A machine learning approach*.

Musa, A., & Abdullahi, M. (2021). *Sentiment analysis in Hausa: A deep learning approach*.

National Population Commission of Nigeria. (2022). *Nigeria demographic statistics*.

Nwankwo, C., & Eze, P. (2020). *Sentiment analysis in Igbo language: A rule-based approach*.

Nwankwo, C., & Eze, P. (2021a). *Sentiment analysis in Igbo: A deep learning approach*.

Nwankwo, C., & Eze, P. (2021b). *Sentiment analysis in Igbo: A hybrid approach*.

Odebiyi, J., & Adetunji, A. (2020). *Multilingual sentiment analysis: A case study of Nigerian languages*.

Odebiyi, J., & Adetunji, A. (2021). *Sentiment analysis in Nigerian languages: A hybrid approach*.

Odebiyi, J., & Ojo, A. (2020a). *Sentiment analysis in multilingual contexts: A case study of Nigerian languages*.

Odebiyi, J., & Ojo, A. (2020b). Sentiment analysis in Nigerian languages: Challenges and opportunities. *Journal of Language and Linguistic Studies, 16*(1), 1–15.

Ojo, A., & Adeyemi, B. (2021). *Sentiment analysis in Nigerian languages: A comparative study*.

Okonkwo, C., & Eze, P. (2020). *Sentiment analysis in Nigerian Pidgin: A machine learning approach*.

Okonkwo, C., & Eze, P. (2021). *Sentiment analysis in Nigerian Pidgin: A deep learning approach*.

Oluwaseun, A., & Hammed, B. (2023). *Hausa as a lingua franca in Nigeria*.

Orife, I., Ngum, E., & Ikwu, C. (2020). *Low-resource NLP for African languages: A review*.

Roy, T., Kurma, S., & Ogun, A. (2020). *Model-view-controller (MVC) architecture: A comprehensive guide*.

Sammy, H. (2020). *The cultural significance of Nigerian Pidgin*.

Shamsuddeen, A., Muhammed, A., Yusuf, I., & Musa, H. (2022). *Sentiment analysis in Pidgin English: A preliminary study*.

Singh, R. (2019). *Software design patterns: MVC and beyond.*

Taboada, M., Brooke, J., & Tofiloski, M. (2011). *Lexicon-based methods for sentiment analysis*.

Ugo, C., & Emeka, O. (2023). *The Igbo language: Cultural and linguistic perspectives*.

World Bank. (2023). *Nigeria economic outlook*.

Appendix

Table 1 Negative Language Data

Language	Paragraph
Igbo	"Ọnụ ụzọ akụ na ụba nke Naịjirịa bụ nnọọ ihe dị egwu n'oge a. Ndị mmadụ na-enwe ihe isi ike ịnweta ọrụ, ebe ọnụahịa nke ihe oriri na-aba n'elu ụbọchị ọ bụla. E wezụga nke ahụ, ndị ọchịchị anaghị arụ ọrụ dị mkpa iji belata ahụhụ ndị mmadụ. Ọ bụ ezie na ụfọdụ ndị mmadụ ka na-ekwenye na mgbanwe ka na-abịa, ọtụtụ ndị anaghị enwe olileanya maka ọdịnihu."
Hausa	"Rayuwa a yau a Najeriya ta yi tsanani, musamman ga matasa da ke neman ayyukan yi. Farashin kaya yana ta hauhawa, yayin da gwamnati ke ci gaba da fama da matsalolin tattalin arziki. Duk da haka, wasu mutane na ganin akwai fata a gaba, yayin da wasu suka daina samun kwarin gwiwa a kan abin da zai biyo baya."
Yoruba	"Ipinle oro aje Naijiria ti dinku pupo ni asiko yii. Aye nira fun awon eniyan lati ri ise, ati pe owo ti nwon n lo fun ohun mimu ati ounje ti po ju ti tele. Bi o tilẹ jẹ pe awon kan n reti pe aye yoo dara si, opo eniyan ti padanu ireti nipa ojo iwaju."
Pidgin	"Life for Naija no easy at all. People dey suffer, no work, food prices dey go up every day. Government no fit do wetin dem promise, and e be like say things dey worse. Some people still get hope say better days go come, but many don lose hope for the future."

Table 2 Positive Language Data

Language	Positive Paragraph
Igbo	"Naịjirịa nọ n'usoro ịzụlite akụ na ụba ya. Ndị na-eto eto na-amalite ọtụtụ azụmahịa ọhụrụ, na usoro teknụzụ na-eme ka ihe dịkwuo mfe. Ọtụtụ ndị mmadụ na-enwe olileanya na, n'ime afọ ndị na-abịa, mgbanwe ga-abịa ma ndị mmadụ ga-enweta ohere zuru oke ịga n'ihu n'akụ na ụba."
Hausa	"A Najeriya yanzu, matasa da yawa suna kafa sabbin sana'o'i kuma suna samun ci gaba. Ci gaban fasaha ya kara saukaka rayuwa. Yanzu mutane na da fata sosai cewa tattalin arziki zai bunkasa, kuma za a samu ci gaba mai dorewa a nan gaba kadan."

Language	Positive Paragraph
Yoruba	"Ni Naijiria, opo awon odo ti n se akitiyan ati kikopa ninu oro aje. Awon ilosiwaju ti n waye ninu oro tekinoloji ti n ran awon eniyan lowo lati gbe igbe aye rere. Gbogbo eleyi n fun awon eniyan l'ireti wipe ojo iwaju yoo kun fun ire ati aseyori."
Pidgin	"Naija dey improve well well. Young people dey start their own business, and technology don make life easier for everybody. People dey hopeful say better days dey come, and things go better for everybody soon."