Journal of Computing, Science &Technology
https://focjournal.unidel.edu.ng/
editorjcst@unidel.edu.ng
ISSN:.....Print
ISSN: ..... Online

A Framework of Deep Learning Based Terrorist Classification Model

Adewale Olumide S.[1], Jimoh Ibraheem T.[2], Makinde Ibrahim A.[3] and Adeleye Samuel A.[4]
[1,4]Computer Science Department, School of Computing, Federal University of Technology Akure, Nigeria.[2] Software Engineering Department, School of Computing, Federal University of Technology Akure, [3]Information System DepartmentSchool of Computing Federal University of Technology Akure
adewale@futa.edu.ng[1], itjimoh@futa.edu.ng[2],iamakinde@futa.edu.ng[3],saadeleye@futa.edu.ng[4]

Corresponding Author's Email: adewale@futa.edu.ng

Article Info

Date Received: 21-03-2024
Date Accepted: 30-04-2024

Keywords:
Terrorist, Deep Learning, Prediction, Classification, BERT.

ABSTRACT

Terrorism has claimed many lives and destroyed many homes throughout the world. All levels of government have made significant investments in security, but there has not been much progress in releasing the terrified public from the grip of terrorism. Several agencies have deployed kinetic techniques such as tactical apprehension and killing of terrorists but the menace keeps increasing, hence, the exploration of soft computing which use intelligence gathering, community engagement and collaborative approach to combat the monster jerking the world without using forceful approach against the terrorists. The aim of the work is to propose a framework that predict the terrorism activities and offer useful information to the security agencies to prevent the risks of terrorist attack. The work proposes the use the Bidirectional Encoder Representation from Transformer for the world embedding that will be feed into the Attention-Based Bidirectional Long Short-Term Memory to analyze the tweets from Twitter, a social network service, to predict the activities of the terrorist group and classify the terrorist group that is responsible for each attack and provide necessary information for both the security agency and members of the society. The framework will be implemented with Python, Natural Language Processing and Deep learning packages such as Keras, TensorFlow, sklearn, NumPy, Pandas, Natural Language Tool Kits while the evaluation metrics such as accuracy, precision, recall, specificity, and F1 score will be used.

1.0 INTRODUCTION

Recently, [1] defines terrorism as the use of force to coarse or cause fear, in order to effect political change. The United Nations General Assembly 1994 view terrorism as criminal acts intended to ignite an atmosphere of fear among the masses in order to have some political gains premise on flimsy excuses such religion, creed, philosophy, race, ethnicity and any other points that may be used to justify the heinous acts [2]. Nevertheless, protest or stoppage of work as a result of demonstration cannot be regarded as terrorist act [3]. As the acts of Maitatsine sect became unbearable in the1980s so was the return of democracy in 1999 enveloped with some form of kidnapping, maiming and killing[4].

The Boko Haram spring up in the northeastern part of Nigeria since 2001 with its catastrophic attendants on Nigeria [4] and [5] reported Fulani Herdsmen ranked as fourth deadliest terrorist group. Deep learning is a subset of ML inspired by the function and architecture of the human brain popularly refers to as neural networks (NN) as shown in figure 1, which relies on inference from large dataset. The term "deep" denotes the quantity of intermediary layers in the neural network. Traditional NN contain 1-3 intermediary layers, while Deep Neural Network (DNN) as shown in Figure 2, is the product of networks of neural networks and can have as many as 150 layers.

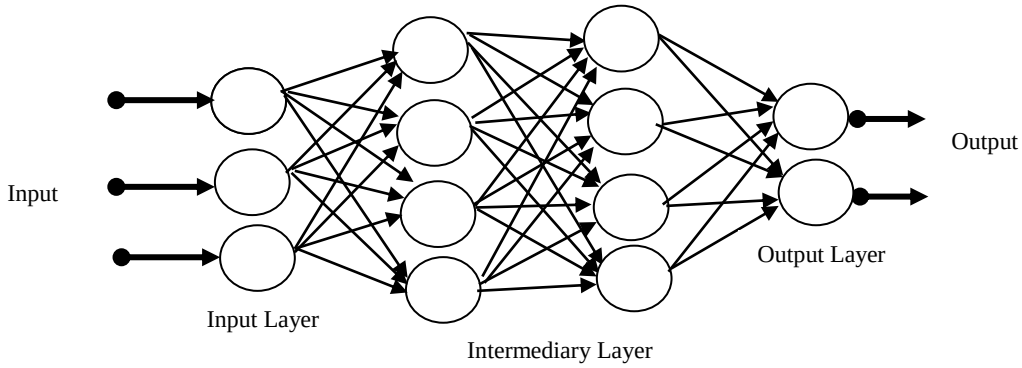

Figure 1: Neural Networks Architecture

DNN can be trained to analyze social media posts, news, surveys and provide valuable insights. Siri, Cortana, Google assistant, and Alexa can respond to questions and

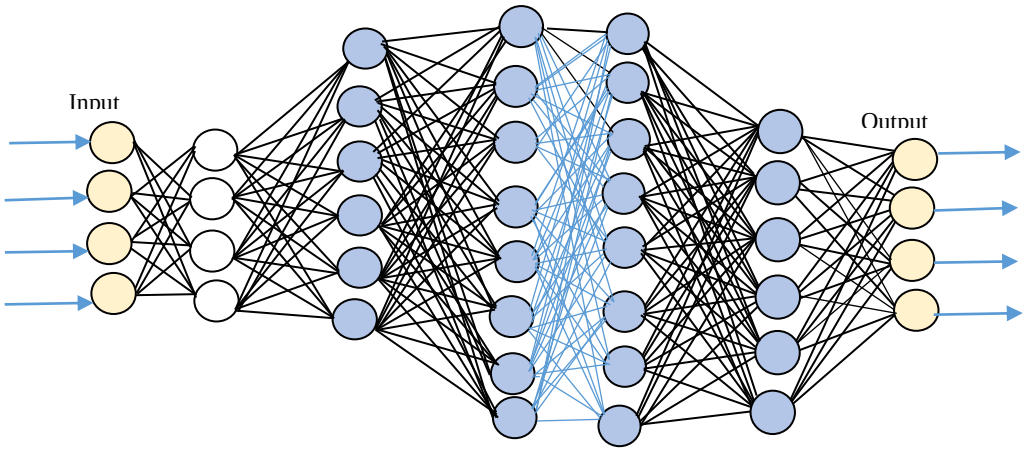

Figure 2: Deep Learning Architecture

adapt to user habits courtesy of deep learning and NLP. Deep learning algorithm can detect threats in advance with better precision than traditional malware solutions in recognizing new, questionable activities rather than responding only to the known threats' database. Deep learning models such as recurrent neural networks, deep belief networks, deep neural networks, convolutional neural networks, Capsule Neural Networks and transformers have been deployed in different sector of human life to solve security problems [6]. 2.0

2.0 RELATED WORK

[7] predicts the period and location of multinational terrorism with hybrid methodology link prediction technique with dataset that cover all transnational terrorist attacks between 1968 – 2002. The approach addresses the source and offer the necessary information required from source to target for safeguarding the target and as well as predicts the emergent of new terrorist group. However, the research work focus on continents. [8] examined whether the incident of the terror attacks can be determined or influenced by demography and explored GTi datasets using random forest, neural network and support vector machine while the scope of the research was limited to between 75˚N and 55˚S region and RF model outperformed other used models. [9] used distributed random forest, gradient boosting, naïve Bayes, deep learning on H_2O platform to explore the PIRUS datasets which consist of demography information close to 1,473 people in the United States who were linked to terror group or have participated in terrorism act from 1948 to 2013. The results support the author's claim that there are identifiable features that uniquely distinguished Islamist radicals from other radicals. However, the work did not consider non-radical personals to help further verify author's claim. [10] investigates terrorism dynamics using temporal meta-graphs extraction from events between 2001 to 2018 to predicts future terrorist targets. The work used Baseline, Convolutional neural networks, Feed forward neural networks, Long short-term memory, Bidirectional long short-term memory, CNN-LSTM on multivariate time-series forecasting task of GTD dataset. The model outperforms shallow Time-series due to its reliance on frequency of feature occurrences and experimental result shows that Bidirectional long short-term memory forecast better compared to other models. [11] used GTD dataset to predict terrorism acts that could threaten the society. The KNN achieves accuracy of 88.74% for Weapon Classifier while Random Forest yielded an accuracy of 90.45% and precision of 89.95% for perpetrator classifier**.** However, the work did not consider social media posts. [12] proposed similarity function based on key features selection graph-based outbreak detection that define susceptible area for violence outbreak. The work achieved more than 90% accuracy for the tactics.

[13] predict terrorist group responsible for the attack in Egypt from year 1996 to 2017. The work used wrapper approaches in a hybridized KNN and RFs that achieved highest classification accuracy using GTD dataset. [14] examines the growth and decay of the terrorist groups by types of attack they carry out, Weapon mastery and availability time, active locations, motive targets, and hidden structures in their activities using logistic regression, SVM, K-NN, where SVM has higher accuracy but the work could be improved upon by exploring other classification algorithms. [15] identify the philosophy of terrorist outfits using open-source dataset to models the behavioural trends of extremist groups through Supervised Profiling and Incident Explanation System, Random Forest Algorithm. Precision, recall, fi-score, accuracy, macro average, weighted average and mean squared error were used for the evaluation. The model achieved an accuracy of 85.45%. Future work should build a complete

system that detect the terrorist group that responsible for an attack using open-source dataset.

[16] forecasts the likelihood of casualties of civilians in the event of terrorist attacks. The work chose features using a unique approach that combines random forest (RF) and principal component analysis (PCA), and XGBoost hyperparameter tuning is done using a genetic algorithm. The technique was assessed using GTD and the Chinese terrorist attack dataset. The model obtains area under curve (AUC) of 87.00% and accuracy of 86.33% for the GTD dataset and sensitivity of 94.00% and AUC of 94.90% for the China terrorist attack dataset based on experimental results. [17] used GTD dataset to obtain hidden pattern and predicts the terrorist groups that is likely to attack a nation using Ensemble methods and some other traditional machine learning model. Random Forest Classification algorithm has best accuracy. [18] predicts the possibilities of terrorist attacks by exploring decision trees and the Random Forest with the GTD from 1970 to 2018. Decision Tree obtained an accuracy of 79.24%. However, the work did not consider post from social networks.

[18] apply Aho-Corasick automata in conjunction with twitter data to predict terrorist attack event. The work used KNN and SVM to predict the occurrence of attack and SVM classified better than KNN. [19] use individual social media posts while introducing the C-Attention and compared with three other machine learning models to automatically identify people who may attempt suicide between thirty days and six months. The greatest F1 score of 0.737 and F2 score of 0.843 for the prediction of suicide six months prior was obtained by C-Attention, while KNN and SVM achieved the best F1 score of 0.741 and F2 score of 0.833 on the prediction of a suicide attempt thirty days prior. [20] uses decision tree, bagging, random forest, extra tree, and XGBoost models with GTD dataset is used to examine terrorist group activity from 1970 to 2017 to classify and predict terrorist organizations. The best accuracy of 97.16% and 96.82%, respectively, was obtained by the XGBoost and random forest models in predicting the 32 terrorist organizations with the highest attack rates. [21] utilized the dataset produced from the suicide bomb blast of Pakistan that was collected from the South Asia Terrorism Portal and use supervised learning to identify the likely target audience of possible suicide bomb blasts. Weka was used to execute the Bayes, Function, Lazy, Meta, Rules, and Tree. [22] created a stack ensemble machine learning model utilizing K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) to forecast the continents where a particular kind of terrorism may manifest using GTD. The stacked model yielded a high accuracy of 97.8%. The work The future work only predicts the continents. [23] presented a method for utilizing the GTD to forecast terrorist attacks. The tests used twelve created feature vectors and applied SVM, DT, RF, LR, KNN, LDA, Bag, AD, and GB to predict nine different types of terrorist incidents related to the GTD dataset. The findings demonstrated that, in comparison to conventional algorithms, combining various text elements with numerical and category variables can significantly improve the ability to classify the type of terrorist attack. [24] implemented logistic regression, SVM, Naïve Bayes a single-layer neural network (NN) and five-layer DNN to predict the future terrorist activities. The DNN outperformed other models with more than 95% in terms of accuracy, precision, recall, and F1-Score, while others could not achieve more than 83% accuracy. [25] combines Nave Bayes (NB), Random Forest (RF), K-nearest neighbor (KNN), Logistic Regression (LR), Stochastic Gradient Descent (SGD), and Decision Tree (DT) with Natural Language Processing (NLP) technologies to forecast rumors from social media. With a 99% accuracy rate, the RF model outperformed all other classifiers. [26] used machine learning to investigates and forecast the pertinent of IT Act 2000 parts from complying to text/subjects. The research data was gathered and processed from news articles, and features were extracted using the Bag of Words model. These features were then fed into the hybrid Ensem_SLDR model. As a result, the results show 100% precision, 92% recall, 96.55% accuracy, and 96% F1 score. [27] categorize terrorist actions using multi-label datasets to forecast upcoming terrorist acts, a hybrid CNN-LSTM model was used. The improved DNN and the proposed CNN-LSTM were compared using [14] as a benchmark in the work. The accuracy of suicide prediction was determined to be 98%, with the DNN model producing 98.6% accuracy and the CNN-LSTM model producing the greatest accuracy of 99%. The threshold was 93% for predicting assault success. The accuracy produced by the CNN-LSTM model was 94.5%, whereas the DNN model produced 93.6%. While DNN achieved a 94% accuracy rate in weapon type prediction, CNN-LSTM could only achieve an 89.7% accuracy rate, falling short of the model. [26] explored machine learning models and daily data collected from the Nigeria Terrorism Database to forecast terrorist operations in Nigeria. A Heterogeneous Neural Network (HETNN) model was employed and contrasted with Random Forest,Boosting, K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Logistic Regression (LR) models.

In prediction, HETNN outperforms the other models. However, the work did not consider the location of terror attacks and social media comments. [29] utilized the GTD dataset, LR, SVM, SGD, DT, RF, ADB, MLP, NB, KNN, GB, and XGB, assault perpetrators' goals are categorized. [30] build an intelligent model with a dataset of 5,503 of terrorist activity in Nigeria to identify patterns of terrorist activity in the country's six geopolitical zones. The average percentage scores obtained were 99.89%,99.96%, 100%, and 99.98% for accuracy, precision, recall, and F1-score, respectively.

3.0 METHODOLOGY

The framework proposes BERT embedding with Attention-based Bidirectional Long Short-Term Memory (BiLSTM) model as shown in figure 3 which consist of data collection, Pre-processing,BERT embedding that is made up of Token, Positional and Segment embedding.The *snsrape*, a module in python used to scraps tweets from Twitter through its API without any restrictions and one does not need a Twitter developer account to scrape tweets to use it. With minimum of Python 3.8, one is good to go with *snscrape*, the Python package dependencies are installed automatically with "! *pip install snscrape"* command. The tweet text is tokenized with *WordPiece* tokenizer which divides the original text into individual pieces called tokens.

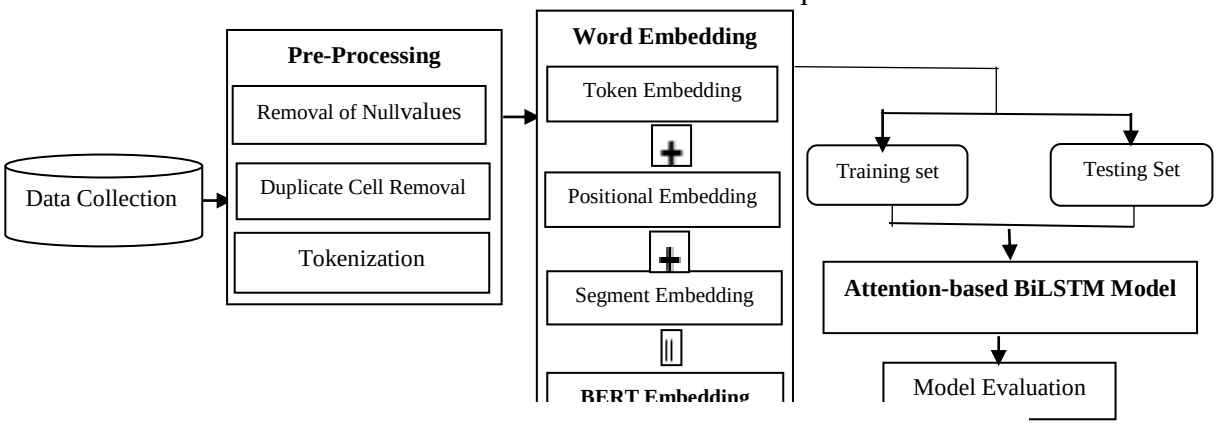

Figure 3: Architecture of the System

Each token is assigned a unique Id to represent it as a number. Eq. 1 ensures that tokenized tweets undergo unity-based normalization to improves the quality of the data and makes it suitable for machines to process.

$$X_{new} = \frac{(X - X_{min})}{(X_{max} - X_{min})} \quad (1)$$

3.1 Bidirectional Encoder Representation from Transformer *(BERT)*

A pre-trained auto encoding language model that employs Next Sentence Prediction and masked language modelling for data vectorization and generate contextual representation of each token, sentences, sentence pairs, or paragraphs. It considers the entire context of a word bidirectionally and resulting in embeddings that capture rich contextual information. Given a tweet input X that consist a set of elements x range from x_1 to x_n.

$$X = \{x_1, x_2, \ldots, x_n\} \quad (2)$$

where $n$denotes the total number of elements in the set X, BERT maps X into a series of word vectors $\overline{X}$ as shown in Eq. 3.

$$\overline{X} = \{e_1, e_2, \ldots, e_n\} \quad (3)$$

To get Eq. 3, words are split into subwords using WordPiece and special tokens;**[CLS]** and **[SEP]** is added to indicate the starting and ending of the sentence respectively and **[PAD]** ensures each sentence have the same dimension.

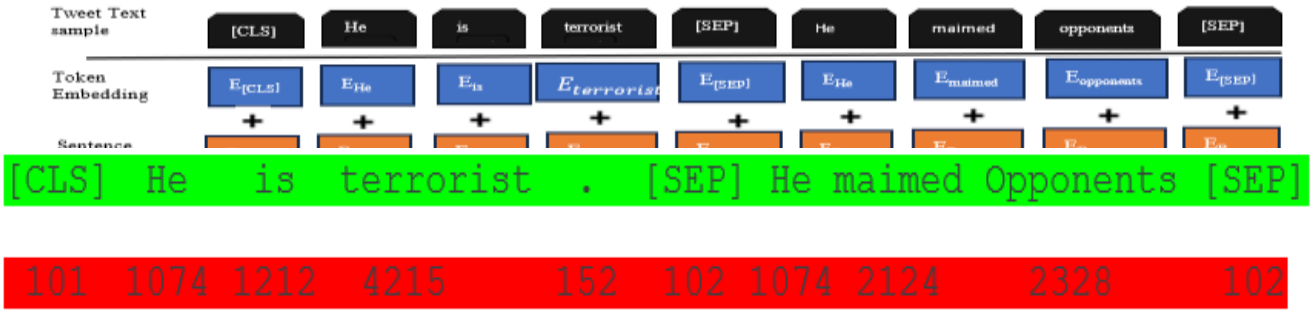

Figure 5: Token Embeddings

3.2 Token Embeddings

BERT Tokenizer convert the data from tweet text into a list of integer Token ID corresponds exactly to a word or portion of a word in the tweet text. e.g. the string "He is terrorist. He maimed opponents" is converted by the Tokenizer into its corresponding token ID as show in figure 5.

BERT use cosine similarity function to assign different vectors to text based on usage information using Eq. 4.

$$\cos(\theta) = \frac{A.B}{\|A\|\|B\|} = \frac{\sum_i^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2}\sqrt{\sum_{i=1}^n B_i^2}} \quad (4)$$

3.3 Position Embeddings

In addition to Token embeddings, which are used to represent every word or subword that the model can comprehend, BERT also uses Position Embeddings, which show where each token is in the input tweets as shown in figure 6.

Figure 6: Positional Embeddings

3.4 Segment Embeddings

Segment embedding is also referred to as sentence embedding and use to determines whether two given sentences *A* and *B*, logically follows each other which is illustrated by figure 7.

3.5 Long Short-Term Memory (LSTM) And Bidirectional Long Short-Term Memory (BILSTM)

Long Short-Term Memory (LSTM) remembers long term information due to long-term dependency. It consists of a memory cell and each cell housed four gates, namely; input gate (i_t), output gate (o_t), forget gate (f_t) and C_t gates that regulate the flow of information through the cell. The f_t decides what should be forgotten from the cell state with the help of the sigmoid function (σ) that squash the output values in closed range [0,1], where 0 means the text is irrelevant to the context and should be remove completely while 1 means the text should be kept for it is relevant. The f_t concatenates at h_{t-1} and x_t, and outputs a value ranges between 0 and 1 for each vector in the cell state C_{t-1}.

Figure 7: Segment Embeddings

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \quad (5)$$

where σ denote element-wise SoftMax function, W_f denotes weight of the encoded vector, h_{t-1} denotes value of the previous hidden layer. where x_t is word embedding of the tweet input at time-step t, b_f is the offset vector and $\odot$ is the element-wise product

$$\sigma(x_t) = SoftMax(x_t) = \frac{e^{x_t}}{\sum_{t=1}^T e^{x_t}} \quad (6)$$

σ learns which of the tweet x_t is important or not based on Normalized Exponential transformation that ensures any input, outputs mut all be positive and they must sum to unity.

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \quad (7)$$

i_t, decides what new data should be added to updates the cell state with word embedding of the tweet x_t, this is done with SoftMax function that decide what exactly to change and observes pointwise product operation with candidate gate.

Where;

W_i is the weight of the encoded input vector, x_t is the current input from the tweet data, h_{t-1} is the previous hidden state and $h_{t-1}, x_t \rightarrow \sigma$. Candidate cell state $\hat{c}_t$ of x_t is the mathematically represented as follows;

$$\hat{c}_t = \tanh.(W_{\hat{c}}.[h_{t-1}, x_t] + b_{\hat{c}}) \quad (8)$$

Where;

$tanh$ is the Tanh function that creates the candidate values by compress the values in range -1 to 1 and added to the cell state denoted by Eq. 9.

$$f(x_t) = tanh(x_t)\frac{e^{x_t} - e^{-x_t}}{e^{x_t} + e^{-x_t}} \quad (9)$$

$$c_t = f_t * c_{t-1} + i_t * \hat{c}_t \quad (10)$$

Where:

c_t denote the cell state, which store and transmit x_t information from one time step to another. The incoming output from cell state c_{t-1} obeys Eqs. (7) and (10) in order to update cell state c_t, f_t were used for point-wise multiplication with c_{t-1} to forget information from context vector and $i_t * \hat{c}_t$ is new candidate values scaled by how much they need to be updated.

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \quad (11)$$

o_t denotes the output gate in Eq. (12), o_t is the influenced and modified version of the cell state and its result is multiplied with Tanh function, resulting in hidden state output h_t. That is, o_t decides the current hidden state h_t.

Hidden state/layer h_t of LSTM is obtained as follows;

$$h_t = o_t * c_t \quad (12)$$

Hence, BiLSTM isemployed to gather the semantic feature bidirectionally and concatenate their representation together as an output in Eqs. (13) - (15) and as shown in figure 8.

$$\vec{h}_t = LSTM(x_t, \vec{h}_{t-1}), \quad (13)$$

$$\overleftarrow{h}_t = LSTM(x_t, \overleftarrow{h}_{t-1}) \quad (14)$$

$$h_t = [\vec{h}_t \oplus \overleftarrow{h}_t] \quad (15)$$

3.6 ATTENTION MECHANISMS

The attention mechanism averts the bottleneck problem of decoder to get information from all the intermediary states of the encoder, not just the last hidden state. The deliverables of BiLSTM will serves as input to the attention layer and because the Attention mechanisms can obtain long-distance semantic information and solve distance limitation of LSTM, it focuses differently on each word by assigning them with a score and align it with all the words in the sentence. To obtain the context vectorc_i, which may be produced as a set of attention weights denoted by$\alpha_1, \alpha_2, \ldots \alpha_T$, determined by the relevant input that produced the corresponding output, each score normalized using SoftMax scores. Then, aggregate the encoder hidden states using a weighted sum of the encoder hidden states. The context vector c_i for the output word h_j is generated using the weighted sum which is illustrated in figure 9.

The three primary parts are used by the general attention mechanism: the queries (Q), the keys (K), and the values (V).

1. Each query vector, $q = s_{t-1}$ matched against a collections of tweet text of keys k to obtain a score value. The dot product of query under consideration with each key vector, k_i is obtain bythe matching operation given as Eq. 16.

$$e_{q,k_i} = q.k_i \qquad (16)$$

2. A softmax operation is applied to the scores in order to produce the attention weights as shown in Eq. 17.

$$\alpha_{q,k_i} = softmax(e_{q,k_i}) = \frac{\exp(a(q,k_i))}{\sum_1^j \exp(a(q,k_j))} \qquad (17)$$

3. Next, a weighted sum of the value vectors, v_{k_i}, is used to calculate the generalized attention. Where each value vector has a matching key as shown in Eq. 18.

$$\alpha_T(q,K,V) = \sum_{i=1}^{T} \alpha_{q,k_i} v_{k_i} \qquad (18)$$

3.7 Self-Attention

In Self-attention, the query, key, and value vectors are created by feeding the tweet text into three distinct fully connected layers. Dot-product matrix multiplication is used to determine the relationship between query q and key k. The resulting similarity score is then fed through a linear layer. As a result, self-attention models are able to evaluate every word in the tweet text against other words and this is obtained with Eq. 19 and as shown in figure 10.

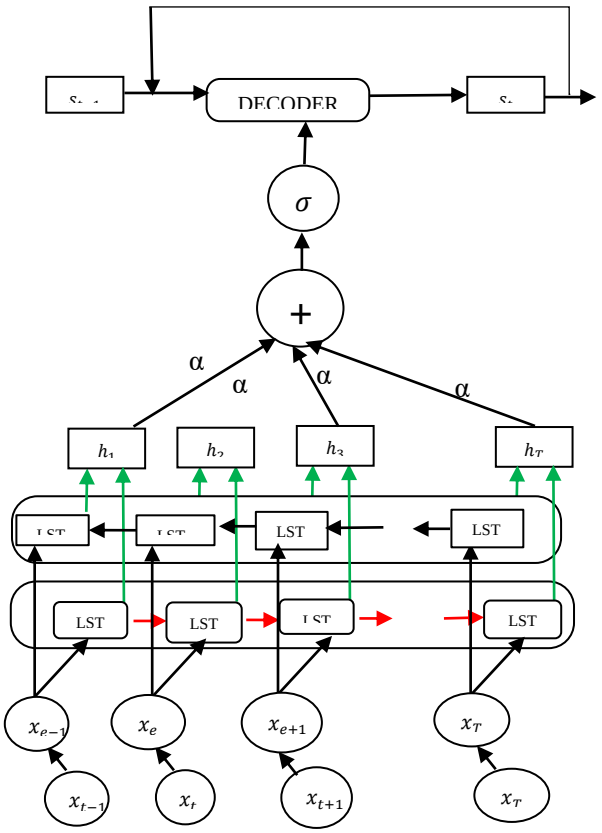

Figure 9: Architecture of the Attention-Based BiLSTM Model

$$\text{Similarity score} = QK^T \qquad (19)$$

query key Similarity Score/Attention Filter

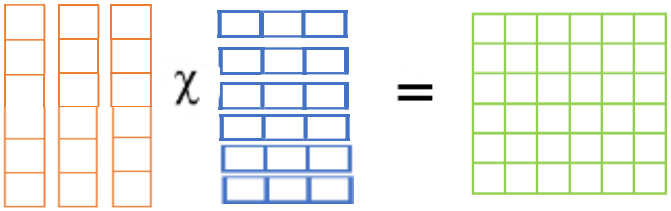

Figure 10: Schematic Representation of Similarity Score

Since each word has a score that correlates to other words in the time-step, the similarity score specifies the degree of emphasis that should be placed on other words in the tweet text. The mapping of the queries to the keys is displayed in Figure 11.

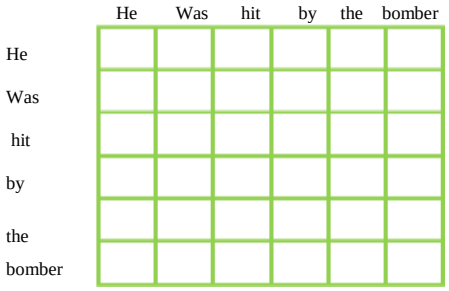

Figure 11: Similarity scores from the dot product.

3.8 Scaling Down the Attention Scores

The larger size of the dot product is kept from dominating and overshadowing the others by attention score scaling. To ensure more steady gradients, the score is divided by the square root of the query and key dimensions as shown in Eq. 20. This is done to prevent the multiplying numbers from likely explosive effects.

$$\text{Scaling of Similarity score} = \frac{QK^T}{\sqrt{d_k}} \qquad (20)$$

3.9 Softmax of the Scaled Scores

To obtain the attention weights and probability values between 0 and 1, the scaled score is SoftMax. SoftMax elevates higher scores while lowering lower ones. By utilizing Eq. 21, this enables the model to be more certain about which term from the twitter data needs what attention.

$$\alpha_{ij} = \sigma(\frac{QK^T}{\sqrt{d_k}}) \qquad (21)$$

The last main step in this network is to multiply α_{ij} with the value matrix that we left out at the start.

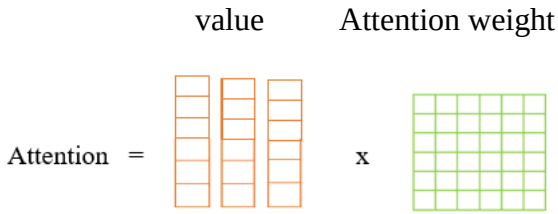

Figure 12: Schematic Diagram of Attention Calculation

$$Attention(Q, K, V) = \sigma(\frac{QK^T}{\sqrt{d_k}})\mathrm{V} \qquad (22)$$

4.0 Operational Environment Requirements and Future Direction

The propose framework will be implemented with Python programming language in Jupital Notebook using NLP and Deep learning packages such as Keras, TensorFlow, sklearn, NumPy, Pandas, NLTKwhile accuracy, precision, recall, specificity, and F1 score will be use as evaluation metrics in the future work.

REFERENCES

[1] Black's Law Dictionary, 8th Edition.

[2] Bridgelall, Raj. 2022. An Application of Natural Language Processing to Classify What Terrorists Say They Want. Social Sciences 11: 23. https://doi.org/10.3390/socsci11010023.

[3] QC, D. A. (2012). THE TERRORISM ACTS IN 2011.

[4] Akpomera, E & Omoyibo, K. (2013). Boko Haram Terrorism in Nigeria: The Paradox and Challenges of Big Brother Foreign Policy.AFRREV IJAH: An International Journal of Arts and Humanities. 2 (1):94-113.

[5] Egbe, O. D. J., and A. A. Muhammad. "Introduction: Interrogating Democratisation Deficits in Nigeria's Fourth Republic." *ODJ Egbe & A. A. Muhammad (Eds.), Nigeria's Democracy in the Fourth Republic* (2024): 1-16.

[6] Liu, H., & Lang, B. (2019). Machine learning and deep learning methods for intrusion detection systems: A survey. *applied sciences*, *9*(20), 4396.

[7] Desmarais, B. A., & Cranmer, S. J. (2013). Forecasting the locational dynamics of transnational terrorism: A network analytic approach. Security Informatics, 2(1), 1-12.

[8] Ding F, Ge Q, Jiang D, Fu J, Hao M (2017) Understanding the dynamics of terrorism events with multiple-discipline datasets and machine learning approach. PLoS ONE 12(6): e0179057. https://doi.org/10.1371/journal.pone.0179057.

[9] Global Terrorism Index: Nigerian Fulani militants named as fourth deadliest terror group in world (2018). Retrieve 12th March, 2024 from https://www.independent.co.uk/news/world/africa/global-terrorism-index-nigerian-fulani-militants-named-as-fourth-deadliest-terror-group-in-world-a6739851.html

[10] Campedelli, G. M., Bartulovic, M., & Carley, K. M. (2021). Learning future terrorist targets through temporal meta-graphs. Scientific reports, 11(1), 8533.

[11] Kalaiarasi, S., Mehta, A., Bordia, D., & Sanskar, D. (2019). Using global terrorism database (GTD) and machine learning algorithms to predict terrorism and threat. International Journal of Engineering and Advanced Technology, 9(1), 5995-6000.

[12] Soliman, G. M., & Abou-El-Enien, T. H. (2019). Terrorism Prediction Using Artificial Neural Network. Rev. d'Intelligence Artif., 33(2), 81-87.

[13] Uddin M. I., Zada N, Aziz F, Saeed Y, Zeb A, and Shah S. A. Prediction of Future Terrorist Activities Using Deep Neural Networks. Hindawi Complexity. 2020; 2020:1–16.

[14] Maniraj, S. P., Chaudhary, D., Deep, V. H., & Singh, V. P. (2019). Data aggregation and terror group prediction using machine learning algorithms. International Journal of Recent Technology and Engineering, 8(4), 1467-1469.

[15] Ankit Tewari. 2020. Decoding the Black Box: Interpretable Methods for PostIncident Counter-terrorism Investigations. In 12th ACM Conference on Web Science (WebSci '20 Companion), July 6–10, 2020, Southampton, United Kingdom. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3394332. 3402839

[16] Feng, Y., Wang, D., Yin, Y., Li, Z., & Hu, Z. (2020). An XGBoost-based casualty prediction method for terrorist attacks. Complex & Intelligent Systems, 6, 721-740.

[17] Rajesh P, Babitha D, Mansoor Alam, Mandour TahemezhadinManika A, 2020. Machine Learning and Statistical Analysis Techniquesnon Terrorism. Open Access byy Ios Press and Disytributed undr the terms of Creative Common Attribution. 210 - 222

[18] Huamaní, E. L., Alva, M. A., & Roman-Gonzalez, A. (2020). Machine learning techniques to visualize and predict terrorist attacks worldwide using the global terrorism database. International Journal of Advanced Computer Science and Applications, 11(4).

[19] Sarker, A., Chakraborty, P., Sha, S. S., Khatun, M., Hasan, M. R., & Banerjee, K. (2020). Improvised technique for analyzing data and detecting terrorist attack using machine learning approach based on twitter data. Journal of Computer and Communications, 8(7), 50-62.

[20] Wang, J., Qin, J. H.; Xiang, X. Y.; Tan, Y.; Pan, N. (2019): CAPTCHA recognition based on deep convolutional neural network. Mathematical Biosciences and Engineering vol. 16, no. 5, pp. 5851-5861.

[21] Pan, X. (2021). Quantitative analysis and prediction of global terrorist attacks based on machine learning. Scientific Programming, 2021, 1-15.

[23] Olabanjo, O. A., Aribisala, B. S., Mazzara, M., & Wusu, A. S. (2021). An ensemble machine learning model for the prediction of danger zones: Towards a global counter-terrorism. Soft Computing Letters, 3, 100020.

[24] Abdalsalam, M., Li, C., Dahou, A., & Noor, S. (2021). A Study of the Effects of Textual Features on Prediction of Terrorism Attacks in GTD Dataset. Engineering Letters, 29(2).

[25] Mahmood, N., & Ghani Khan, M. U. (2022). Prediction of Extremist Behaviour and Suicide Bombing from Terrorism Contents Using Supervised Learning. Computers, Materials & Continua, 70(3).

[26] Zayno, M., & Radhi, A. M. (2022). Data Mining Methods for Extracting Rumors Using Social Analysis Tools. Iraqi Journal of Science, 3618-3627.

[27] Hemakshi Pandey, Riya Goyal, Deepali Virmani, Charu Gupta, "Ensem_SLDR: Classification of Cybercrime using Ensemble Learning Technique", International Journal of Computer Network and Information Security (IJCNIS), Vol.14, No.1, pp.81-90, 2022. DOI: 10.5815/ijcnis.2022.01.07.

[28] Saidi, F., & Trabelsi, Z. (2022). A hybrid deep learning-based framework for future terrorist activities modeling and prediction. *Egyptian Informatics Journal*, *23*(3), 437-446.

[29] Odeniyi, O. A., Adeosun, M. E., & Ogundunmade, T. P. (2022). Prediction of terrorist activities in Nigeria using machine learning models. Innovations, 71, 87-96.

[30] George, U. D., Udoh, S. S., & Obot, O. U. (2023). An intelligent pattern recognition model for assessment of terrorists' activities in Nigeria. International Journal of Science and Research Archive, 9(2), 231-244.