Performance Analysis of Deep Learning Models to Detect Phishing Attacks in Edge Devices

Ushus Maria Joseph, Dr.Mendus Jacob
[1]Research Scholar, [2]Professor and director MCA,
[1]Lincoln University College, Malaysia

Abstract - Phishing is the deceitful use of electronic communications to trick and take advantage of customers. Phishing attacks aim to obtain sensitive, protected data, such as usernames, passwords, MasterCard information, network credentials, and more. By acting like a real individual or foundation by means of telephone or email, digital aggressors utilize social designing to maneuver casualties toward performing explicit activities — like tapping on a noxious connection or connection — or determinedly disclosing private data. Phishing is a reserved and hence hostile to practical movement, which has a significant social and monetary aspects. Cell phones are well known with programmers since they're intended for quick reactions in light of negligible relevant data. The primary goal of this work is to foster a model that can distinguish and forestall conceivable phishing assaults progressively. A clever phishing assault location system, when carried out to handheld gadgets expands the transfer speed of assurance in the worldwide advanced town. Forestalling phishing it assist with achieving manageable turn of events. Various calculations like decision trees and SVM are utilized to recognize phishing assaults. This kind of calculation needs enormous computational ability to run and the client information is shipped off the server for the location of phishing assaults. The fundamental burden is that our perusing information is shipped off one more server for examination. In this present circumstance an outsider approaches our perusing information which will prompt a protection issue and discovery can be subject to different elements like organization transmission capacity and so on. To defeat the security issue we can utilize constant forecast on the client-side. For distinguishing phishing assaults in low computational gadgets we can utilize a quantized model. This research focuses on continually detecting SMS phishing attempts with the use of BERT in edges and relies on NN and LSTM networks to identify phishing attacks.

Index Terms - Neural Networks, Phishing Attacks, Binary cross entropy loss value, , TensorFlow lite, Quantization, Bidirectional Encoder Representations from transformers, softmax, Distillation loss , DistillBERT, Adam Optimizer, Sigmoid Activation function

I. INTRODUCTION

Phishing is the interesting use of electronic correspondences to dumbfound and take advantage of clients. Phishing attacks try to get fragile, confidential information, for instance, usernames, passwords, Visa information, network licenses, and that is just a hint of something larger. By behaving like a certifiable individual or foundation through phone or email, computerized aggressors use social planning to control losses into performing express exercises — like tapping on a dangerous association or association — or emphatically revealing grouped information. Phishing is a sort of well-disposed planning attack consistently used to take client data like login nuances, Visa numbers and other sensitive information. Aggressors endeavor to copy a real association or a person to take information. For instance, this should be possible by mirroring your financial help to send you an email expressing "your record is in a difficult situation, you want to change your secret word" with a connection to change the secret word. Somebody who knows nothing about the danger might tap on the connection to present their login accreditations to the assailant. This is the most well-known phishing assault and only one strategy among a great deal of other phishing assaults. Phishing is a standoffish and accordingly against reasonable movement, which has a significant social and monetary aspects. Cell phones are standard with programmers since they're planned for quick responses subject to inconsequential sensible information.

Hoodlums trust on trickery and making a longing to move rapidly to put forth progress with their phishing attempts. During a crisis, people are tense. They need information and are looking for course from their directors, the public power, and other significant subject matter experts. A lot of game plans has been proposed for the distinguishing proof and evasion of phishing attacks, still the peril isn't decreased. Email phishing attacks have spiked in light of Crown disease. A successful phishing attack can have shocking implications for the setbacks inciting money related mishaps and discount misrepresentation. Nevertheless, guaranteeing ourselves against phishing is both a potential and central development. With the reliably extending usage of messages and improvement of progressions, danger of losing significant information to fraudsters has in like manner been growing .This examination centers around identifying phishing assaults continuously with the assistance of brain organizations, LSTM and BERT.

II. LITERATURE SURVEY

Usually, Hackers will influence the users through phishing in order to gain access to the organization’s digital assets and networks. With security breaches, cybercriminals execute ransomware attack, get unauthorized access, and shut down systems and even demand a ransom for releasing the access. Anti-phishing software and techniques are circumvented by the phishers for dodging tactics. Though threat intelligence and behavioral analytics systems support organizations to spot the unusual traffic patterns, still the best practice to prevent phishing attacks is defended in depth.

Phishing attacks have been increasing recently. Attackers use clever social engineering techniques to convince their victims into clicking a malware or deceptive login-based webpages. Most solutions for this particular problem focus more on helping desktop computer users than mobile device users. Mobile device users are more vulnerable than their desktop counterparts because they are online most of the time and they have device limitations such as smaller screen size and low computational power.

“Phishing attacks are generally carried on using the four steps listed below:

1. Attackers configure a forged web site that impersonates the legitimate one. They also apply the Domain Name System and configure the web server.
2. A bulk pool of spoofed e-mails are dispatched to end users making them to input their personal details.
3. These victims open or click on the spoofed links assuming that they are from legitimate companies and organizations and provide their personal details.

4. This confidential information from the receivers are acquired and frauds are performed by the phishers.

Phishing websites is one such area where administrators need new techniques and algorithms to protect naïve users from getting exploited. Phishing is an attempt of fraud aimed at stealing our information, which is mostly done by emails. These phishing emails mostly come from trusted sources and try to retrieve our valuable information, for instance our passwords, bank details or even SSN. Many a times, these attacks come from sites where we have not even any type of account. A lot of solutions have been proposed for the detection and prevention of phishing attacks, still the threat is not alleviated. Blacklisting, Uniform Resource Locator (URL) based detection, static detection, and heuristics techniques are various methods used for detecting phishing attacks.

Recent years have witnessed the increasing threat of phishing attacks on mobile computing platforms. In fact, mobile phishing is particularly dangerous due to the hardware limitations of mobile devices and mobile user habits. Existing schemes designed for web phishing attacks on PCs cannot effectively address the various phishing attacks on mobile devices.

As compared to desktop users, mobile device users are at least three times more vulnerable to phishing attacks and the reason for this vulnerability is small screen size, lack of identity indicators, inconvenience of user input, switching between applications, habits and preferences of mobile device users. By exploiting the hardware limitations of these devices and careless behaviour of the users, an attacker can easily carry out phishing attack on mobile phones**”.**

III. Objective of the Study

The primary goal of this work is to foster a model that can distinguish and forestall conceivable phishing assaults continuously. A clever phishing assault discovery system, when carried out to handheld gadgets builds the transmission capacity of assurance in the worldwide advanced town. Forestalling phishing itself assist with achieving feasible turn of events. Various calculations like choice trees and svm are utilized to distinguish phishing assaults. This kind of calculation needs huge computational ability to run and the client information is shipped off the server for the location of phishing assaults. In this present circumstance an outsider approaches our perusing information which will prompt a security issue and discovery can be subject to different variables like organization transmission capacity and so on.

Conventional ai models can be gigantic to the point that it runs on cloud servers, which makes it less effective. The forecasts can take such a long time that it may not be ongoing consistently. This defer in expectation and advance notice can cause malware to influence the framework before we can do anything. Figuring out is additionally impacted in ai models. Aggressors recreated the models for extricating plan data from them. To do this expectation locally in the end-point framework, the model should be smaller and advanced to the point of working continuously limiting the framework asset utilization. With the assistance of dl, each time a danger occurs, the calculation advances without anyone else and forestalls the chance of being taken advantage of.

IV. Proposed Methodology

To conquer the security issue we can utilize continuous forecast on the client-side. The central concern is that right now utilizing identification calculations needs enormous computational power for making expectations and this kind of calculation isn't reasonable for running on the client-side. For distinguishing phishing assaults in low computational gadgets we can utilize a quantized model. The quantized model size is extremely low and it is reasonable for running in low computational gadgets and we can recognize the phishing assaults continuously. The principal benefit of sending the model on the client-side is that the client information isn't shipped off an outsider and there is no security issue in the framework, our perusing information is protected on the client-side. While the model is running on the client-side the speed of the identification is exceptionally quick contrasted with the information shipping off the server for recognition of phishing assaults.

1 Features

After referring to available literature, we have selected and defined a set of features that capture the characteristics of phishing emails.

1. Using the IP Address

If an IP address is used as an alternative of the domain name in the URL, such as “http://125.98.3.123/fake.html”, users can be sure that someone is trying to steal their personal information. Sometimes, the IP address is even transformed into hexadecimal code as shown in the following link “http://0x58.0xCC.0xCA.0x62/2/paypal.ca/index.html”.

2. Using URL Shortening Services “TinyURL”

URL shortening is a method on the “World Wide Web” in which a URL may be made considerably smaller in length and still lead to the required webpage. This is accomplished by means of an “HTTP Redirect” on a domain name that is short, which links to the webpage that has a long URL. For example, the URL “http://portal.hud.ac.uk/” can be shortened to “bit.ly/19DXSk4”.

3. having “@” Symbol

Using “@” symbol in the URL leads the browser to ignore everything preceding the “@” symbol and he real address often follows the “@” symbol.

4. Domain Registration Length

Based on the fact that a phishing website lives for a short period of time, we believe that trustworthy domains are regularly paid for several years in advance. In our dataset, we find that the longest fraudulent domains have been used for one year only.

5. HTTPS (Hyper Text Transfer Protocol with SSL)

The existence of HTTPS is very important in giving the impression of website legitimacy, but this is clearly not enough. Certificate Authorities that are consistently listed among the top trustworthy names include: “GeoTrust, GoDaddy, Network Solutions, Thawte, Comodo, Doster and VeriSign”. Furthermore, by testing out our datasets, we find that the minimum age of a reputable certificate is two years.

6. Favicon

A favicon is a graphic image (icon) associated with a specific webpage. Many existing user agents such as graphical browsers and newsreaders show favicon as a visual reminder of the website identity in the address bar. If the favicon is loaded from a domain other than that shown in the address bar, then the webpage is likely to be considered a Phishing attempt.

7. Port

This feature is useful in validating if a particular service (e.g. HTTP) is up or down on a specific server. In the aim of controlling intrusions, it is much better to merely open ports that you need. Several firewalls, Proxy and Network Address Translation (NAT) servers will, by default, block all or most of the ports and only open the ones selected. If all ports are open, phishers can run almost any service they want and as a result, user information is threatened.

8. The Existence of “HTTPS” Token in the Domain part of URL

The phishers may add the “HTTPS” token to the domain part of a URL in order to trick users. For example, http://https-www-paypal-it-webapps-mpp-home.soft-hair.com/.

9. URL of Anchor

An anchor is an element defined by the <a> tag. This feature is treated exactly as “Request URL”. However, for this feature we examine: If the <a> tags and the website have different domain names. This is similar to request URL feature. If the anchor does not link to any webpage, e.g.:

<a href=“#”>
<a href=“#content”>
<a href=“#skip”>
<a href=“JavaScript ::void(0)”>

10. Links-tags

Given that our investigation covers all angles likely to be used in the webpage source code, we find that it is common for legitimate websites to use <Meta> tags to offer metadata about the HTML document; <Script> tags to create a client side script; and <Link> tags to retrieve other web resources. It is expected that these tags are linked to the same domain of the webpage.

11. Server Form Handler (SFH)

SFHs that contain an empty string or “about: blank” are considered doubtful because an action should be taken upon the submitted information. In addition, if the domain name in SFHs is different from the domain name of the webpage, this reveals that the webpage is suspicious because the submitted information is rarely handled by external domains.

12. Abnormal URL

This feature can be extracted from WHOIS database. For a legitimate website, identity is typically part of its URL.

13. Redirect

The fine line that distinguishes phishing websites from legitimate ones is how many times a website has been redirected. In our dataset, we find that legitimate websites have been redirected one-time max. On the other hand, phishing websites containing this feature have been redirected at least 4 times.

14. On Mouse over

Phishers may use JavaScript to show a fake URL in the status bar to users. To extract this feature, we must dig-out the webpage source code, particularly the “onMouseOver” event, and check if it makes any changes on the status bar.

15. Using Pop-up Window

It is unusual to find a legitimate website asking users to submit their personal information through a pop-up window. On the other hand, this feature has been used in some legitimate websites and its main goal is to warn users about fraudulent activities or broadcast a welcome announcement, though no personal information was asked to be filled in through these pop-up windows.

2 Neural networks

A neural network is a computational model that has a network architecture. This design is comprised of counterfeit neurons. This design has explicit boundaries through which one can change it for playing out specific errands. A neural network has numerous layers. Each layer plays out a particular capacity, and the complex the organization is, the more the layers are. That is the reason a neural organization is additionally called a multi-facet perceptron.

The purest form of a neural network has three layers:

The input layer
The hidden layer/secret layer
The output layer

As the names recommend, every one of these layers has a particular reason. These layers are comprised of hubs. There can be various secret layers in a neural organization as per the prerequisites. The information layer gets the info signals and moves them to the following layer. It assembles the information from the rest of the world.

An example of a neuron showing the input (x_{1}.-x_{n}), their corresponding weights (w_{1} -w_{n}), a bias (b) and the activation function f applied to the weighted sum of the inputs.

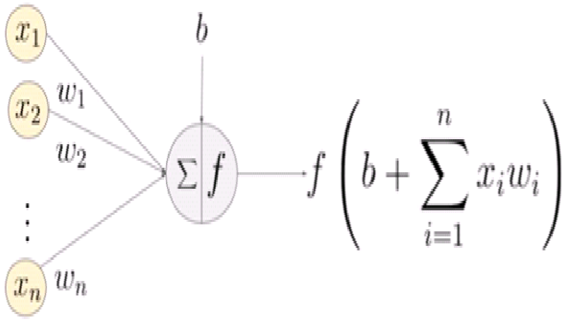

Fig 1: Working of a Simple Neuron

For computation, each neuron considers weights and bias. Then, the combination function uses the weight and the bias to give an output (modified input). It works through the following equation:

combination = bias +weights * inputs

After this, the activation function produces the output with the following equation:

output = activation(combination)

Information is fed into the input layer which transfers it to the hidden layer. The interconnections between the two layers assign weights to each input randomly. A bias added to every input after weights are multiplied with them individually. The weighted sum is transferred to the activation function. The activation function determines which nodes it should fire for feature extraction. The model applies an application function to the output layer to deliver the output. Weights are adjusted, and the output is back-propagated to minimize error. The model uses a cost function to reduce the error rate. You will have to change the weights with different training models. The model compares the output with the original result. It repeats the process to improve accuracy.

3 *ReLU (Rectified Linear Unit) activation function*

The rectified linear activation function or ReLU is a linear function that will output the input directly if it is positive, otherwise, it will output zero.

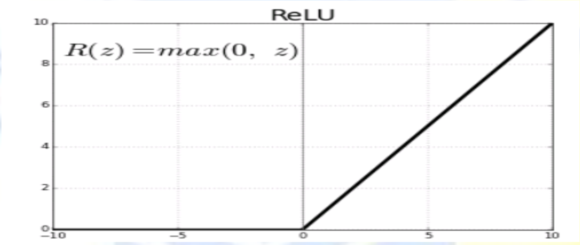

Fig 2: Representation of ReLU

Equation: f(x) = max (0, x)

Range: (0 to infinity)

Advantages of using ReLU are the function and its derivative both are monotonic, due to its functionality it does not activate all the neuron at the same time and it is efficient and easy for computation.

4 *Binary cross entropy loss function*

Binary cross entropy is a loss function that is used in binary classification tasks. These are tasks that answer a question with only two choices (yes or no, A or B, 0 or 1, left or right).

```
model. compile (optimizer='adam',
            loss=tf.keras.losses.binary_crossentropy,
            metrics=['accuracy'])
```

5 *Sigmoid Activation Function*

A neural network is contained layers of hubs and figures out how to plan instances of contributions to yields. For a given hub, the information sources are duplicated by the loads in a hub and added together. This worth is alluded to as the added enactment of the hub. The added enactment is then changed by means of an actuation work and characterizes the particular yield or "initiation" of the hub. The least difficult initiation work is alluded to as the direct enactment, where no change is applied by any stretch of the imagination.

An organization included just straight enactment capacities is extremely simple to prepare, yet can't learn complex planning capacities. Straight initiation capacities are as yet utilized in the yield layer for networks that foresee an amount.

Nonlinear initiation capacities are liked as they permit the hubs to learn more perplexing constructions in the information. Generally, two broadly utilized nonlinear enactment capacities are the sigmoid and exaggerated digression actuation capacities.

The sigmoid actuation work, additionally called the strategic capacity, is customarily an exceptionally famous initiation work for neural organizations. The contribution to the capacity is changed into a worth somewhere in the range of 0.0 and 1.0. Data sources that are a lot bigger than 1.0 are changed to the worth 1.0, comparatively, values a lot more modest than 0.0 are snapped to 0.0. The state of the capacity for all potential information sources is a S-shape from zero up through 0.5 to 1.0. The exaggerated digression work, or tanh for short, is a comparable formed nonlinear enactment work that yields esteems between - 1.0 and 1.0. An overall issue with both the sigmoid and tanh capacities is that they soak. This implies that huge qualities snap to 1.0 and little qualities snap to - 1 or 0 for tanh and sigmoid separately. Further, the capacities are simply truly touchy to switches up their mid-point of their feedback, for example, 0.5 for sigmoid and 0.0 for tanh.

6 *Recurrent Neural Networks*

They are networks with circles in them, permitting data to endure. In the outline, a piece of neural organization, sees some information xt and yields a worth ht. A circle permits data to be passed starting with one stage of the organization then onto the next.

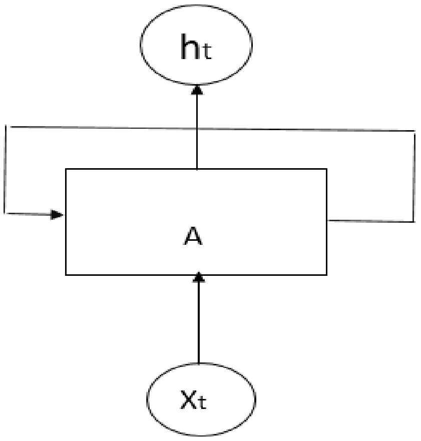

Fig. 3. RNN

7 *LSTM Networks*

All RNNs have the form of a chain of repeating modules of neural network. Repeating module in an LSTM contains four interacting layers. On step t there is a hidden state and a cell state. The cell stores long term information. The LSTM can erase, write and read information from the cell. The selection of which information is erased written or read is controlled by the corresponding gates. The gates are dynamic their value is computed based on current contents.

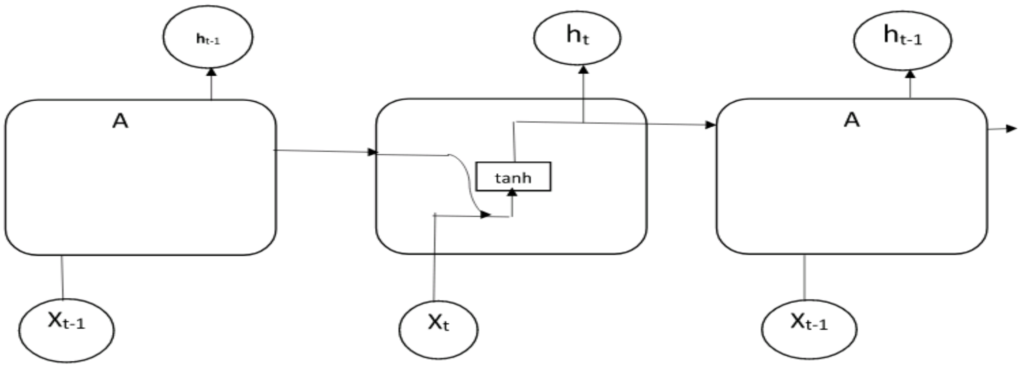

Fig. 4. The repeating module in a standard RNN contains a single layer.

9 *Features of neural network model to make inference at the edge*

Light-weight**:** Edge devices have limited resources in terms of storage and computation capacity. Deep learning models are resource-intensive, so the models we deploy on edge devices should be light-weight with smaller binary sizes.

Low Latency**:** NN models at the Edge should make faster inferences irrespective of network connectivity. As the inferences are made on the Edge device, a round trip from the device to the server will be eliminated, making inferences faster.

Secure**:** The Model is deployed on the Edge device, the inferences are made on the device, no data leaves the device or is shared across the network, so there is no concern for data privacy.

Optimal power consumption**:** Network needs a lot of power, and Edge devices may not be connected to the network, and hence, the power consumption need is low.

Pre-trained**:** Models can be trained on-prem or cloud for different deep learning tasks like image classification, object detection, speech recognition, etc. and can be easily deployed to make inferences at the Edge."

10 *Quantization*

When we save the TensorFlow Model, it stores as graphs containing the computational operation, activation functions, weights, and biases. The activation function, weights, and biases are 32-bit floating points. Quantization reduces the precision of the numbers used to represent different parameters of the TensorFlow model and this makes models light-weight. Quantization can be applied to weight and activations. Weights with 32-bit floating points can be converted to 16-bit floating points or 8-bit floating points or integer and will reduce the size of the Model. Both weights and activations can be quantized by converting to an integer, and this will give low latency, smaller size, and reduced power consumption.

```
converter = tf. lite. TFLiteConverter.from_keras_model(model)
tflite_model = converter. Convert ()
```

11 *Optimization*

Adam is a stochastic slope plunge technique that figures individual versatile taking in rates for various boundaries from evaluations of first-and second-request snapshots of the inclinations. Finally, the Adam enhancer is picked to restrict the adversity and make the model meet. The Adam estimation continuously changes the learning rate for each limit reliant upon the essential solicitation second measure and the second-demand second check of the slant of each limit subject to the disaster work. We picked Adam considering the way that the learning step size of each cycle limit has a particular reach, and will not cause a gigantic learning venture because of a tremendous slant and the limit regard is reasonably consistent. Right when the mishap regard is adequately diminished, the model joins and the readiness closes.

```
modelp. compile (optimizer='adam',
loss=tf.keras. losses. binary_crossentropy,
metrics=['accuracy'])
```

12 *Transformers*

The central piece of all state of the art NLP introducing approaches presented in this work is the general treatment of customary language through transformers. Transformers were at first developed to perform machine translation.

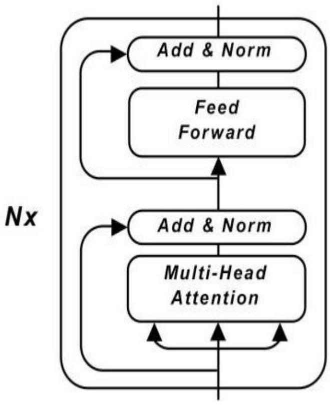

Fig. 5. Encoder Model

Attention mechanism is the part of neural network architecture that is used to dynamically learn relevant features of the input data that contains sequence of textual elements, It can be directly applied to the raw input or to its higher level representation.

$\text{Attention}(Q,K,V)=\text{softmax}((QK^T)/\sqrt{d_k})V$ (1)

Q: L Queries of size d

K: L Keys of size d

V: L Values of size d

L: length of sequence

D: depth of attention

13 BERT

BERT, which implies Bidirectional Encoder Representations from Transformers. Not under any condition like continuous language depiction models, is BERT planned to pre-train significant bidirectional depictions from unlabeled text by together trim on both left and right setting in all layers. Subsequently, the pre-arranged BERT model can be changed with just a single additional yield layer to make top tier models for a wide extent of tasks, for instance, question tending to and language inference, without critical task unequivocal plan changes. BERT's designing develops top of transformer encoders.

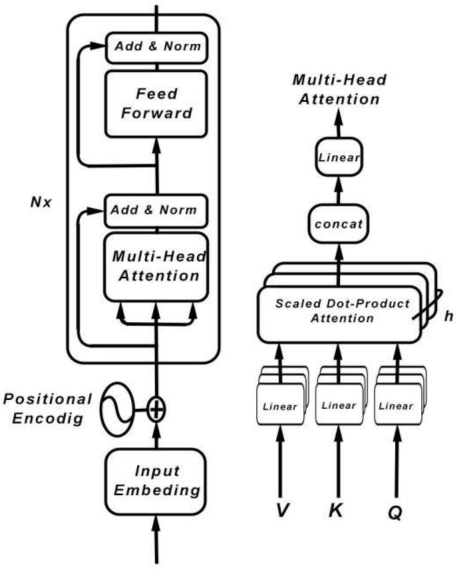

Fig. 6. Attention Mechanism

14 DistillBERT

DistillBERT, a refined form of BERT: more modest, quicker, less expensive and lighter uses Knowledge Distillation. Information Distillation involves a bigger model as an educator for a more modest one which attempts to duplicate its results and sub-layer actuation for a given arrangement of data sources. It empowers bigger models, or outfits of models, to be consolidated into a solitary, more modest understudy organization. This empowers use of cutting edge models to be conveyed where the computational climate is restricted. Refining misfortune. The result probabilities of the instructor (t) are joined with the ones from the understudy (s). To limit this misfortune the understudy organization, along these lines, need to have higher sureness (likelihood) for similar results the educator gave huge probabilities.

Cosine embedding loss: A distance measure between stowed away portrayals for understudy and educator. This basically empowers the understudy to mirror the educator in a larger number of layers than only the result layer, instinctively empowering the production of a superior model. Masked Language Modelling loss: A similar misfortune as utilized while preparing BERT to anticipate the right token covered from the grouping.

V. RESULTS AND CONCLUSIONS

To evaluate the model accuracy 20% test data is used and the weights of the model are quantized for running in edge devices. Using this quantized model, we can run the model in the edge device and the predictions are real time. Accuracy is defined as the number of correct predictions divided by total number of predictions made. When the model training dataset is small and lightweight architecture there will arise situations like overfitting and underfitting in the model. When the hidden units in the model are increased it will affect the speed of the model running in the edge device and computational cost also increases. This model can be deployed either as a browser extension or an application .

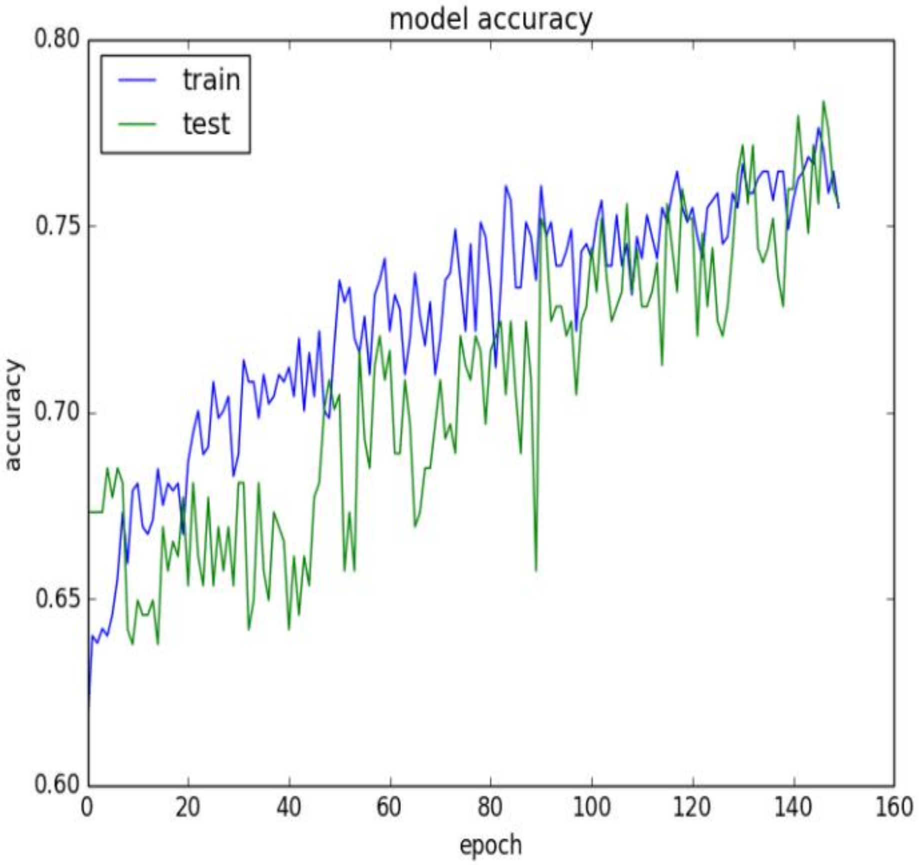

Fig. 7. Model Accuracy using ReLU(NN)

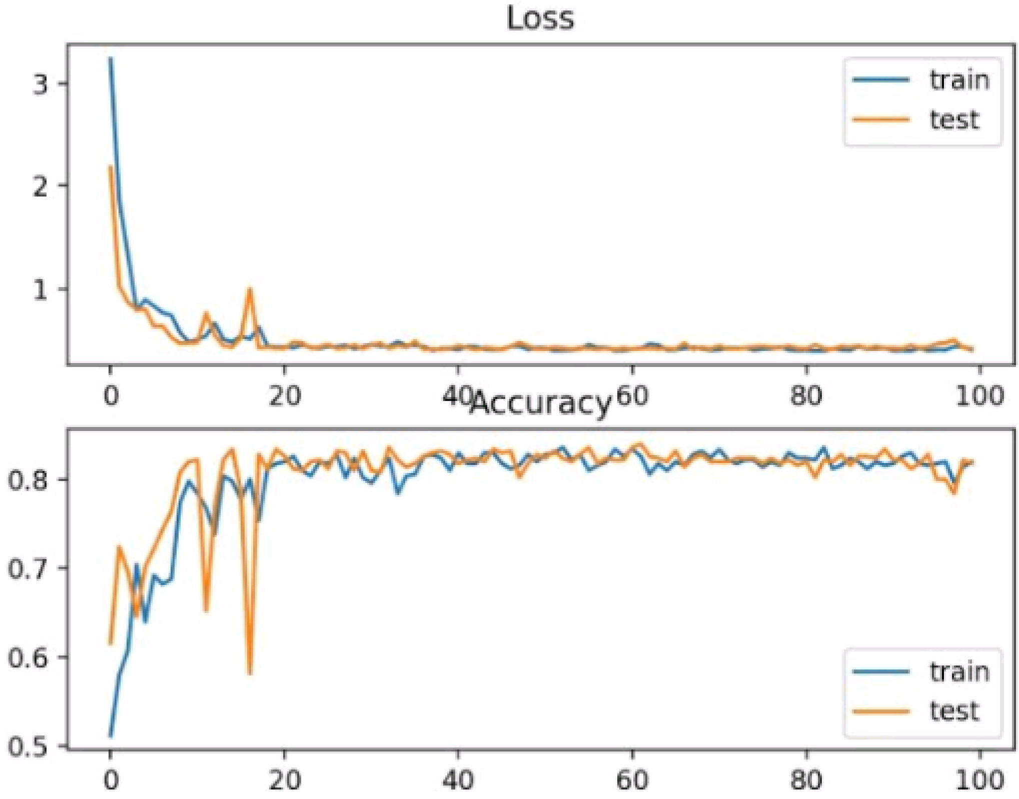

Fig. 8. Model Accuracy using ReLU (LSTM)

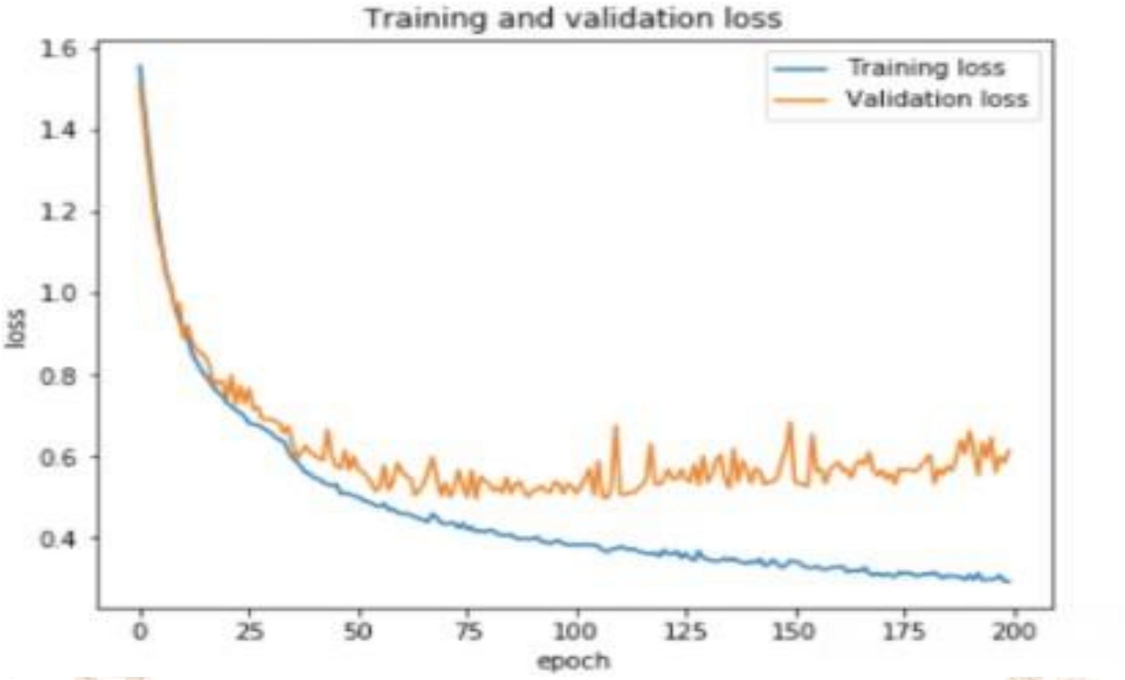

Fig. 9. Training and validation loss(BERT)

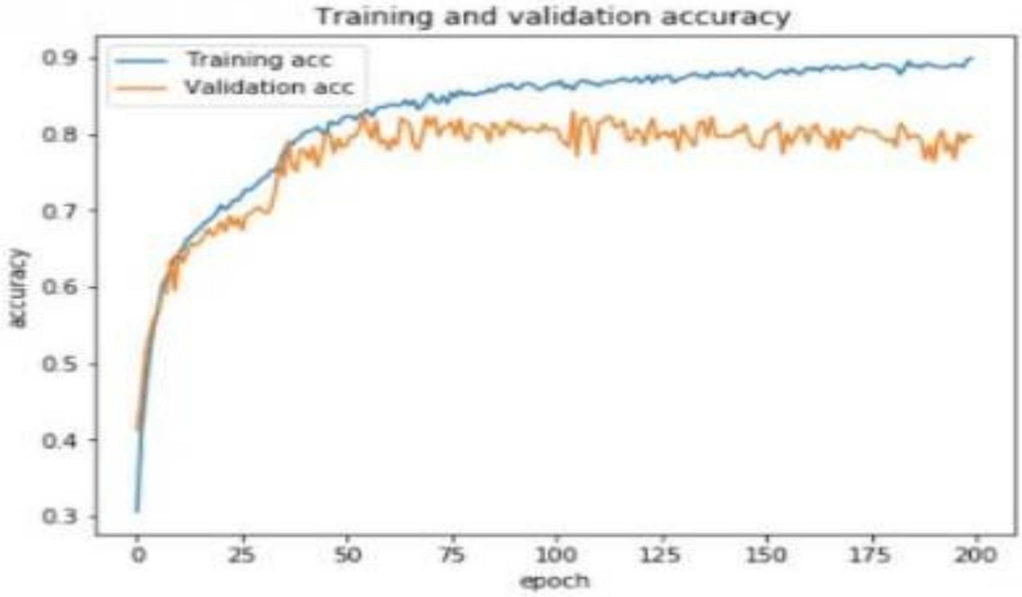

Fig.10 Training and validation accuracy (BERT)

VI. FUTURE WORKS

As what’s to come works, to lessen the execution time and augmentation the efficiency of the edge future work will require upgrading the engineering to diminish the memory size of the model. By decreasing the memory size, we can run the model on many edge gadgets for constant recognition without utilizing a focal server, and client information is dependably on the edge side.

VII. REFERENCES

[1]D. j. Ushus Maria Joseph, "Real Time Detection of Phishing Attacks in Edge Devices," in International Journal of Engineering Research & Technology (IJERT), 2021.
[2]D. S. F. L. S.-E.-U. H. Mohammad Nazmul Alam, "phishing attacks detection using machine learning approach," in Third international conference on smart systems and inventive technology, 2020.
[3]Y. K. b. Jema David Ndibwile, "UnPhishMe: Phishing Attack Detection by Deceptive Login Simulation through an Android Mobile App," in The 12th Asia Joint Conference of Information Security (AsiaJCIS)At: Seoul, South Korea Volume: 978-1-5386-2132-5/17 $31.00 © 2017 IEEE, 2017.
[4]M. M. K. S. H. S. U. S. N. Dr. Reshmabanu, "Phishing Attacks Detection using Machine Learning Approach," in Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2019), 2019.

[5].J. S. S. S. J. S. S. S. IshantTyagi, "Detection of Phishing Attacks using Machine Learning," in 5th International Conference on Signal Processing and Integrated Networks (SPIN), 2018.
[6].A. K. J. Daksha Goel, "Mobile phishing attacks and defense mechanisms: State of art and open research: State of art and open research," computers & security , vol. 73, pp. 519-544, 2018.
[7].X. D. S. M. I. a. J. W. F. I. Longfei Wu, "Effective Defense Schemes for Phishing Attacks on Mobile Computing Platforms," IEEE Transactions on Vehicular Technology, vol. 65, no. 8, 2016.
[8].X. D. a. J. W. Longfei Wu, "MobiFish: A Lightweight Anti-Phishing Scheme for," in International Conference on Computer Communications and Networks (ICCCN), 2014.
[9]."PDRCNN: Precise Phishing Detection with Recurrent Neural Networks," Hindawi security and communication networks, vol. 2019, p. 15, 2019.
[10]."https://www.kaggle.com/akashkr/phishing-url-eda-and-modelling,"[Online]. Available: https://www.kaggle.com/akashkr/phishing-url-eda-and-modelling.
[11].https://www.kaggle.com/akashkr/phishing-url-eda-and-modelling.
[12].J. Brownlee, "A Gentle Introduction to the Rectified Linear Unit (ReLU)," Deep Learning, 2019.
[13]."https://colah.githu D. j. Ushus Maria Joseph,"Developing a real time model to detect SMS phishing attacks in edges using BERT", IEEE Xplore digital library,2022.
b.io/posts/2015-08-Understanding-LSTMs/," August 2015. [Online].
[14].Y. S. Tianrui Peng.Ian G Harris, "Detecting Phishing Attacks Using Natural Language Processing and Machine Learning," in IEEE International Conference on Semantic Computing, 2018.
[15].S. a. D. Soni, "A security model to detect smishing through sms content analysis and url behavior analysis," in Future Generation computer systems-the International Journel of Escience, 2020.
[16].D. j. Ushus Maria Joseph, "Real Time Detection of Phishing Attacks in Edge Devices using LSTM networks", AIP Journals, 2022.
[17]. D. j. Ushus Maria Joseph,"Developing a real time model to detect SMS phishing attacks in edges using BERT", IEEE Xplore digital library,2022.