Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 3, Issue 12, 2025

A COMPARATIVE ANALYSIS OF DEEP LEARNING METHODS FOR SLANG DETECTION IN TWITTER DATA

Muhammad Asim[1], Muhammad Waqar[2], Iftikhar Alam[*3]

[1,2,*3]*Departmet of Computer Science, City University of Science and Information Technology, Peshawar, Pakistan*

[1]mrasimkhan2003@gmail.com, [2]mwaqar.personal@gmail.com, [*3]iftikharalam@cusit.edu.pk

DOI: https://doi.org/10.5281/zenodo.17906419

Keywords
Slang word, Twitter, Deep Learning, Sentiment Analysis

Article History
Received: 15 October 2025
Accepted: 25 November 2025
Published: 12 December 2025

Corresponding Author: *
Iftikhar Alam

Abstract

Informal communication is expanding online, particularly on social networking sites, such as Twitter. This caused the widespread use of slang, short speech, and non-standard lingo, which makes basic Natural Language Processing (NLP) techniques harder to execute. This study addresses automated slang detection in tweets with state-of-the-art machine learning (ML) and deep learning (DL)-based models, such as Logistic Regression, BERT, ALBERT, Decision Tree, Random Forest, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), DistilBERT, and RoBERTa. The data normalization process is done by cleaning, lemmatization, and tokenization activities to standardize the input. Accuracy, precision, recall, and F1-score are used to evaluate how well a model performed over the conditions. The criterion of measuring performance was AUC-ROC and AUC-ROC. Results showed that RoBERTa was found to be the best among all the models in terms of accuracy, as it achieved. 93.99%, followed closely by DistilBERT (92.63%) and GRU (82.65%), demonstrating the advantage of transformer architectures and sequential models being better at informal language and nigger English tweets. Traditional ML models such as Logistic Regression (78.56%) and Decision Tree (71.13%) showed moderate performance but provided baseline interpretability. The results state that the context-sensitive deep learning models, and particularly transformer-based deep learning models such as RoBERTa and DistilBERT, have garnered great success in terms of slang detection. Tasks. The study establishes baseline, conducts a multi-comparative analysis of several models, and provides a potent perspective on investigating the automated slang category from various angles.

1. INTRODUCTION

A large amount of user-generated content has been produced, rich with informal language, slang, and abbreviations. [1]. Sentiment analysis, an approach in Natural Language Processing (NLP), focuses on detecting and classifying sentiments into categories such as positive, negative, or neutral. Still, the pervasive use of slang and other non-standard vocabulary on social-media platforms, such as Twitter, poses significant challenges. This requires specialized approaches that can understand the unique linguistic features and structure of social media language, emphasizing the need for customized sentiment analysis frameworks tailored specifically to handle informal communication. [2].

The common forms of informal language include lexical variants such as abbreviations (e.g., ”TMI” for ”too much information”), phonetic spellings

(e.g., "4eva" for "forever"), and jargon (e.g., "bday" for "birthday"). These variations deviate from standard language, complicating sentiment detection in social media content. [3]. Though the hurricane-strength flow of this data is vital to sentiment analysis and numerous NLP tasks, it likewise calls for powerful, feature-rich platforms and algorithms that can ingest, process, and evaluate the data in real time. This capability is crucial for applications like real-time sentiment tracking and event detection [4, 5].

Unlike text classification, which is usually concerned with document classification, sentiment analysis seeks to determine the polarity or emotional sentiment and is usually found in shorter or incomplete statements, and that does not necessitate a full grasp of a sentence. Slang and non-standard expressions

further increase this complexity, where traditional lexicons and rigid machine learning models often fall short [6].

To this extent, lexicon-based approaches rely on a set of fixed sentiment lexicons to assign a rigid emotional charge to every word. The social media is capable of processing the information of word of mouth, which can be further converted into information that can be analyzed through the application of methods of feature extraction, Bag-of-Words (BoW), Part-of-Speech (POS) tagging, n-gram, and Hashtag Analysis into information that can be analyzed and can forward the process of acquiring context, emotive content, and term saliency. [7]. Beyond technical complexity, sentiment analysis plays a vital role across various sectors, including politics, business, healthcare, and finance, where understanding public sentiment is crucial for informed decision-making, trend forecasting, and policy development. [8].

The natural language understanding was elaborated through language modeling and paradigms like the BERT (Bidirectional Encoder Representations from Transformers) paradigm. Besides, the single architecture of BERT, with its ability to fine-tune, does not require any task-specific model architecture or manually created features, making it to be highly general to multiple NLP tasks, such as slang detection, opinion mining, and emotion classification, especially in noisy settings such as social media [9]. This has led to growing interest in more holistic approaches, which combine symbolic reasoning (like semantic networks) with data-driven models (like neural networks), aiming to move beyond surface-level pattern recognition toward deeper language understanding [10].

To this end, the conducted experiment gave us a clear sense of the ability of each structure to identify slang on Twitter, thus revealing the more refined nuances of the vernacular that social media is shot through with. They implemented the Logistic Regression, Naive Bayes, Support Vector Machine, Random Forest, and, most recently, deep learning models, including Convolutional Neural Network (CNN), long short-term memory (LSTM), Gated Recurrent Unit (GRU), and Bidirectional long short-term memory (BiLSTM) [11].

This work also aimed to improve the performance of the state-of-the-art transformer models BERT, DistilBERT, and RoBERTa through individual fine-tuning. As a parallel line of inquiry, we built and evaluated hybrid ensemble structures CNNBERT and LRRF, which combine feature-extraction with awareness of the surrounding context in a bid to boost performance and introduce a new twist to the landscape of ensembles. In each case, we compiled and tested a slang-annotated and curated Twitter corpus divided into two 8020 train splits. This paper contributes to research as follows:

- We have designed and evaluated a dozen classifiers, ranging from traditional machine learning to deep learning and transformer-based approaches, to recognize slang in social media posts.
- The findings showed that transformer-based models, i.e., RoBERTa and DistilBERT, are strongly superior to all other methods in the process of slang classification.
- We proposed two ensemble-based frameworks (CNN-BERT and LR-RF) that combine the strengths of different architectures and achieved the highest performance across evaluation metrics.

The rest of the paper is divided into five sections. Section 2, which gives a literature review through

discussing the previous studies on the research problem and mentioning the major contributions in the area. Section 3 puts into detail the research methodology, writing how the tools, techniques, and approaches are going to be used in addressing the identified problem. Going along with Section 4, this is the section of presenting results and experimentation, backed up with tables, graphs, and illustrations in a way that is easy to read. Lastly Section 5 also provides a conclusion to the study, recommendations (how one can advance the findings in the future), and conclusions to reach out to other researchers who may take up this field in the future.

2. LITERATURE REVIEW

Unlike formal texts, these platforms are filled with informal language, abbreviations, emojis, and slang, presenting significant challenges for traditional Natural Language Processing (NLP) systems. This shift has made the development of robust, adaptable sentiment analysis frameworks a critical focus in both computational social science and artificial intelligence research [12].

Developing on this, the highest semiotic variability and expressivity are characteristic of slang commonly referred to as the language of the streets. These are words such as sick, which may relate to the negative or the positive context. Slang usage can typically be divided into two categories: (1) newly extended senses of existing words, and (2) newly created terms. To address this, a study [13], proposed a model that detects slang both at the sentence level and token level, marking a foundational step toward training NLP models to handle socially-evolving, informal language [13].

Bonta et al. [14], evaluate the utility of lexicon-based sentiment analysis tools, specifically VADER, TextBlob, and NLTK, for processing social media text. VADER can be seen as the most successful, as it has an F1-score of 81.6 percent and an accuracy of 77 percent, because of special treatment of emoticons, intensifiers, punctuation, and slang. Singh et al. [15] Provided both a theoretical and empirical evaluation of the influence of diverse preprocessing strategies on sentiment classification within the context of Twitter data. The study made use of the Sentiment140 dataset and even compared the use of stop-word removal, stemming, tokenization, and noise filtering. It directed its focus toward engaging with the distinct social media "white noise" user tags, hashtags, and casual language units.

Zhao et al.[16], conducted a comprehensive evaluation of preprocessing techniques in sentiment analysis, particularly on Twitter data. The study assessed standard steps such as stemming and lemmatization alongside social media-specific adjustments like acronym expansion and negation handling. Experiments across five datasets demonstrated that adapting preprocessing pipelines to the informal nature of social media significantly boosted classifier performance, with Random Forest achieving the highest accuracy of 85.6%.

To sum up the discussion, Table 1 provides a brief overview of the most useful literature on sentiment analysis and slang detection with the aim of succinctly presenting their main contributions.

Table 1: The literature about Sentiment Analysis and Slang detection

Study	Focus / Objective	Methodology / Model Used	Dataset / Source	Key Findings
[8]	Hybrid optimization for Twitter sentiment classification	PSO + Cuckoo Search + SVM	Twitter data	Achieved 91.91% accuracy; metaheuristic feature selection significantly improved classification.

[14]	Lexicon-based sentiment analysis on social media	VADER, TextBlob, NLTK	Social media posts	VADER achieved an F1score of 81.6%, best for slang, emojis, and informal structures.
[17]	A hybrid framework using optimization in sentiment classification	Naïve Bayes + SVM with PSO and ACO	Standard sentiment datasets	NB + PSO achieved 77.3% accuracy; optimization enhanced model effectiveness.
[13]	Slang detection using deep learning	BiLSTM + CRF + MLP, POS + CNN embeddings	Online Slang Dictionary, Wall Street News	Sentence-level F1-score 0.7971; slang shows syntactic irregularities.
[18]	Aspect-level sentiment classification with BERT	BERT, BERTFC, AEN-BERT, BERT-PT	SemEval2014, Twitter datasets	BERT significantly improved performance: neutral/mixed sentiment is still challenging.
[19]	Sociolinguistic Impact of slang on Indonesian youth language	Surveys and secondary sources	Indonesian youth users	95.7% use slang frequently; it affects grammatical norms and encourages group identity.
[20]	Review of sentiment and emotion detection on social media	Deep learning (LSTM with attention), lexicon methods	Mixed (survey-based)	Hybrid approaches suggested; LSTM is better for informal content.

Multiple works indicate the need to utilize hybrid methods, i.e., the integration of machine learning mechanisms and algorithms of optimisation, such as PSO or ACO, to make the classification systems domain-specific and, when the solutions are multilingual, multilingual. The rapidly developing models of deep learning [21], as well as transformer networks and specifically the BERT and its extensions, demonstrate the patterns of a trend towards context-series architecture that can master in the context of extracting rich sentiment information captured in informal, emotional, or sarcastic text [22].

3. METHODOLOGY

The methodology combines natural language processing (NLP), custom pre-processing for slang-rich content, and transformer-based modeling using RoBERTa. The diagram in Figure 1 demonstrates the layout of the whole research methodology adopted in the current study.

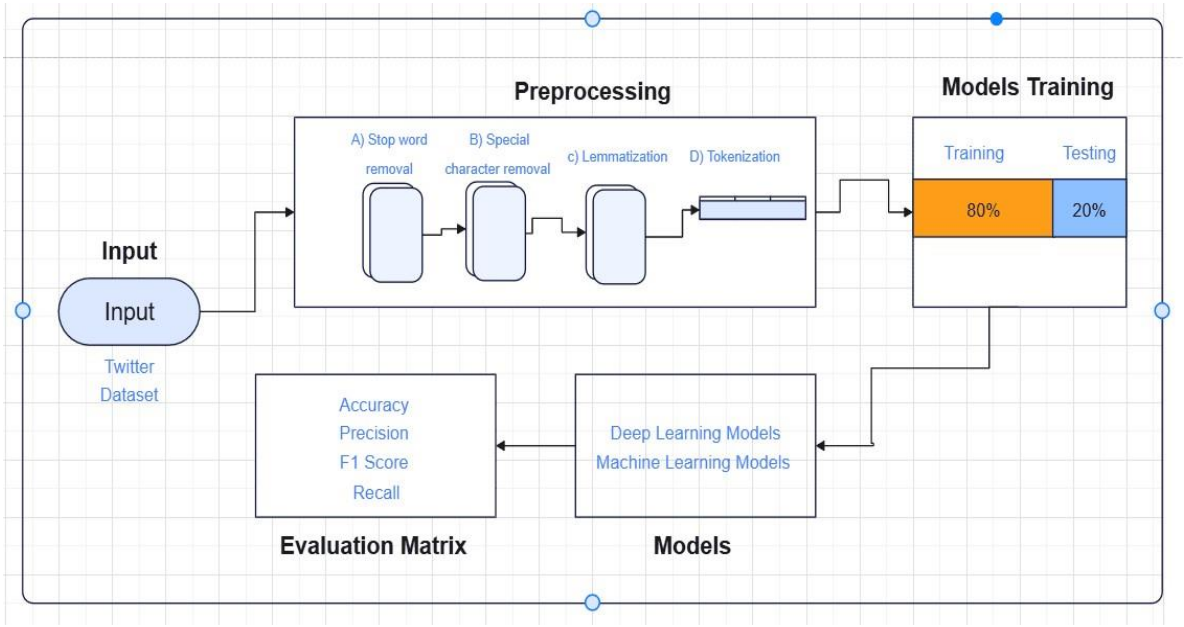

Figure 1: Research Methodology

The solution suggested is to combine the recently introduced Natural Language Processing (NLP) method with the deep learning-based technique (RoBERTa) to identify slang words and sentiment recognition. However, unlike other earlier-model slang detectors, the mechanism in the system, which fails to recognize informal speech, works to the benefit of the system in terms of raising the sentence classification rate. The detailed procedure of the use of the suggested methodology is the following, as it is depicted in Figure 2.

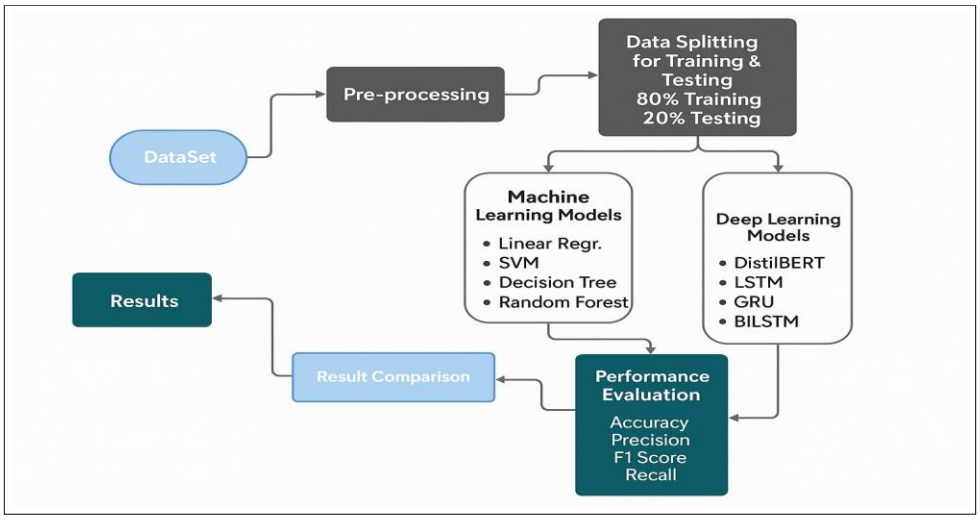

Figure 2: Workflow Diagram

3.1 Dataset Description

The dataset used in this research is sourced from the public GitHub repository [23]. This dataset was specifically designed for the task of binary slang classification in tweets and short text messages. It contains 8,071 samples, where each row represents an individual sentence along with

its corresponding slang label. The dataset comprises two main attributes, as shown in Table 2. Each sentence is a standalone entry and is pre-annotated based on the presence or absence of slang. Examples include casual phrases, internet abbreviations, explicit expressions, and urban vernacular. Some lines contain code-switched or stylized content, such as:

- "That took a lot of EF4T to complete." → Label: 1 (slang)
- "Blow his fucking head off." → Label: 1 (slang)
- "At the moment of maximum roll did I grasp what was going on." → Label: 0 (normal)

Table 2: Attributes Description for Slang Detection Dataset

Attribute	Description
Text (Unnamed column 1)	The actual tweet or sentence was written in informal English.
Label (Unnamed column 2)	Binary classification label: 1 indicates slang is present, 0 indicates normal text.

3.2 Preprocessing

Data preprocessing involves assessing, filtering, modifying, and encoding information so that a machine learning algorithm can comprehend it and utilize the output derived thereof. The key objective of data preprocessing is to improve data quality and is to ensure that data is fit to be used in machine learning by removing data problems with missing data, special cases, and special characters like punctuations and digits, converting case to lower case, breaking long text into small sized words or tokens and reducing words to their root form (as base or stem form) also known as lemmatization and stemming. To enhance the model performance, the dataset undergoes rigorous preprocessing, particularly tuned for social media (Twitter) data, where slang and informal usage are common.

3.3 Framework Training and Testing

The proposed slang detection framework was evaluated in terms of its training and evaluation via a designed pipeline of structured experiments across both the deep learning (DL) and traditional machine learning (ML) frameworks. The last objective was to establish a useful binary system of classification that will perform its part in discriminating between tweets that are full of slang and tweets that are not characterized by slang. To support this, it has embraced a holistic approach in the establishment of data, training, validation, analysis, and testing.

3.4 Performance Assessment Measures

To quantify the performance of the suggested pattern of sentiment analysis that will be utilized to identify the usage of slang words in tweets, a number of conventional measures of performance were used. All these metrics can give a good picture of the classifying power of the model, particularly in light of the binary nature of our task (slang vs. normal tweets). It was assessed with the help of a set of tests, which were not trained during training (20unbiased evaluations). The metrics that were calculated are as follows:

a. Accuracy

One of the most popular measures of evaluation in classification tasks is called accuracy. It represents the proportion of correctly predicted observations (tweets) to the total number of predictions. In this study, the percentage of correctly classified tweets as slang or otherwise.

It can be expressed as a formula, as demonstrated by Equation.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

A high accuracy value indicates that the framework is making fewer classification mistakes across both classes.

b. Precision

Precision measures the number of the tweets written as slang, which are slang. As part of this project, it is useful to identify the reliability of the model when it comes to recognizing slang.

It is of special importance in cases when false positives are expensive (in our case, this could be false flagging a normal tweet as slang, which is an issue in an automated moderation system). It can be evaluated by the formula as portrayed in the Equation. It can be calculated using the formula as shown in the Equation.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

High precision means that most of the tweets labeled as slang are indeed slang, minimizing false alarms.

c. Recall

Recall (also called Sensitivity or True Positive Rate) measures how many actual slang tweets the model was able to detect. It obtained a response to the inquiry of how many slang tweets the model has replied to in terms of the total number of slang tweets. This is critical in our task because missing slang tweets (i.e., false negatives) could lead to offensive or inappropriate content being overlooked. Using substitutions, we are left with the following as the formula:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

A high recall value indicates that the model captures many actual slang instances, which is essential in tasks involving harmful or toxic language.

d. F1-Score

These harmonic means of Precision and Recall are called the F1 Score and therefore represent the balanced measures of the two factors. This is particularly useful in a scenario where the data is skewed, and this might be the case with us since there are more slang tweets than normal tweets.

$$F - measurement = 2 \cdot \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

A high F1 implies that the model will have good performance in terms of identifying slang tweets correctly, as well as classifying the tweets that are not slang incorrectly.

e. AUC - ROC (Area Under the Curve - Receiver Operating Characteristic)

The point where such attains AUC 1.0 would be a candidate perfect classification object, with points each directly equivalent to being slang or a typical tweet, and 0.5 would model that doesn't mean anything performance. High value corresponds to close to 1.0. Measures Such as AUC-ROC are Intrinsic to this study. This performance measure, though not accuracy, precision, or recall, is very useful to complement other measures of accuracy, precision, recall, and F1-score.

4. RESULTS ANALYSIS AND DISCUSSION

Within this section, the outcomes of the experiments obtained through the application of several machine learning (ML) and deep learning (DL) models to the task of slang detection in tweets are thoroughly analyzed and discussed. The objective of this assessment will be to determine how best to label what tweets are, whether slang or normal, with reference to the results taken by each of the models trained according to the methodology described earlier.

4.1 Result Analysis

The details of the performance of each of the models employed in this study were tabulated. It will include the confusion table, precision, recall, F1-score, accuracy, and any other comments based on the behaviour of the model on the test set. It has the same value partitions and training configuration; therefore, it levels the playing ground to evaluate all models. The results of the current study demonstrate the evolution and the comparative advantage of various machine learning and deep learning systems with slang identification on Twitter.

The following sub-sections demonstrate the results and performance of numerous algorithms used for this study. The overall comparison is also presented following this section.

4.1.1 RoBERTa

The results of performance metrics of the RoBERTa model are listed in Table 3.

Table 3: Performance Metrics of the RoBERTa Model for Slang Detection

Metric	Score (%)
Accuracy	93.99
Precision	94.01
Recall	93.99
F1-Score	93.99
AUC-ROC Score	98.22

4.1.2 ALBERT

ALBERT (A Lite BERT) achieved a validation accuracy of 93.56%, making it a strong contender in transformer-based architectures. The performance of the attributable performance of the ALBERT model is shown in Table 4.

Table 4: Performance Metrics of the ALBERT Model for Slang Detection

Metric	Value
Accuracy	93.56%
Precision	93.56%
Recall	93.56%
F1-Score	93.56%
AUC-ROC Score	97.93%

4.1.3 BERT

The BERT model performance metrics are shown in Table 5.

Table 5: Performance Metrics of the BERT Model for Slang Detection

Metric	Score (%)
Accuracy	93.31
Precision	93.39
Recall	93.31
F1-Score	93.31
AUC-ROC Score	97.69

4.1.4 DistilBERT

A performance metric (s) calculated for the DistilBERT model is presented in Table 6.

Table 6: Performance Metrics of the DistilBERT Model for Slang Detection

Metric	Value
Accuracy	92.63%
Precision	92.73%
Recall	92.63%
F1-Score	92.63%
AUC-ROC Score	98.62%

4.1.5 TinyBERT

The performance of the TinyBERT model is shown in Table 7.

Table 7: Performance Metrics of the TinyBERT Model for Slang Detection

Metric	Value
Accuracy	90.40%
Precision	90.40%
Recall	90.40%
F1-Score	90.40%
AUC-ROC Score	96.76

4.1.6 LSTM

Its performance with LSTM was evaluated, and the results are shown in Table 8.

Table 8: Performance Metrics of the LSTM Model for Slang Detection

Metric	Value
Accuracy	82.90%
Precision	83.21%
Recall	82.90%
F1-Score	82.86%
AUC-ROC Score	91.55%

4.1.7 GRU

Gated Recurrent Unit (GRU) delivered strong performance with a validation accuracy of 82.65%. Table 9 demonstrates the performance metrics that were achieved as a result of the GRU model work.

Table 9: Performance Metrics of the GRU Model for Slang Detection

Metric	Value
Accuracy	82.65%
Precision	82.66%
Recall	82.65%
F1-Score	82.65%
AUC-ROC Score	91.51%

4.1.8 BiLSTM

Bidirectional LSTM (BiLSTM) secured an accuracy of 82.53%, leveraging two LSTM layers that read text both forward and backward. Its performance with BiLSTM was evaluated, and the results are shown in Table 10.

Table 10: Performance Metrics of the BiLSTM Model for Slang Detection

Metric	Value
Accuracy	83.53%
Precision	82.67%
Recall	82.53%
F1-Score	82.51%
AUC-ROC Score	91.26%

4.1.9 Logistic Regression

The average validation accuracy of logistic Regression (LR) was 78.56 percent, as shown in Table 11. LR is a linear classifier designed to be effective on data whose features are linearly separable, so it is an effective baseline model here.

Table 11: Performance Metrics of the Logistic Regression Model for Slang Detection

Metric	Value
Accuracy	78.56%
Precision	78.73%
Recall	78.56%
F1-Score	78.53%
AUC-ROC Score	86.83%

4.1.10 Random Forest

The best performing non-deep-learning algorithm in our study was the Random Forest (RF) traditional ensemble machine learning model, with a validation accuracy of 76.27. Table 12 demonstrates the performance metrics that were achieved because of the Random Forest model's work.

Table 12: Performance Metrics of the Random Forrest Model for Slang Detection

Metric	Value
Accuracy	76.27%
Precision	76.41%
Recall	76.27%
F1-Score	76.24%
AUC-ROC Score	83.45%

4.1.11 Decision Tree

The simplest model in our experiment was a Decision Tree with a maximal validation accuracy of 71.13%. Although intuitive and easily seen, Decision Trees tend to overfit and are prone to any minor changes in data. Its performance with the Decision Tree was evaluated, and the results are shown in Table 13.

Table 13: Performance Metrics of the Decision Tree Model for Slang Detection

Metric	Value
Accuracy	71.13%
Precision	71.16%
Recall	71.13%
F1-Score	71.12%
AUC-ROC Score	73.23%

4.1.12 Comparative Analysis of Frameworks

As one can see, the various trained frameworks are presented in Table 14 below. To be exact, the most successful framework that has demonstrated the most beneficial metrics on all parameters is RoBERTa. It is sufficiently cooked with backgrounds of converging to create contexts and non-verbal elucidation of language with tweets. A general summary of the performance of those models according to different categories is shown in Table 14.

Table 14: Model Category-Wise Performance Summary

Category	Frameworks	Observation
Best Performer	RoBERTa	Achieved the highest accuracy, precision, recall, and F1-score due to deep contextual understanding of slang in tweets.

Strong DL Models	BERT, DistilBERT, GRU, LSTM	All demonstrated robust performance, especially GRU (efficient memory usage) and DistilBERT (lightweight, fast, yet accurate).
Efficient & Lightweight	CNN, DistilBERT	Delivered faster training and effective local pattern detection; good for real-time scenarios.
Traditional Base-lines	Random Forest, Logistic Regression	RF showed reliable ensemble-based accuracy; LR served as a lightweight baseline but lacked deeper contextual modelling.
Underperformer	Decision Tree	Lowest accuracy and prone to overfitting; lacks context modelling capability required for slang-heavy and informal social media text.

4.2 Comparative Analysis of the Proposed Frameworks

Among all the trained models, the most successful and with the highest accuracy and F1-score value (93.99 percent) of the Slang Detection model is the one that uses RoBERTa. One of its strengths, which I believe made it very successful in identifying the slang phrases in the noisy and improper information on Twitter, is its high sensitivity to contextual relationships, not to mention its sensitivity to linguistic attributes.

RoBERTa performed better compared to not only the standard machine learning classifiers (Logistic Regression, Random Forest, and Decision Tree) but also the deep learning architectures (LSTM, GRU, and BiLSTM).

This aspect that the model possesses very many layers of the finetune transformer in conjunction with the fact that the model is greatly contextual, had also predetermined the fact that the model turned out to be really successful as regards both normal or those which contain slang stringent material as being effectively modeled and as well as can be recalled properly and to the maximum extent.

Though the raw performance of DistilBERT was lower than that of RoBERTa (at 92.63 percent accurate), it produced comparable results at a much smaller scale. Even this simplified form assumed that distilBERT could at least make more inferences per unit time, at least to the extent that it required less memory per unit time, and thus would be an optimal model when limited resources are made available in an implementation, or where it might be hoped that the model would be run in a real-world implementation.

Both DistilBERT and RoBERTa supported our initial hypothesis, according to which language models based on transformers are more sufficient to represent small text and informational data, i.e., tweets. However, the fact that RoBERTa wins in all respects is probably the reason why this framework should be selected as the most suitable one in our analysis.

4.3 Time and Resource Trade-Off Analysis

Transformer models inherently require considerable computational power due to their layered self-attention mechanisms and sequence modeling capabilities. RoBERTa, while delivering the best accuracy (93.99%), demands significant resources in terms of training time, GPU memory, and inference latency. With 12 transformer layers and a large pretraining corpus, RoBERTa's complexity is suitable for high-performance settings but may not be practical for low-resource environments.

The model selection at the end of the day is subject to the deployment objectives, namely performance performance-centric or resource-efficient. The category-wise performance across the models is listed in Figures 3 and 4.

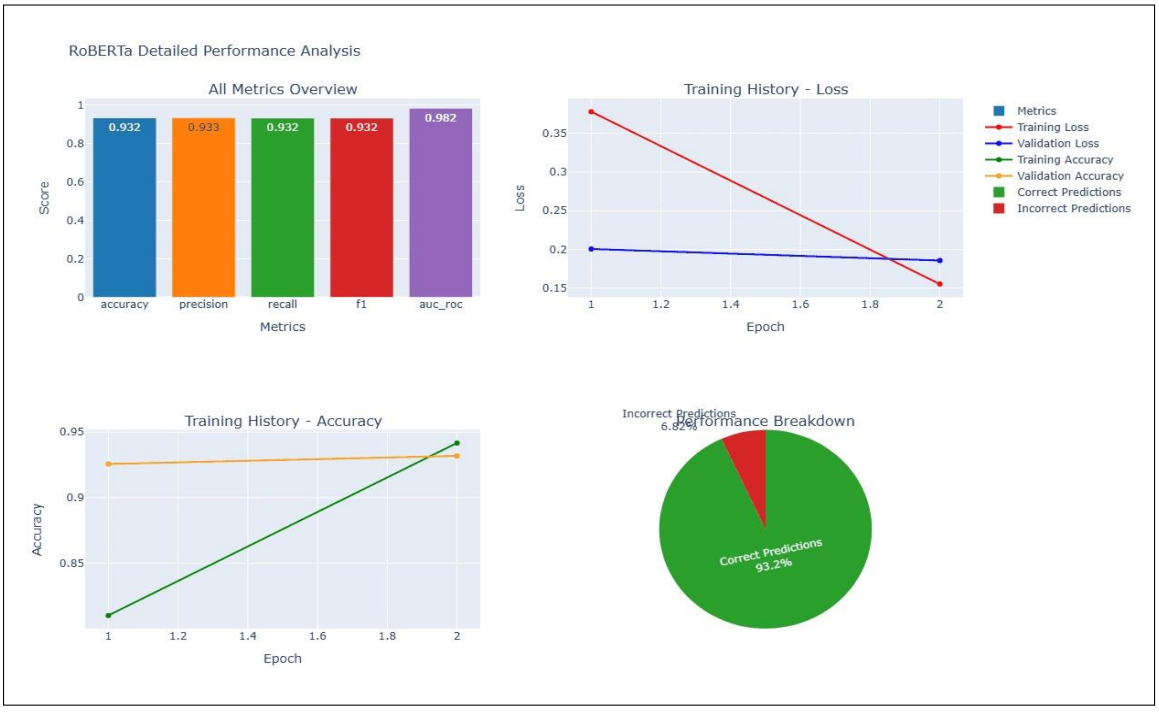

Figure 3: RoBERTa details performance Analysis

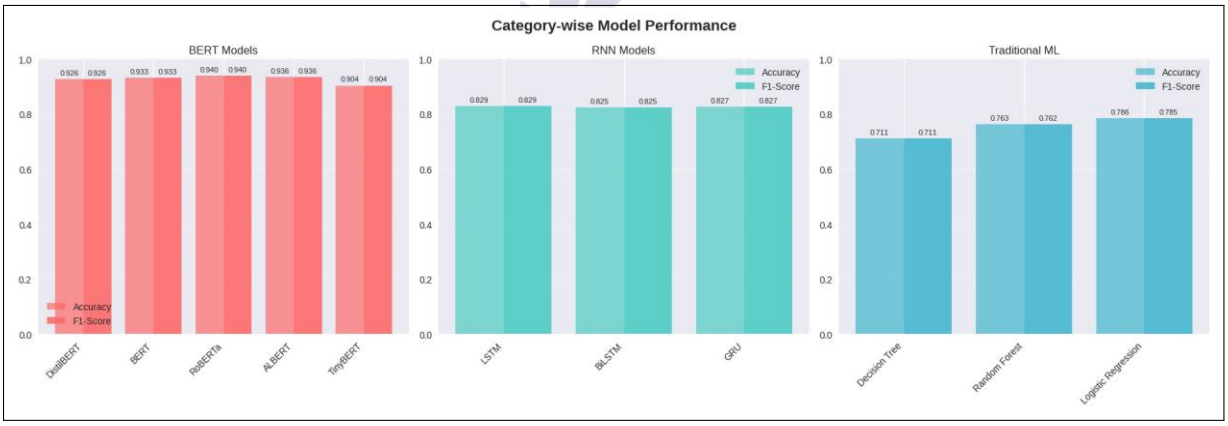

Figure 4: Category Model Performance

4.4 Time Complexity Analysis

The following complexity analysis explains how the scaling of each framework is measured. The detailed comparison of time complexity values between frameworks is presented in Table 15, and the time spent on training and performance is illustrated in Figure 5.

Table 15: Time Complexity Comparison of Different Frameworks

Framework	Time Complexity Description
Logistic Regression	O(nd) – Efficient for linearly separable datasets with small to moderate feature sets.

Random Forest	O(T ·m·log(m)·d) – More complex due to ensemble averaging, but still manageable.
CNN	O(n·f ·k) – Fast and efficient for feature extraction over fixed-length text windows.
GRU / LSTM	O(n·h^2) – Sequential, depends on sequence length n and hidden size h.
DistilBERT	O(L · h^2 · n) – Transformer-based but reduced compared to BERT due to fewer layers.
RoBERTa / BERT	O(L · h^2 · n) – Highest due to deeper architecture and full attention computations.

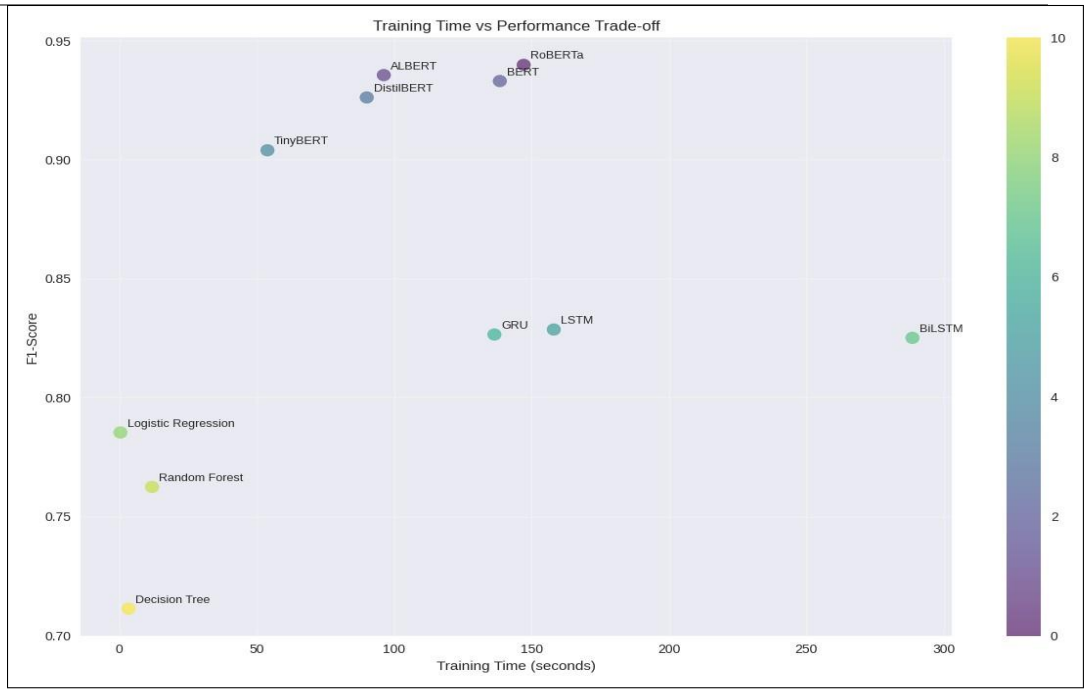

Figure 5: Training time and Performance time

4.5 Comparative Results: Standard vs. Proposed Frameworks

The framework categories, along with their key observations, are outlined in Table 16. Finally, the overall performance comparison summary of frameworks is detailed in Table 17.

Table 16: Framework Categories and Key Observations

Category	Key Observations
Best Performers	RoBERTa achieved the highest overall accuracy (93.99%), showing deep contextual understanding.

Efficient Transformers	DistilBERT delivered fast and accurate results (92.63%) with reduced complexity.
Strong DL Models	GRU (82.65%) and LSTM (82.90%), and traditional ML, proving the value of sequential memory.
Traditional Baselines	Random Forest (76.27%) and Logistic Regression (78.56%) performed decently but lacked nuance handling.
Underperformer	Decision Tree (71.13%) struggled due to overfitting and low context awareness.

Table 17: Performance Comparison Summary of Frameworks

Framework	Accuracy (%)	Precision	Recall	F1-Score
RoBERTa	93.99	94.01	93.99	93.99
ALBERT	93.56	93.56	93.56	93.56
BERT	93.31	93.39	93.31	93.31
DistilBERT	92.63	92.73	92.63	92.62
TinyBERT	90.40	90.40	90.40	90.40
LSTM	82.90	83.21	82.90	82.86
GRU	82.65	82.66	82.65	82.65
BiLSTM	82.53	82.67	82.53	82.51
Logistic Regression	78.56	78.73	78.56	78.53
Random Forest	76.27	76.41	76.27	76.24
Decision Tree	71.13	71.16	71.13	71.12

5. DISCUSSION

The classical models Logistic Regression (LR) and Decision Tree (DT) demonstrated the possibility to achieve the baseline objective of approximately 78.56 percent and 71.13 percent criteria, respectively. They are powerful because they are easy to comprehend and cheap to enjoy. However, the thing is that they are not even capable of identifying multifaceted modelings in writing in general, and slang expressions, abbreviations, and constructions that people tend to use on social media. Their reliance on shallow feature representations limits their generalization to anything beyond their explicit keywords or heuristics based on word frequency features.

On the other hand, the ensemble-based learning model, such as the Rand Field (RF), increased the accuracy rate to 76.27 per cent because of using

many decisions made by learners. The nature of the voting system and the random selection of the features to manipulate allowed RF to adapt to the change in the format of the tweets and the use of slang words between the features manipulated. It, however, could not understand contextual elements that are applied in creating the differences between weak slang utterances that are grouped and interpreted as normal language within a short-texting system. The deep learning systems brought a tremendous revolution. Models (LSTM, BiLSTM, and GRU) able to understand the contextual relationships in a tweet could manage word embeddings and sequential representations without the need to elaborate on their functions. GRU did 82.65% (and LSTM 82.90). It is a simple gating mechanism that allowed GRU to retain valuable information of the past whilst maintaining the issues of vanishing gradients, which are very common in RNNs.

6. CONCLUSION AND FUTURE WORK

First, transformer networks used today during the DistilBERT and RoBERTa deep learning architecture showed and described the study has provided an outstanding machine learning-based barely detecting slang in tweets, and today with deep learning (transformer). This has been facilitated by the creation of informative and context-sensitive programming on social media such as Twitter, which developed the specific slang used in captivity, applying mostly to applications such as content moderation systems, sentiment detection, and language recognition of abuse. This is not to mention general speech recognition. Although the current framework evokes positive outcomes, several avenues toward the improvement and development of the system need to be taken into consideration: Detecting slang in real-time is probably one of the most useful follow-ups to this study and can essentially be achieved by running the model in a ready production environment. The other promising future is the arrangement of a multi-class slang categorization system. Rather than categorizing the tweets into either affirmative or negative, including being defined as merely slang or normal, the system may be conditioned to determine a particular kind of slang, e.g., abusive, humorous, sarcastic, or cultural. Lastly, incorporating explainable AI (XAI) techniques is a crucial enhancement that can increase the model's transparency and user trust.

REFERENCES

[1] J. Jiang, A. Way, and R. Haque, "Translating user-generated content in the social networking space," in Proceedings of the 10th Conference of the Association for Machine Translation in the Americas: Commercial MT User Program, 2012.

[2] A. Balahur, "Sentiment analysis in social media texts," in Proceedings of the 4th workshop on computational approaches to subjectivity, sentiment and social media analysis, 2013, pp. 120-128.

[3] A. A. Adedamola, A. Modupe, and O. J. Dehinbo, "Development and evaluation of a system for normalizing Internet slang in social media texts," in Proceedings of the World Congress on Engineering and Computer Science, 2015, vol. 1, pp. 21-23.

[4] A. G. Prasad, S. Sanjana, S. M. Bhat, and B. Harish, "Sentiment analysis for sarcasm detection on streaming short text data," in 2017 2nd International Conference on Knowledge Engineering and Applications (ICKEA), 2017: IEEE, pp. 1-5.

[5] I. Khan, S. Khusro, and I. Alam, "Smartphone distractions and their effect on driving performance using vehicular lifelog dataset," in 2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), 2019: IEEE, pp. 1-6.

[6] L. Vu and T. Le, PhD, "A lexicon-based method for Sentiment Analysis using social network data," in Proceedings of the International Conference on Information and Knowledge Engineering (IKE), 2017: The Steering Committee of The World Congress in Computer Science, Computer ..., pp. 10-16.

[7] N. K. Singh, D. S. Tomar, and A. K. Sangaiah, "Sentiment analysis: a review and comparative analysis over social media,"

Journal of Ambient Intelligence and Humanized Computing, vol. 11, no. 1, pp. 97-117, 2020.

[8] D. Sharma and M. Sabharwal, "Sentiment analysis for social media using SVM classifier of machine learning," Int J Innov Technol Exploring Eng (IJITEE), vol. 8, no. 9, pp. 39-47, 2019.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), 2019, pp. 4171-4186.

[10] E. Cambria, S. Poria, A. Gelbukh, and M. Thelwall, "Sentiment analysis is a big suitcase," IEEE Intelligent Systems, vol. 32, no. 6, pp. 74-80, 2018.

[11] I. Alam, A. Hameed, and R. A. Ziar, "Exploring sign language detection on smartphones: A systematic review of machine and deep learning approaches," Advances in Human-Computer Interaction, vol. 2024, no. 1, p. 1487500, 2024.

[12] B. Pang and L. Lee, "Opinion mining and sentiment analysis," Foundations and Trends® in information retrieval, vol. 2, no. 1–2, pp. 1-135, 2008.

[13] Z. Pei, Z. Sun, and Y. Xu, "Slang detection and identification," in Proceedings of the 23rd conference on computational natural language learning (CoNLL), 2019, pp. 881-889.

[14] V. Bonta, N. Kumaresh, and N. Janardhan, "A comprehensive study on lexicon-based approaches for sentiment analysis," Asian Journal of Computer Science and Technology, vol. 8, no. S2, pp. 1-6, 2019.

[15] T. Singh and M. Kumari, "Role of text pre-processing in Twitter sentiment analysis," Procedia Computer Science, vol. 89, pp. 549-554, 2016.

[16] Z. Jianqiang and G. Xiaolin, "Comparison research on text pre-processing methods on Twitter sentiment analysis," IEEE Access, vol. 5, pp. 2870-2879, 2017.

[17] E. Badr, M. A. Salam, M. Ali, and H. Ahmed, "Social media sentiment analysis using machine learning and optimization techniques," International Journal of Computer Applications, vol. 975, p. 8887, 2019.

[18] Z. Gao, A. Feng, X. Song, and X. Wu, "Target-dependent sentiment classification with BERT," IEEE Access, vol. 7, pp. 154290-154299, 2019.

[19] R. Maulidiya, S. E. Wijaya, C. Mauren, T. P. Adha, and M. G. R. Pandin, "Language Development of slang in the Younger Generation in the Digital Era."

[20] P. Nandwani and R. Verma, "A review on sentiment analysis and emotion detection from text," Social Network Analysis and mining, vol. 11, no. 1, p. 81, 2021.

[21] I. Alam, A. Basit, and R. A. Ziar, "Utilizing Age-Adaptive Deep Learning Approaches for Detecting Inappropriate Video Content," Human Behavior and Emerging Technologies, vol. 2024, no. 1, p. 7004031, 2024.

[22] A. Mansoor et al., "Enhancing thyroid ultrasound diagnosis with a hybrid CNN and graph attention network," Spectrum of Engineering Sciences, pp. 95-105, 2025.

[23] S. A. Mohammed. "Slang-Detection-and-Identification." https://github.com/Sohaila-Abdulsattar-Mohammed/Slang-Detection-and-Identification/blob/main/datasets/slang_detection_final_dataset.csv (accessed 10-Dec-2025)