JOURNAL OF SCIENCE TECHNOLOGY AND EDUCATION 10(3), SEPTEMBER, 2022
ISSN: 2277-0011; Journal homepage: www.atbuftejoste.net

GAX and GAS Modes: A New Hybrid Deep Learning Method for Phishing URL Detection using GRU, Attention Mechanisms, SVM and XGBOOST Algorithms

Hajara, Musa[1, 3], M, S, Adamu[1], A. Y. Gital[1], Usman Ali A[2], A. M. Kwami[2], F. U. Zambuk[1] and A. A. Aminu[3]
[1,3] *Department of Mathematical Sciences, Gombe State University, Gombe, Nigeria*
[1,2,3] *Department of Mathematical Sciences, Abubakar Tafawa Balewa University Bauchi, Nigeria*
[2]*Department of computer Science, Federal Collage of Education Technical Gombe, Nigeria.*

ABSTRACT
Many approaches have been proposed for classifying and detecting of phishing URLs. While these methods have recorded great successes, many research are based on content based approaches which has poor generalization ability against new unseen URLS and it requiring domain knowledge experts and substantial future engineering. To cover these limitations, recently, our approaches which offer the ability to automatically capture the semantic or sequential patterns in URLs have been proposed by several studies. This paper reviews and compares the state-of-the-art approaches for phishing URLs classification. It has been observed from the experimental results on 131,072 URLs dataset that Gated Recurrent Unit–attention mechanism- support vector machine (GRU-ATT-SVM) model outperforms the basic from other in terms of accuracy, precision, recall, training, and testing time. Hence the GRU-ATT-SVM model can be the best for the phishing URL classification in the future work.

INTRODUCTION

Phishing is a cyber-crime which involves the fraudulent act of illegally capturing private information like credit card details, usernames, password, account information by pretending to be authentic and esteemed in instant messaging, email and various other communication channels. The traditional approaches used by majority of the email filters for identifying these emails are static which make it weak to deal with latest developing patterns of phishing since the defrauders are dynamic in actions and keep on modifying their activities to dodge any kind of detection [2].

Phishing presents a diversified development trend, which poses new detection challenges. While phishers are pernicious and hide, security experts and researchers have dedicated many efforts in terms of phishing website detection. Phishing is a very popular method used in network attacks and leads to privacy leaks, identity theft and property damage [3]. The spread of phishing is no longer limited to traditional modalities such as e-mail, SMS, and pop-ups. Though the prosperity of the mobile Internet and social networks have brought convenience to users, they have also been employed to spread phishing, such as code phishing, spear phishing and spoof mobile applications [4], [5] and [59] etc.

Reducing the risk pose by phishers and other cybercriminals in the cyber space requires a robust and automatic means of detecting phishing websites, since the culprits are constantly coming up with new techniques of achieving their goals almost on daily basis. Phishers are constantly evolving the methods they used for luring user to revealing their sensitive information. The Main aim of the attacker is to steal banks account credentials. However, due to the dynamic nature of attackers and the challenging nature of the problem, it still lacks a complete solution [6-18][19-20].

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

JOURNAL OF SCIENCE TECHNOLOGY AND EDUCATION 10(3), SEPTEMBER, 2022
ISSN: 2277-0011; Journal homepage: www.atbuftejoste.net

Traditionally, phishing URLs are detected through blacklisting techniques, Although these methods just need to perform a quick database search to determine if a given URL is harmful or not and are predicted to have very low false positive rates, their main limitation is that they can only identify malicious URLs that have already been blacklisted. This is a serious concern as new URLs are almost generated on daily basis [1-21].

To solve these challenges, Machine Learning techniques which offer the ability to generalize well against new unseen URLs have been employed by several studies [1] While the traditional approaches have solved the problem associated with the blacklisting methods, their main limitation is the need for substantial manual features engineering, since they cannot directly learn representations from the URL's sequences of characters [7]. Most of the proposed traditional approaches require expert knowledge to determine the best feature set for malicious URL detection and they are unable to effectively capture the sequential patterns in URLs.

In this paper we come up with the hybrid deep learning to address above issues. DL is a set of ML algorithms that have the ability to detect the URL's patterns globally by learning the complex hierarchical feature representations and hidden sequential relationships from large-scale data. These characteristics contribute significantly to the success of DL algorithms in a variety of artificial intelligence (AI) tasks, including speech recognition, image processing, natural language processing, and many more [8].

To address the above issues, Gated Recurrent Unit –Attention mechanisms-Support vector machine (GRU-ATT-SVM) model have been proposed. The proposed approaches receive the URL (sequence of characters) as an input and applies the GRU-ATT-SVM at the character level in the URL. The models first identifies the individual characters in the training corpus and then represents each character as a fixed-length vector using one-hot encoding. Using this, input sequence characters are converted into an embedding representation, where the models can be applied. Experimental results showed that all the models obtained great statistical results. However, the GRU-ATT-SVM model has shown superior performance over the other models in terms of accuracy, precision, recall, training, and testing time.

METHODOLOGIES

GRU

The Gated recurrent unit (GRU) is another modification of RNN and an alternative to LSTM networks that works in a similar fashion to LSTM. Like the LSTM, it is capable of effectively retaining long-term dependencies in sequential data. However, compared to the LSTM, GRU has a simpler structure. GRU structure consists of two gates, the Reset Gate and the Update Gate. It merges the Memory and output state [9]. The reset gate (r) controls the information that flows out or is discarded. The forget gate and input gate are combined to form an update gate (z) for regulating the information to be retained from previous memory and also the new memory to be added. By reducing the parameters GRU computation is a little more efficient, simpler, and faster. A basic GRU structure is shown in Figure 1. Equations (1) to (4) illustrate the modeling procedure to compute the output of GRU at step t.

$$r_t = \sigma\,(w_r\,.\,[h_t - 1, x_t] + b_r) \quad (1)$$

$$z_t = \sigma(w_z\,.\,[h_t - 1, x_t] + b_z) \quad (2)$$

$$g_t = tanh\big(w\,.\,[r_t \times h_t - 1, x_t] + b_g\big) \quad (3)$$

$$h_t = (1 - z_t) \times h_t - 1 + z_t \times g_t \quad (4)$$

Where W and b are the trainable weights and biases respectively, and r and z denote the reset and update gates respectively. g denotes the hidden state which is used to control the removal and addition of new information decided by the update gate and h_t is the output of the GRU at step t.

In general, the LSTM and the GRU both preserve important features using a variety of gating functions, ensuring that they won't be

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

JOURNAL OF SCIENCE TECHNOLOGY AND EDUCATION 10(3), SEPTEMBER, 2022
ISSN: 2277-0011; Journal homepage: www.atbuftejoste.net

lost when long-term propagation takes place. They can therefore resolve the RNN's vanishing gradient problem through their gate controlling [10]. Additionally, the LSTM has one more gate function than the GRU, making the GRU the simplest RNN structure in this comparative analysis.

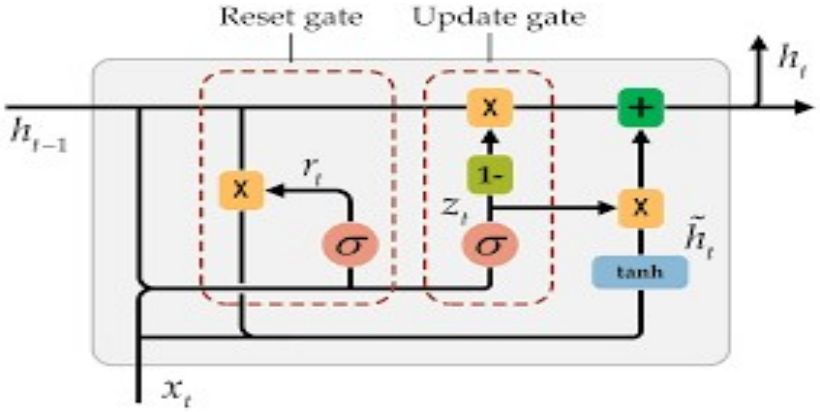

Figure 1: GRU memory cell [24]

ATTENTION MECHANISM

It is introduced to learn the correlation between sequences to strengthen the learning of key useful information of the model [11]. Attention mechanism have proven to have a great positive effect in this research. Different locations of URLs have different specification requirements. Therefore, for a URL, in order to fully learn the dependency between a word or a symbol and the words in other locations of the URL, this paper introduces an attention mechanism. This allows the model to focus more on the key information of URLs, while other irrelevant content will be ignored by the model. The introduction of an attention mechanism can facilitate more effective use of data, thereby improving the accuracy of the model [11]. The specific calculation formula is as follows

$$e_{t=w^T\sigma(w_l * x_t)} \quad (5)$$

$$q^t = \frac{\exp(e_t)}{\sum_{t=1}^{T}\exp(e_t)} \quad (6)$$

$$X_t^* = \sum_{t=1}^{T} q_t * x_t \quad (7)$$

In the formula, x_t is the input information at t time w_l , w^T is the learned weight, σ is the tanh activation function, and X_t^* is the output information after t time passes through the attention layer.

XGBOOST

XGBOOST is tree based model that aggregates trees using the boosting technique. In XGBOOST we used the training data xi to predict the target variable yi iteratively until the parameters of the model are optimized.

XGBOOST model have high predictive power as well as to have a simple in nature (deals with less number of features). As we know minimizing loss function ((Θ)) encourages predictive models as well as optimizing regularization (Ω (Θ)) encourages simpler model to have smaller variance in future predictions, making prediction stable [12-21]. The closed form of the objective is given below:

$$obj(\Theta) = l(\Theta) + \Omega(\Theta) \quad (8)$$

$$\Theta = \{f_1, f_2, \dots \dots f_k\} \quad (9)$$

$$obj(\Theta) = \sum_{i}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \quad (10)$$

Using Taylor expansion,

$$obj^{(t)} = \sum_{i=1}^{n}\left[l\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)\right] + \Omega(f_t) + constant \quad (11)$$

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

JOURNAL OF SCIENCE TECHNOLOGY AND EDUCATION 10(3), SEPTEMBER, 2022
ISSN: 2277-0011; Journal homepage: www.atbuftejoste.net

Objective, with constants removed, therefore the new form of optimizing goal is:

$$obj^{(t)} = \sum_{i=1}^{n}\left[g_i f_t(x_i) + \frac{1}{2}h_i f_t^2(x_i)\right] + \Omega(f_t) \qquad (12)$$

Where

g_i and h_i comes from definition of loss function, the learning of function only depend on the objective via g_i and h_i

$$g_i = \partial\hat{y}_i^{(t)} l\left(y_i, \hat{y}_i^{(t-1)}\right), \qquad g_i = \partial_{\hat{y}^{(t-1)}}(\hat{y}^{(t-1)} - y_i)^2 = 2(\hat{y}^{(t-1)} - y_i)$$

$$h_i = \partial^2\hat{y}_i^{(t)} l\left(y_i, \hat{y}_i^{(t-1)}\right), \qquad h_i = \partial^2{}_{\hat{y}^{(t-1)}}(\hat{y}^{(t-1)} - y_i)^2 = 2$$

XGBOOST approximates f(x) by an additive expansion of t regression trees, but instead of minimizing just a lost function, an objective function with two parts is defined, a lost function over the training set as well as a regularization term to prevent overfitting.

Support Vector Machine (SVM)

SVM can best be explained in practice by using a hypothetical classifier called Maximal-Margin Classifier. The numerical input variables (x) in the data (the columns) form an n-dimensional space [13-14]. A line that splits the input variable space is known as a hyperplane. The hyperplane is selected in SVM to best separate the points in the input variable space by their class, either class 0 or class 1. In n-dimension say two, you can visualize this as a line and assume that all our input points can be separated by this line. For example:

$$B0 + (B1 * X1) + (B2 * X2) = 0 \qquad (13)$$

Where the coefficients (B1 and B2) determine the slope of the line and the intercept, (B0) is found by the learning algorithm, and X1 and X2 are the two input variables. You can make classifications using this line. By plugging in input values into the line equation, you can calculate whether a new point is above or below the line.

Real in practice can be messy and cannot be separated accurately with a hyperplane. The constraint of maximizing the margin of the line that separates the classes must be relaxed. This is often called the soft margin classifier. This change allows some points in the training data to violate the separating line. An additional set of coefficients are introduced that gives the margin wiggle room in each dimension. These coefficients are sometimes called slack variables. This increases the complexity of the model as there are more parameters for the model to fit the data to provide this complexity.

It is characterized by the usage of kernels, the sparseness of the solution, and the capacity control gained by acting on the margin, or several support vectors, etc. The capacity of the system is controlled by parameters that do not depend on the dimensionality of feature space. SVM algorithm operates natively on numeric attributes, as a result, it uses a z-score normalization on numeric attributes. In regression, Support Vector Machines algorithms use epsilon-insensitivity loss function to solve regression problem [15].

Modeling Phishing Models by the use of GRU-ATT-SVM

In order to compare the proposed approaches for phishing classification, character level embedding has been used in the proposed methodologies. We learn a representation directly from the character sequence of the URL rather than manually extracting the features. Each character sequence exhibits correlations, that is, nearby characters in a URL are likely to be related to each other. These sequential patterns are important because they can be exploited to improve the performance of

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

predictors [16]. GRU-ATT-SVM Model, in contrast to other approaches, are able to learn the representation from the URL's character order, eliminating the need for handcrafted feature extraction, which is labor-intensive and requires domain knowledge experts [17]. The proposed networks compose of an embedding layer for URL representation, GRU layer for optimal feature extraction, we introduce another mechanism called attention mechanisms which allows the model to focus more on the key information of URLs, while other irrelevant content will be ignored by the model. The introduction of an attention mechanism can facilitate more effective use of data, thereby improving the accuracy of the model [11]. And we used XGB and SVM for classification.

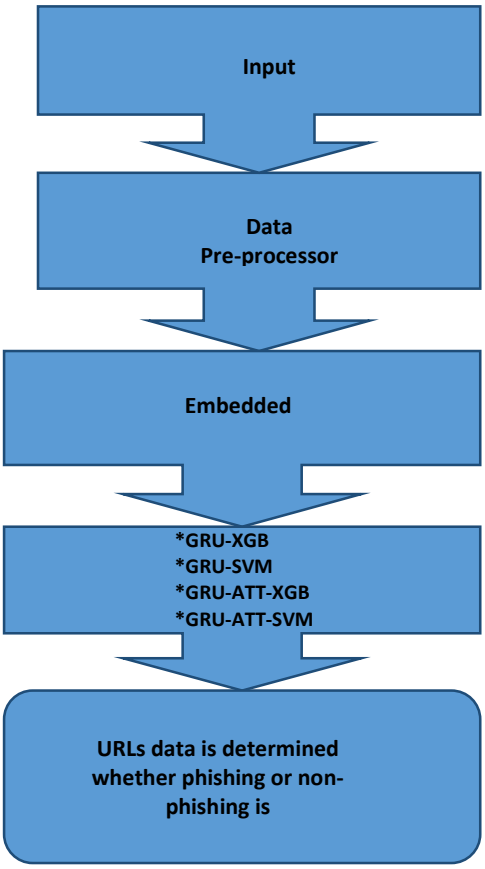

Figure 2: Architecture of the proposed models

Figure. 2 illustrates the architecture of the proposed models. One-hot embedding is used to translate each input, and fed into the GRU-ATT layer. Finally, the classification is performed using XGBoost and SVM function.

EXPERIMENTAL SETUP

Dataset Description

The data set used in this paper is collected from the malicious phish URLs. The malicious phish URLs (https://www.kaggle.com/datasets/sid321axn/malicious-urls-dataset) come from the Kaggle community and are from the collection of URL data, collected by Manu Siddhartha across

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

multiple communities. In this experiment, 65,536 benign URLs and 65,536 malicious URLs were randomly selected, for a total of 131,072 URLs. This article divides the URL data into a training set, a validation set, and a test set. To perform the experiments, we choose a recent dataset from kaggle community which is shown in Table 1.

Table 1: Number of samples in each data subset.

URLs	Training Set	Validation Set	Test Set	Aggregate
Phishing URLs	45,956	5,879	13,701	65,536
Benign URLs	45,795	5,917	13,824	65,536
Aggregate	91,751	11,796	27,525	131,072

Experimental Design

The objective of this paper is to empirically assess between four models for classifying malicious URLs, not to produce cutting-edge models. Therefore, the setup for our experiments is kept to a minimum, and the comparisons are made fairly. As shown in Table 2, the generated datasets are divided into two separate sets to create the training set and test set, respectively, in order to evaluate the performance of the models. All the models were then trained and evaluated using the same training and testing datasets. The evaluation is done using standard classification evaluation measures, as described below:

$$Accuracy = \frac{TP+TN}{(TN+TN+TP+FN)} \quad (19)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (20)$$

$$Precision = \frac{TP}{(TP+FP)} \quad (21)$$

$$F-score = \frac{2*(Precision*Recall)}{(Recall+Precision)} \quad (22)$$

Where TP is the True Positive, which denotes the number of URLs that are phishing and correctly classified as phishing, TN represents True Negative which denotes the number of benign URLs recognized as benign ones, FP represent False Positive, which denotes the number of benign URLs which are wrongly classified as phishing, and FN is the False Negative that denotes the number of phishing URLs identified as benign ones. We define the phishing URLs as positive and the non-phishing URLs as negative.

EXPERIMENTAL RESULTS

Table 2: Hyper-parameters

Hyper-parameters	Values
Epochs	30
Learning Rate	0.001
Validation-split	0.20
Batch-Size	256
Dropout Rate	0.2

In this section, we present the experimental results of the proposed models. All the experiments were conducted on a laptop computer with Intel Core (TM) i7-7700HQ CPU @ 2.80GHz x 4, 16GB of DDR3 RAM, and NVIDIA GeForce GTX 1050M 4GB DDR5 GPU. The hyper-parameters used for all models were assigned by hand, and not through hyper-parameter optimization/tuning (see Table 2). All models were trained on 70% of phishing and

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

non-phishing URLs data for 30 epochs. Afterward, the trained models were tested to classify with 30% phishing and non-phishing URLs. The performances of the different models in terms of accuracy, precision, recall, and F1-score are summarized in Table 3, 4 and 5. As can be seen from Table 3, 4, and 5, it can be observed that the GRU-ATT-SVM performed best in classifying the URLs as either phishing or non-phishing in comparison to GRU-XGBoost, GRU-SVM and GRU-ATT-XGBoost, models. We also evaluated the different models.

Table 3: Summary of test results without the attention mechanism

MODELs	Accuracy	Precision	Recall	F-score
GRU-XGBoost	0.9912	0.9896	0.9906	0.9899
GRU-SVM	0.9924	0.9946	0.9879	0.9912

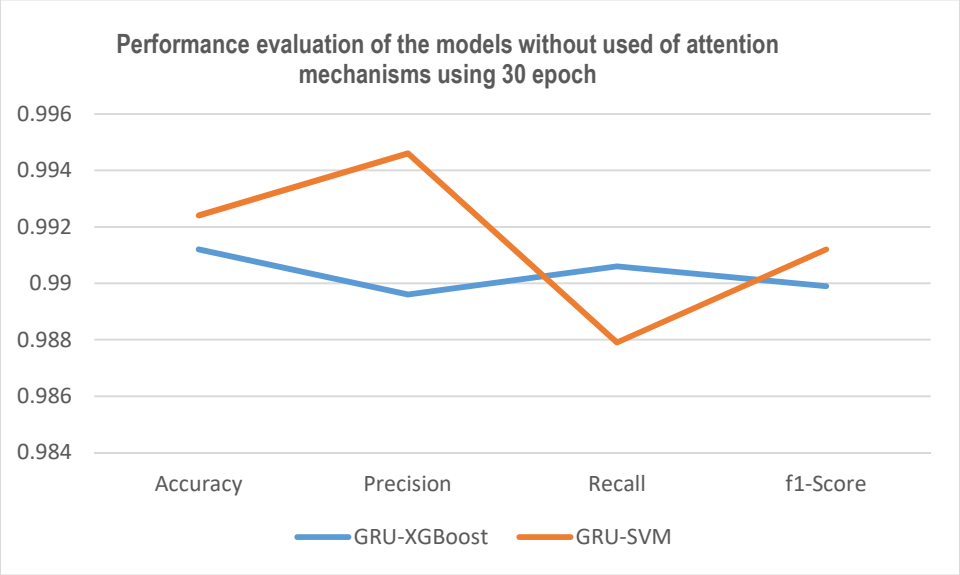

Figure 2: Performance evaluation curve of the models without used of attention mechanisms.

The above figure 2 shows that the GRU-SVM demonstrate that the proposed method can achieve better classification results in malicious URL detection than GRU-XGBoost method.

Table 4: Summary of test results without the attention mechanism

MODELs	**Accuracy**	**Precision**	**Recall**	**F-score**
GRU-ATT-XGBoost	0.9915	0.9887	0.992	0.9903
GRU-ATT-SVM	0.9929	0.9933	0.9907	0.9919

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

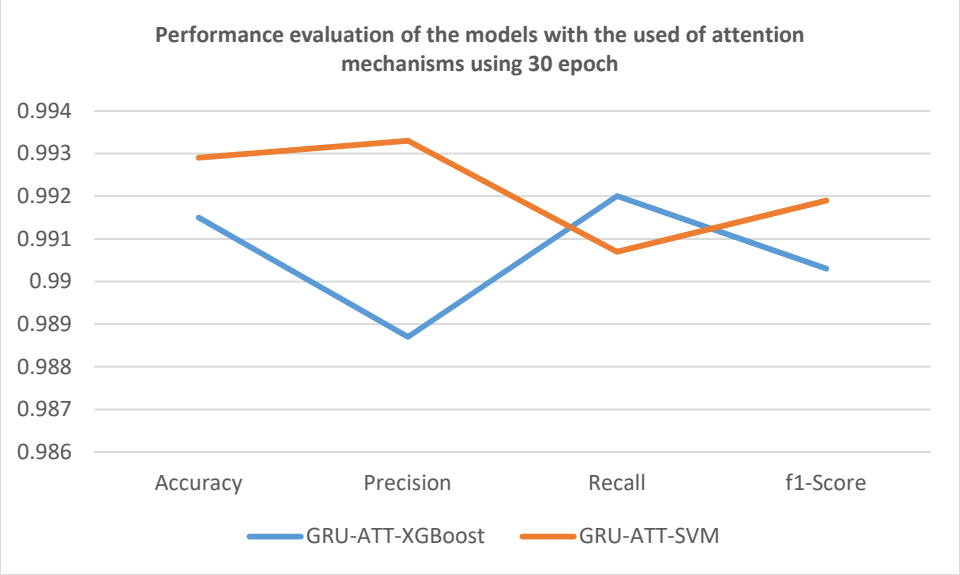

Figure 3: Performance evaluation curve of the models with the used of attention mechanisms.

The above figure 3 shows that the attention mechanism was introduce in order to extract the main key important features of the URLs so that we come up with high accuracy than the other models, therefore GRU-ATT-SVM demonstrate that the proposed method can achieve better classification results in malicious URL detection, which has high significance for practical applications than the other models.

Table 3: Summary of test results of all the proposed models

MODELs	Accuracy (%)	Precision (%)	Recall (%)	F-score (%)
GRU-XGBoost	99.12	98.96	99.06	98.99
GRU-ATT-XGBoost	99.15	98.87	99.2	99.03
GRU-SVM	99.24	99.46	98.79	99.12
GRU-ATT-SVM	99.29	99.33	99.07	99.19

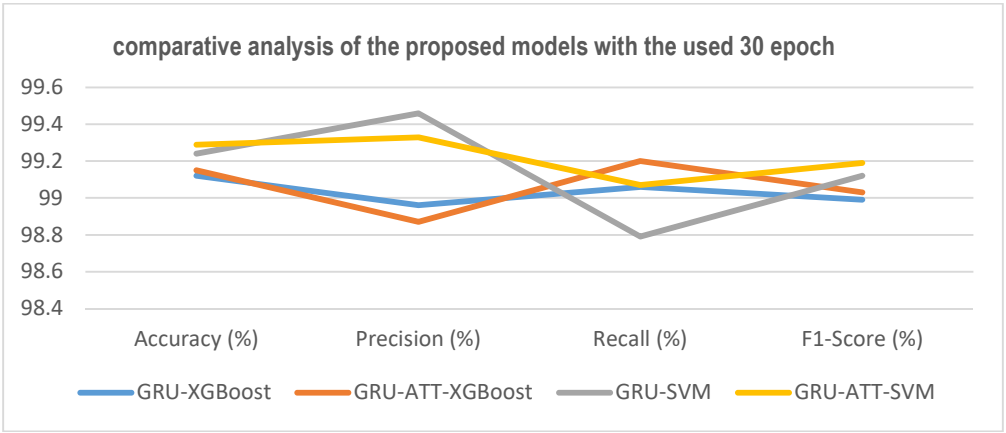

Figure 4: Comparative analysis of the proposed models curve with and without attention mechanisms.

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

Looking at the above figure 4, the proposed model GRU-ATT-SVM shows that the accuracy and precision outperformed the others models. Hence the GRU-ATT-SVM model can be the best for the phishing URL classification.

Finally, to shows the models' performances during training and evaluations, we present the models' learning curves in figure 5(accuracy vs epoch) with and without attention mechanism. Also figure 6 shows the curve of (loss vs epoch) with and without the used of attention mechanism. As can be seen from both figures, in just few epochs, the validation accuracies and losses of the proposed models converge, increasing the accuracies of each model to over 98% from 5 epochs onward and reducing their losses to below 8%. Models exhibit constant increase in performance with only slight fluctuations in their accuracies during training the testing. On the other hand, the models had fluctuating in losses during the training testing process due to some over-fitting experienced by the two models GRU and SVM. However, in case of accuracies, all the models show steady rise in performance until the 3th epoch. Comparing the training and validation performances of the models, it can be observed that all the models showed constant performances during training and testing and some fluctuations during the loss training and testing time. Overall, the training and validation performances of all the models in classifying malicious URLs are acceptable.

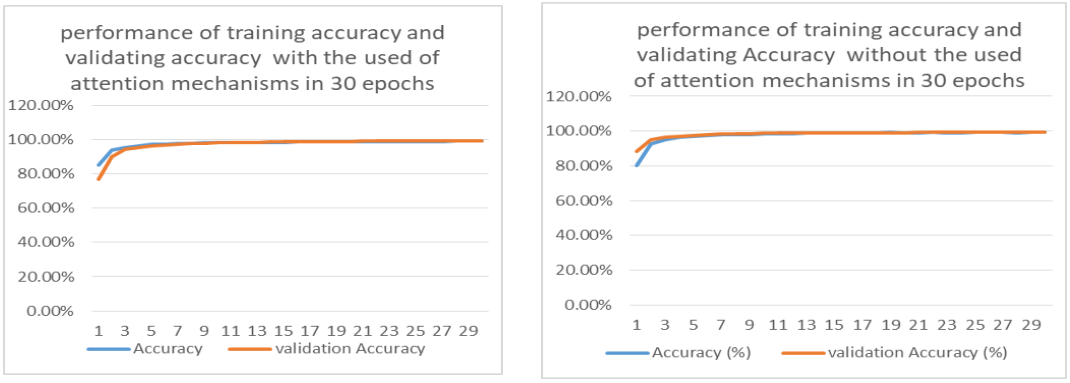

Figure 5: model training and testing performance (accuracy vs epochs)

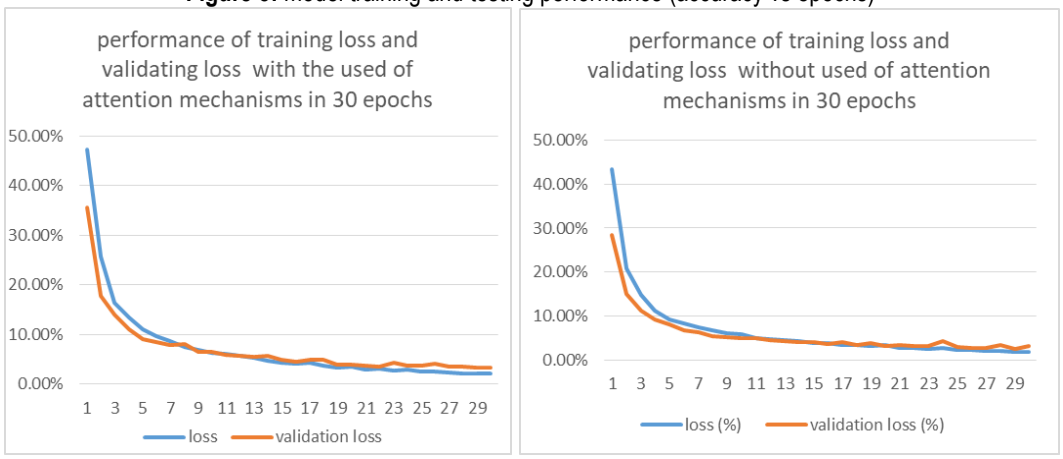

Figure 6: Model training and testing performance (loss vs epochs)

DISCUSSIONS AND FUTURE DIRECTIONS

Phishing URL detection is an essential component of the majority of security applications. Most of the traditional machine learning methods used for phishing URLs detection present on hand design features, which

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

are labor-intensive to construct and lack some of the attributes necessary to build effective phishing detection systems. To overcome with this challenge, recent studies have resorted to the use of hybrid deep learning approaches which have the ability to extract the necessary abstract feature representations by acting on raw input data. To exploit on this, we apply for the task of phishing URLs detection with GRU-XGB, GRU-SVM, GRU-ATT-XGB, GRU-ATT-SVM. The empirical evidence presented suggests that GRU-ATT-SVM outperforms other models in the task of phishing and non-phishing URLs classification. However, the results of the other models are also acceptable. Additionally, we have used the various networks with their simple and basic architectures. To gain more performance, their complex architectures can be explore using advanced hardware in a distributed environment. The characteristics of malicious threats are evolving in nature. At the same time, the URLs are also changing with time. We make a concrete statement such that deep learning mechanisms are most representative to deal with the drifting nature of URLs. Thus, require a larger study by transforming supervised learning to semi-supervised to unsupervised learning in deep learning mechanisms. This can be considered another significant future direction.

CONCLUSION

In this paper, we explored how well we can discern phishing URLs from legitimate URLs using GRU-XGB, GRU-SVM, GRU-ATT-XGB, and GRU-ATT-SVM based on character-level embedding. In order to evaluate the approaches, themalicious phish URLs (https://www.kaggle.com/datasets/sid321axn/malicious-urls-dataset) come from the Kaggle community and are from the collection of URL data, collected by Manu Siddhartha across multiple communities. In this experiment, 65,536 benign URLs and 65,536 malicious URLs were randomly selected, for a total of 131,072 URLs. The proposed approaches receive the URL (sequence of characters) as an input and applies the GRU-XGB, GRU-SVM, GRU-ATT-XGB, and GRU-ATT-SVM at the character level in the URL. The models first identifies the individual characters in the training corpus and then represents each character as a fixed-length vector using one-hot encoding. Using this, the URL sequence characters are converted into an embedding representation, where the models can be applied. The embedding layer followed by other deep networks layer extracts features implicitly and those optimal features are fed to other algorithms for classification in the supervised classification setting. Detailed experiments reveal that all models showed great statistical results. However, the GRU-ATT-SVM model has shown superior performance over the other models in terms of accuracy, precision, recall, training, and testing performance. This work can be considered preliminary research towards the evaluation of various proposed models for malicious URLs detection and classification. Moreover, adopting deep learning based malicious URL detection in real time should require an extended data sets to avoid the state.

REFERENCES

[1] Musa. H., Ya, A., Alkali, A. H., & Umar, A. (2022).). “Phishing URLs Classification Using Deep Learning Approach”. *Journal of Science Technology and Education 10*(2), 413–424.

[2]Yadav, D. P., Paliwal, P., KumaD, D., and Tripathi, R. (2017). A Novel Ensemble Based Identification of Phishing E-Mails, 2–6.

[3] Yang, P., Zhao, G., & Zeng, P. (2019). Phishing website detection based on multidimensional features driven by deep learning. *IEEE Access*, *7*. https://doi.org/10.1109/ACCESS.2019.2892066

[4] Ahmad A. Y, Selvakumar M., Mohammed A., Mohammed A. and Samer A. S. (2016). TrustQR: A New Technique for the Detection of Phishing Attacks on QR Code, *Adv. Sci. Lett.*, vol. 22, no. 10, pp. 2905-2909, Oct. 2016.

[5] Inez C. C. and Baruch F. (2018). Setting Priorities in Behavioral Interventions: An Application to Reducing Phishing Risk, *Risk Anal.*, vol. 38, no. 4, pp. 826-838, Apr. 2018.

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.

[6] Musa, H., Gital, A. Y., Zambuk, F. U., Umar, A., Umar, A. Y., & Waziri, J. U. (2019). A comparative analysis of phishing website detection using XGBOOST algorithm. *Journal of Theoretical and Applied Information Technology*, *97*(5).

[7] A. Ali, Q. Jiang, Q. Qu, M. Huang, and J. Niyigena, "An effective phishing detection model based on character level convolutional neural network from URL," *Electronics* 9, no. 9 (2020): 1514.

[8] L. Yann, Y. Bengio, and G. Hinton. "Deep learning." *nature* 521, no. 7553 (2015): 436-444.

[9] C. Kyunghyun, B V. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078* (2014)

[10] B. Yun, J. Xie, C. Liu, Y.Tao, B. Zeng, and C. Li, "Regression modeling for enterprise electricity consumption: A comparison of recurrent neural network and its variants," *International Journal of Electrical Power & Energy Systems* 126 (2021): 106612.

[11] Wu, T., Wang, M., Xi, Y., Zhao, Z. (2022): *Malicious URL Detection Model Based on Bidirectional Gated Recurrent Unit and Attention Mechanism*. Appl. Sci. 2022, 12, 12367. https://doi.org/10.3390/ app122312367

[12] Tianqi Chen Oct. 22 2014. Introduction to Boosted Trees, university of Washington.

[13] brahim, A., Kashef, R., Li, M., Valencia, E., & Huang, E. (2020). *Bitcoin Network Mechanics : Forecasting the BTC Closing Price Using Vector Auto-Regression Models Based on Endogenous and Exogenous Feature Variables*.J. Risk Financial Manag., MDPI, 13, 189; doi:10.3390/jrfm13090189.

[14]Cohen, G. (2020). *Forecasting Bitcoin Trends Using Algorithmic Learning Systems Learning Systems,* Entropy J., MDPI vol. i, 2020.

[15]Brownlee, J. (2019). *Master Machine Learning Algorithms. Discover How they Work and Implement Them from Scratch*. Edition, v1.1 http://MachineLearningMastery.com.

[16] R. Tomas, and L. Dovydaitis, "Detection of phishing URLs by using deep learning approach and multiple features combinations," *Baltic journal of modern computing* 8, no. 3 (2020): 471-483.

[17] B. A. Correa, E. C. Bohorquez, S. Villegas, J. Vargas, and F. A. González, "Classifying phishing URLs using recurrent neural networks," In *2017 APWG symposium on electronic crime research (eCrime)*, pp. 1-8. IEEE, 2017.

[18]Musa, H., Aminu, A. A., & Muhammad, A. N. (2019). Investigation the Effect of Dataset Size on the Performance of Different algorithm in Phishing Website Detection*, International Journal of Artificial Intelligence and Applications (IJAIA), Vol.10, No.3, May 2019 pp*, 53–59. https://doi.org/10.9790/7388-0901025359

[19]Musa, H., Gital, A. Y., Bitrus, M. G., Juma, F., & Balde, M. A. (2020). *Boosting the Accuracy of Phishing Detection with Less Features Using XGBOOST*. *8*(2), 81–90.

[20]Musa, H., Gital, A. Y., Zambuk, F. U., Umar, A., Umar, A. Y., & Waziri, J. U. (2019). A comparative analysis of phishing website detection using XGBOOST algorithm. *Journal of Theoretical and Applied Information Technology*, *97*(5).

[21]Musa H., Gital, A. Y., A, U. A., & Kwami, A. M. (2023). *A comprehensive Review on phishing website detection using machine learning and deep learning techniques. 8*(2), 188–198.

Corresponding author: Hajara Musa ✉ mhajara86@gmail.com ✉ Department of Mathematical Sciences, Gombe State University, Gombe.