JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES
www.journalimcms.org

ISSN (Online) : 2454 -7190 Vol.-15, No.-6, June (2020) pp 89-103 ISSN (Print) 0973-8975

A PARALLEL AVERAGED NEURAL NETWORK APPROACH FOR DETECTING SMARTPHONE PHISHES

E Sudarshan[1], Seena Naik Korra[2], P. Pavan Kumar[3], S Venkatesulu[4]

[1]Sumathi Reddy Institute of Technology for Women, Warangal, India.

[2,4]S R Engineering College, Warangal, India.

[3]Faculty of Science and Technology, IFHE Hyderabad, India

Corresponding Author: E Sudarshan

Email: medasare@gmail.com

https://doi.org/10.26782/jmcms.2020.06.00008

Abstract

Smartphone with the Internet is the most common item today and it provides the best online platform for businesses to trade their goods. Customers are more comfortable with online shopping and banking transactions, which are enough for hackers to cheat. Phishing attacks are now very common for smart phones. These attacks come in a variety of ways to steal customer sensitive information and payment information through fake Short Message Service(SMS) or E-Mail or Uniform Resource Locator (URL) links or applications(APPs). Therefore, the end user needs to know a few precautions to avoid phishing attackers. This paper explicitly discusses phishing attacks by their behavior and proposes a parallel defending approach to classifying messages as harm or spams using the Graphics Processing Unit (GPU) platform, which is achieved in logarithmic time of O(n log n) and also discusses the future scope.

Keywords: Smartphone, Phishing, Mobile Security, GPU, Parallel avNNet, Smishing.

I. Introduction

Phishing is a cyber security threat with the help of social engineering techniques to trick internet users into disclosing personal and confidential information. Phishing attacks on smart phones are increasing day by day. In 2019, 5,4,8,288 phishing sites have reported to the APWG. Agari documented that 62% of BEC (Business E-mail Compromise) attacks demanded funds in the form of gift cards, 56% in the third quarter of 2019 and 65% in Q2. 16% of attacks requested payroll diversions, down from 25% in Q3. 22% of BEC attacks include requests for direct bank transfer [IV]. Most of the areas Sensitive user credentials are essential for every end user to enjoy using the Internet for e-commerce and online banking

transactions. Compared to computer system users, smartphone users are at least 3X more prone to phishing attacks and because of smaller screen, lack of identification indicators, input inconvenience to users, switching between apps, habits and preferences of mobile device users [III]. Many hackers easily attacked by phishing attacks by using software, hardware restrictions, and consumer carelessness. Manyusers do not know the phishing attacks and how to prevent it [XLVIII]. As per Symantec, 2017 study 44% of users doesn't know the smart phone security system [XLVI].

Attackers attempt to steal information by sending SMS (Short Message Service) messages or E-Mails or URL links or APPs and redirect the user to unauthorized websites, whereby they may collect bank credentials or personal information [XI]. These unauthorized sites are difficult for users to find because it is very similar to the original sites. The user is facing a major problem with embedded malware, and they are pre-packaged applications where they can unknowingly collect sensitive information from users and pass it on to hackers. Additionally, many users are downloading promoted malicious apps and are unaware of the phishing malware it contains [XVI].

Manufacturers of consumer smart phone devices do not verify the authenticity of URLs to run their businesses [IX]. The phishing attackers use the user's inabilities. Most smart phones have an Android operating system, so phishing attacks mainly attack the Android operating system [XXVII]. Most smart phone users not technically equipped and are unaware of phishing attacks on phones. According to researchers, the causes of phishing attacks are: 1) Most portable devices have difficulty checking hyperlinks. 2) Consumers are not aware of options to prevent phishing attacks. 3) Requires user credentials for application interfaces that an attacker can easily follow [XL]. The foregoing causes continue and linked to ongoing problems with phishing attacks on phones. As attacks continue to evolve, detection methods must search everywhere from vandalism and log data to databases [XXI].

Over ten months, Cui et al. [XVI] and other monitored 19,066 phishing attacks reported on PhishTank. Of those, about 90% of attacks are similar, but are duplicates or deviations of earlier recorded attacks that already stored in the database. At present, there is a great deal of restriction on the need to examine the current state of mitigation methods and understand the limitations and implement a new solution. In an instance, Wardman et al. [L]has tested on PayPal's payment for phishing attacks through the defense mechanism's in real time campaigns.Goel et al. [XXV] discuss the different smart phone phishing attacks and defense practices.

This paper classifies anti-phishing policies and also discusses current solutions to reduce the impact of phishing attacks on smart phones. In addition, this paper focuses on detecting phishing attacks as early as possible.

II. Motivation

To get a broader view of the safety requirements discussed below, we conducted a literature survey of papers relating to Phishing and Smishing.

History

In 1996, it's called "phishing" because of a major fraud involving a registered case of fake credit cards on the American Online (AOL) website [XXXVIII]. Attackers use AOL system as a resource because AOL accepts credit card accounts with no authentication. AOL has found that for paying services, these credit cards are not very valid and the accounts are fake. As a result, it withholds the accounts. Following this, AOL began verifying credit cards correctly. This caused the attacker to find other ways to get AOL accounts. Subsequently, without using fake accounts, the attackers stole AOL users' passwords through emails or messages from AOL employees and by using various services on behalf of lawful users [XXIX].

Phishing attacks have increased over time in the history of hacking. Here's how the evolution of phishing attacks has used the Internet. Algorithm-Based Fishing in 1990: In the early 1990s, AOL detected phishing attacks. The first phishing attackers implemented an algorithm to generate random credit card numbers from AOL accounts to get actual card matching. Once matched phishing attacks were beginning, and after 1995 the scandal erupts. Email Phishing in 2000: Most of the phishers are more sophisticated and technologically advanced. They have adopted inexpensive methods to attack customers.In the same year, a phishing email hit PayPal asking users to compromise their accounts and verify credentials. Phishing by HTTPS in 2018: In the past, phishing went through two main ways: email phishing and domain spoofing. Over time, attackers are using new types of phishing methods. This paper provides the full overview of phishing attacks [XXVIII].

Phishing Life Cycle

It involves the following steps in phishing life cycle [Fig. 1]:***Planning phase:*** An attacker selects a communication channel to begin a phishing attack. This channel may be a phishing webpage, phishing application, email or an SMS with a malicious link.

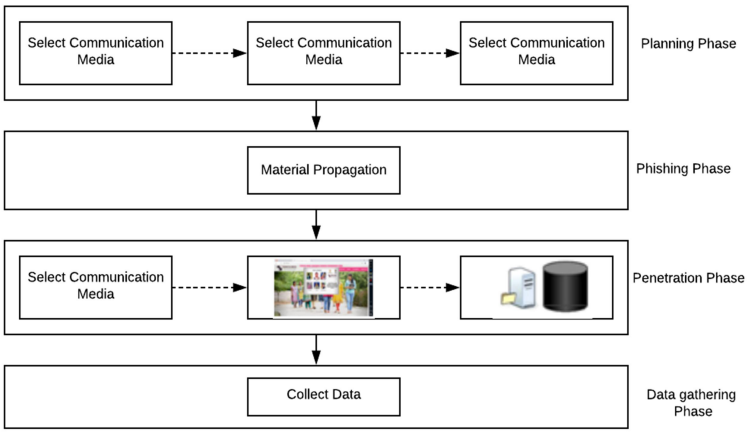

Fig. 1: Phishing life cycle.

An attacker sets a target for a person, company or country, and collects information about them by physically visiting or monitoring them. Once that attack method has selected, it would be identified as a malicious app or SMS.

Phishing Phase: At this stage, it communicates information to the victim. An attacker sends phishing matter to the user's device using illusory SMS or emails posing as a lawful source.

Penetration Phase: After the user opens the published content, the login page appears which redirects to a phishing webpage and asks for personal information or downloads a malicious application, making it vulnerable to an attacker.

Data-Gathering Phase: An attacker gains access to the device, collecting user information through malicious applications for a fake login page. They install if malware on the device, the attacker will collect the information that he wants from the device and Attackers can use gained information for financial benefits or other intent.

Phishing Categories with Behavior

It classifies the attacks according to the phishing channel as shown in Fig.2 [XXV]. Here, it discusses social engineering phishing practices and their behavior: 1) *Vishing*: Vishing attacks usually done by phone calls, where it used for voice attacks, which we call a "vishing" combination of voice and phishing.

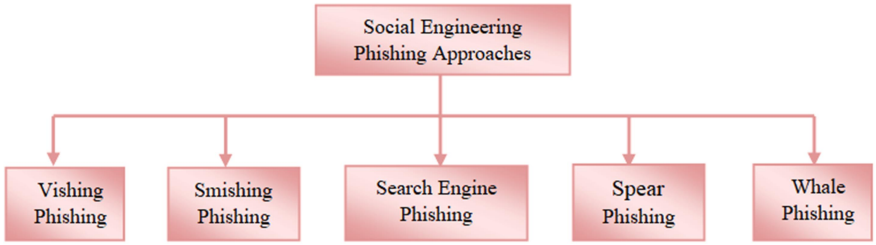

Fig.2: Mobile Phishing Categories

Phishers communicate confidently with the user after receiving some information about them from the social network. That way hacker can hack some sensitive information from users. 2) *Smishing*: Smishing is one of the easiest phishing attack and which would do by SMS alerts. User may receive fake order details or gifts SMS with a cancellation link from attackers to gather sensitive information. 3) *Search Engine Phishing*: Attackers create specific or attracted keywords in the search engine to attract users' attention where fake web pages are. It opens the affiliate link after the user clicks the keyword and it hooks them up to the attacker. This attack has called search engine phishing.4) *Spear Phishing*: This attack differs from traditional phishing. They usually send emails to millions of users targeting a specific user.These attacks were severe because the attacker collects user's information from a variety of sources, such as social media profiles, organizations, and company websites.5)

Whale: Whale attack is like spear phishing, but the target group is more specific and limited in this phishing attack.

In this attack often choose top management positions such as CEO, CFO, COO or others. They have usually called "whales" in terms of phishing.As per APWQ report, the most targeted sector for phishing here is technology, banking and healthcare because of two main factors: it shows large numbers of consumers and high reliance on data in Fig.3 [IV].

Mobile Phishing Protection Policies

In this section, we discuss different approaches for identifying and protecting phishing attacks on smart phones. Different anti-phishing solutions are presented Table 7 in [XXV].

Most of the smart phones are having problems with the internet. Therefore, consumer education is important here to gain an understanding of phishing attacks. Education-based policies include online training through alerts and games. They can alert by identifying passive and active users. Passive users may not pay attention to warnings, and this will be less effective and active users will be more effective. Proper guidance and training required for the customer through mobile games is essential to prevent phishing attacks.

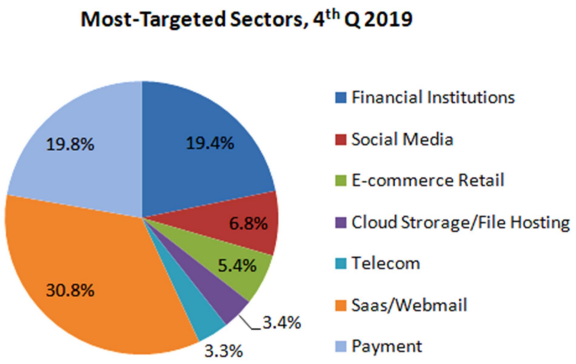

Fig. 3: Most targeted sectors Q4 in 2019

There are various methods to train the customers. Various mobile games are being developed to educate consumers about the concept of phishing attacks.As the use of Internet technology has increased, the risk of piracy attacks on mobile device users has increased. User gets guided by playing a game which being developed to aware the phishing attack [V]. These games educate users about phishing emails and URLs to detect fake and legitimate emails and URLs. Anti-phishing game developed in Google App Inventor Emulator to educate the user. It fixes this game to prevent phishing attacks [VIII]. They create gaming by combining conceptual and procedural knowledge to educate consumers [VII]. This approach integrates "self-efficacy" with the anti-phishing education game to improve user behavior to prevent phishing attacks and other approaches have discussed in [VI, LII].

Related Works

SMS (short messaging service) is a standard communication mechanism for a user without mobile internet and is a very common tool for mobile worldwide. This is because it is a much more affordable and more effective channel than email services [XV]. However, the attacker has focused on this service to perform malicious activities and has caused trouble for consumers and service organizations [II, XXXVII]. Over the past few years, researchers have developed a variety of techniques to detect SMS phishing using machine learning techniques. However, most of the procedures for classifying spam SMS are in the early stages [XXX, XXXIII].In this section; we discussed recently published spam detection methods with pros and cons.

Zainal et al. [LIII] introduced the Bayesian method using the Rapid Miner and Weka tool; they used two freeware tools to experiment. Their research showed that the two tools yield nearly identical results in the same dataset with the same clustering and classification methods. El-Alfy et al. [XX] Spam Detection has introduced on email and SMS platforms with different methods to achieve best results with less complexity. They developed it using the methods of Support Vector Machine (SVM) and Naive Bayes. Eventually, they analyzed the performance of their methods across five SMS and email datasets. Nuruzzaman et al. [XLVII] proposed for SMS spam detection technique. They have found performance for the evaluation of the text spam detection technique of text classification technology. They have developed it in their filtration training and update process. It well shows the result of the method is well when SMS is low storage. Chan et al. [XII] has proposed two approaches for identifies the SMS spam and this targets on weight and length of the messages.Uysal et al. [XLIX] proposed an approach called hybrid method of chi-square and information gain algorithm for feature selection purposes for filtering the spam SMS.For this purpose they have taken the Android platform and based on two different Bayesian classification methods. It categorizes messages as harm and spam.

Serrano et al. [XXXIX] Introduced new technology to filter spam according to external writing style. Here, they protect the writing styles of spam and harm using sequential labeling and term clustering extraction techniques. They achieved good classification on the WEKA platform. Junaid[XXXI] introduced a method for detecting spam messages using evolutionary learning classification. They developed this method with a hexadecimal format to extract two properties. They also examined evolutionary and non-evolutionary classifications. Finally, 89% accuracy classification was performed in their approach. Hidalgo et al. [XXVI] developed Bayesian detection methods for English and Spanish datasets to classify spam in emails and messages. Chowdhury and Jain [XIII] implemented a system with high performance for identifying SMS spam andfor their experiment the True Positive (TP) rate, False Positive (FP) rate; Accuracy and F-measurement have calculated. Here, they presented various classification techniques out of them the Random Forest technique with the best TP rate of 96.1%.Suleiman and Al-Nayamats [XLV] have developed a new method for filtering spam messages using the HBO framework, and the detection of spam messages with feature selection on various machine learning

algorithms have done. Finally, the Random forest algorithm achieves the highest classification accuracy with 96%.

Here is review recently proposed procedures for identifying and filtering SMS spam messages with their vulnerabilities and limitations. According to the literature, there is no comprehensive approach to addressing the problem effectively, which means they are not accurate enough. Therefore, this paper proposes a high performance machine learning method for detecting SMS spam messages with the best classification accuracy.

III. The Proposed Method

Research on machine learning in GPUs precedes the resurgence of recent deep learning. CUDA's creator, Ian Buck, experimented with 2-layer fully connected neural networks in 2005, before joining NVIDIA in 2006 [XLII]. Subsequently, convolutional neural networks have developed on the graphics processing unit (GPU) platform, and here we have observed high performance over high-optimized CPU implementations [XIV]. The release of the first CUDA toolkit brought general-purpose parallel computing to GPUs. At the beginning, CUDA was only accessible through C, C++, and FORTRAN interfaces, but in 2010 the upstream library made CUDA available through Python [XXXIV]. Typical Structure of CUDA As shown in [XLIV], there are many types of memory used to construct the structure.

In this section, we first describe the dataset and feature extraction process, followed by the proposed method.

Data Collection

Here, we have used a dataset with 5,574 text messages, which have classified as spam or harm (legal). These data are publicly available in the UCI Machine Learning Repository [XIX]. Of them, 747 messages are of spam type and 4,827 are of harm type. It shows the results in two parts, the content of the message and its type.

Feature Selection and Extraction

This is a critical step because the feature selection and extraction algorithm can significantly affect the performance. Therefore, it is effective to categorize the most challenging task to identify the useful features from the messages. Therefore, the selection and extraction of the best features should improve and produce the detection rate [XXIV].

Kaldi [XVII] has two neural network stages. It develops these two stages in parallel in this paper on parallel computing platform. In the first stage they implemented a parallel pre-training method [I, XXXII] called Restricted Boltzmann with sequence-discriminative [XXII]. This method uses the general-purpose computing on graphics processing units (GPGPU) [XXXVI] for the parallel training of Natural Gradient Stochastic Gradient Descent (NG-SGD) [XVIII]. In the second stage [LI] the training dataset comprises multiple CPUs. It has developed through a set of GPUs for training and it uses layer-wise non-discriminatory pre-training. The PyCUD Aprogramming language [XXXV] is used to develop the parallel average

neural network model (avNNet)[XLIII, XLI]. This allows the GPGPU environment to run. Henceforth, select prominent features from the message to identify harm and spam, for which we need some sample dataset or feature set, as specified in [XXIII].

It has adapted this method adapted to artificial neural networks because it made up of neurons and helps solve many real-world problems. The main purpose of this approach is to detect SMS messages as harm or spam. This has done with high accuracy and high performance with the help of a parallel averaged neural network model.

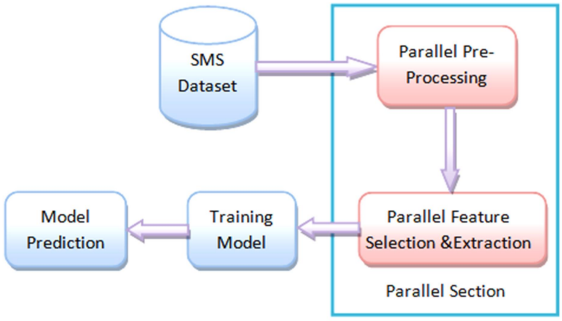

Fig. 4: A structure of Parallel Proposed model

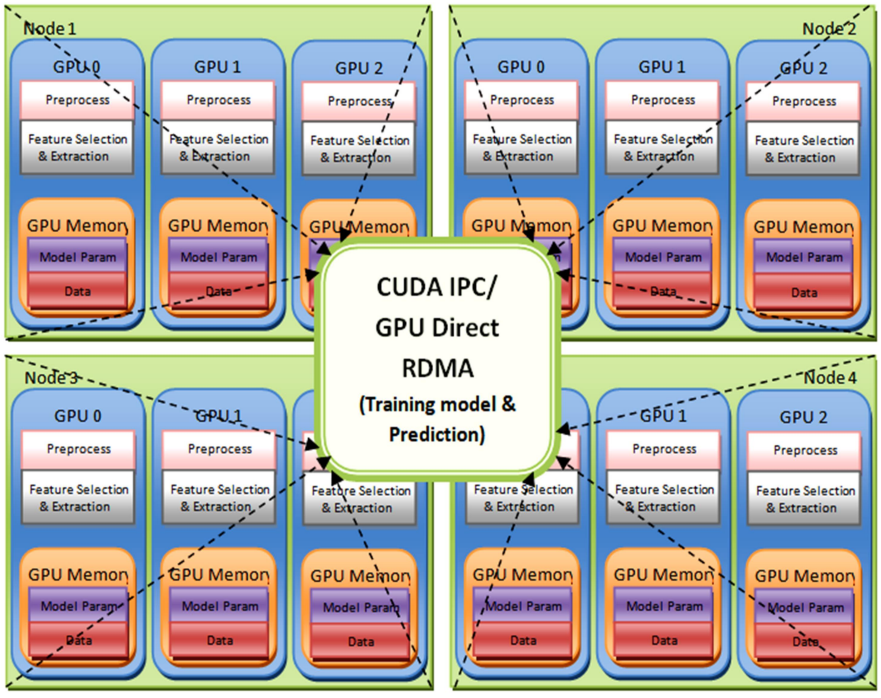

Fig. 5: Architecture of parallel processing in CUDA

Therefore, in the first stage, we gathered the dataset and finalized the experimental features. Subsequently, the selected and extracted features from all the SMS messages in the dataset were used to create feature vectors in parallel, which were used to train and test the proposed model. Finally, we applied the features to classify SMS messages by the proposed model, shown in Fig 4. Figure 5 is a general purpose distribution with GPU computing architecture.

Initially, it pre-processes SMS spam data on all nodes and detects the feature selection and extraction step. Since each node simultaneously activates the data and forwards it to the training model section. We can interpret the SMS spam message as harm and spam.

Experimental Setup

It developed the proposed model using the PyCUDA programming language on the GPGPU platform. The parallel avNNet method has implemented and it contains a hidden layer which concludes the decision using the SMS spam message information received from the predictors. Eventually the average output in Fig.4 is the output of five individual models. We executed two out of five in parallel.

We know that neural networks triggered by hypermeters, such as the number of hidden neurons in hidden layers/ values, limit the weight of neuronal connections. We applied grid searches to find the decay variable value and the correct set of hidden neurons. The achieved result of this process was used to tune the model. Other parameters such as size, bagging, regularization and maximum iterations have used.

To get reliable results, we divide 80 percent of the data for training and 20 percent for validation. To improve model performance in predicting unknown cases, we used the well-known k-fold cross-validation method with model k = 10 for model training. As in [XII, LII] we considered, the predictive measurements have performed to determine performance, accuracy, recall value, and F-measure.

IV. Result Analysis

We calculated the performance of the proposed model based on a series of experiments to detect SMS messages. The main aim of the model is to separate messages into harm or spam in a short time with high performance. Initially, we need to identify the features of harm and spam messages. With that help, we create the feature vector. Extracts features from subsequent messages and applies them to the proposed procedure to get performance metrics. Finally, the output of the proposed model has compared to the same dataset with several latest classification algorithms.

It presents the result of all methods of Table 4 in [XXV]. Performance metrics show that the parallel average neural network model achieves high performance in a short time. The proposed model achieves 0.9884% accuracy and 0.9929% F-measure rate, respectively, and it performs better than other classification methods. It described the time complexity of a parallel gradient batch per unit in [X].

$$b.\left(\frac{tn}{N} + \lceil ld(N)\rceil(4l-1)T\left(\sqrt{n/l}\right)\right) \quad (1)$$

$$\left\lceil\frac{b}{N}\right\rceil.tn + 2\lceil ld(N)\rceil T(n) \quad (2)$$

Where in eq.1&2, b has denoted as the number of samples in a batch, N is for computing nodes, n is the number of network connections in a batch, l is the number of layers, t is the time for computing a connection in a forward neural network.

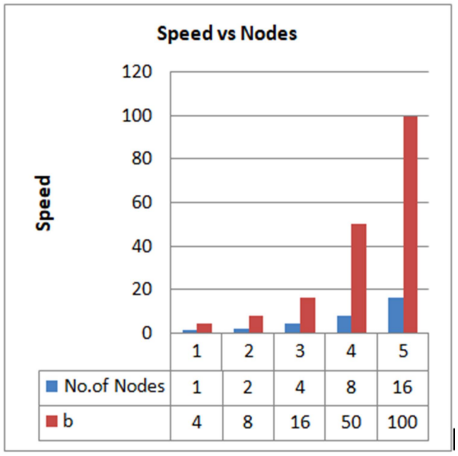

Fig. 6: An example of parallelism with speedup

T is the communication time between the network nodes and the paired nodes in the data size n. Let $T=T_0+\alpha.n$. The load has distributed evenly for the computation and communication of the nodes. The time complexity of this algorithm becomes logarithmic of O(n log n).

We can see the correspondence between the two sets of lines can be. Peaks arise along the node axis, either by adding another node in the tree or by dividing the b patterns into computing nodes. Communication costs become less important as batch size increases to achieve better efficiency in the parallel.The expected speed were there at N = 50 with b = 100 in this example, as see in fig.6.This approach works best in a large dataset rather than a small one.

V. Conclusion

This paper aims to determine whether SMS messages are of harm or spam type after applying a content-based approach to parallel averaged neural networks. The UCI SMS Spam Dataset was used to extract features from 5000 messages in this experiment, and we did these in parallel. We have applied the method of parallel average neural networks to classify harm and spam messages into two categories on captured features. Experimental results show that the message class has highly correlated and that the Parallel Average Neural Network approach has efficiently characterized with improved accuracy and high F-measurement rate. In addition, we have compared the proposed model to recently proposed average neural network

algorithms for classification accuracy and F-measurement in the same dataset along with the time complexities. The time complexity of this algorithm is O (n log n).

The paper's results show that classifying spam messages based on content-based features is a more useful metric because most spammers use suspicious content in their messages. One limitation of this research is that it improves the accuracy and performance of the proposed model by assessing the use of large datasets.

References

I. Abi-Chahla, Fedy. "Nvidia's CUDA: The End of the CPU?'." Tom's Hardware (2008): 1954-7.

II. Almeida, T.A., Hidalgo, J.M.G. and Yamakami, A., "Contributions to the study of sms spam filtering: New collection and results", in Proceedings of the 11th ACM symposium on Document engineering. (2011), 259-262.

III. Amrutkar C, Kim YS, Traynor P. Detecting mobile malicious webpages in real time. IEEE Trans Mobile Comput 2017;16(8):2184–97.

IV. APWG, APWG. "Phishing Activity Trends Report: 4th Quarter 2019." Anti-Phishing Working Group. Retrieved December 12 (2019): 2019.

V. Arachchilage, Nalin, Steve Love, and Michael Scott. "Designing a mobile game to teach conceptual knowledge of avoiding'phishing attacks'." International Journal for e-Learning Security 2, no. 1 (2012): 127-132.

VI. Arachchilage, NalinAsankaGamagedara, and Melissa Cole. "Design a mobile game for home computer users to prevent from "phishing attacks"." In International Conference on Information Society (i-Society 2011), pp. 485-489. IEEE, 2011.

VII. Arachchilage, NalinAsankaGamagedara, and Mumtaz Abdul Harmeed. "Integrating self-efficacy into a gamified approach to thwart phishing attacks." arXiv preprint arXiv: 1706.07748 (2017).

VIII. Arachchilage, NalinAsankaGamagedara, and Steve Love. "A game design framework for avoiding phishing attacks." Computers in Human Behavior 29, no. 3 (2013): 706-714.

IX. Basnet, Ram B., and TenzinDoleck. "Towards developing a tool to detect phishing URLs: a machine learning approach." In 2015 IEEE International Conference on Computational Intelligence & Communication Technology, pp. 220-223. IEEE, 2015.

X. Besch, Matthias, and Hans Werner Pohl. "Flexible data parallel training of neural networks using MIMD-computers." In Proceedings Euromicro

XI. CAPEC. CAPEC-164: mobile phishing; 2017. Available from https:// capec.mitre.org/data/definitions/164.html. [Accessed June 2017].

XII. Chan, P.P., Yang, C., Yeung, D.S. and Ng, W.W., "Spam filtering for short messages in adversarial environment", Neurocomputing, Vol. 155, (2015), 167-176.

XIII. Choudhary, N. and Jain, A.K., "Towards filtering of spam messages using machine learning based technique", in International Conference on Advanced Informatics for Computing Research, Springer. (2017), 18-30.

XIV. Cirecsan, D.; Meier, U.; Gambardella, L.M.; Schmidhuber, J. Deep big simple neural nets excel on hand-written digit recognition. arXiv: 1003.0358 v1 2010.

XV. Cormack, G.V., "Email spam filtering: A systematic review", Foundations and Trends® in Information Retrieval, Vol. 1, No. 4, (2008), 335-455.

XVI. Cui, Qian, Guy-Vincent Jourdan, Gregor V. Bochmann, Russell Couturier, and Iosif-ViorelOnut. "Tracking phishing attacks over time." In Proceedings of the 26th International Conference on World Wide Web, pp. 667-676. 2017.

XVII. D. Povey, A. Ghoshal, G.Boulianne, L. Burget, O.Glembek, N. Goel, M. Hannermann, P.Motl´ıˇcek, Y. Qian, P. Schwartz, J. Silovsk´y, G. Stemmer, and K. Vesel´y, "The kaldi speech recognition toolkit," in ASRU. IEEE, 2011.

XVIII. Daniel Povey, Xiaohui Zhang, and SanjeevKhudanpur, "Parallel training of deep neural networks with natural gradient and parameter averaging," arXiv preprint arXiv:1410.7455, 2014.

XIX. Dua, D. and Graff, C., Uci machine learning repository. 2017.

XX. El-Alfy, E.-S.M. and AlHasan, A.A., "Spam filtering framework for multimodal mobile communication based on dendritic cell algorithm", Future Generation Computer Systems, Vol. 64, (2016), 98-107.

XXI. Fan, Chun-I., Han-Wei Hsiao, Chun-Han Chou, and Yi-Fan Tseng. "Malware detection systems based on API log data mining." In 2015 IEEE 39th annual computer software and applications conference, vol. 3, pp. 255-260. IEEE, 2015.

XXII. Geoffrey E Hinton, Simon Osindero, and Yee-WhyeTeh, “A fast learning algorithm for deep belief nets,” Neural computation, vol. 18, no. 7, pp. 1527–1554, 2006.

XXIII. Gharvirian, F. and Bohloli, A., "Neural network based protection of software defined network controller against distributed denial of service attacks", International Journal of Engineering, Transactions B: Applications, Vol. 30, No. 11, (2017), 1714-1722.

XXIV. Gholami, M., "Islanding detection method of distributed generation based on wavenet", International Journal of Engineering, Transactions B: Applications, Vol. 32, No. 2, (2019), 242-248.

XXV. Goel, Diksha, and Ankit Kumar Jain. "Mobile phishing attacks and defence mechanisms: State of art and open research challenges." Computers & Security 73 (2018): 519-544.

XXVI. Gómez Hidalgo, J.M., Bringas, G.C., Sánz, E.P. and García, F.C., "Content based sms spam filtering", in Proceedings of the 2006 ACM symposium on Document engineering., (2006), 107-114.

XXVII. Grace, Michael, Yajin Zhou, Qiang Zhang, ShihongZou, and Xuxian Jiang. "Riskranker: scalable and accurate zero-day android malware detection." In Proceedings of the 10th international conference on Mobile systems, applications, and services, pp. 281-294. 2012.

XXVIII. https://en.wikipedia.org/wiki/Phishing#History

XXIX. IMPERVA. Cross site scripting attacks; 2017. Available from https://www.incapsula.com/web-application-security/cross-site-scripting-xss-attacks.html. [Accessed June 2017].

XXX. Ji, H. and Zhang, H., "Analysis on the content features and their correlation of web pages for spam detection", China Communications, Vol. 12, No. 3, (2015), 84-94.

XXXI. Junaid, M.B. and Farooq, M., "Using evolutionary learning classifiers to do mobilespam (SMS) filtering", in Proceedings of the 13th annual conference on Genetic and evolutionary computation, (2011), 1795-1802.

XXXII. KarelVesel`y, ArnabGhoshal, Luk´asBurget, and Daniel Povey, “Sequence-discriminative training of deep neural networks,” in INTERSPEECH, 2013, pp. 2345–2349.

XXXIII. Kim, S.-E., Jo, J.-T. and Choi, S.-H., "Sms spam filterinig using keyword frequency ratio", International Journal of Security and Its Applications, Vol. 9, No. 1, (2015), 329-336.

XXXIV. Klöckner, A. PyCuda: Even simpler GPU programming with Python. GPU Technology Conf. Proceedings, Sep. 2010, 2010.

XXXV. Klöckner, Andreas, Nicolas Pinto, Yunsup Lee, Bryan Catanzaro, Paul Ivanov, Ahmed Fasih, A. D. Sarma, D. Nanongkai, G. Pandurangan, and P. Tetali. "PyCUDA: GPU run-time code generation for high-performance computing." Arxiv preprint arXiv 911 (2009).

XXXVI. Owens, John D., David Luebke, Naga Govindaraju, Mark Harris, Jens Krüger, Aaron E. Lefohn, and Timothy J. Purcell. "A survey of general-purpose computation on graphics hardware." In Computer graphics forum, vol. 26, no. 1, pp. 80-113. Oxford, UK: Blackwell Publishing Ltd, 2007.

XXXVII. ParandehMotlagh, F. and KhatibiBardsiri, A., "Detecting fake websites using swarm intelligence mechanism in human learning", International Journal of Engineering, Transactions A: Basics, Vol. 31, No. 10, (2018), 1642-1650.

XXXVIII. Rekouche, Koceilah. "Early phishing." arXiv preprint arXiv: 1106.4692 (2011).

XXXIX. Serrano, J.M.B., Palancar, J.H. and Cumplido, R., "The evaluation of ordered features for sms spam filtering", in Iberoamerican Congress on Pattern Recognition, Springer., (2014), 383-390.

XL. Shahriar, Hossain, TulinKlintic, and Victor Clincy. "Mobile phishing attacks and mitigation techniques." Journal of Information Security 6, no. 03 (2015): 206.

XLI. Sheikhi, S., M. T. Kheirabadi, and A. Bazzazi. "An Effective Model for SMS Spam Detection Using Content-based Features and Averaged Neural Network." International Journal of Engineering 33, no. 2 (2020): 221-228.

XLII. Steinkraus, D.; Buck, I.; Simard, P. Using GPUs for machine learning algorithms. Eighth International Conference on Document Analysis and Recognition (ICDAR'05). IEEE, 2005, pp. 1115–1120.

XLIII. Su, Hang, and Haoyu Chen. "Experiments on parallel training of deep neural network using model averaging." arXiv preprint arXiv:1507.01239 (2015).

XLIV. Sudarshan, E., and K. SeenaNaik. "A Parallel Approach for Maximum Quantization of Descendants Of Wavelet Trees."

XLV. Suleiman, D. and Al-Naymat, G., "Sms spam detection using h2o framework", Procedia Computer Science, Vol. 113, (2017), 154-161.

XLVI. Symantec. Symantec internet security threat report 2014, Vol. 19; 2017a.

XLVII. TaufiqNuruzzaman, M., Lee, C., Abdullah, M.F.A.b. and Choi, D., "Simple sms spam filtering on independent mobile phone", Security and Communication Networks, Vol. 5, No. 10, (2012), 1209-1220.

XLVIII. Tewari A, Jain AK, Gupta BB. Recent survey of various defense mechanisms against phishing attacks. J Info Privacy Sec 2016;12(1):3–13.

XLIX. Uysal, A.K., Gunal, S., Ergin, S. and Gunal, E.S., "A novel framework for sms spam filtering", in 2012 International Symposium on Innovations in Intelligent Systems and Applications, IEEE., (2012), 1-4.

L. Wardman, Brad, Michael Weideman, JakubBurgis, Nicole Harris, Blake Butler, and Nate Pratt. "A practical analysis of the rise in mobile phishing." In Cyber Threat Intelligence, pp. 155-168. Springer, Charm, 2018.

LI. Xiaohui Zhang, Jan Trmal, Daniel Povey, and SanjeevKhudanpur, "Improving deep neural network acoustic models using generalized maxout networks," in Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE, 2014, pp. 215–219.

LII. Yang, Weining, AipingXiong, Jing Chen, Robert W. Proctor, and Ninghui Li. "Use of phishing training to improve security warning compliance: evidence from a field experiment." In Proceedings of the hot topics in science of security: symposium and bootcamp, pp. 52-61. 2017.

LIII. Zainal, K., Sulaiman, N. and Jali, M., "An analysis of various algorithms for text spam classification and clustering using rapidminer and weka", International Journal of Computer Science and Information Security, Vol. 13, No. 3, (2015), 66.