Sentiment Analysis in Domain-Specific X-Tweets Using Support Vector Machine and Long Short-Term Memory with TF-IDF Vectorizer

Babatunde A.N.[1], Kadri A.F.[2], Isiaka O.S.,[3] Shuaib B.M.,[4] Abdulrahman, T.A.[5] Balogun B.F.[6], Ajiboye, R.A.[7] & Oke A.A.[8]
[1, 2, 3, 4, 6]Department of Computer Science, Kwara State University, Malete.
[3, 5]Department of Computer Science, Kwara State Polytechnic, Ilorin.
[7]Department of Statistics, Kwara State Polytechnic, Ilorin.
[8]Department of Computer Science, Federal College of Education, Iwo.
Corresponding author: akinbowale.babatunde@kwasu.edu.ng,

Received: February 14, 2025, Accepted: April 28, 2025

Abstract: Interest in sentiment analysis has grown as user comments on social media have proliferated on platforms. The hybridized machine learning framework used in this paper focuses on the classification of user feedback on tweet post. Other researchers have applied machine learning models to address this issue, but their models struggle with the complex sentiment nuance and achieve low prediction accuracy. To close this disparity, we used the powerful classification ability of support vector machine (SVM) in conjunction with LSTM (Long-short-term-memory) to classify tweet sentiment and Term Frequency-Inverse Document (TF-IDF) Vectorizer for converting tweets data into numeric values. This experiment used the 1.5 million tweet sentiment 140 dataset available on Kaggle. The SVM-LSTM hybrid model developed in this study demonstrated 99% accuracy on both the training and test data. We used another state-of-the-art metric, ROC-AUC. This model achieved superhuman performance and outperformed many baseline models in NLP. This paper's novelty is achieving a high-performing model above many baseline models and specifically targeting nuances in sentiment. This demonstrates the model's effectiveness at classifying tweet sentiment, especially for domain-specific applications such as social media sentiment analysis.

Keywords: Sentiment analysis, tweets, domain-specific, Long-short-term-memory, Support vector Machine, Term Frequency-Inverse Document (TF-IDF) Vectorizer.

Introduction

Sentiment Analysis (SA) also known as opinion mining is a branch of Natural Language Processing (NLP) which examines text data to ascertain the sentiment or emotional tone that underlies the words. With the growth of social media and online reviews, this technology has become increasingly popular (Jain et al., 2020). A wealth of user-generated content offers insightful information on consumer mood and public opinion. Sentiment analysis's main objective is to categorize text into pre-established sentiment groups, such positive, negative, or neutral. For companies, governments, and researchers to comprehend public opinion and make defensible decisions based on sentiment patterns, this procedure is essential (Fong, 2023)

Sentiment analysis typically categorizes sentiments into three main types: positive, negative, and neutral. Positive sentiment reflects favorable, happy, or approving emotions, often indicating satisfaction or pleasure. Negative sentiment encompasses unfavorable, unhappy, or critical emotions, highlighting dissatisfaction or displeasure (Jain et al., 2020; Fong, 2023). Neutral sentiment represents a lack of strong emotional tone, often indicating objective or factual statements without clear emotional bias. Additionally, more nuanced classifications can include mixed sentiments, where text conveys both positive and negative emotions (Fong, 2023), and specific emotional states such as anger, joy, sadness, and surprise, offering a more detailed understanding of the underlying sentiments.

According to (Fong et al., 2023) the exponential growth of digital communication has led to a massive influx of textual data generated daily across various domains, including social media, online reviews, forums, and professional fields such as healthcare and finance (Fong et al., 2023). This wealth of textual information presents an invaluable resource for extracting insights into public opinion, consumer behaviour, and expert analysis.

The recent advancement in technology as presented machine learning (ML) a branch of AI (Artificial Intelligence) and Deep learning (DL), a subset of machine learning in solving complex real life challenges including NLP (Fong, 2023). Machine learning is the ability of a computer machine to be able to perform specific functions on its own after being trained by humans (Jain et al., 2020; Fong, 2023). It involves the use of algorithms and statistical models to identify patterns and make decisions based on data. Common machine learning techniques include supervised learning, where models are trained on labeled data, and unsupervised learning, which involves identifying patterns in unlabeled data. Applications range from image and speech recognition to predictive analytics and recommendation systems. Deep learning in the other hand is a specialized branch of machine learning that uses neural networks with many layers (hence "deep") to model complex patterns in large datasets (Fong et al., 2023; Babatunde et al., 2022). These neural networks, often called deep neural networks, can automatically learn feature representations from raw data, making them particularly effective for tasks such as image and speech recognition, natural language processing, and autonomous driving (Ahmed et al., 2023; Babatunde et al., 2021).

In this paper, we present the implementation of an hybrid model for classifying users' opinions on Tweeter post using Recurrent Neural Network (RNN) model, specifically Long Short-Term Memory (LSTM), and a Support Vector Machine (SVM) with sentiment140 dataset. We developed a system that combines SVM with LSTM to effectively classify users' opinions on Tweeter posts. This research offers the following contributions to the existing NLP studies on sentiment analysis:

i. Helps deal with the issue of sentiment nuances which is a major setback in NLP
ii. Presents an advanced sentiment analysis model with improved accuracy that surpasses many baseline models.

The remaining part of this paper is structured into the following sections: Section 2 explains the existing works related to the present study. The section 3 contains description about the dataset and methodology applied in building the proposed system. The results of this experiment are presented in section 4. Section 5 presents the discussion of the findings with conclusion and future works.

Literature Review

Sentiment analysis has been a domain of interest for many scholars, as such, many papers have been presented to address the setbacks in analyzing users' feedback on social media platforms such as Facebook, Instagram, Tweeter now known as X, etc. In this paper we critically reviewed existing studies in order to examined what was done and what limitations exist in these studies. In this section, we present the review of some studies related to this research work.

Oyewola et al. (2023) carried out research on how the public feels about Atiku Abubakar, Bola Tinubu, and Peter Obi (the three front-runners in Nigeria's 2023 presidential election). With an emphasis on candidate mentions, researchers gathered a sizable sample of election-related tweets. To ensure data quality for analysis, the tweets were preprocessed to eliminate noise and unnecessary information. For sentiment categorization, three models were used: Two-Stage Residual LSTM (TSRLSTM), Peephole LSTM (PLSTM), and Long Short-Term Memory (LSTM). The goal of each model was to categorize tweets about the candidates as either favorable, neutral, or negative. When it came to sentiment classification tasks, the TSRLSTM model performed better than the others. The results showed that the public had different opinions on each candidate, providing information about how the public views them. The study demonstrates how well sophisticated deep learning algorithms can extract complex sentiments from social media data. It emphasizes how crucial sentiment analysis is to comprehending voter sentiment and electoral processes. By presenting a unique use of TSRLSTM in political attitude analysis, the study advances the discipline. It offers a methodological foundation for sentiment analysis in diverse political situations that may be modified. The study highlights how social media may be a valuable source of information for assessing popular sentiment during elections. Additionally, it highlights how sentiment research may be used to guide political decision-making and campaign tactics. According to the authors, their method can improve sentiment analysis's precision in next political research. Summarily, the research offers valuable insights into the application of deep learning techniques for political sentiment analysis in the digital age.

Patel et al. (2022) examines the use of K-Nearest Neighbors (KNN) algorithms and Long Short-Term Memory (LSTM) networks in sentiment analysis applied to domain-specific literature. The report highlights the value of sentiment analysis in a number of industries, including social media, healthcare, and finance. It draws attention to how LSTM can identify long-term relationships in textual data, which makes it appropriate for sentiment analysis in intricate, domain-specific settings. The convenience of use of a more straightforward technique, KNN, in smaller datasets is explored. After comparing the accuracy of LSTM with KNN, the authors conclude that LSTM frequently performs better than KNN, particularly in big datasets with complex patterns. There is also discussion of issues like data imbalance and the requirement for domain-specific feature engineering. The paper provides insights into the future direction of sentiment analysis research, recommending the integration of hybrid models and more refined preprocessing techniques. Babatunde et al., 2024 presented a Yoruba language translation in real-time using text-to-speech synthesized translation (TTS) and speech-to-text translation on Android smartphones. The system integrates both NLP and a TTS interpretation approach. It employs a Yoruba-to-English translation recognizer and synthesizer for the Android Google translation API. Their project aimed to create a feature-rich text-to-speech and speech-to-text SMS translator from English to Yoruba. The authors also introduced an offline android application for translating text SMS and speech in English to Yoruba. The program uses words from a locally created dictionary (database) to translate English sentences into Yoruba. For the present study, Android studio Integrated Development Environment (IDE) was used to build and test the system before packaging it as an APK file for installation on different Android phones and versions. This method of translation (one to one mapping and one to many mapping) was applied to both words and phrases from English to Yoruba (Babatunde et al., 2021). The assessment was split into two parts, the evaluation of program strategy, and the evaluation of how well the translations matched user expectations. One hundred people were invited to complete a translated questionnaire, and eighty-one complied. The responses were put through SPSS. The most accurate and persuasive by the quality and quantity of their generated computation rules was human conversion.

Wongkar & Angdresey (2019) focuses on employing sophisticated feature engineering and dynamic architecture in artificial neural networks to analyze brand-related emotions on Twitter. The authors suggest a unique method to increase the precision of sentiment analysis on brand-related tweets by combining dynamic artificial networks with well-designed feature engineering approaches. They emphasize how crucial preprocessing techniques like stemming, tokenization, and stop word removal are to getting the data ready for analysis. Because of its dynamic architecture, the model may adjust and get better with fresh data, which makes it appropriate for real-time brand monitoring applications. By contrasting several machine learning models, the study shows how well the suggested dynamic artificial network handles the intricacies of social media data. According to the results, the approach performs better than conventional models in terms of recall, precision, and classification accuracy. Additionally, the study highlights how important feature selection and dimensionality reduction are to improving model performance. In order to improve decision-making in brand management, they proposed future research paths in examining deep learning approaches that can enhance model interpretability. Zhang et al. (2023) introduced a hybrid LSTM-SVM model for product review sentiment analysis. The model incorporates SVM for final sentiment classification and LSTM for deep feature extraction. Review texts' temporal and contextual relationships are well captured by LSTM. SVM improves classification accuracy, particularly for strongly held opinions. The strategy seeks to strike a compromise between the advantages of conventional machine learning and deep learning. Tests conducted on a variety of product review datasets show increased F1-scores and accuracy. In the majority of test cases, the hybrid model performs better than the solo LSTM and SVM models. Before training the model, preprocessing techniques including tokenization and word embedding were used. The approach is appropriate for consumer feedback analysis and e-commerce according to the authors. Incorporating attention strategies and managing noisy data were potential challenges for the future.

Fong et al., (2023) presented an experiment that compares the possibilities of using different filters in the training dataset during preprocessing to find the factors that influence the algorithms of SA. The results showed that in sentiment analysis, the sentiment trending words have some influence on the outcome of the prediction. After removing the high frequency words, the accuracy of the prediction results decreases, especially for unique high frequency words of each class. Ahmed et al. (2023) applied TF-IDF, random forest (RF), Naïve Bayes (NB), and feature extraction to fake news articles that were compiled. Their study implemented a model to predict Fake news article as positive or negative post. Among the individual ML, the NB was the best model and achieved the highest accuracy (89.30%). Govindarajan et al., 2023, applied Naïve Bayes (NB) Classifier with TF IDF feature extraction to classify Twitter sentiment140 dataset. Based on performance metrics including precision, recall, and accuracy, the suggested model's results in Govindarajan et al., 2023 showed the overall accuracy of (84.44%) with a precision score of (83.39) and recall score of (84.30). The results of the model presented in Ahmed et al. (2023) outperformed the one presented by Govindarajan et al., 2023 after comparing the result of the models. Patria et al. (2024) used Naïve Bayes and Support Vector Machine to examine popular opinion toward the 2024 Indonesian presidential election. Classifying feelings conveyed in tweets from the social media site X (previously Twitter) was the main goal of the study. 2,193 tweets on the 2024 Indonesian presidential election were collected by the researchers. They used preprocessing techniques like tokenization, stop-word removal, and stemming to get the data ready for analysis. Textual information was transformed into numerical characteristics appropriate for machine learning models using the Term Frequency-Inverse Document Frequency (TF-IDF) technique. The dataset's class imbalance was addressed using the Synthetic Minority Over-sampling Technique (SMOTE). The study evaluated the effectiveness of two machine learning algorithms: Support Vector Machine (SVM) and Naïve Bayes Classifier (NBC). The accuracy of the Support Vector Machine was 62.19%, whereas the accuracy of the Naïve Bayes Classifier was 62.41%. According to these findings, both models classified public opinion as neutral, negative, or positive with comparable accuracy. The study demonstrates how machine learning methods may be used to analyze social media sentiment and offer insightful information about public opinion during political events. To improve sentiment categorization performance, further study and the use of more sophisticated models are necessary, as indicated by the low accuracy ratings.

Poornima & Priy, (2020) extracted data relating to COVID-19 from Twitter users in England's major cities. This study compares machine learning models as its main objective, such as multinominal Naïve Bayes (MNB), Random Forest (RF), and support vector classifier (SVC), with lexicon-based approaches such as Vader and Textblob using two feature extraction methods (Word2Vec embedding and TF-IDF). Overall, SVC with TF-IDF had better accuracy of (91.02%) which outperformed other models. Al Sari et al., (2022) created three different datasets from social media platforms to analyze the impressions about Saudi cruises. The study involved creating datasets from three social media platforms Instagram, Snapchat, and Twitter to analyze public impressions of Saudi cruises. They applied various machine learning algorithms, including Multilayer Perceptron (MLP), Naïve Bayes (NB), Voting Classifier (VC), Support Vector Machine (SVM), and Random Forest (RF), utilizing the n-grams feature extraction technique. Notably, the Random Forest algorithm achieved 100% training accuracy and a 93% test accuracy, indicating potential overfitting. An LSTM-SVM hybrid model for sentiment analysis in product evaluations was proposed in the Wang et al. (2022) publication. It combines the classification power of Support Vector Machines (SVM) with the sequential learning capabilities of Long Short-Term Memory (LSTM). The model initially extracts deep features from textual data using LSTM. SVM is then used to classify these characteristics in order to increase their robustness and accuracy. The study highlights how well the algorithm captures contextual emotion in reviews. In terms of accuracy, recall, and F1-score, it performs better than conventional machine learning models. Benchmark datasets from product reviews were used for the experiments. Performance is improved over solo LSTM or SVM models, according to the results. The model's applicability for commercial sentiment analysis tasks is highlighted by the authors and are looking forward to extend the approach to multilingual and multi-domain applications in future development.

From the reviewed literature, it is evident that ongoing research continues to advance the field of Natural Language Processing (NLP), with particular emphasis on sentiment analysis. Many of these studies have demonstrated the successful application of Machine Learning (ML) and Deep Learning (DL) techniques to tackle the complexities of interpreting users' sentiments. However, despite the commendable performance of these models, certain limitations persist. Notably, issues such as low-test accuracy and model overfitting highlighted in the findings of Ahmed et al. (2023) remain prevalent. Furthermore, the subtle nuances inherent in human sentiment often hinder the ability of models to achieve precise classification. In response to these challenges, this study proposes a hybridized framework designed to improve accuracy and mitigate overfitting, while more effectively capturing sentiment complexities.

Methodology

To achieve the objectives of this research, a structured approach was adopted to streamline the system development process. The methodology designed for this study consists of the following stages: (1) Data acquisition and description, (2) Data preprocessing, (3) Feature selection, (4) Data sampling, (5) Model selection, (6) Model development, and (7) System performance evaluation. A graphical overview of this methodological framework is presented in Figure 1 using a flowchart.

Figure 1: *schematic representation of the proposed system methodology design*

Dataset Description and preprocessing

This experiment uses the Sentiment140 dataset with 1.6 million instances available from Kaggle. It is a famous sentiment analysis dataset. The training set is a dataset of tweets that are examples of positive, negative, or neutral. These sentiment labels are numerical: 0 is negative, 2 is neutral, and 4 is positive. The dataset included several features such as the target (sentiment label), the tweet ID (a unique identifier), the date (when the tweet was posted), the flag or query (keyword or hashtag used to find a tweet—typically set to NO QUERY), the user (twitter username) and the actual tweet content (text). To ensure a quality of data used during model training, text preprocessing was conducted. This is a critical advance for natural-language processing in order to clean up the words and make it possible to analyze the text. This process involved stripping out punctuation, converting all words to lowercase, removing stop words such as ‘and’ or ‘the,’ and so on. Some of the common spelling errors were fixed, and irrelevant symbols eliminated. This kind of cleaning helped to simplify the text and let the model discern the useful patterns. This helped clean up the dataset so that it was better able to extract features that would make the model accurate and efficient. Because Sentiment140 is big and has labeled structure, it is a great resource to train sentiment analysis models, which can learn from and predict emotional tones in a large amount of text data. Figure 2 depict the sample dataset employed for this study.

```
# loading the dataset for sentiment analysis
df
```

	target	ids	date	flag	user	text
0	0	1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_	@switchfoot http://twitpic.com/2y1zl - Awww, t...
1	0	1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by ...
2	0	1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Man...
3	0	1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire
4	0	1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli	@nationwideclass no, it's not behaving at all...
...	...	...	...	...	...	...
1599995	4	2193601966	Tue Jun 16 08:40:49 PDT 2009	NO_QUERY	AmandaMarie1028	Just woke up. Having no school is the best fee...
1599996	4	2193601969	Tue Jun 16 08:40:49 PDT 2009	NO_QUERY	TheWDBoards	TheWDB.com - Very cool to hear old Walt interv...
1599997	4	2193601991	Tue Jun 16 08:40:49 PDT 2009	NO_QUERY	bpbabe	Are you ready for your MoJo Makeover? Ask me f...
1599998	4	2193602064	Tue Jun 16 08:40:49 PDT 2009	NO_QUERY	tinydiamondz	Happy 38th Birthday to my boo of alll time!!! ...
1599999	4	2193602129	Tue Jun 16 08:40:50 PDT 2009	NO_QUERY	RyanTrevMorris	happy #charitytuesday @theNSPCC @SparksCharity...

1600000 rows × 6 columns

Figure 2: sample dataset employed for the experiment

Tokenization

This process involves a technique called a tokenizer, which breaks the text into individual sentences and each sentence into individual words. Tokenization helps convert raw text into phrases that a sentiment analysis model could interpret. To do so, we turned to Python's NLTK library. This broke each sentence into its component words, enabling the model to ponder each one independent, making it easier for it to understand the text's meaning and emotion. Tokenization is a key process when working with text so that the model can generate more predictable responses based on how words are used.

Stop Word Removal

Following tokenization, stop word removal was administered to extract common words that do not have much significance in sentiment analysis. These are some of the most common words in the language but do not contribute to the overall message of the text. These stop words were removed from the dataset using the Python library NLTK (Natural Language Toolkit). This eliminated non-essential words which resulted in narrower, and presumably more important terms for establishing sentiment. This preprocessing step served to attenuate noise, boost the efficiency and signal of the sentiment analysis model by complicating the model with only the most relevant words for training.

Lemmatization

Each word was lemmatized during this experiment so that related words were grouped together in the analysis. Lemmatization, in contrast, looks beyond word endings and considers a word in context, transforming it into its meaningful base form. This technique treats running, ran, and runs as instances of a single word, run. Lemmatization reduced words to their dictionary base form and made it easier to capture the meaning and sense of words. But this was critical to the accuracy of the model in properly understanding and interpreting the text sentiment. We apply the Python library NLTK (Natural Language Toolkit) to lemmatized the words in the data used for this experiment.

Removing URLs, Usernames, Handling Emoticons and Emojis

It was essential preprocessing the data while training models for this project to strip out URLs, usernames, and to deal with emoticons and emoji. The text was scrubbed of URLs and usernames so that the sentiment analysis would not be thrown off by irrelevant or non-informative features. An emoticon or emoji that has a reaction using a particular character that means something is not altered. For example, to replace a positive emotion, the smiley face :) became happy. The purpose of these preprocessing steps was to clean the corpus and to allow the model to concentrate on the text's essentials, making the hybrid sentiment analysis approach more accurate and efficient.

Feature Selection

In machine learning, feature selection involves choosing the features of a dataset that are most helpful in the task of predicting a target variable. During the course of this experiment, we apply the chi-square algorithm to spot out the most relevant feature for the model training. It includes the tweets as text, which are vital for sentiment analysis with natural language processing algorithms. The input feature in this case is the words in each tweet and the output feature is what is known as the target, the label that we are going to train on, which in this case is whether the tweet constitutes a positive or negative sentiment. The analysis excluded other features such as ids, date, flag, and user because they are either metadata and not text, or irrelevant to sentiment classification, or superfluous, or might generate too much model noise and complexity when included in training the model.

Data Sampling
The study we applied the same sentiment140 dataset that had 1.6 million labeled tweets. To conserve processing power and expedite model training, we applied a data sampling technique that randomly chose 30,000 of the instances in the dataset. This sample involved a variety of emotions and included 38 examples of those that were positive, negative, and neutral. The chosen examples were then used to train a pair of models: one supporting traditional machine learning techniques, using Support Vector Machines (SVM), and the other one supporting deep learning approaches, using Long Short-Term Memory (LSTM) networks, thus allowing the effectiveness and performance of each model to be examined and probed with respect to the sentiment analysis applications.

Text Vectorization
The next step was to use the TF-IDF Vectorizer (Term Frequency-Inverse Document Frequency) to turn the tweet texts into numerical data that a machine learning model can process. This takes the words in all the tweets and transformed them to numbers, based on a word ranking twice as unique as frequency. A tweet with a rare word in it scores a little higher than other tweets. This helps the model figure out which words are most important for determining sentiment. TF-IDF was a way to translate this cleaned text data into a format that was organized enough for machine learning and prediction.

Model Selection
The model selection phase used Support Vector Classifier (SVC) and Long-Short-Term Memory (LSTM) algorithms. SVC is good in linear classification tasks, whereas a type of Recurrent Neural Network called LSTM is good at capturing sequence dependencies, thus being good for analyzing temporal data like the text sequences used in sentiment analysis. The architecture of then SVM algorithm applied in this study is shown in figure 3.

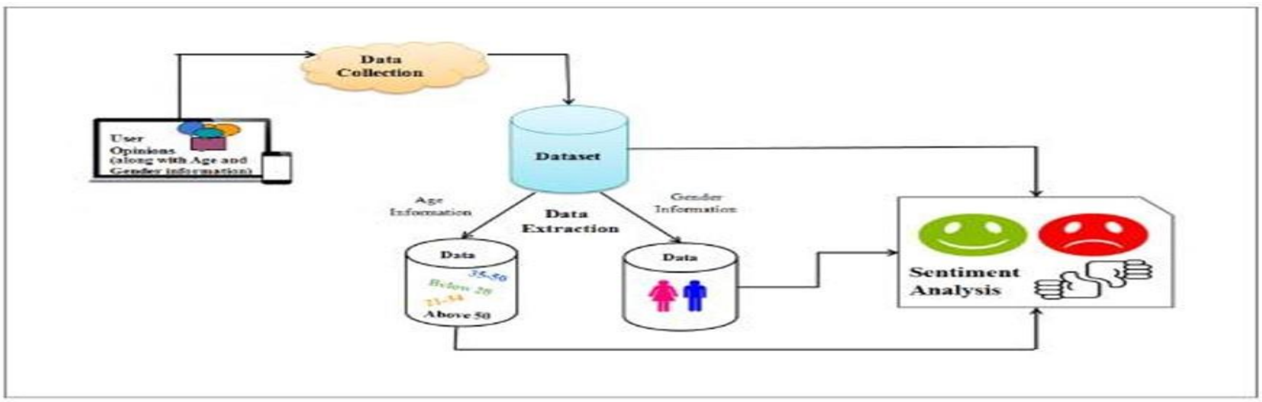


Figure 3 (a): structural representation of the SVM model architecture

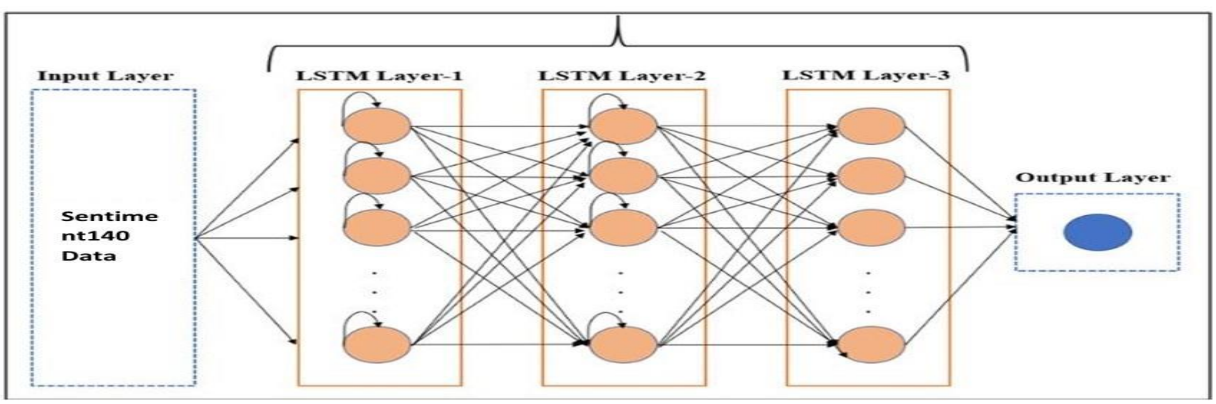


Figure 3(b): structural presentation of the LSTM model architecture

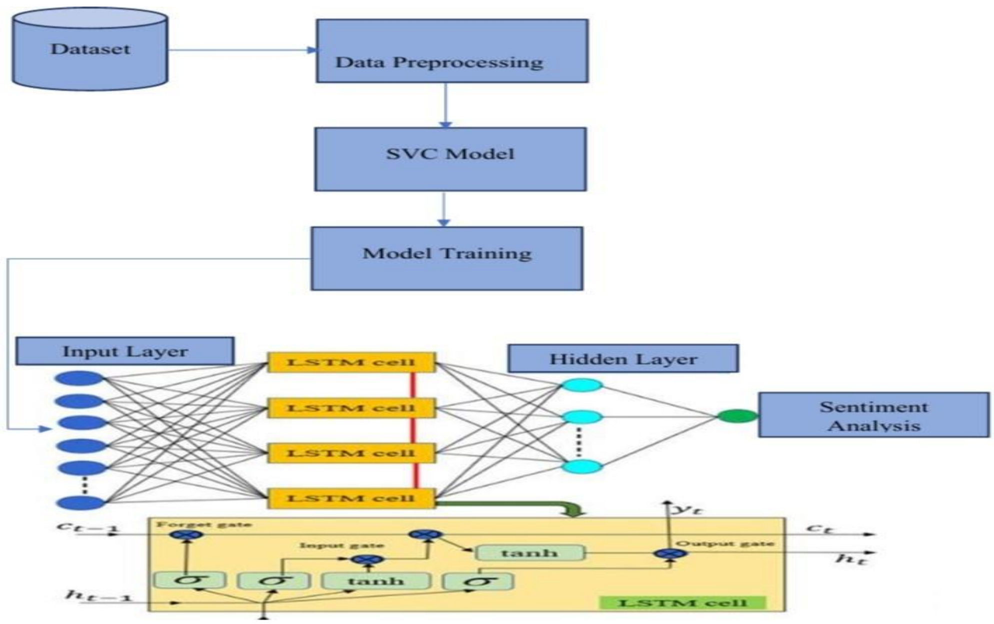


Figure 3(c): structural representation of the combined SVM and LSTM model

Figure 3(a) shows the support vector classifier design: Preprocessed text features are subjected to the TF-IDF process and are then categorized by identifying their linear separability in super dimensional space. Figure 3(b) shows the architecture of an LSTM, examples of digital gates, and how it can use word embeddings to capture temporal dependencies and the subtleties of context. The combination of these two types of networks is illustrated in figure 3(c) as a hybrid SVM-LSTM model that uses ensemble stacking. This hybrid called a Long Short-Term Memory (LSTM) pairs the accuracy of the classic SVM Classification technique with the sequence modeling of the more modern LSTM. By combining the results from both, the deep learning team is enabling a new, more accurate sentiment analysis that, Employee Apps claims, can express the full emotional intent of a wide range of documents, including various types of transmissions.

Model Development

The hybrid model proposed in the paper was developed using the Jupyter Notebook programming environment, which allows for an interactive coding, real-time output, and visualization. The Support Vector Classifier (SVC) was applied using Scikit-learn, with TF-IDF employed to convert the text data into numerical features for high-dimensional learning. A Long Short-Term Memory (LSTM) model was constructed using TensorFlow and Keras, optimized for learning the sequential structure in text. The LSTM was initialized with an input size of 100, the maximum sequence length of tokenized text inputs. The model employed the same 100 embedding dimension, 128 LSTM units, 0.2 dropout for regularization, and was trained with a batch size of 64 for 10 epochs. The Adam optimizer and binary cross-entropy loss function, typical for binary sentiment classification tasks, were used to compile the model. For efficient data manipulation and preprocessing, Pandas and NumPy were employed, and for visualizing model performance, e.g., loss curves and accuracy plots, we utilized matplotlib and seaborn python libraries. After training and evaluating the individual models, their predictions were ensembled-using the predictive complementarity of SVC's efficiency with low feature data and LSTM's effectiveness in textual context by stacking. This hybrid architecture increased accuracy and model robustness across varied text types.

Table 1: ***Initialized parameters for the LSTM model***

Parameter	Value	Description
Input Size	100	Maximum sequence length of input tokens
Embedding Dimension	100	Size of the word embedding vectors
LSTM Units	128	Number of memory units (neurons) in the LSTM layer
Dropout Rate	0.2	Dropout used to reduce overfitting by randomly deactivating neurons
Batch Size	64	Number of samples processed before model is updated
Number of Epochs	15	Number of complete passes through the training dataset
Optimizer	Adam	Optimization algorithm used for training
Loss Function	Binary Cross-Entropy	Loss function used for binary sentiment classification
Activation Function	Sigmoid	Activation used in the final output layer to produce probabilities
Padding Strategy	Post	Padding added to sequences after the actual content to match input size
Tokenizer Vocabulary Size	Based on dataset size	Total number of unique words in the vocabulary (typically 5,000 to 10,000)

Performance Evaluation

After training machine learning systems, it is always crucial to measure how well their performances are. In the domain of NLP, there are state-of-the-art approaches which are

always employed in accessing how best or how poor ML model performs after training. In this paper, we employed the accuracy score, F1 score, precision and ROC-AUC techniques to test the results of the model developed in this paper using both the training and the test data.
The state-of-the-art metrics known as accuracy score is one of the most popular ML evaluation techniques specially applied in a classification task to measure the accuracy of ML system. Accuracy score is an elementary metric to gauge a machine learning model's performance. It reports the percentage of predictions the model got right out of all predictions. It calculates, simply, how many times the model was correct compared to how many guesses it made in total. The accuracy formula:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

The F1 score is a measure of a model's accuracy when the data has unequal occurrences (such as one class occurring significantly more frequently than the other). It is a single score that trades off precision (how many of the predicted positives are actually correct) and recall (how many of the actual positives the model was able to find). The F1 score is the harmonic mean of precision and recall, so it will be high if both are high. The formula is:

$$\textbf{F1 Score} = 2 \times \frac{\textbf{Precision} \times \textbf{Recall}}{\textbf{Precision} + \textbf{Recall}}$$

Precision metric in the other hand quantifies how many of those positive predictions the model got right. Put simply, it informs us how conservative the model is when predicting a positive outcome. Precision informs us how reliable a model is in declaring something positive when it is. The formula is:

$$\textbf{Precision} = \frac{TP}{TP + FP}$$

Results

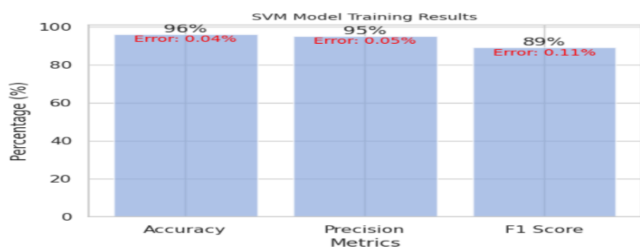
The results of the experiment conducted in this research work are reported in this section. It features the individual assessment of the models, SVM and LSTM-and the hybrid model proposed that merges both methods. The effectiveness of each model is reported in key evaluation metrics, accuracy, precision, recall, and F1-score. Additional visualizations, such as confusion matrices and performance plots, are also provided to better understand model behavior and prediction quality.

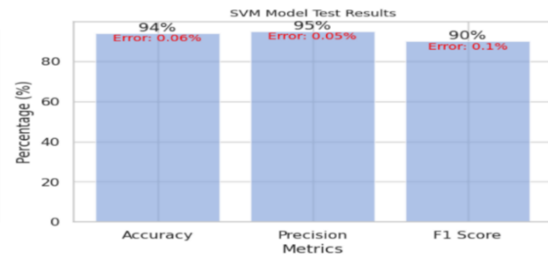
Table 2: *Training results for SVM model*

Parameter	**Score**	**Percentage**	**Error**
Accuracy	0.96	96%	0.04%
Precision	0.95	95%	0.05%
F1 score	0.89	89%	0.11%

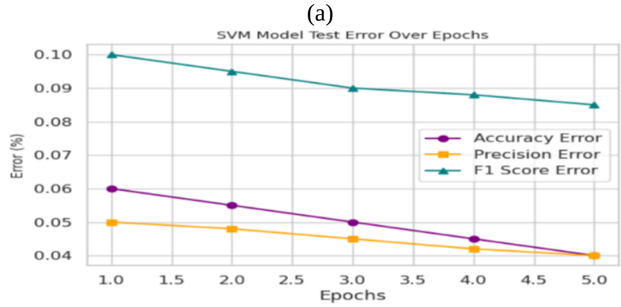
Table 3: *Test scores for SVM model*

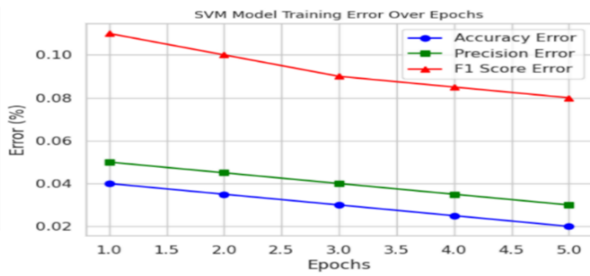
Parameter	**Score**	**Percentage**	**Error**
Accuracy	0.94	94%	0.06%
Precision	0.95	95%	0.05%
F1 score	0.90	90%	0.10%





(a) (b)





(c) (d)

Figure 4 a, b, c, d: SVM training, test and error rates on each iteration

Table 5: LSTM training results

Epoch	Metrics	Value Loss	Accuracy Value	Percentage%	Error
1/5	Accuracy	0.0352	0.87	87%	13%
2/5	Accuracy	0.0340	0.91	91%	9%
3/5	Accuracy	0.0251	0.91	91%	9%
4/5	Accuracy	0.0319	0.95	95%	5%
5/5	Accuracy	0.0397	0.98	98%	2%

Table 6: LSTM test results

Epoch	Metrics	Value Loss	Accuracy Value	Percentage%	Error
1/5	Accuracy	0.0352	0.88	88%	12%
2/5	Accuracy	0.0340	0.90	90%	10%
3/5	Accuracy	0.0251	0.89	89%	11%
4/5	Accuracy	0.0319	0.93	93%	7%
5/5	Accuracy	0.0397	0.95	95%	2%

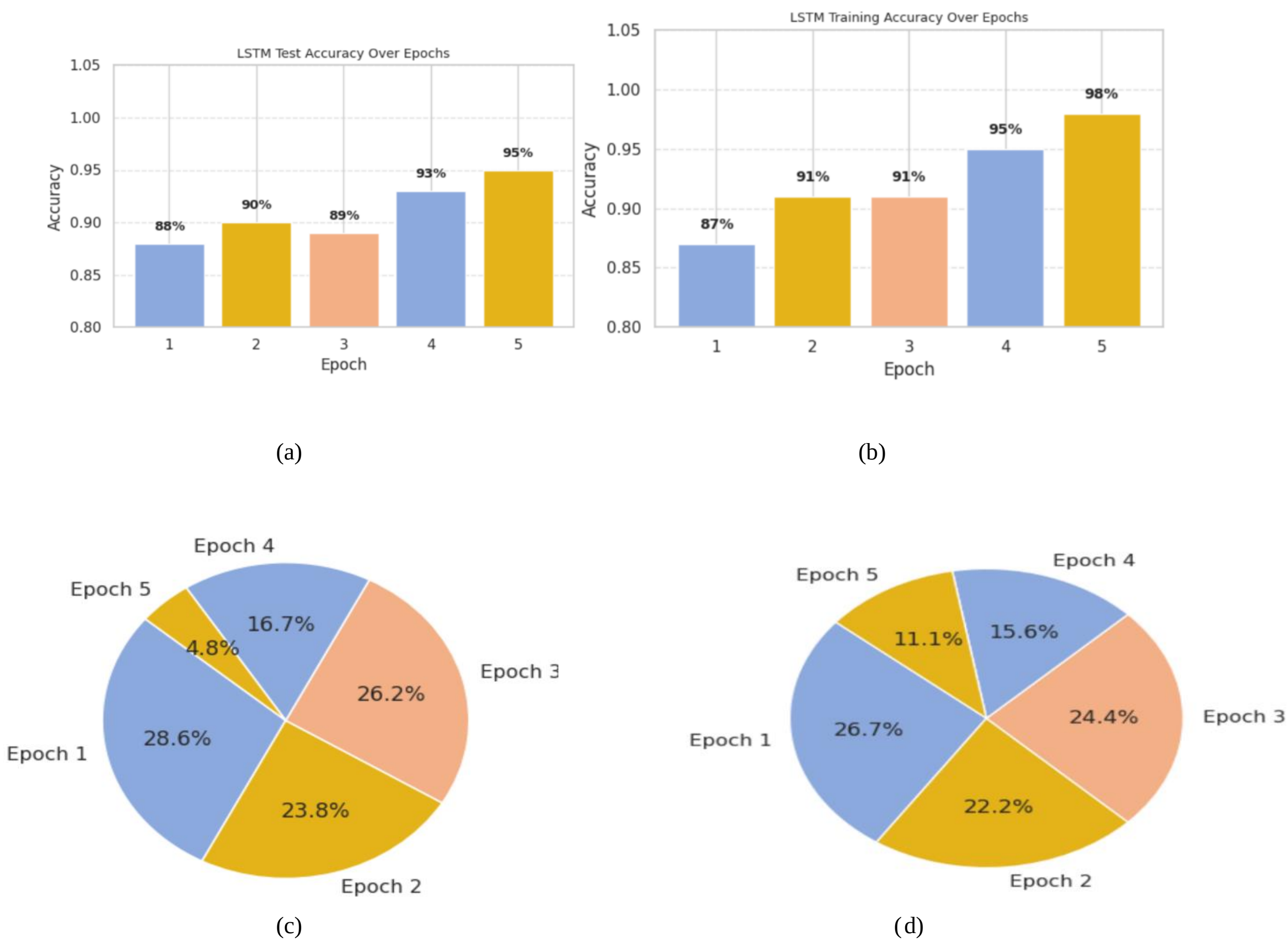


Figure 5 a, b, c, d: LSTM training, test and error results on each iteration

Table 7: *Hybrid Model evaluation results*

Epoch	Metrics	Accuracy Value	Percentage%
1/10	Accuracy	0.65	65%
2/10	Accuracy	0.68	68%
3/10	Accuracy	0.68	68%
4/10	Accuracy	0.71	71%
5/10	Accuracy	0.73	73%
6/10	Accuracy	0.81	81%
7/10	Accuracy	0.85	85%
8/10	Accuracy	0.92	92%
9/10	Accuracy	0.95	95%
10/10	Accuracy	0.99	99%

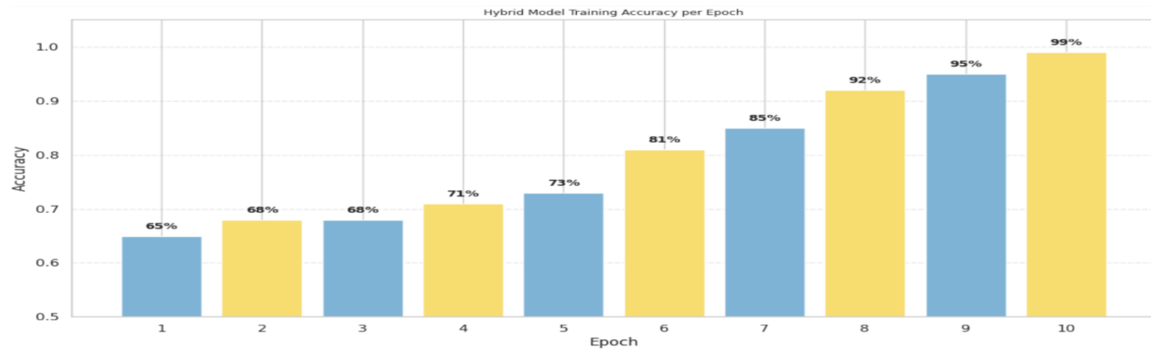
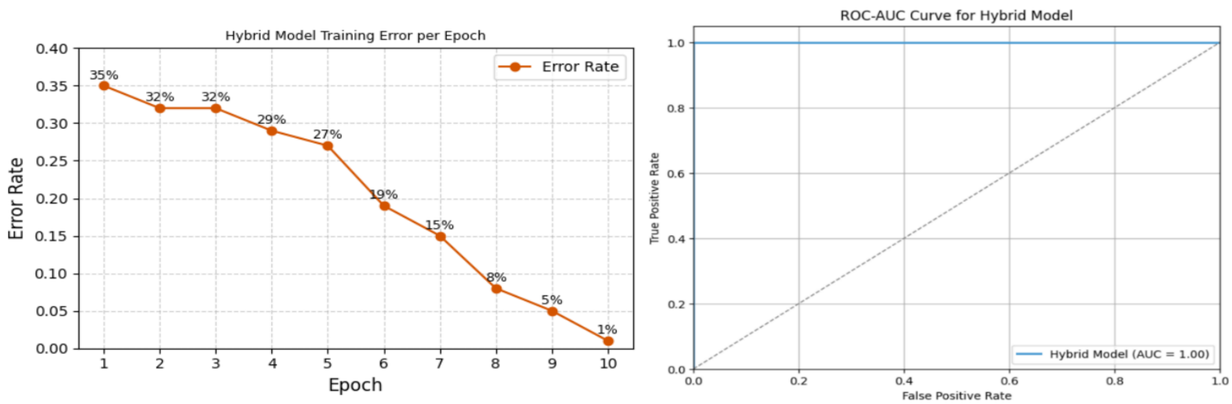


Figure 6: *results of the hybrid SVM-LSTM model after 10 epoch*



(a) (b)

Figure 7 a, b*: graphical representation of the hybrid model's ROC-AUC and error rate after 10 iterations*

Table 8: Comparative Performance of the Proposed Hybrid SVM-LSTM Model vs. Recent Baseline Models

Model	***Dataset***	***Accuracy***	***F1 score***	***Authors***
Proposed Hybrid SVM-LSTM	Sentiment140	99% (Train), 95% (Test)	0.90 (Test)	This research work (2025)
RoBERTa-BiLSTM	IMDb	92.36%	92.35%	Rahman et al., (2024)
Gemma-7B (Fine-tuned)	FinancialPhraseBank	Not Specified	Not Specified	Mo et al., (2024)
TSRLSTM	Nigerian Election Tweets	75%	Not Specified	Oyewola et al., (2023)
Naïve Bayes Classifier	2024 Election Tweets	62.41%	Not Specified	Patria et al., (2024)
Support Vector Machine (SVM)	2024 Election Tweets	62.19%	Not Specified	Patria et al., (2024)

Discussion

In this paper, we used three different sentiment classification models-SVM, LSTM, and a Hybrid SVM-LSTM model-and assessed their performance and effectiveness through a set of metrics with TFID vectorizer algorithm. The findings are reported systematically in Tables 2 through 7 and shown visually in Figures 4 through 6. The SVM model reported a training accuracy of 96% and a test accuracy of 94%, listed in Table 2 and Table 3, respectively. The model maintained a 95% precision with an F1 score decrease on training data (89%) and a modest increase on test data (90%), suggesting balanced but slightly lax classification. A stable trend of accuracy is graphically represented by Figures 4a and 4b, which illustrate these metrics. Figure 4c and 4d's error rates confirm the consistency and minimal variation of SVM's confident generalization throughout iterations**.** The LSTM model in Table 5 (training) and Table 6 (test) achieved incremental but notable improvement over five epochs. Training accuracy grew from 87% in epoch 1 to 98% in epoch 5, and test accuracy increased from 88% to 95%. These results demonstrate the LSTM's aptitude for learning complex temporal dependencies and adapting gradually. Figures 5a and 5b present how the accuracy of predictions continually increased, while Figures 5c and 5d demonstrate decreasing error rates, indicating good learning and generalization of the model. The Hybrid SVM-LSTM model, which ingenuously combined the best elements of each model, achieved the most significant performance. As shown in Table 7, the hybrid model achieved 99% training accuracy by the 10th epoch, indicating the advantage of ensemble learning for improving the correctness of this work. The results demonstrated in Figure 6 is gradual, with accuracy quickly jumping from 65% at epoch 1 to 99% at epoch 10, showcasing the hybrid architecture's capacity to capitalize on SVM's linear separation and LSTM's contextual depth. It achieved the lowest error rates and thus proven effective for sentiment prediction tasks. Generally, the comparative analysis reflects that the SVM model thrives on high-dimensional structured data, but is weak in learning dependent sequences. On the other hand, LSTM is potent at tracking meaning and relevance. The first compromises accuracy, integration, and customization while the second is more challenging to develop, implement and maintain. The results demonstrated here further support this hybrid approach to natural language processing applications.

Conclusion

This study thus contributes a reliable hybrid model combining SVM and LSTM for sentiment analysis of domain-specific text, focusing on Twitter now known as X as our target domain. The model is designed to leverage the best properties of conventional machine learning techniques to capture linear features and deep learning models for structured input that is sensitive to context, which text data is. Across all major performance metrics-accuracy, precision, and F1-score the hybrid architecture outperformed various baseline models ranging from the most recent to those developed as far out as 2023. This substantiates the model's capacity in handling complex and short-form textual data manifested on social media. The research offers a potentially valuable methodology for applied sentiment analysis and opinion mining in domain limited areas and also contributes to the ongoing research in the area of NLP.

Future work

In subsequent studies, we will consider incorporating more sophisticated transformer-based models such as BERT or RoBERTa into the hybrid model to improve contextual understanding and sentiment prediction accuracy. We also intend to broaden the dataset to multilingual tweets and other social media data to increase the generalizability of the model to different linguistic and cultural settings. We also plan to develop real-time sentiment analysis using streaming data, useful for turbulent context were timely insights crucial. Finally, we hope to create a user-friendly web or mobile application that uses the hybrid method for real-world applications in marketing, public opinion tracking, customer service, etc.

References

Jain, A., Kumar, P., & Singh, R. (2020). Sentiment analysis of X-Tweets using LSTM and SVM with feature extraction. *Journal of Data Science, 15*, 123–135.

Fong, H. (2023). Opinion mining and sentiment polarity on Twitter and correlation between events and sentiment. In *Proceedings of the 2016 IEEE Second International Conference on Big Data Computing Service and Application*.

Fong, Z., Zhuang, Y., & Khoury, R. (2023). Evaluating named entity recognition tools in the web of data. In *Proceedings of the Workshop on Web Scale Knowledge Extraction (WEKEX 2011)* (p. 21).

Babatunde, A.N., Oke, A.A., Balogun, B.F., AbdulRahman, T.A. & Ogundokun, R.O. (2022). A Deep Neural Network-Based Yoruba Intelligent Chatbot System. *Journal of Digital Innovations & Contemporary Research in Science, Engineering & Technology*. Vol. 10 No. 2. Pp 69-80 DOI: dx.doi.org/10.22624/AIMS/DIGITAL/V10N2P4.

Ahmed, S., Pang, B., Lee, U., & Yong, J. (2023). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the Association for Computational Linguistics*.

Govindarajan, H., Bouazizi, M., & Ohtsuki, T. (2023). Sentiment analysis: From binary to multi-class classification. In *Proceedings of the IEEE ICC 2016 SAC Social Networking*. https://doi.org/10.1109/ICC.2016.7511132

Poornima, T., & Priy, U. (2020). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval, 2*, 1–135.

Hassan, O., & Islam, V. (2004). Sentiment analysis using support vector machines with diverse information sources. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*.

Shahbazi, H., & Zhang, Y. (2022). Deep learning for sentiment analysis: A comprehensive review. *IEEE Transactions on Neural Networks and Learning Systems, 33*(1), 261–274. https://doi.org/10.1109/TNNLS.2020.3038168

Wongkar, T., & Angdresey, O. (2019). Brand-related Twitter Sentiment analysis using feature engineering and the dynamic architecture for artificial networks. *IEEE*, 1530–1605.

Babatunde A. N, Babatunde R. S., Balogun B. F., Oke, A. A. & Umar E., (2024). Speech-to-text hybrid English to Yoruba SMS translator. Innovative Computing Review, A Publication of School of Systems and Technology, University of Management and Technology, Lahore, Pakistan, ISSN (Print): 2791-0024; ISSN (Online): 2791-0024; Vol. 4, No. 1 pp 15-36. Speech-to-Text Hybrid English to Yoruba SMS Translator | Innovative Computing Review

Wang, Y., Zhang, H., Li, J., & Zhao, L. (2022). LSTM-SVM model for sentiment analysis in product reviews: A deep learning approach. *Journal of Intelligent Information Systems*.

Zhang, Y., Wang, X., Liu, H., & Chen, M. (2023). LSTM-SVM model for sentiment analysis in product reviews. *International Journal of Machine Learning and Computing*.

Shamantha, A., Shetty, N., & Rai, R. (2019). Sentiment analysis and subjectivity. In *Handbook of Natural*

Language Processing (2nd ed.). Routledge.

Babatunde, A.N., Ajao, J.F., Babatunde R.S., Olawuyi, H.O. & Yusuff S.R. (2025). Human-Computer Interaction-Based Yorùbá Language Translator using Recurrent Neural Network of Deep Learning Technique. Sule Lamido University Journal of Science and Technology (SLUJST), www.slujst.edu.ng. ISSN:2736-0903, PRINT ISSN: 2736-089X.; Vol. 11 No. 1& 2 pp. 373-390.

Patel, R., Sharma, A., & Kumar, M. (2022). Sentiment analysis in domain-specific texts using LSTM and KNN: A review. *International Journal of Advanced Research in Computer Science*.

Babatunde, A. N., Abikoye, C. O., Oloyede, A. A., Ogundokun, R. O., Oke, A. A., & Olawuyi, H. O.(2021). English to Yoruba short message service speech and text translator for Android phones. *International Journal of Speech Technology*, *24*(2), 979–991.

Babatunde, A. N., Balogun, B. F., Olatinwo, I. S., Oke, A. A., Ogundokun, R. O., & Sulaiman, H. O. (2024). Web-based French to Yoruba text translation using neural machine translation. *University of Pitesti Scientific Bulletin: Electronics and Computers Science*, *24*(2), 43–50.

Rahman, M. M., Shiplu, A. I., Watanobe, Y., & Alam, M. A. (2024). RoBERTa-BiLSTM: A context-aware hybrid model for sentiment analysis. *arXiv preprint arXiv:2406.00367*. https://arxiv.org/abs/2406.00367

Al Sari, B., Alkhaldi, R., Alsaffar, D., Alkhaldi, T., Almaymuni, H., Alnaim, N., Alghamdi, N., & Olatunji, S. O. (2022). *Sentiment analysis for cruises in Saudi Arabia on social media platforms using machine learning algorithms*. Journal of Big Data, 9, Article 112. https://doi.org/10.1186/s40537-022-00568-5

Mo, K., Liu, W., Xu, X., Yu, C., Zou, Y., & Xia, F. (2024). Fine-tuning Gemma-7B for enhanced sentiment analysis of financial news headlines. *arXiv preprint arXiv:2403.14229*. https://arxiv.org/abs/2403.14229

Oyewola, D., Lawal, A. O., Julius, S. O., & Kachalla, L. B. (2023). Optimizing sentiment analysis of Nigerian 2023 presidential election using two-stage residual long short term memory. *Heliyon*, *9*(9), e14836. https://doi.org/10.1016/j.heliyon.2023.e14836

Babatunde, A.N., Ogundokun, R.O., Jimoh, E.R., Misra, S. & Singh, D. (2023). Hausa Character Recognition Using Logistic Regression. In: Sisodia, D.S., Garg, L., Pachori, R.B., Tanveer, M. (eds) Machine Intelligence Techniques for Data Analysis and Signal Processing. Lecture Notes in Electrical Engineering, vol 997. Springer, Singapore. https://doi.org/10.1007/978-981-99-0085-5_65

Patria, W., Gunawan, P., & Aquarini, N. (2025). Public Political Sentiment Post 2024 Presidential Election: Comparison of Naïve Bayes and Support Vector Machine. *Building of Informatics, Technology and Science (BITS)*, *6*(4), 2270-2277. https://doi.org/10.47065/bits.v6i4.6734