

# Development and Evaluation of Life-Threatening Statement Detection Models for Social Media Platforms Using Supervised Machine Learning Algorithms

Ifeoluwa Michael Olaniyi<sup>1</sup>, Jide Ebenezer Taiwo Akinsola<sup>1</sup>, Maryann Gold Azodo<sup>2</sup>,  
Fathia Oluwadamilola Onipede<sup>1</sup>, Emmanuel Ajayi Olajubu<sup>3</sup>,  
Ganiyu Adesola Aderounmu<sup>3</sup>

<sup>1</sup> Department of Computer Sciences, Abiola Ajimobi Technical University, Ibadan, 200255, Nigeria.

<sup>2</sup> Department of Computer Science, University of Ibadan, Ibadan, 200005, Nigeria.

<sup>3</sup> Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, 220282, Nigeria.

## Article Info

### Article history:

Received March 21, 2025

Revised May 05, 2026

Accepted May 20, 2026

### Keywords:

Cyberbullying

Deep learning

Large language model

Life-threatening statement

Machine learning

Social media platform

## ABSTRACT

In today's digital age, threats to life have become alarmingly common, infiltrating everything from social media platforms to personal communications. About 11.3% of people experienced suicidal ideation over the past year, 5.5% of adults in the United States reported serious suicidal thoughts in the past year, and 10-14% of children and adolescents experienced cyberbullying globally. This study developed machine learning models to detect life-threatening statements across different model categories by employing hyperparameter tuning and Adam optimization. Six models were built using RF, SVM, FFNN, LSTM, pre-trained BERT, and T5, and evaluated using benefit and cost measures and confusion matrices. The outcome indicates that the six models performed well, but the FFNN model outperformed all other models by achieving an accuracy of 98.86%, precision of 98.86%, MCC of 97.72%, Recall of 98.86%, MAE of 0.0113, and RMSE of 0.1067. It also achieved the highest number of accurately classified instances and the lowest number of misclassified instances, which demonstrates that the FFNN model exhibits robustness and precision, making it highly reliable for detecting life-threatening statements compared to other models. Having achieved a very high accuracy, the method still lacks an in-depth understanding of contextual information. Therefore, further studies may focus on incorporating contextual information for more accurate classification of life-threatening statements and integrating the developed model into a real-time detection system.

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.*



**Corresponding Author:** Jide Ebenezer Taiwo Akinsola (e-mail: [akinsolajet@gmail.com](mailto:akinsolajet@gmail.com))

## 1. INTRODUCTION

The global world relies more on the internet for communication and interaction because broadcasting and transferring of information is faster and more efficient with it. Almost every individual around the world uses one or more social media platforms to access and share information online. Social media platforms such as social networking (Facebook, LinkedIn), microblogging (Twitter, Tumblr), photo sharing (Instagram, Snapchat, Pinterest), and video sharing (TikTok, YouTube, Facebook Live) are some of the most used platforms to perform these actions. Social media platforms have become the core of information access and dissemination, and they have been of great value to numerous individuals.

However, the negative effect of it is an increase in cyberbullying. Cyberbullying was defined by Hinduja & Patchin [1] as recurring and willful harm imposed through the means of electronic text. It may include swearing, life-threatening statements, broadcasting pictures without consent, blackmailing or extorting through chat, and spreading pornographic images or private information on the internet. The

uniqueness of bullying on social media is in the extent to which it can be anonymous and its potential to accelerate the contribution of several people in harassing the victims through functions like comments and reposts. Hence, victims often exhibit emotional distress [2], low self-esteem [3], loneliness [4], and social anxiety [5]. These victims also reported more suicidal thoughts and are more likely to attempt suicide, and the actions lead to social and psychological harm among students [6]. A report from Twitter shows that over 1.1 million accounts spread hateful content in the second half of 2020, representing a 77% increase compared to the first half of 2023 [7]. Similarly, a report shows that 50% of sampled students in 2025 experienced cyberbullying, and about 10-14% of children and adolescents experienced cyberbullying globally. In a similar regard, Vidgen et al. [8] highlight the circulation of violent and hateful statements online. A lot of lives have been threatened through cyberbullying, resulting in death, suicide, murder, and severe injury.

These violence-related behaviors remain a global public health concern, with recent reports indicating rising trends among adolescents in the year 2025 [9]. While several studies have explored related areas such as cyberbullying detection [10], abusive content detection [11], suicidal ideation prediction [12], and hate speech prediction [13] using machine learning algorithms and techniques like XGBoost, Logistic Regression, Random Forest, and BERT model, most approaches rely on static text classification and fail to capture crucial patterns, limiting their effectiveness for real-time intervention. Beyond these, existing studies mainly focused on hate speech and abusive statement classification, which results in limited attention to life-threatening statements identification and prediction. In addition, people use negative polarity items and rhetorical strategies like metaphor and irony to mask hateful statements [14] with the aim of deceiving existing detection models. This gap is therefore critical because delayed and inaccurate detection reduces the window for timely intervention and thereby limits the effectiveness of carrying out mitigation strategies in real-world scenarios.

Early identification of life-threatening statements is of great benefit to individuals and society. Detecting statements that indicate self-harm or mischief to others on social media platforms can bring about proper intervention before any mischief occurs. To address these limitations, this study proposes the development of a predictive framework for the identification of life-threatening statements with a focus on negative polarity, using different classes of ML algorithms, deep learning techniques, and pre-trained LLM models. The proposed contribution of this research includes the following:

- Development of supervised ML, deep learning (DL), and LLM-based models for predicting life-threatening statements.
- Fine-tuning of pre-trained LLMs for safety-critical classification tasks.
- Automatic detection of life-threatening statements helps support early intervention efforts.
- Improvement in the recognition of harmful linguistic patterns enhances public safety monitoring.
- The provision of machine-learning models capable of identifying suicidal or violent intent strengthens digital risk-assessment tools.
- Application of deep learning for threat prediction improves speed and accuracy over traditional methods.
- Utilization of pre-trained language models to enhance the ability to understand subtle, context-dependent threats.
- Exploring the effect of negative polarity items in linguistic classification tasks.
- Development of benchmark models to create a foundation for future real-time safety systems.

Upon this motivation, this work builds an accurate and precise model for classifying life-threatening statements utilizing ML techniques.

The rest of this paper is structured as follows: Section 2 examines the related works, which were used to gather all the information essential for successful research, and various works that have been done by different researchers. Section 3 reports a detailed explanation of the materials and techniques utilized in this work. Section 4 gives a thorough analysis of the performance evaluation results, while Section 5 gives the conclusion reached by the authors based on the performance results of the ML models.

## 2. LITERATURE REVIEW

The vast transition to using social media platforms for daily communication has brought about well-known negative cyber-actions that can threaten human life and well-being. Cyber-victimization can result in loneliness among adolescents as well as psychosocial challenges, including depressive symptomatology among the youth. The victims of hate speech on and off social media are threatened mentally [15], and this presents serious safety risks to modern society. Bullying is one of the most relevant risk factors connected with suicide, which has been the second most prominent reason behind the death of youths and young adults in the United States [16]. Cyber-victimization also exists among adults using life-threatening statements and hate speech, and this can lead to violence, severe injury, mental illness, suicide, and murder.

Researchers have made significant contributions to the detection of these statements using various techniques. A Manual approach, such as the use of counterhate speech, was discussed as having a significant impact in fighting and limiting the dissemination of hateful content. Deep learning and supervised learning algorithms are proficient in performing classification tasks efficiently, and their applications are not limited to agriculture [17], healthcare, spam detection, and digit classification [18].

Machine learning methods are also crucial techniques in detecting life-threatening statements. A Participant-vocabulary consistency method for identifying cyberbullying was proposed. The model simultaneously discovers victims, instigators, and word vocabularies that indicate bullying. Similarly, the performance of Extreme Gradient Boosting (XGBoost), Random Forest, and two DL models, BERT and Long Short-Term Memory (LSTM), in classifying hate speech. The result shows that the machine learning models were all bested by deep learning models, achieving an accuracy of 89.59%. The study reported that one of the challenges faced was data imbalance, which can be solved by employing SMOTE. Masking a statement to trick an ML model is possible using negative polarity. Ciaburro et al. [19] employed a Recurrent Neural Network (RNN) for a sentiment analysis-based method to identify the polarity of the message contents of the popular WhatsApp messaging app. The study indicates that the method can be used as a tool to monitor conversations between young people.

Large language models are a great option for classification tasks as they can adapt to contextual information within the dataset. The MAHGA framework was proposed to tackle the drawbacks of contextual relationships in harmful speech detection on social networks [20]. Similarly, [21] focuses on integrating contextual information to enhance the competence of hateful statements classification models. The Arabic BERT-Mini model (ABMM) was introduced for identifying hateful statements in the Arabic language used by social media users. The study fine-tuned the BERT model for classifying data collected on Twitter (X) into hate speech, abusive, and normal, resulting in an accuracy of 0.986. The use of FastText embedding and the BERT model for cyberbullying detection was also employed by Joshi et al. [22]. The result shows that the proposed fusion framework achieved an accuracy of 96.58%. Furthermore, the pre-trained BERT model was employed to handle the challenges of multilingual threatening text classification throughout the Urdu and English fields [23]. Table 1 shows the gap in existing literature. The analysis indicates that there is limited and inadequate research on the classification and detection of life-threatening statements.

While advanced approaches have explored hate speech detection in memes using a contrastive language-image pre-training model [24] and a fusion-based model. Bidirectional Transformer-based models were also employed for detecting life-threatening texts. Three open-source LLMs, including Llama-2, Gemma, and Mistral, were fine-tuned for [25]. The fine-tuned Llama-2 model achieved an accuracy of 98.26%. The study indicates that the dataset used is extremely imbalanced, which could cause the model to be biased. Traditional text-based methods often overlook critical aspects such as contextual information and cultural cues [26]. In this regard, [27] proposed the use of the T5 model due to its text summarization features. [28] discussed the use of the BERT pre-trained model for capturing complex and contextual information in preventing cyberbullying. The model achieved an accuracy of 93.4%.

Table 1 presents a summary of related works and supports the increasing use of machine learning and deep learning methods to identify harmful online content, but, upon a more detailed comparison, the sources of datasets and scale vary greatly, as the studies were based on platforms like Reddit, Twitter, and blogs, with dataset instances varying between 2100 and 60188. Although social media sites such as X (Twitter) can be a valuable source of big data, they are usually noisy and short text-based, and this might not help us understand the context. On the contrary, blog-based datasets can provide more context, but are frequently smaller and less heterogeneous. While some models performed excellently, their practical use remains limited, and they fail in real-time applications because they mostly fail to capture the conceptual nuance that may exist in the grammar when people try to mask a statement using negative polarity items.

Table 1. Gap Analysis Related Works

| S/N | Study | Data source/size | Method used                                     | Acc. (%) | Target threat | Gap  |
|-----|-------|------------------|---|----------|---------------|--|
| 1   | [29]  | Reddit/2100      | Su-RoBERTa                                      | 69.85    | Suicide       | Low accuracy for real-time integration.                                      |
| 2   | [12]  | Twitter/10000    | XGBoost, Random Forest, and Logistic Regression | 99.6     | Suicide       | The study focuses only on classifying suicidal ideation using ML techniques. |

|   |      |               |   |       |                  |   |
|---|------|---------------|---|-------|------------------|---|
| 3 | [28] | Socia media   | RF, LinearSVC, LR, SGD, MultinomialNB, CNN, LSTM, BERT (fine-tuned) | 93.4  | Cyberbullying    | The limitations are multilingualism, dataset imbalance, and the fluid and sarcastic nature of offensive language. |
| 4 | [30] | Twitter/60188 | KNN, Naive Bayes, Decision Tree (DT), SVM, and Random Forest.       | 96.22 | Hate speech      | The study focuses only on hate speech, and only supervised machine learning algorithms were explored.             |
| 5 | [22] | Blog/10000    | mBERT   | 80.42 | Cyberbullying    | The study highlights potential dataset biases. It also focuses generally on cyberbullying.                        |
| 6 | [25] | TEL corpus    | Llama-2, Gemma and Mistral  | 98.26 | Life-threatening | The study indicated that the dataset suffered from extreme dataset imbalance.                                     |

### 3. METHOD

This section discusses the material used in performing the analysis on the dataset used in this study, as well as the methods and techniques employed accordingly.

#### 3.1. Proposed Model Framework

This section discusses the proposed system architecture and the specific process flow diagram for building the models. The architecture of the system is presented in Figure 1. After the data was collected. The collected data was preprocessed accordingly and separated into the training and testing sets using a 70:30 ratio. The training set was utilized to train the model with a grid search technique for hyperparameter tuning to obtain the best parameters for each model. After training, the models' performances were assessed using several evaluation metrics to identify the golden model, and then the golden model was integrated for classifying new data. Figure 2 illustrates the process flow of the models developed.

#### 3.2. Dataset

Data is the most important factor driving a machine learning-based system. Therefore, this study carefully employed different techniques and sources for data collection. Figure 3 illustrates the methods employed for data collection. Three data collection techniques were used. Data were scraped from X (formerly Twitter) using the X API in accordance with the platform's developer and data usage policies. Data were also collected from the TEL Corpus [31], and since a moderately large quantity of data is needed to adequately train a model, synthetic data were carefully collected and scrutinized. Data from these three sources was then merged into a single data bucket. The scraped data on X is labeled as "X API," and it contains 200 instances. The X-API data were carefully analyzed and classified accordingly. 0 represents a non-life-threatening statement while 1 represents a life-threatening statement. The data collected from the TEL corpus contains 299 instances and was labeled as "TEL Corpus". The generated synthetic data, which contains 4500 instances, was labeled as "Synthetic Data". Classes were generated with the data, and they were carefully analyzed to ensure each statement conforms to the assigned class. The total number of data instances collected before preprocessing was performed is 4999 instances. These data were merged and stored in a dataset labeled as "Life Threatening Statement Dataset". Table 2 shows the data distribution per source before preprocessing.

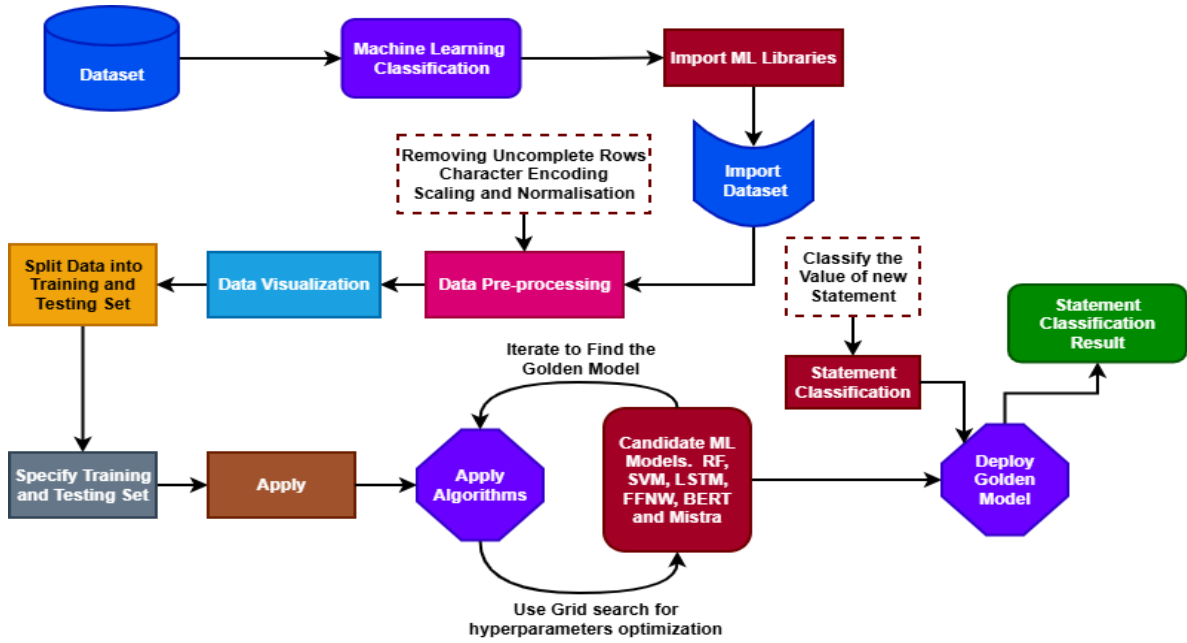


Figure 1. Life-Threatening Statement Detection System Architecture

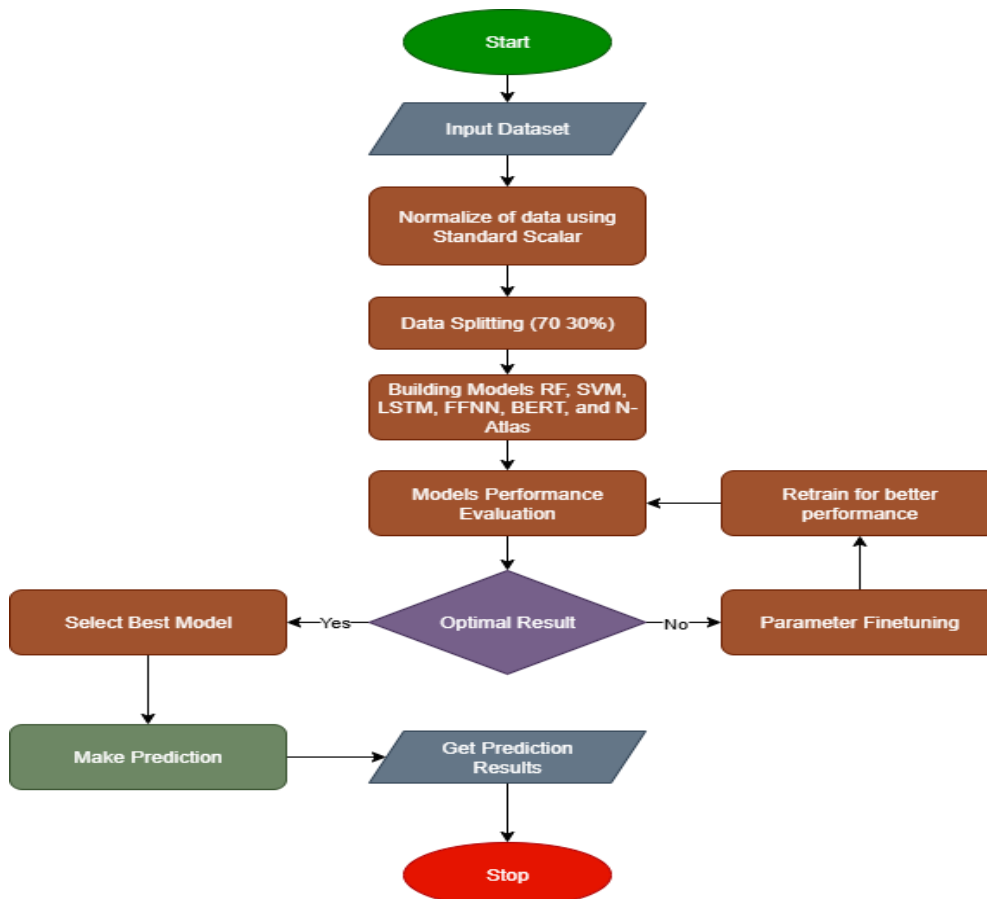


Figure 2. Life-Threatening Statement Detection Process Flow

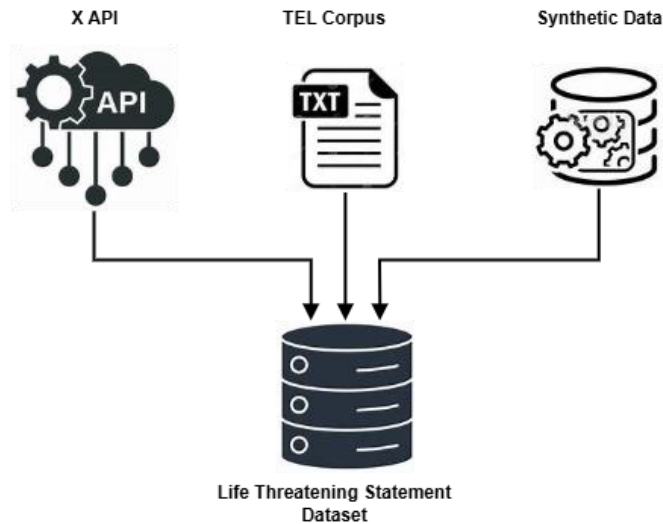


Figure 3. Data Collection for Life-Threatening Statement Implementation

Table 2. Data Distribution per Source

| S/N | Source         | Number of Samples |
|-----|----------------|-------------------|
| 1.  | X API          | 200               |
| 2.  | TEL Corpus     | 299               |
| 3.  | Synthetic Data | 4500              |
|     | Total          | 4,999             |

### 3.3. Application of Models for Life-Threatening Statement Detection

This study employed two supervised ML algorithms, two deep learning algorithms, and two LLM models for the classification and detection of life-threatening statements. Multiple algorithms were employed to ensure that the results are not based on a single approach or assumption. Different algorithms learn patterns in different ways. These algorithms and models include Random Forest (RF), Support Vector Machine (SVM), Long Short-Term Memory (LSTM), Feedforward Neural Network (FFNN), Bidirectional Encoder Representations from Transformers (BERT), and Text to Text Transfer Transformer (T5).

#### 3.3.1. Random Forest

RF is a common and effective supervised learning algorithm, capable of solving regression and classification problems. One of the advantages of RF is that overfitting of the training set is less of an issue [32]. It works by building multiple DTs during training and uses the result of each decision tree to make a better prediction. The relevance of features on a DT is determined as shown in (1).

$$f_i = \frac{\sum_{j:\text{node } j \text{ splits on feature } i} n_{ij}}{\sum_{k:\text{all nodes}} n_{ik}} \quad (1)$$

Here,  $f_i$  denotes the significance of parameter  $i$ , and the significance of node  $j$  is denoted by  $n_{ij}$ . The RF algorithm was optimized using a grid search to identify a suitable combination of hyperparameters. The parameter grid explores variations in the number of trees, tree depth, and node splitting criteria to balance model complexity, generalization, and avoid overfitting. A 3-fold cross-validation scheme was employed to ensure robust performance evaluation across different data partitions. Model selection was based on classification accuracy as the primary evaluation metric. To enhance computational efficiency, parallel processing was enabled by utilizing all available processor cores, and a fixed random state was applied to ensure reproducibility of results, as shown in Figure 4. Table 3 shows the hyperparameter settings for the RF algorithm.

Table 3. Hyperparameter Settings for the RF Algorithm

| S/N | Parameters           | Value  |
|-----|----------------------|--|
| 1.  | GidSearch parameters | n_estimators: [100, 200],<br>max_depth: [None, 20, 50],<br>min_samples_split: [2, 5] |
| 2.  | Random state         | 42   |
| 3.  | cv                   | 3  |
| 4.  | verbose              | 2  |
| 5.  | Number of jobs       | -1   |
| 6.  | Scoring              | Accuracy   |

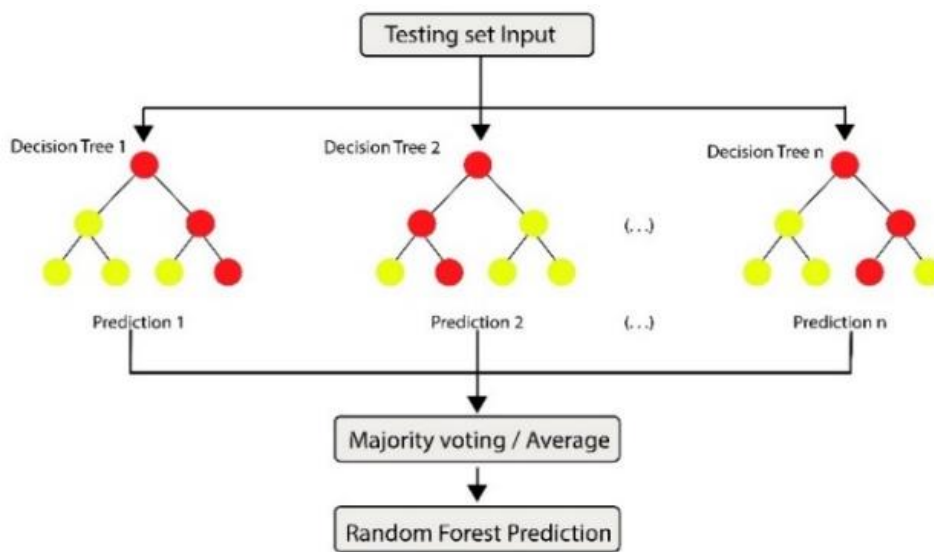


Figure 4. Random Forest Architecture

**3.3.2. Support Vector Machine**

SVM’s primary goal is to discover the best hyperplane that linearly divides the instances into two components during margin optimization [33]. Figure 5 shows the mathematical interpretation of the optimal hyperplane. The linear hyperplane equation is expressed in (2).

$$w^T x + b = 0 \tag{2}$$

Here, the normal vector (w) is perpendicular to the hyperplane, and b is the offset or bias term that represents the gap of the hyperplane from its starting point along the normal vector (w).

Table 4. Hyperparameter Settings for SVM Algorithm

| S/N | Parameters           | Value  |
|-----|----------------------|--|
| 1.  | GidSearch parameters | C: [0.1, 1, 10],<br>kernel: ["linear", "rbf"],<br>gamma: ["scale", "auto"] |
| 2.  | cv                   | 3  |
| 3.  | verbose              | 2  |
| 4.  | Number of jobs       | -1   |
| 5.  | Scoring              | Accuracy   |

Table 4 shows the parameter settings for the SVM algorithm used to train the model. Grid search was employed for the best hyperparameter configuration. The regularization parameter ( $C$ ) was varied across values of 0.1, 1, and 10 to control the trade-off between maximizing the margin and minimizing classification error. Two kernel functions, linear and radial basis function (RBF), were considered to evaluate both linear and non-linear decision boundaries, while the gamma parameter defines the influence of individual training samples in the RBF kernel. The model selection was also performed using 3-fold cross-validation and all available processor cores.

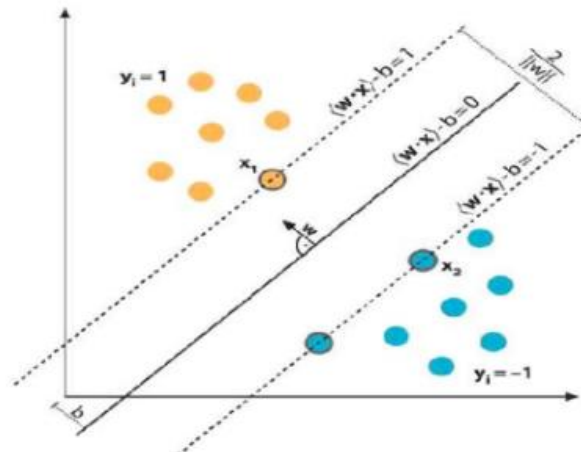


Figure 5. SVM Optimal Hyperplane

### 3.3.3. Long Short-Term Memory

LSTM is known to be a powerful RNN model built to avoid and eradicate the vanishing and slope problems that arise naturally during dependency learning. Such problems are long-term, even when the minimal lag time is too long. The error signal between each cell's units is prevented using a constant error carousel (CEC). The cells are recurrent networks themselves with an architecture that is fantastic in the way that there is an extension of the CEC with extra features, which forms the cell of the memory through the input and output gates. The feedback is illustrated using connections that are self-recurrent with a one-time lag step [34]. The architecture is presented in Figure 6.

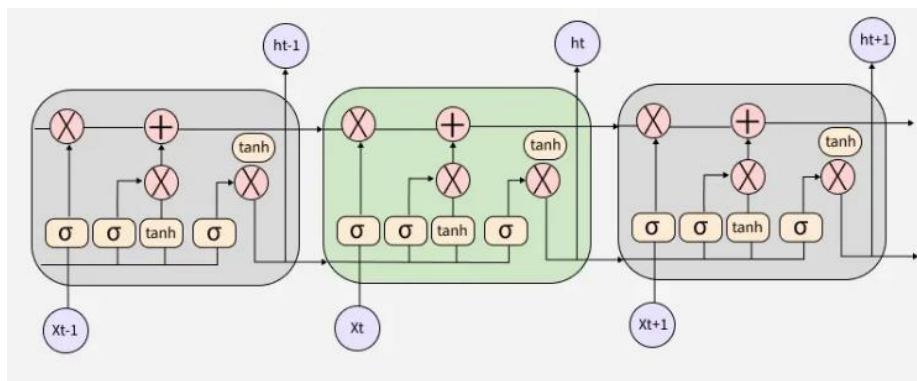


Figure 6. LSTM Architecture

The LSTM modules consist of three internal gates, which are the input, output, and the forgotten gate. If the forgetting gate case, it incorporates a tangent hyperbolic layer. The forgotten gate permits the addition or removal of information to the cell state, known to be a link that allows the transfer of information from one module of LSTM to another. The entry of other information is controlled by the input gate. Forgotten information is controlled by the forgotten gate by permitting discrimination between important data and unnecessary data by the cell state, while providing space for fresh data. Because of this, a hyperbolic tangent layer is brought in combination with the sigmoid layer. The memory outcome that is saved within the cell state is regulated by the output gate. The cell state possesses the function for optimizing weight depending on the output error result of the network, which controls the individual gate. The state of the cell and the output value obtained by the module are moved to the next module of the LSTM [35]. Table 5 presents the hyperparameter settings for the LSTM algorithm. The model begins with an embedding layer,

which transforms each word into a dense vector representation, allowing the network to learn semantic relationships between words. A spatial dropout layer is applied to reduce overfitting by randomly dropping entire word embeddings during training. A bidirectional LSTM is used to capture context from both past and future words in the sequence, which improves understanding of sentence meaning. The learned features are then passed through a dense layer for further refinement, followed by dropout for regularization. Finally, a softmax output layer predicts the class probabilities. The model is trained using the Adam optimizer, which provides efficient and adaptive learning.

Table 5. Hyperparameter Settings for the LSTM Algorithm

| S/N | Parameters         | Value   |
|-----|--------------------|---|
| 1.  | Embedding Layer    | input_dim = 20,000, output_dim = 128 (embedding size), input_length = 200 |
| 2.  | Spatial Dropout    | 0.2   |
| 3.  | Bidirectional LSTM | 128 units, return_sequences = False                                       |
| 4.  | Dense Layer        | 128 neurons, ReLU activation, Dropout = 0.3                               |
| 5.  | Output Layer       | num_classes neurons, Softmax activation                                   |
| 6.  | Optimizer          | Adam  |
| 7.  | Epochs             | 6   |

### 3.3.4. Feedforward Neural Network

FFNN is a kind of ANN in which the flow of information is directly from the input layer, goes through the hidden layer, and goes to the output layer. It is a kind of neural network (NN) with connected nodes that do not form a circle. The functions in an FFNN include the cost function, log loss function, gradient learning algorithm, and output unit. Figure 7 illustrates the algorithm architecture, while Table 6 shows the hyperparameter settings for the FFNN model. The model uses a TF-IDF vectorizer to convert text into numerical features, capturing both single words and two-word phrases (bigrams) while limiting the vocabulary to the 10,000 most important terms [36].

The feedforward neural network (FFNN) is designed with two hidden layers to learn patterns from these features. The first layer is relatively large (512 neurons) to capture complex relationships, while the second layer (256 neurons) refines these learned patterns. Dropout layers are introduced to prevent overfitting by randomly disabling some neurons during training. The model is trained using the Adam optimizer, which adapts learning efficiently, and uses sparse categorical cross-entropy as the loss function since the labels are integers.

Table 6. Hyperparameter Settings for FFNN Algorithm

| S/N | Parameters           | Value   |
|-----|----------------------|---|
| 1.  | TF-IDF Vectorization | max_features = 10,000, ngram_range = (1, 2)   |
| 2.  | Hidden Layer 1       | 512 neurons, ReLU activation, Dropout 1 = 0.3 |
| 3.  | Hidden Layer 2       | 256 neurons, ReLU activation, Dropout 2 = 0.2 |
| 4.  | Output Layer         | num_classes neurons, Softmax activation       |
| 5.  | Optimizer            | Adam  |
| 6.  | Epochs               | 6   |

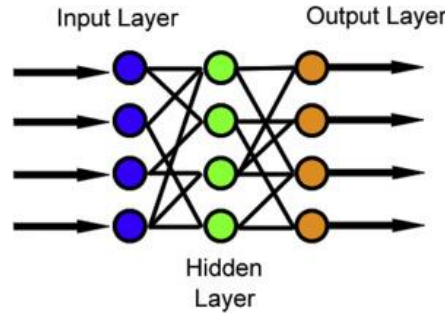


Figure 7. General model of an FFNN

**3.3.5. BERT Model**

The BERT model is a large language model that uses the encoder-only transformer architecture [37]. BERT is a pile of transformer encoder layers [38] comprising several self-attention heads. Each head calculates key, value, and query vectors for each input token in a sequence, which are then utilized to produce a weighted representation [39]. A fully connected layer is created by combining the heads’ outputs of a matching layer. Layer normalization comes after wrapping each one of them with a skip connection. The pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering, language inference, and classification [40]. Figure 8 shows the BERT architecture, and Table 7 presents the hyperparameter settings for the model.

A pre-trained BERT (bert-base-uncased) model is fine-tuned for text classification. Instead of building representations from scratch, the model leverages knowledge learned from large-scale text data. The text is tokenized using BERT’s tokenizer, which converts sentences into token IDs while ensuring all inputs have the same length through padding and truncation. Attention masks are also created to help the model distinguish between actual text and padded values. The classification model adds a final layer on top of BERT to predict the target classes. A relatively small batch size is used due to the computational cost of BERT, and the model is trained for a few epochs to fine-tune its pre-trained knowledge without overfitting.

Table 7. Hyperparameter Settings for BERT Model

| S/N | Parameters            | Value             |
|-----|-----------------------|-------------------|
| 1.  | Pre-trained model     | bert-base-uncased |
| 2.  | Tokenizer             | BERT tokenizer    |
| 3.  | Padding               | max_length        |
| 4.  | Truncation            | True              |
| 5.  | Train Batch Size      | 8                 |
| 6.  | Evaluation Batch Size | 8                 |
| 7.  | Epochs                | 4                 |

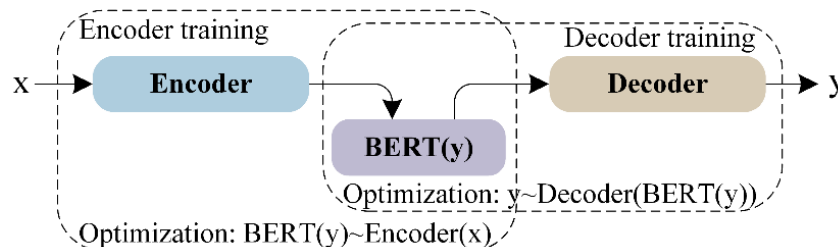


Figure 8. BERT supervised training architecture

**3.3.6. T5 Pre-trained Model**

The Text-to-Text Transfer Transformer (T5) model is a pre-trained model with an encoder-decoder architecture that has been pre-trained on a diverse set of both unsupervised and supervised tasks. Its unique design allows it to handle various natural language processing (NLP) tasks by converting each task into a

consolidated text-to-text representation. In contrast to the standard approaches that use task-based architectures, T5 generalizes across multiple tasks using a single framework.

One of T5's key features is its ability to differentiate tasks using specific prefixes added to the input text. For example, when performing machine translation, a prefix such as "translate English to German:" is prepended to the input text, guiding the model toward the expected task. Similarly, for summarization tasks, the input is prefixed with "summarize:" to indicate that the model should generate a concise summary of the given text. Due to its flexible and scalable architecture, T5 performs well on a wide range of NLP applications, including text generation, summarization, translation, and question-answering. By leveraging transfer learning and large-scale pre-training, the T5 model has been a grounded state-of-the-art model, making it a powerful and adaptable model for various text-based AI tasks. Figure 9 shows the T5 model architecture, while Table 8 illustrates the hyperparameter settings.

A pre-trained T5 (t5-small) model is used for text classification by framing the task as a text-to-text problem. Each input sentence is prefixed with "classify: " to guide the model toward the classification task. The tokenizer converts both the input text and the target labels into token IDs, ensuring fixed lengths through padding and truncation. The model itself is a sequence-to-sequence transformer, meaning it reads the input text and generates the corresponding label as output. During training, the model learns to map input sequences to their correct textual labels. For evaluation, the generated predictions are decoded back into text and compared with the true labels using accuracy. Like the BERT setup, training uses small batch sizes due to computational cost.

Table 8. Hyperparameter Settings for T5 Model

| S/N | Parameters            | Value   |
|-----|-----------------------|---|
| 1.  | Pre-trained model     | t5-small                                      |
| 2.  | Task format           | Text-to-text ("classify: <text>" - "<label>") |
| 3.  | Padding               | max_length                                    |
| 4.  | Truncation            | True  |
| 5.  | Train Batch Size      | 8   |
| 6.  | Evaluation Batch Size | 8   |
| 7.  | Epochs                | 4   |

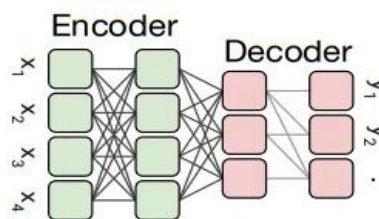


Figure 9. T5 model architecture

#### 4. RESULTS AND DISCUSSION

The models developed were analyzed using performance evaluation metrics commonly used for classification algorithms. These metrics include F1-score, accuracy, precision, and Recall. The experiment in this study was conducted on Kaggle Notebook with Keras provision, and the TensorFlow library was employed for building the two deep learning models.

##### 4.1. Data Preprocessing and Exploratory Data Analysis (EDA)

Data preprocessing was performed to prepare the dataset for model development. The presence of null entries was discovered, and the affected instances were removed accordingly. The total number of null entries removed is 169, and this results in a total of 4830 instances of data left for model development. Table 9 shows the normalization techniques employed to normalize the data before training for each model.

Table 9. Normalization Techniques

| S/N | Model | Technique   |
|-----|-------|---|
| 1.  | RF    | TF-IDF + L2 normalization                                   |
| 2.  | SVM   | TF-IDF + L2 normalization                                   |
| 3.  | LSTM  | Tokenization + padding/truncation                           |
| 4.  | FFNN  | TF-IDF + L2 normalization                                   |
| 5.  | BERT  | WordPiece tokenization + pretrained embedding normalization |
| 6.  | T5    | Subword tokenization + pretrained embedding normalization   |

Exploratory data analysis was carried out on the collected dataset to analyze patterns, trends, and word usage. Different techniques and data visualization tools were used to perform this operation. Exploratory Data Analysis (EDA) plays an important role in shaping how a model is developed because it helps in understanding the dataset before training begins. By examining the data, issues like missing values, noise, or class imbalance can be spotted, which then guide the data cleaning process. It also reveals patterns and relationships within the data, which can help in deciding the most suitable feature representation, such as whether to use TF-IDF or embeddings. In addition, insights from EDA can influence the choice of model, since the nature and size of the data often determine whether a classical machine learning approach or a deep learning model is more appropriate. Overall, EDA ensures that decisions made during model development are based on actual data characteristics, leading to more accurate and reliable results. Table 10 and Figure 10 show the sample distribution across the two classes available in the dataset.

Table 10. Distribution Sample per Class for Dataset

| S/N | Class                           | Number of Samples |
|-----|---------------------------------|-------------------|
| 1.  | Life-Threatening Statements     | 2,436             |
| 2.  | Non-Life-Threatening Statements | 2,394             |
|     | Total                           | 4,830             |

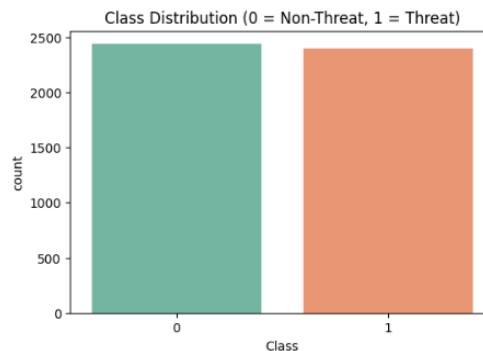


Figure 10. Class Distribution

The sample distribution presented in Table 10 shows that the dataset used for this study contained 4830 instances, of which 2436 were life-threatening statements, and 2394 were non-life-threatening statements. This indicates a strong balance between the two classes. Analyses were also performed at the feature level. Figure 11 (a) and Table 11 show the word count distribution by class. The average word count for threatening statements is approximately 7 words, while the average for non-threatening statements is approximately 13 words. The character count for threatening and non-threatening statements is approximately 44 and 88, respectively. This indicates that life-threatening statements tend to have shorter sentences compared to non-threatening statements and may be more direct and concise. Life-threatening statements show higher negative polarity, which indicates that the statements are perceived to be more forceful, and the user may intentionally use negative polarity items to deceive the existing traditional model.

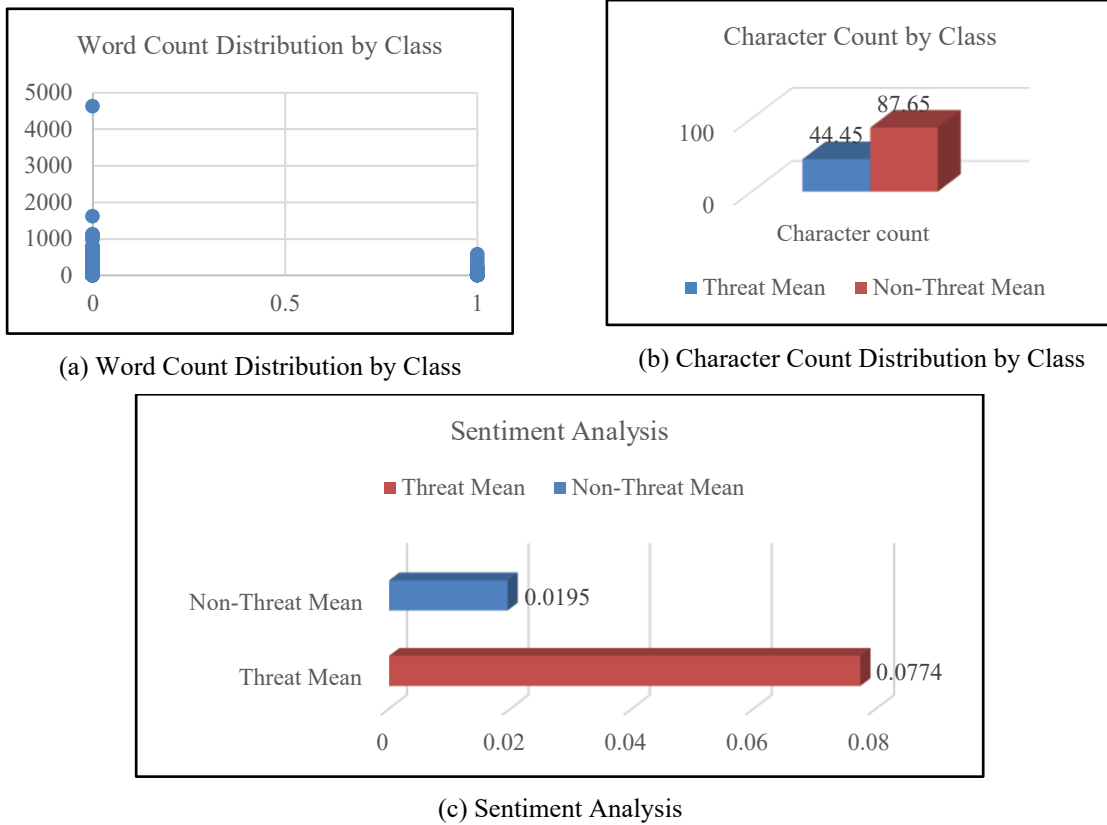


Figure 11. These are figures for feature-level analyses. (a) shows the graphical representation of word count by class; (b) shows the graphical representation of character count by class, and (c) shows the graphical representation of the sentiment analysis regarding negative polarity.

Table 11. Feature-Level Analyses

| S/N | Feature         | Threat Mean | Non-Threat Mean |
|-----|-----------------|-------------|-----------------|
| 1.  | Word count      | 7.25        | 13.25           |
| 2.  | Character count | 44.45       | 87.65           |
| 3.  | Sentiment       | 0.0774      | 0.0195          |

#### 4.2. Performance Evaluation

The models were evaluated using different performance evaluation metrics. These metrics include the benefit metrics, cost metrics, and confusion matrix. Table 12, Figure 12, and Figure 13 show the results of the evaluation.

Table 12. Models' Performance Evaluation

| S/N | Models | Benefit Measures |           |        |        |          | Cost Measures |        |        |
|-----|--------|------------------|-----------|--------|--------|----------|---------------|--------|--------|
|     |        | Accuracy         | Precision | Recall | MCC    | F1-Score | MAE           | MSE    | RMSE   |
| 1   | RF     | 0.9855           | 0.9857    | 0.9855 | 0.9712 | 0.9855   | 0.0144        | 0.0144 | 0.1203 |
| 2   | SVM    | 0.9844           | 0.9844    | 0.9844 | 0.9689 | 0.9844   | 0.0155        | 0.0155 | 0.1246 |
| 3   | FFNN   | 0.9886           | 0.9886    | 0.9886 | 0.9772 | 0.9886   | 0.0113        | 0.0113 | 0.1067 |
| 4   | LSTM   | 0.9689           | 0.9693    | 0.9689 | 0.9382 | 0.9689   | 0.0310        | 0.0310 | 0.1762 |
| 5   | BERT   | 0.9855           | 0.9855    | 0.9855 | 0.9710 | 0.9855   | 0.0144        | 0.0144 | 0.1203 |
| 6   | T5     | 0.9803           | 0.9803    | 0.9803 | 0.9860 | 0.9803   | 0.0196        | 0.0196 | 0.1402 |

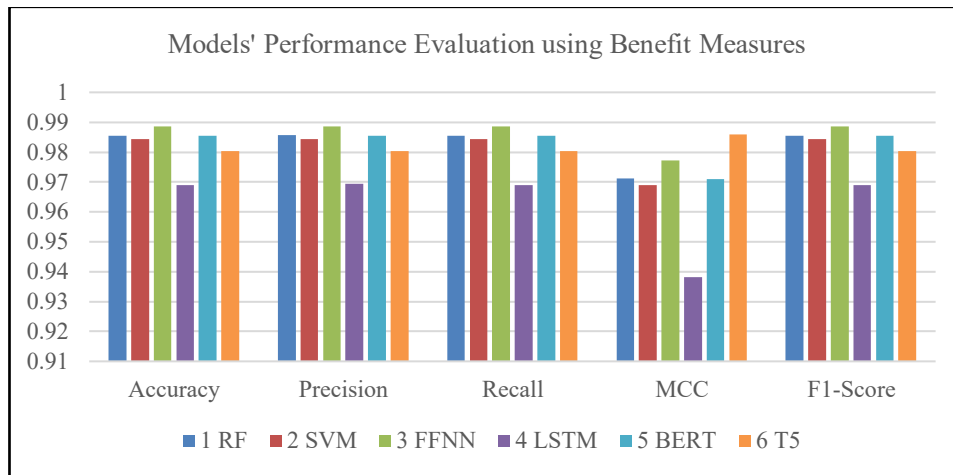


Figure 12. Models' Performance Evaluation using benefit measures

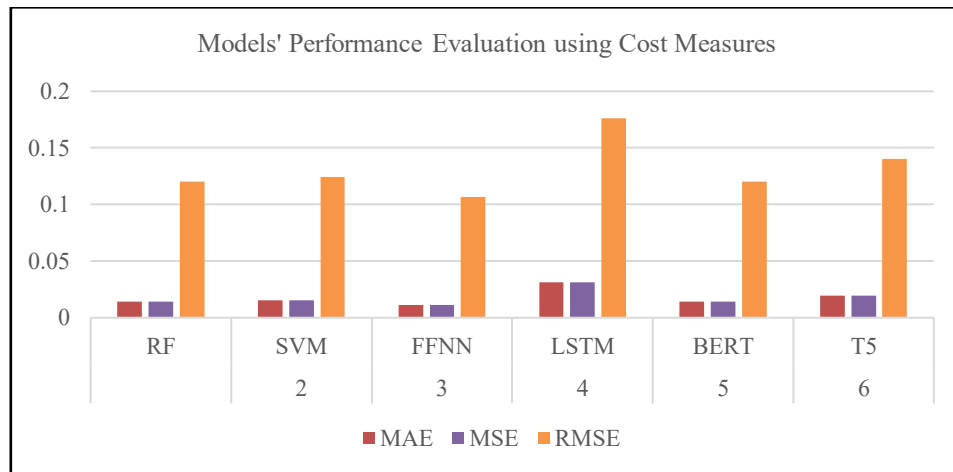


Figure 13. Models' Performance Evaluation using cost measures

All the models achieved an accuracy above 96%, indicating good performance across the models. The result shows that RF model built achieved an 98.55% accuracy, 98.57% precision, 0.0144 MAE, and RMSE value of 0.1203; the SVM model attained an accuracy of 98.44%, 98.44 precision, 0.0155 MAE, and 0.1246 RMSE; the FFNN model attained 98.86% accuracy, 98.86 precision, 0.0113 MAE, and 0.1067 RMSE; the LSTM model attained 96.89% accuracy, 96.93 precision, 0.0310 MAE, and 0.1762 RMSE; the BERT model achieved an accuracy of 98.55%, precision of 98.55, MAE of 0.0144, and RMSE value of 0.1203; and the T5 model attained 98.03% accuracy, 98.03 precision, 0.0196 MAE, and 0.1402 RMSE value. Furthermore, the models were evaluated using confusion matrices. Figure 14 illustrates the confusion matrices of the developed models, and a summary of the figures is presented in Table 13.

Table 13. Models' Performance Evaluation using the Confusion Matrix

| S/N | Models | TP  | FP | FN | TN  |
|-----|--------|-----|----|----|-----|
| 1   | RF     | 467 | 2  | 12 | 485 |
| 2   | SVM    | 471 | 7  | 8  | 480 |
| 3   | FFNN   | 472 | 4  | 7  | 483 |
| 4   | LSTM   | 471 | 22 | 8  | 465 |
| 5   | BERT   | 468 | 5  | 9  | 484 |
| 6   | T5     | 466 | 8  | 11 | 481 |

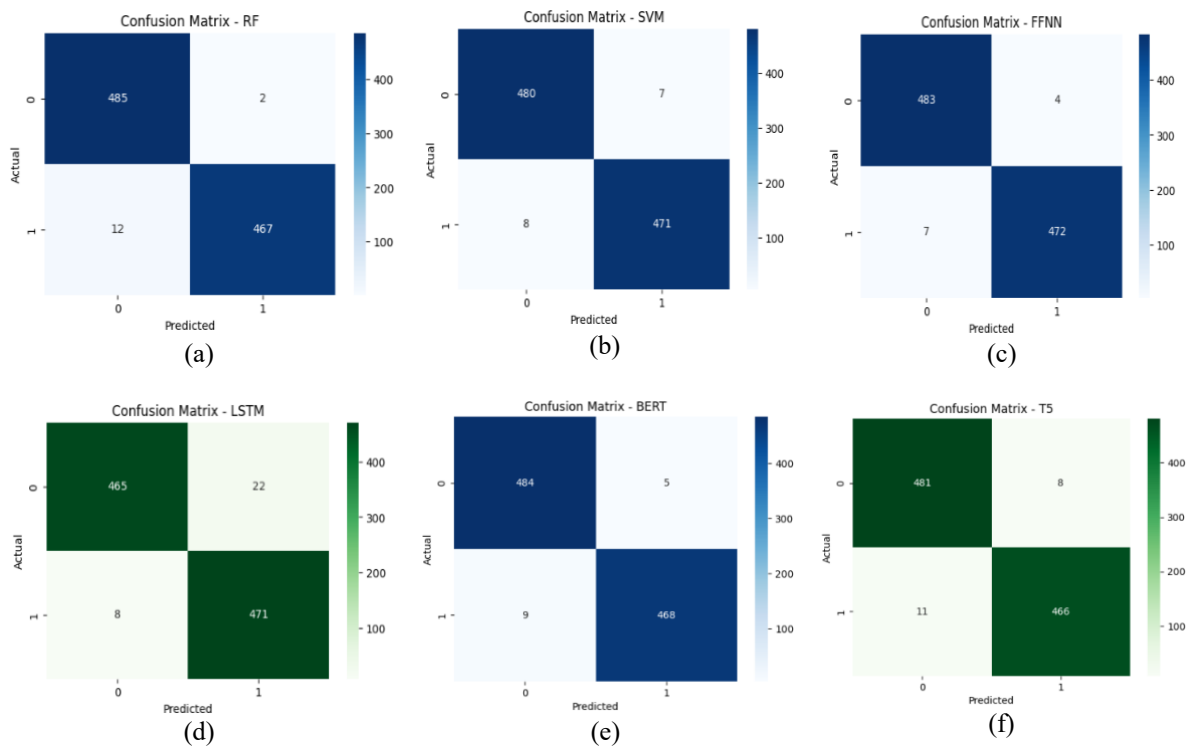


Figure 14. These are figures for the confusion matrices. (a) shows the CM for the RF model; (b) shows the CM for the SVM model; (c) shows the CM for the FFNN model; (d) shows the CM for the LSTM model; (e) shows the CM for the BERT model; and (f) shows the CM for the T5 model.

Table 13 shows that the RF model was unable to classify 14 instances accurately, the SVM model misclassified 15 instances, the FFNN model misclassified 11 instances, the LSTM model misclassified 30 instances, the BERT model misclassified 14 instances, and the T5 model misclassified 19 instances.

#### 4.3. Discussion of Findings

The exploratory data analysis result shows that life-threatening statements have fewer words compared to non-life-threatening statements. This indicates that rather than telling the actual cause of the circumstance or the actions that warrant the statement, people who engage in threatening others or themselves on social media platforms often use statements that are concise and direct, stating only the harm they intend to cause. According to the results of the performance evaluation metrics, the FFNN model provides the best performance score with respect to benefit and cost measures by achieving the highest benefit measures and lowest cost measures compared to other models evaluated. It also surpassed the other model in terms of performance evaluation using confusion matrices. It has the highest instances accurately classified and the lowest misclassified instances. This indicates that the FFNN model demonstrates better robustness and precision, making it highly reliable for detecting life-threatening statements compared to other models.

The superior performance of the FFNN model can be attributed to the characteristics of the dataset and the feature representation used. Since the dataset was relatively structured and represented using TF-IDF features, the FFNN was able to effectively learn from these high-quality sparse representations. In contrast, transformer-based models such as BERT and T5 typically require larger datasets and more extensive fine-tuning to fully leverage their contextual learning capabilities. The limited dataset size and relatively short training epochs may have constrained their performance in this study. Additionally, the computational setup and hyperparameter configuration may not have been fully optimized for deep transformer models. Overall, these factors likely contributed to the observed result where the simpler FFNN model achieved better performance in this specific experimental setting. Table 14 presents the comparison between this study and existing related work. The Golden models from the literature reviewed were compared to the Golden model developed in this study.

Table 14. Benchmark with Existing Literature

| S/N | Literature         | Golden Model | Dataset source  | Accuracy | F1 score |
|-----|--------------------|--------------|---|----------|----------|
| 1   | [41]               | BERT-CNN     | X (Twitter)   | 92.34    | -        |
| 2   | [25]               | Llama-2      | Dynamically generated hate speech" dataset and TEL corpus | 96.96    | 89.77    |
| 3   | [29]               | BERT         | Multi-source (Mandl, Basile, Dynamic)                     | 86.67    | 79.13    |
| 4   | [22]               | Fusion model | Hinglish blog corpus                                      | 96.58    | 96.59    |
| 5   | The proposed model | FFNN         | X (Twitter), TEL corpus, and synthetic dataset.           | 98.86    | 98.86    |

The result shows that the Feedforward Neural Network (Golden) model built in this work outperformed all models in the existing literature in terms of accuracy and F1-score. The use of grid search is of great advantage in model building as it helps to find the most effective configuration for a model in a structured way. Instead of guessing which parameters work best, it tests all possible combinations within a given range and identifies the one that produces the best result. This makes the model more accurate and reliable, as it is tuned based on evidence rather than trial and error. It also leads to more optimized and robust model performance, especially when combined with cross-validation for the reduction of model overfitting and improvement of generalization.

## 5. CONCLUSION

This study applied multiple categories of machine learning and deep learning algorithms with large Language Models for classifying life-threatening and non-threatening statements. The results show that the FFNN model achieved an accuracy of 98.86%, precision of 98.86%, MCC of 97.72%, Recall of 98.86%, MAE of 0.0113, and RMSE of 0.1067. The confusion matrix shows that the FFNN model was able to classify 955 of 966 instances of the testing set correctly, which is more than other models. This confirmed that the Feedforward Neural Network model consistently outperformed the others, including the BERT and T5 models, which are LLMs. In conclusion, the analysis of the life-threatening dataset demonstrates the critical role of ML techniques in comprehending, identifying, and detecting the class a statement may belong to, which could be of great importance in scrutinizing and protecting the society we have found ourselves in. By leveraging machine learning for classification and interpretability, we can uncover hidden patterns that support timely decision-making and intervention. The study highlights that, beyond achieving high accuracy, data quality and in-depth understanding of contextual information are crucial in tasks such as detecting life-threatening statements to avoid high false positives or false negatives, as they could trigger the system to give a false alarm. The limitations of the study include a limited dataset size, the inability to fully capture contextual information, and the real-time deployment of the golden model. Therefore, it is recommended that further studies focus on increasing the dataset to capture more diverse patterns, integrating explainability techniques for model interpretation, incorporating advanced techniques for contextual understanding to build a more reliable model for classifying and detecting life-threatening statements, and real-time implementation.

## DATA AVAILABILITY STATEMENT

The data presented in this study are available on request from the corresponding author.

## CONFLICTS OF INTEREST

The authors declare that they have no conflict of interest.

## REFERENCES

- [1] S. Hinduja and J. W. Patchin, "Cyberbullying: An Exploratory Analysis of Factors Related to Offending and Victimization," *Deviant Behavior*, vol. 29, no. 2, pp. 129–156, Jan. 2008, <https://doi.org/10.1080/01639620701457816>.
- [2] M. L. Ybarra and K. J. Mitchell, "Online aggressor/targets, aggressors, and targets: a comparison of associated youth characteristics," *Journal of Child Psychology and Psychiatry*, vol. 45, no. 7, pp. 1308–1316, Oct. 2004, <https://doi.org/10.1111/j.1469-7610.2004.00328.x>.
- [3] J. W. Patchin and S. Hinduja, "Measuring cyberbullying: Implications for research," *Aggression and Violent Behavior*, vol. 23, pp. 69–74, Jul. 2015, <https://doi.org/10.1016/j.avb.2015.05.013>.
- [4] M. ŞAHİN, "The relationship between the cyberbullying/cybervictimization and loneliness among adolescents,"

- Children and Youth Services Review*, vol. 34, no. 4, pp. 834–837, Apr. 2012, <https://doi.org/10.1016/j.childyouth.2012.01.010>.
- [5] J. Juvonen and E. F. Gross, "Extending the School Grounds?—Bullying Experiences in Cyberspace," *Journal of School Health*, vol. 78, no. 9, pp. 496–505, Sep. 2008, <https://doi.org/10.1111/j.1746-1561.2008.00335.x>.
- [6] B. P. Nkwanyana, "Psychosocial effects of cyberbullying in selected secondary schools, KwaZulu-Natal, South Africa," *Interdisciplinary Journal of Education Research*, vol. 7, no. s1, p. a12, Apr. 2025, <https://doi.org/10.38140/ijer-2025.vol7.s1.12>.
- [7] A. Albanyan, A. Hassan, and E. Blanco, "Not All Counterhate Tweets Elicit the Same Replies: A Fine-Grained Analysis," in *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (\*SEM 2023)*, 2023, pp. 71–88, <https://doi.org/10.18653/v1/2023.starsem-1.8>.
- [8] B. Vidgen *et al.*, "Detecting East Asian Prejudice on Social Media," in *Proceedings of the Fourth Workshop on Online Abuse and Harms*, 2020, pp. 162–172, <https://doi.org/10.18653/v1/2020.alw-1.19>.
- [9] E. Jackson, M. Boyes, B. Blundell, J. Hirn, and S. Leitão, "Managing cyberbullying among adolescents with neurodevelopmental disorders: A scoping review," *International Journal of Speech-Language Pathology*, vol. 27, no. 3, pp. 341–356, May 2025, <https://doi.org/10.1080/17549507.2025.2451305>.
- [10] E. Raisi and B. Huang, "Cyberbullying Identification Using Participant-Vocabulary Consistency," Jun. 2016, <https://doi.org/10.48550/arXiv.1606.08084>.
- [11] M. Almaliki, A. M. Almars, I. Gad, and E.-S. Atlam, "ABMM: Arabic BERT-Mini Model for Hate-Speech Detection on Social Media," *Electronics*, vol. 12, no. 4, p. 1048, Feb. 2023, <https://doi.org/10.3390/electronics12041048>.
- [12] E. Rajesh Kumar, K. V. S. N. Rama Rao, S. R. Nayak, and R. Chandra, "Suicidal ideation prediction in twitter data using machine learning techniques," *Journal of Interdisciplinary Mathematics*, vol. 23, no. 1, pp. 117–125, Jan. 2020, <https://doi.org/10.1080/09720502.2020.1721674>.
- [13] N. Maqbool, A. Anjum, M. Z. Hussain, T. Aslam, A. Yaseen, and M. Z. Hasan, "FROM DETECTION TO PRECISION: ELEVATING HATE SPEECH CLASSIFICATION WITH CUTTING-EDGE MODELS," *Spectrum of Engineering Sciences*, vol. 3, no. 5, pp. 143–158, 2025.
- [14] C. Ren, D. Jeong, M. Wu, Y. Huang, Y. Gao, and Y. Li, "A survey of multimodal hate meme detection," *Expert Systems with Applications*, vol. 323, p. 132507, Aug. 2026, <https://doi.org/10.1016/j.eswa.2026.132507>.
- [15] K. Solovev and N. Pröllochs, "Moralized language predicts hate speech on social media," *PNAS Nexus*, vol. 2, no. 1, Jan. 2023, <https://doi.org/10.1093/pnasnexus/pgac281>.
- [16] S. Arnon *et al.*, "Association of Cyberbullying Experiences and Perpetration With Suicidality in Early Adolescence," *JAMA Network Open*, vol. 5, no. 6, p. e2218746, Jun. 2022, <https://doi.org/10.1001/jamanetworkopen.2022.18746>.
- [17] J. O. Okesola *et al.*, "Predictive analytics on crop yield using supervised learning techniques," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 36, no. 3, p. 1664, Dec. 2024, <https://doi.org/10.11591/ijeecs.v36.i3.pp1664-1673>.
- [18] J. E. T. Akinsola, M. A. Olatunbosun, I. M. Olaniyi, M. A. Adeagbo, E. A. Olajubu, and G. A. Aderounmu, "Application of Artificial Intelligence on MNIST Dataset for Handwritten Digit Classification for Evaluation of Deep Learning Models," *Acadlore Transactions on AI and Machine Learning*, vol. 4, no. 3, pp. 219–234, 2025, <https://doi.org/https://doi.org/10.56578/ataiml040305>.
- [19] G. Ciaburro, G. Iannace, and V. Puyana-Romero, "Sentiment Analysis-Based Method to Prevent Cyber Bullying," 2022, pp. 721–735.
- [20] R. Yoshida, S. Yoshida, and M. Muneyasu, "MAHGA: Multi-Aspect Heterogeneous Graph Analysis for Harmful Speech Detection on Social Networks," *IEEE Access*, vol. 13, pp. 106673–106689, 2025, <https://doi.org/10.1109/ACCESS.2025.3581214>.
- [21] J. M. Pérez *et al.*, "Assessing the Impact of Contextual Information in Hate Speech Detection," *IEEE Access*, vol. 11, pp. 30575–30590, 2023, <https://doi.org/10.1109/ACCESS.2023.3258973>.
- [22] M. Joshi, D. Pandey, V. Pandey, and M. W. Khan, "A fusion framework for Hinglish cyberbullying detection using mBERT and FastText," *International Journal of Engineering in Computer Science*, vol. 7, no. 1, pp. 7–14, Jan. 2025, <https://doi.org/10.33545/26633582.2025.v7.i1a.149>.
- [23] M. Rehan, M. S. I. Malik, and M. M. Jamjoom, "Fine-Tuning Transformer Models Using Transfer Learning for Multilingual Threatening Text Identification," *IEEE Access*, vol. 11, pp. 106503–106515, 2023, <https://doi.org/10.1109/ACCESS.2023.3320062>.
- [24] G. Arya *et al.*, "Multimodal Hate Speech Detection in Memes Using Contrastive Language-Image Pre-Training," *IEEE Access*, vol. 12, pp. 22359–22375, 2024, <https://doi.org/10.1109/ACCESS.2024.3361322>.
- [25] T. T. Nguyen, C. Wilson, and J. Dalins, "Large Language Models for Detection of Life-Threatening Texts," 2025, pp. 311–323.
- [26] I. Priyadarshini, S. Sahu, and R. Kumar, "A transfer learning approach for detecting offensive and hate speech on social media platforms," *Multimedia Tools and Applications*, vol. 82, no. 18, pp. 27473–27499, Jul. 2023, <https://doi.org/10.1007/s11042-023-14481-3>.
- [27] A. B. Mathew, "Novel Artificial Intelligence Model for Chapter-wise Summarization from Books," *International Journal for Research in Applied Science and Engineering Technology*, vol. 12, no. 6, pp. 2216–2228, Jun. 2024, <https://doi.org/10.22214/ijraset.2024.63469>.
- [28] P. Nagvekar, N. Bhanushali, D. Bhandary, and D. S. Tamboli, "Cyberbullying: Analyzing its Impact and Prevention Strategies," *IJNRD2504557 International Journal Of Novel Research And Development*, vol. 10, no. 4, pp. 517–519, 2025.

- [29] C. Tank *et al.*, “Su-RoBERTa: A Semi-supervised Approach to Predicting Suicide Risk through Social Media using Base Language Models,” Dec. 2024, <https://doi.org/10.48550/arXiv.2412.01353>.
- [30] S. Das, K. Bhattacharyya, and S. Sarkar, “Performance Analysis of Logistic Regression, Naive Bayes, KNN, Decision Tree, Random Forest and SVM on Hate Speech Detection from Twitter,” *International Research Journal of Innovations in Engineering and Technology*, vol. 07, no. 03, pp. 07–03, 2023, <https://doi.org/10.47001/IRJET/2023.703004>.
- [31] Y. Zheng and Y. Zhao, “A critical review on the study of threatening in English,” *Forum for Linguistic Studies*, vol. 2, no. 1, p. 71, Sep. 2020, <https://doi.org/10.18063/fls.v2i1.1206>.
- [32] Renju K and Brunda V, “Optimizing Crop Yield Prediction through Multiple Models: An Ensemble Stacking Approach,” *International Journal of Data Informatics and Intelligent Computing*, vol. 3, no. 2, pp. 52–58, Jun. 2024, <https://doi.org/10.59461/ijdiic.v3i2.120>.
- [33] M. Alipio and M. Bures, “Intelligent Network Maintenance Modeling for Fixed Broadband Networks in Sustainable Smart Homes,” *IEEE Internet of Things Journal*, vol. 10, no. 20, pp. 18067–18081, Oct. 2023, <https://doi.org/10.1109/JIOT.2023.3277590>.
- [34] G. Van Houdt, C. Mosquera, and G. Nápoles, “A review on the long short-term memory model,” *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5929–5955, Dec. 2020, <https://doi.org/10.1007/s10462-020-09838-1>.
- [35] J. M. Navarro, R. Martínez-España, A. Bueno-Crespo, R. Martínez, and J. M. Cecilia, “Sound Levels Forecasting in an Acoustic Sensor Network Using a Deep Neural Network,” *Sensors*, vol. 20, no. 3, p. 903, Feb. 2020, <https://doi.org/10.3390/s20030903>.
- [36] J. A. L. Marques, F. N. B. Gois, J. P. do V. Madeiro, T. Li, and S. J. Fong, “Artificial neural network-based approaches for computer-aided disease diagnosis and treatment,” in *Cognitive and Soft Computing Techniques for the Analysis of Healthcare Data*, Elsevier, 2022, pp. 79–99.
- [37] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.” 2019, <https://doi.org/10.18653/v1%2FN19-1423>.
- [38] A. Vaswani *et al.*, “Attention Is All You Need,” Aug. 2023, <https://doi.org/10.5555/3295222.3295349>.
- [39] M. Alruily, A. Manaf Fazal, A. M. Mostafa, and M. Ezz, “Automated Arabic Long-Tweet Classification Using Transfer Learning with BERT,” *Applied Sciences*, vol. 13, no. 6, p. 3482, Mar. 2023, <https://doi.org/10.3390/app13063482>.
- [40] Y. Dong, G. Ifrim, D. Mladenić, C. Saunders, and S. Van Hoecke, Eds., *Machine Learning and Knowledge Discovery in Databases. Applied Data Science and Demo Track*, vol. 12461. Cham: Springer International Publishing, 2021.
- [41] G. Anitha, G. Karthik, R. Chadge, M. Dinesh, T. D. Subha, and A. Al Sherideh, “A Novel Methodology Design to Predict Hate Speech on Social Media Using Machine Learning Principles,” *2024 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)*, pp. 1–6, Dec. 2024, <https://doi.org/10.1109/ICSES63760.2024.10910624>.

## BIOGRAPHIES OF AUTHORS



**Ifeoluwa Michael Olaniyi** earned his B.Sc. degree in Computer Science from Abiola Ajimobi Technical University (Formerly First Technical University), Ibadan, Nigeria, where he graduated with First Class honors. Also, a M.Sc. degree in Computer Science at the Department of Computer Science, University of Ibadan, Nigeria. His major research focus is on artificial intelligence, machine learning, and human-centered computing to ensure technology serves real-world needs. He's shaping a future where technology empowers. He can be contacted at email: [olaniyiifeoluwa12@gmail.com](mailto:olaniyiifeoluwa12@gmail.com).



**Jide Ebenezer Taiwo Akinsola** is a lecturer in the Department of Computer Sciences at the Abiola Ajimobi Technical University (formerly First Technical University), Ibadan, Nigeria, and is currently the Acting Head of the Department of Computer Sciences. His expertise spans both academia and industry. He holds a B.Sc., M.Sc., and Ph.D. in Computer Science, with specializations in Artificial Intelligence and Data Security. His contributions span various fields, including research in AI-driven health assessment models, data security, blockchain integration, digital forensics, cloud security, cryptography, and intelligent user interfaces. His current research focuses on “Innovative Data and Modeling Approaches to Measure Women’s Health,” leveraging AI and blockchain technologies to improve the accuracy, security, and accessibility of women's health data. This work applies machine learning techniques to identify health patterns and risk factors, ensuring data integrity and informed decision-making in healthcare. A member of the Nigeria Computer Society (NCS), the Computer Professional Registration Council of Nigeria (CPN), and a fellow of the Institute of Business Administration and Knowledge Management; his professional standing is reinforced by achievements such as the Blockchain Developer Mastery Award winner; An alumnus of MSM, Maastricht, The Netherlands, and a Fellow of the Netherlands Universities Foundation for International Cooperation (NUFIC). He has a strong international research background that supports ongoing advancements in technology and data science. He can be contacted at: [akinsolajet@gmail.com](mailto:akinsolajet@gmail.com).



**Maryann Gold Azodo** is a dedicated researcher and software developer with a strong foundation in Mathematics and Computer Science, holding a B.Sc. and M.Sc. in Computer Science. Gold's work centers on the cutting edge of bioinformatics and machine learning. Specializing in user-centric design, gold leverages advanced computational methods to drive innovations in software development that make complex tools accessible to a broader audience. Passionate about the intersection of technology and life sciences, Gold is committed to creating impactful, user-friendly solutions that address real-world challenges. She can be contacted at email: goldzodo@yahoo.com.



**Fathia Oluwadamilola Onipede** is a first-class graduate and research scholar in the Department of Computer Sciences at Abiola Ajimobi Technical University, Ibadan, Nigeria, with a bias in Software Engineering. She has an admirable capacity for originality and creative thinking for solving complex problems, especially in training, testing, and evaluating machine learning and deep learning models. She has worked on several projects, such as combinational and sequential circuits using LOGISM, object detection using deep learning models, diabetes prediction, stock prediction, image classification, computer vision, and many others. Her research interests are Artificial Intelligence and its applications, Machine learning, big data, Data science, Requirement discovery, Requirement specification, and Software development life cycle models. She is a promising researcher. She has some peer-reviewed scientific journal articles to her credit. She can be contacted at email: fathiaoluwadamilola46@gmail.com.



**Emmanuel Ajayi Olajubu** is a Professor of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria, where he obtained a BSc in Computer Science (with Economics), M.Sc., and Ph. D in Computer Science. He is a member of the Nigerian Computer Society (NCS), Computer Professional Registration Council of Nigeria (CPN), and International Association of Engineers (IAENG). His research interests are in the areas of distributed systems, cyber-physical systems, and cybersecurity, including network security. He was a former Acting Head of the Department of Computer Science and Engineering at OAU. He can be contacted at email: emmolajubu@oauife.edu.ng.



**Ganiyu Adesola Aderounmu** is a Professor of Computer Science and Engineering with several years of research, teaching, leadership, and project funding experiences within and outside Nigeria. He has attracted several research grants, and he is currently the Center Leader of the African Center of Excellence, OAU Knowledge-driven ICT park at Obafemi Awolowo University, Ile-Ife, Nigeria, and spearheads the partnership program of the Digital Science and Technology Network (DSTN) project for the OAK-Park, Nigeria, and the African Center of Excellence in Mathematics, Applications and Physical Sciences (ACE-SMIA), University of Abomey-Calavi, Benin Republic. He is a former Acting Head of the Department of CSE; former Director of the Information Technology and Communications Unit at OAU. He is a member of the Screening and Monitoring subcommittee of the Tertiary Education Trust Fund (TETFUND) research fund. He served as a member of curriculum development for the National Open University of Nigeria, and a member of COREN, CPN, and NUC accreditation teams, respectively, for various universities in Nigeria. He is a visiting research fellow at the University of Zululand, Republic of South Africa. He is the former National President of the Nigerian Computer Society. He can be contacted at email: gaderoun@oauife.edu.ng.