Sentiment Analysis on Social Media Using Machine Learning

Rudra Bhadoriya, Saksham Tomar, and Shekhar Kansana

Department of Computer Science and Engineering

Institute of Technology and Management, Gwalior

{rudra.b, saksham.t, shekhar.k}@itm.edu

***Abstract*—**The rapid growth of user-generated content on social media platforms has created a pressing need for automated tools to extract and interpret public opinion at scale. This paper presents a comparative evaluation of machine learning approaches for sentiment classification of social media posts. We experiment with Naïve Bayes, Logistic Regression, Support Vector Machine (SVM), Random Forest, and a bidirectional LSTM on a balanced sample of 50,000 Twitter posts drawn from the Sentiment140 dataset. Our results indicate that LSTM achieves the highest accuracy (88.3%) while linear SVM (85.6%) provides a compelling balance between performance and computational efficiency. We discuss the implications of these findings for real-world sentiment monitoring applications.

Index Terms*—sentiment analysis, machine learning, social media, natural language processing, Twitter, opinion mining.*

I. INTRODUCTION

Social media platforms such as Twitter, Facebook, Instagram, and Reddit have become primary channels through which billions of users share opinions, express emotions, and discuss world events. This explosion of user-generated content presents both an opportunity and a challenge for researchers: the sheer volume of data makes manual analysis infeasible, yet the embedded sentiments carry immense value for businesses, policymakers, and social scientists.

Sentiment analysis—also known as opinion mining—is the computational task of identifying and categorizing subjective information in text. When applied to social media, it enables automatic classification of posts as positive, negative, or neutral, and in more nuanced systems, identifies specific emotions such as joy, anger, or sadness.

This paper presents a comparative study of machine learning approaches for sentiment classification on social media data. We evaluate traditional models (Naive Bayes, Logistic Regression, SVM) alongside a deep learning baseline (LSTM) on a Twitter dataset. Our goal is to identify an effective pipeline that balances accuracy, interpretability, and computational cost for real-world deployment.

II. LITERATURE REVIEW

Early work by Pang et al. [1] demonstrated that machine learning outperforms hand-crafted rules for sentiment classification in movie reviews, establishing bag-of-words with SVM as a strong baseline. Go et al. [2] applied this paradigm to Twitter data using distant supervision, showing that emoticons could serve as noisy labels to build large-scale training sets without manual annotation.

C. Deep Learning Baseline

For comparison, we implement a bidirectional LSTM with 128 units per direction. Word embeddings are initialized with 100-dimensional GloVe vectors pre-trained on Twitter data. The network is trained for 10 epochs with a batch size of 64, using the Adam optimizer and binary cross-entropy loss. A dropout of 0.4 is applied to reduce overfitting.

V. EXPERIMENTS AND RESULTS

A. Experimental Setup

The dataset is split into 80% training and 20% testing sets with stratified sampling. All models are evaluated on accuracy, macro F1-score, and area under the ROC curve (AUC). Experiments are run in Python 3.9 using scikit-learn 1.2 and TensorFlow 2.11.

B. Results

Table I summarizes the performance of all classifiers.

TABLE I

Performance Comparison of Classifiers

Algorithm	Accuracy	F1-Score	AUC
Logistic Regression	83.2%	0.831	0.89
Naïve Bayes	80.7%	0.804	0.86
SVM (Linear)	85.6%	0.854	0.91
Random Forest	82.1%	0.819	0.88
LSTM	88.3%	0.881	0.94

LSTM achieves the highest accuracy (88.3%) and F1-score (0.881), confirming that sequence models capture contextual patterns that bag-of-words representations miss. Among traditional models, linear

Pak and Paroubek [3] specifically studied Twitter sentiment, noting that the informal, abbreviated language of tweets poses unique preprocessing challenges. Liu [4] provided a comprehensive survey of opinion mining, covering both lexicon-based and machine-learning-based methods. More recently, deep learning models—especially recurrent architectures and attention mechanisms—have substantially advanced the state of the art [5].

III. DATASET AND PREPROCESSING

A. Dataset

We use the Sentiment140 dataset, comprising 1.6 million tweets labeled via emoticons as positive (1) or negative (0). For this study we sample 50,000 tweets (25,000 per class) to maintain class balance while keeping experiments manageable.

B. Text Preprocessing

Raw tweets require substantial cleaning before feature extraction. Our pipeline applies the following steps in order: (1) removal of URLs, @mentions, and hashtag symbols; (2) lowercasing; (3) removal of numbers and special characters; (4) tokenization; (5) stop word removal using NLTK's English stop-word list; and (6) stemming using the Porter Stemmer. Negations such as "not good" are handled by prepending a negation flag to subsequent tokens until a punctuation boundary.

IV. METHODOLOGY

A. Feature Extraction

For traditional models we extract TF-IDF weighted unigram and bigram features with a maximum vocabulary of 20,000 terms. We additionally compute a small set of handcrafted features: tweet length, positive emoticon count, negative emoticon count, and a lexicon-based sentiment score derived from the VADER sentiment dictionary.

B. Machine Learning Models

We train and evaluate four classifiers. Naive Bayes (NB) assumes feature independence and applies Laplace smoothing. Logistic Regression (LR) is trained with L2 regularization (C = 1.0) using the lbfgs solver. Support Vector Machine (SVM) employs a linear kernel optimized via stochastic gradient descent. Random Forest (RF) uses 200 decision trees with a maximum depth of 20.

SVM performs best (85.6%), consistent with findings in prior literature. Naive Bayes, despite its simplicity, achieves a competitive 80.7%, making it a useful baseline for resource-constrained settings.

VI. DISCUSSION

The gap between SVM and LSTM (~2.7%) is meaningful but modest given the substantially higher compute cost of training neural models. For production systems where inference latency and model explainability matter—such as brand monitoring dashboards or customer service triage—SVM remains an attractive choice.

A key limitation of this work is the reliance on emoticon-labeled data, which may not fully represent neutral or sarcastic tweets. Sarcasm detection remains an open research challenge. Future work should incorporate transformer-based models such as BERTweet, explore multilingual corpora, and investigate aspect-level sentiment to move beyond document-level polarity.

VII. CONCLUSION

This paper presented a comparative analysis of machine learning methods for sentiment analysis on social media. Our experiments on 50,000 tweets demonstrate that while LSTM outperforms traditional classifiers, linear SVM offers a strong accuracy-efficiency trade-off suitable for real-world deployment. The preprocessing pipeline—combining TF-IDF features with lexicon scores—proves effective across all models. We conclude that the choice of model should be guided by the operational constraints of the target application.

REFERENCES

[1] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in Proc. EMNLP, 2002, pp. 79–86.

[2] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," Stanford CS224N Project Rep., 2009.

[3] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," in Proc. LREC, vol. 10, 2010.

[4] B. Liu, "Sentiment analysis and opinion mining," Synthesis Lectures on Human Language Technologies, Morgan & Claypool, 2012.

[5] S. Rosenthal, N. Farra, and P. Nakov, "SemEval-2017 Task 4: Sentiment analysis in Twitter," in Proc. SemEval, 2017, pp. 502–518.