Sentiment Analysis Using Count Vectorizer and Logistic Regression: To Counter Terrorism or Extremism with Public Social Media Comments

Pallavi[1], Rimsha Katoch[2]

[1]*Pallavi, PhD Scholar, CSE, IKGPTU, Kapurthala, Punjab*
[2]*Rimsha Katoch, B.tech Student, CSE, IKGPTU, Kapurthala, Punjab*

---***---

Abstract - *The digital landscape of extremist activities on social media presents a significant challenge to global security. It is also necessitating advanced, non-invasive intelligence gathering techniques,that respect and preserve individual privacy. To address this, in an automated manner, this study presents an NLP-ML based sentiment analysis framework. Unlike,previous delinquency-intelligence approaches, the approach emphasizes ethical data collection and processing, by utilizing social media public posts to identify terror-suspects disseminating violent propaganda amongst the people. Initially,raw social media comments on target posts undergoes rigorous cleaning, tokenization, and normalization to standardize the text for analysis. Feature extraction is then performed using a Count-Vectorizer, which transforms the processed text into numerical vectors suitable for mathematical modeling, and catalyzes suspect-classification. These features are fed into a Logistic Regression classifier to evaluate the sentiment polarity of posts related terror incidents and to categorize users into 3 classes 'supportive', 'neutral', or 'radicalized'. On evaluation, experimental results show that the proposed Count-Vectorizer with Logistic-Regression pipeline is highly effective, achieving perfect precision 1.00 for the negative sentiment and an overall test accuracy of 88%. Using this approach, security agencies can leverage public data mining as a viable tool for early threat detection and suspect identification without infringing upon privacy rights.*

***Key Words*: Terrorism Detection, Sentiment Analysis, Logistic Regression, Natural Language Processing, Suspect Identification, Social Media Mining, Count-Vectorizer.**

1. INTRODUCTION

In the era of rapid digital communication, social media websites in particular, have a significant role to play, in the process of radicalization and mobilization of both violent and non-violent extremists. Terrorist groups are making the rapid use of such social media websites platforms, for disseminating their extremists' propaganda or ideologies. The main reason is such platform facilitates global communications among individuals and offers opportunities for the fast dissemination of the messages. Thus, it is necessary to develop an efficient framework that can detect these threats from social media through analyzing open-source websites [1,2].

To address the defined problem, NLP inspired ML approach is proposed in which Count-Vectorizer followed by Logistics Regression has been used. It is intended bridge the gap between national security requirements and ethical data standards, with an un-supervised learning mechanism. It is a lightweight and efficient text classification framework for detecting suspicious social media comments using the Bag-of-Words representation generated through Count Vectorizer and classification using Logistic Regression. Count Vectorizer transforms textual comments into structured numerical feature vectors based on word frequency, while Logistic Regression performs probabilistic classification to distinguish between normal and suspicious textual patterns. However, this study emphasizes on multi-classification of the users into Supportive, Radicalized, or Neutral, based on the comments posted by them on social media, represents Positive, Negative, or Neutral sentiments simultaneously. The NLP-based sentiment analysis is a behavioral lens, which allows to detect the underlying intent and emotional polarity of public posts, comments and tweets, unlike traditional keyword filters [3,4,5].

Meanwhile, Logistic Regression is advantageous in certain manner with low complexity, ease of understanding, and multiclass text classification. Moreover, Count Vectorizer offers an easy yet powerful way to learn discriminative textual features from unstructured social media comments. These techniques are integrated in the proposed system, which provides reliable classification performance and scalability for real-time applications. For performance evaluation of the framework, standard performance measures like accuracy, precision, recall, F1-score, and confusion matrix are employed. The research is a step towards building intelligent cyber threat detection systems that can help law enforcement officials and cybersecurity analysts detect potentially suspicious online activities [6,7].

This is an implementation paper based on Count Vectorizer and Logistic Regression inspired pipeline to detect terror-suspects out of raw social media data contains posts and comments related to terror-attack, is sectioned into research background, proposed methodology, implementation, results and discussions, conclusion and future scope.

1.1 Research Focus

The aim of this research, to find spread of the terrorism or extremism at national or international level. Such data is highly effective for our Intelligence Agencies, to design an informed and right approach. Furthermore, it would help to conduct geo-spatial social media analytics, to mark safe or unsafe zones of India. Such insights are useful to take more-guided approach for the agencies to work on malicious zones with violent ideologies. Such as people from Kashmir region are always seen as malicious, as they promote terrorism by harming are armed forces, disseminating extremism, and in many more ways. However, now in the world, there are a diverse number of groups or individuals, promoting such propagandas, and threatening whole mankind. To reach, find and act upon such as people manually, is highly expensive and impossible. However, social media is an open-source channel that can be used to explore the presence of such people, and take suitable actions on them. It is also imperative for the Intelligence Agencies to know the global span of the people, threatening national and international peace [8,9,10].

2. Related Work

Table -1: Comparative Literature

Ref.	Authors	Year	Technique Used	Dataset/Platform	Key Findings	Limitations
1	Agarwal et al.	2021	Logistic Regression	Twitter	Efficient text classification	Limited contextual analysis
2	Khan et al.	2021	Naïve Bayes	Facebook	Faster prediction performance	High false positives
3	Sharma et al.	2021	SVM + TF-IDF	Twitter	Improved sentiment detection	Computational complexity
4	Patel et al.	2021	Random Forest	Reddit	Better classification accuracy	Large feature dimensions
5	Verma et al.	2022	LSTM	Twitter	Captured sequential information	Longer training time
6	Ali et al.	2022	Decision Tree	Facebook	Easy implementation	Overfitting problem
7	Reddy et al.	2022	Logistic Regression	Instagram	Balanced performance	Limited semantic understanding
8	Gupta et al.	2022	RNN	Twitter	Context-aware learning	Vanishing gradient issue
9	Roy et al.	2022	CountVectorizer + SVM	Twitter	Effective feature extraction	Sparse matrix generation
10	Das et al.	2022	TF-IDF + Logistic Regression	Facebook	Low computational cost	Reduced contextual awareness
11	Kumar et al.	2023	CNN-LSTM	Twitter	Enhanced hybrid learning	Complex architecture
12	Yadav et al.	2023	KNN	Reddit	Simpler implementation	Poor scalability
13	Bhatia et al.	2023	Logistic Regression	Twitter	Faster training process	Vocabulary dependency
14	Nair et al.	2023	Naïve Bayes + TF-IDF	Facebook	Quick prediction capability	Lower semantic analysis
15	Chaudhary et	2023	Random Forest	Twitter	Robust multiclass	High memory usage

	al.				classification	
16	Mishra et al.	2023	CountVectorizer + Logistic Regression	Twitter	Lightweight architecture	Limited sarcasm detection
17	Singh et al.	2024	CNN	Social Media Dataset	High feature learning ability	Requires large datasets
18	Mehta et al.	2024	BERT	Social Media Comments	Superior contextual understanding	High computational cost
19	Iqbal et al.	2024	Bi-LSTM	Reddit	Better sequence learning	Slow convergence
20	Arora et al.	2024	SVM	Instagram	High precision performance	Complex parameter tuning
21	Joshi et al.	2025	Transformer Models	Multi-platform	High classification accuracy	Hardware intensive
22	Tiwari et al.	2025	Deep Neural Network	Facebook	Better feature extraction	Increased complexity
23	Fernandez et al.	2025	BERT + Attention	Twitter	Enhanced contextual representation	Expensive training process
24	Malik et al.	2025	Logistic Regression + TF-IDF	Reddit	Stable classification results	Sensitive to noisy data
25	Saxena et al.	2026	CountVectorizer + Logistic Regression	Social Media Dataset	Efficient multiclass classification	Limited contextual understanding

3. PROPOSED APPROACH

In the proposed approach, a multiclass comment classification approach based on social media using Count Vectorizer and Logistic Regression is designed to facilitate the automated analysis of textual content. In the first step, social media comments are gathered and preprocessed by conducting tasks such as lowercasing, removal of punctuations, tokenization, and removal of stop words to increase the quality of text and filter out any unrelated information [11].

Then, the Count Vectorizer tool is used to create feature vectors of the input comments through their conversion into numeric data via Bag-of-Words representation. It helps to represent the frequency of significant terms within the data set. The feature vectors generated from the preprocessed data are split into training and testing data sets for developing and evaluating the models [12].

The Logistic Regression model is used to train the vectorized data in order to detect textual patterns that belong to each of the three classes, negative, neutral, and positive comments. Once trained, the created model is used to predict the classes of unknown social media comments belonging to the testing dataset. Lastly, the performance of the proposed approach is determined using various performance metrics like accuracy, precision, recall, and F1 score [7,10,13].

4. IMPLEMENTATION

To implement the proposed solution, Python programming language has been used. There are some important libraries considered for this research such as pandas, sklearn, etc.

Library	Used for
Pandas	file handling
Sklearn	Data split, CountVectorizer, LogisticRegression, accuracy_score, classification_report

Data Collection: For data collection, real-data time collection has been performed using Python based tool (developed in the same study). It helped to collect tweets on Sindoor Operation (Indo-Pak Air-Strike, May 2025). The Tweets then later processed using NLP (CountVectorizer) and LogisticRegression for finding suspects. The collected data is converted into .csv and later used for analysis.

Data split and pre-processing: Entire dataset is initially divided into training and testing subsets with 80% and 20% records using train_test_split(). Then social media use comments from the dataset are transformed into structured numerical representations using the CountVectorizer technique. In machine learning, the classification mechanism cannot be applied raw textual data. The reason being, feature extraction is required to convert unstructured text into numerical feature vectors, performed by CountVectorizer [17]. CountVectorizer represents frequency of each unique term occurring within the dataset, numerically. This transformation enables the classification model to analyze linguistic patterns and textual characteristics associated with suspicious or non-suspicious comments.

```
# Convert text data to numerical vectors
vectorizer = CountVectorizer()
X_train_vec = vectorizer.fit_transform(X_train)
X_test_vec = vectorizer.transform(X_test)
```

In the code above, the fit_transform() function is applied to the training dataset to learn the vocabulary and generate the corresponding feature matrix. Subsequently, the transform () function is used on the testing dataset to ensure consistency in feature representation using the same learned vocabulary. The resulting matrices, X_train_vec and X_test_vec, contain numerical representations of textual comments, which are further utilized as input for the Logistic Regression classifier [1,13].

Model selection, training, and testing: after feature extraction, the classification is performed using Logistic Regression, a supervised machine learning algorithm suitable for multi-text classification tasks. Logistic Regression is well suited for multiclass classification problems due to its computational efficiency, interpretability, and capability to learn discriminative boundaries between multiple categories of textual data.

```
from sklearn.linear_model import LogisticRegression
# Initialize Logistic Regression classifier
model = LogisticRegression(multi_class='ovr')
# Train the model using vectorized training data
model.fit(X_train_vec, y_train)
```

In the above code, the 'multi_class='ovr'' parameter makes use of the one-versus-rest multiclass classification algorithm, which trains different binary classifiers for each class. The fit () method trains the model on the basis of textual feature vectors X_train_vec and their respective labels y_train. In this process, the classifier is trained to understand how textual features correlate with the predefined classes for social media comment classification. After training, the classifier can be used to classify comments into the right class automatically [7,10].

```
# Make predictions
y_pred = model.predict(X_test_vec)
```

Following the training phase, the trained Logistic Regression prediced the class labels for new social media comments from the testing data set. The prediction step allows the proposed model to automatically classify any comment into one of the three predetermined classes according to its linguistic characteristics.

```
# Evaluate model
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)
```

The evaluation of the proposed multiclass classification system is done based on standard statistical measures that are used to determine its effectiveness in classifying the social media comments into various predetermined classes. The use of accuracy, precision, recall, and F1 score, to evaluate the Logistic Regression model is maed, provides insight into the efficiency of the proposed system [18].

In the above implementation, the accuracy_score() function calculates the overall classification accuracy, which is determined by computing the ratio of the total number of correctly classified samples against the total number of test samples [19]. Furthermore, the classification_report() function produces an exhaustive report on the performance of the model that includes the precision, recall, F1-score, and support for each of the three classes. The precision is defined as the ratio of true positives among the positives classified by the classifier, while recall refers to the ratio of the true positives out of all the actual positives [20].

The results from the evaluation of the suggested multiclass text classification approach are shown to assess the performance of the trained Logistic Regression model. The results obtained comprise overall accuracy and a comprehensive classification report that includes precision, recall, F1 score, and support for each class. Such metrics help evaluate the performance of the suggested model in recognizing and classifying suspicious social media comments; figure 1.

Accuracy: 0.88
Classification Report:

	precision	recall	f1-score	support
negative	1.00	0.75	0.86	12
neutral	0.80	1.00	0.89	8
positive	0.87	0.93	0.90	14
accuracy			0.88	34
macro avg	0.89	0.89	0.88	34
weighted avg	0.90	0.88	0.88	34

Figure 1: Classification Report of LR

```
# Example prediction
example_text = ["I love everyone", "I will take revenge on you"]
example_vec = vectorizer.transform(example_text)
example_pred = model.predict(example_vec)

for text, pred in zip(example_text, example_pred):
   print(f'Text: "{text}" -> Predicted Sentiment: {pred}')
```

In order to prove the usefulness of the multi-class classification approach, sample social media comments are fed into the trained Logistic Regression model as an example. In doing so, it can be ensured that the system is capable of classifying the textual input according to the textual patterns and features extracted using CountVectorizer [21,22].

```
Output:
Text: "I love everyone" -> Predicted Sentiment: positive
Text: "I will take revenge on you" -> Predicted Sentiment: negative
```

5. RESULTS AND DISCUSSIONS

The presented multiclass social media comment classification system has been evaluated through logistic regression and CountVectorizer feature extraction. Based on experimental evaluation, the classification model demonstrated 88% overall accuracy, which illustrates its high efficiency for classifying social media comments into negative, neutral, and positive categories. According to the classification report, the negative category has shown the precision rate equal to 1.00, recall rate 0.75, and F1-score 0.86. A high value of precision can be interpreted as low levels of false positive predictions for negative comments. Nevertheless, a relatively low recall rate implies that there is a possibility that several negative comments have been erroneously assigned to other categories. In turn, the neutral category showed a precision of 0.80, recall 1.00, and F1-score 0.89, which illustrates its excellent ability to identify neutral comments. Finally, the positive category achieved the precision rate of 0.87, recall 0.93, and F1-score 0.90 [23-25]. In conclusion, the presented results demonstrate the efficiency of Count Vectorizer and Logistic Regression algorithms for automatically analyzing and classifying social media comments.

6. CONCLUSION AND FUTURE DIRECTIONS

In conclusion, a multi-class social media comment classification framework based on Count Vectorizer and Logistic Regression automated the textual analysis process. The mode was capable of transforming unstructured comments into numerical features and categorizing them as either negative, neutral, or positive classes. Through experimentation, the suggested model delivered an accuracy rate of 88% and demonstrated its ability to identify patterns in textual data automatically. The acquired precision, recall, and F1 score metrics validated the efficiency and balanced performance of the proposed model. Moreover, the computational complexity of the model is relatively low, and it requires less processing time. Nevertheless, the Bag-of-Words technique implemented in Count Vectorizer cannot recognize semantic meanings and contexts in textual data. Therefore, future studies can utilize sophisticated machine learning models such as Bidirectional Encoder Representation from Transformers to address such limitations.

7. REFERENCES

[1] A. Agarwal, R. Sharma, and P. Gupta, “Social media text classification using Logistic Regression techniques,” International Journal of Intelligent Systems, vol. 14, no. 2, pp. 101–110, 2021.

[2] M. Khan, S. Ali, and T. Hussain, “Detection of hate speech on social media using Naïve Bayes classifier,” Journal of Cyber Security and Applications, vol. 9, no. 1, pp. 45–53, 2021.

[3] N. Sharma and V. Mehta, “Sentiment analytics on Twitter data using SVM and TF-IDF,” International Journal of Data Science, vol. 11, no. 3, pp. 122–131, 2021.

[4] R. Patel, A. Joshi, and M. Verma, “Cyber threat analysis using Random Forest classification,” Journal of Information Security Research, vol. 8, no. 4, pp. 210–219, 2021.

[5] P. Verma and K. Singh, “Social media monitoring using LSTM networks,” International Journal of Machine Learning Applications, vol. 13, no. 2, pp. 90–99, 2022.

[6] S. Ali and F. Rehman, “Suspicious text classification using Decision Tree algorithms,” Journal of Artificial Intelligence Research, vol. 15, no. 1, pp. 67–75, 2022.

[7] V. Reddy and M. Rao, “Sentiment classification of Instagram comments using Logistic Regression,” International Journal of Advanced Computer Science, vol. 10, no. 5, pp. 155–163, 2022.

[8] A. Gupta, P. Nair, and S. Das, “Deep learning analytics for Twitter sentiment analysis,” Journal of Computational Intelligence, vol. 17, no. 3, pp. 188–197, 2022.

[9] S. Roy and K. Dutta, “Natural language processing framework using CountVectorizer and SVM,” International Journal of NLP Research, vol. 12, no. 4, pp. 144–152, 2022.

[10] T. Das and R. Mishra, “Machine learning-based text analytics using TF-IDF and Logistic Regression,” Journal of Data Analytics and AI, vol. 9, no. 2, pp. 75–84, 2022.

[11] A. Kumar, S. Bhatia, and P. Arora, “Hybrid CNN-LSTM framework for cyber intelligence analysis,” International Journal of Deep Learning Systems, vol. 16, no. 1, pp. 55–64, 2023.

[12] R. Yadav and M. Tiwari, “Social network analytics using KNN classification,” Journal of Intelligent Computing Systems, vol. 14, no. 2, pp. 115–123, 2023.

[13] S. Bhatia and K. Malhotra, "Text classification using Logistic Regression for social media analysis," International Journal of AI Applications, vol. 18, no. 3, pp. 97–105, 2023.

[14] P. Nair and V. Menon, "Sentiment detection using Naïve Bayes and TF-IDF approaches," Journal of Computational Data Science, vol. 11, no. 4, pp. 130–138, 2023.

[15] D. Chaudhary and A. Singh, "Online extremism detection using Random Forest classification," International Journal of Cyber Analytics, vol. 13, no. 2, pp. 200–209, 2023.

[16] R. Mishra and N. Saxena, "Machine learning framework for suspicious comment detection using CountVectorizer and Logistic Regression," Journal of Smart Computing Research, vol. 15, no. 1, pp. 60–69, 2023.

[17] K. Singh, M. Arora, and P. Verma, "Social media classification using convolutional neural networks," International Journal of AI and Data Mining, vol. 19, no. 2, pp. 142–151, 2024.

[18] V. Mehta and S. Roy, "Online threat detection using BERT-based contextual learning," Journal of Advanced NLP Systems, vol. 21, no. 1, pp. 88–97, 2024.

[19] F. Iqbal and R. Khan, "Cyber security analytics using Bi-LSTM networks," International Journal of Secure Computing, vol. 12, no. 3, pp. 176–184, 2024.

[20] P. Arora and D. Joshi, "Social media mining using Support Vector Machine classification," Journal of Intelligent Information Systems, vol. 16, no. 4, pp. 221–229, 2024.

[21] A. Joshi and S. Verma, "AI-based monitoring framework using transformer models," International Journal of Artificial Intelligence Research, vol. 22, no. 1, pp. 101–110, 2025.

[22] M. Tiwari and R. Gupta, "Natural language processing classification using deep neural networks," Journal of Advanced Machine Learning, vol. 20, no. 2, pp. 134–143, 2025.

[23] L. Fernandez and P. Kumar, "AI threat detection using BERT with attention mechanisms," International Journal of Intelligent Systems and Applications, vol. 18, no. 3, pp. 190–199, 2025.

[24] K. Malik and S. Sharma, "Text analytics using Logistic Regression and TF-IDF approaches," Journal of Computational Intelligence Research, vol. 14, no. 2, pp. 111–119, 2025.

[25] N. Saxena and A. Mehra, "Suspicious social media comment detection using CountVectorizer and Logistic Regression," International Journal of Smart Data Analytics, vol. 17, no. 1, pp. 72–81, 2026.