Bitcoin Ransomware Detection using Deep Neural Networks

Abhishek Aditya BS*
Department of Computer Science
PES University, Bangalore, India
abhishek.aditya10@gmail.com

Vinay P Naidu*
Department of Computer Science
PES University, Bangalore, India
vinaypurushothamnaidu@gmail.com

Vishal R*
Department of Computer Science
PES University, Bangalore, India
vishalramesh01@gmail.com

Abstract—**Ransomware is a sophisticated malware that has grown quickly in recent years, that prevents users from accessing their data using encryption techniques until a ransom is paid to the attacker. Traditional machine learning algorithms tend to be biased towards the more frequently occurring class categories, and fail to capture the general pattern or structure of the ransomware bitcoin addresses. They tend to over fit to the data provided thereby performing poorly on the real world data instances, which have unknown class category labels. Hence we believe that traditional machine learning algorithms do not perform well enough to be used in such a critical data security problem. Deep Neural Networks have proven to work well with time-series data as well as multi-class classification and clustering problems, and it captures various levels of granularity of the underlying structure in the data set at different layers of the model architecture.**

Index Terms—**Bitcoin, Ransomware, Detection, Deep Neural Networks, Deep Learning, Cryptocurrency**

I. Introduction

Ransomware is a sophisticated malware that has grown quickly in recent years, that prevents users from accessing their data using encryption techniques until a ransom is paid to the attacker. This results in huge losses for businesses and individuals. Ransomwares can be classified into numerous types, the most violent and virulent being crypto ransomware. Crypto ransomware not only encrypts user data, but it also tries to encrypt information on both mapped and unmapped network devices, putting a whole department or company to a standstill if just one machine is infected [1]. In this type of attack, the attacker does not benefit by selling user data on illegal websites. They reap the benefits from the value associated with the victim's data and by the money paid by the victim to release their data. This attack causes temporary or permanent loss of valuable information and blocking the regular operations. Crypto ransomware prefers Bitcoin network as during the ransomware attack, the victim's system remains fully functional thus allowing the victim to pay the ransom in Bitcoins on the system [2].

Bitcoin is a peer-to-peer online communication protocol which was introduced by group of developers [3]. This framework can be used to make electronic payments since it servers as a virtual currency. Bitcoin network is decentralised and the transactions are stored in a public ledger that is distributed to all the nodes in the network. Transactions recorded in the ledger are irreversible and is not under any influence of a single authority. A Bitcoin address consists of alphanumeric characters which are obtained from public and private keys of the user. This makes the user details pseudo-anonymous. The traditional ransom payment methods have a number of drawbacks such as restricted geographic availability reduces the number of paying victims, and they are operated by businesses subject to local laws, which may force them to reverse transactions or monitor ransom receivers [4]. To overcome this the attackers have adapted to Bitcoin.

Like other types of malware, ransomware spreads through a multitude of channels like malicious email attachments, pay-per-install networks and existing weaknesses in network facilities to spread inside a local network [4]. Once the ransomware is executed on the host it encrypts the files and documents and shows a ransom message on the screen saying files will be decrypted upon paying ransom in Bitcoins. The ransom message includes ransom addresses of Bitcoin wallet victims are supposed to pay into. Once the payment is done ransomware decrypts the files and documents which were held for ransom. The ransomware attackers then exchange the bitcoins for cash and other fiat currencies. Various Machine Learning algorithms [5]–[7] have been proposed to detect the Ransomware Bitcoin addresses, but they do not generalize enough to classify Bitcoin addresses belonging to different Malware families and the test results need to be validated on more recent Bitcoin addresses.

In this paper we tried out different machine learning approaches to detect the Bitcoin ransomware addresses and we found out that neural network model performs slightly better compared to other models in metrics like Receiver Operating Characteristic (ROC), F1 score and accuracy. Neural network captures the information granularity at various layers which helps to detect ransomware addresses in a more generalized way.

This paper goes into further details on ransomware works in section II, related works in section III, details on implementation in section IV, experimental results in section V and conclusion in section VI.

II. Background

Ransomware attack begins when the targeted users receive suspicious mail-attachments or if any of their network services, pay-per-install have any vulnerabilities which can be exploited [4]. The ransomware software starts executing on the host system and encrypts the files and documents valuable to the user. Once the encryption is done, the attacker will transmit a Bitcoin address to which the victim needs to send money in Bitcoins. [8]. The ransomware shows a ransom message alert on the user's screen, which asks the user to pay the ransom in Bitcoins. Once ransom is paid, the user's files are decrypted and user then gets access to their files. The address which the attacker send to victim is a ransom address using which we can obtain clustering heuristics in the Bitcoin network.

The ransom message instructs the victim to buy Bitcoins from specific exchanges and online facilities that allow the conversion of Bitcoins to traditional currencies. These exchanges operate either globally or regionally and some of them cab be centralized while others allow direct exchange facility. Some of the Ransomware families like locky and cerber will generate ransom address which is unique and can be used to identify the paying victims, while WannaCry and CryptoDefense ransomware families reuse the same address, but the victim needs to send hashed payment transaction so that the attacker can verify the paying victims. On the payment of ransom the victim's files and documents are either automatically decrypted by the ransomware or the attacker sends the decryption keys which can be used to decrypt their files.

To convert the Bitcoins obtained by ransomware attacks into fiat currencies (eg. USD) the attackers deposit the bitcoins into exchanges. Some of the exchanges require them to provide information about their clients, so they often deposit the Bitcoins into mixers, which are services that intermix bitcoin inflow from various sources, to conceal the bitcoin trails.

Crypto Ransomware can be classified into three types [9], [10]:

- Symmetrical Cryptosystem Ransomware : uses a symmetrical encryption algorithm like AES or DES for encrypting user's files.
- Asymmetrical Cryptosystem Ransomware : uses a public key included inside the ransomware file or obtained during contact with the command and control (C&C) server which encrypts the files.
- Hybrid Cryptosystem Ransomware : creates a symmetric public key dynamically which is encrypted by the public key embedded in the ransomware file.

There has been extensive research and study on detection of ransomware addresses which includes some approaches like using different heuristics for grouping the bitcoin addresses into subsets or clusters which can be linked to ransomware attackers. Multiple-input heuristics [11] states that two input addresses utilised in the same transaction is likely to be executed by the same ransomware attacker. When one input address is used in conjunction with other input addresses in another transaction, they can all be connected to the same ransomware attacker. Modern bitcoin analytic solutions currently enable two key features: grouping of bitcoin addresses and labeling addresses with attribute data (e.g. chainalysis, elliptic, graphsense,bitcluster) [8].

Some other approaches include identifying patterns of unique traits inside malware code or behaviour to differentiate malware from non-malicious apps, for example, NLP could be used to analyse the function calls, installation activities etc [12]. Before executing their damaging payloads, many ransomware variants seek to connect to C&C server, hence network-based tactics can be beneficial in detecting ransomware attacks.

III. Related Works

Masarah Paquet-Clouston et. al, in their paper [8] used a data driven approach to identify and gather data on bitcoin transactions related to suspicious activities influenced by public bitcoin blockchain's digital footprints. Monetary flows were traced by computing network representations and calculating the statistics(like the amount of transactions and their anticipated value) that's flowing between two addresses) for each directed edge. Cluster graphs were used to partition addresses into groups or subsets which belongs to the same ransomware attacker.

Ahmad O. Almashhadani et. al. in their paper [1] designed a network based intrusion detection system with two independent classifiers, packet classifier and flow-based classifier, working in parallel on packet and flow levels to detect the packet-level and flow-level feature vectors coupled with a decision unit which detects any suspicious activity. On a packet-based data set using Random Forest algorithm they were able to achieve a F1 score of 0.979 and an accuracy of 98.72 % . On a flow-based data set, Bayes Net algorithm achieved a F1 score of 0.971 and 99.83% accuracy.

Danny Yuxing Huang et. al. in their paper [4] proposed a system for tracing reported victim's payment (victims who reported on public forums) and group them with previously unidentified victims (victims who have not reported) to filter out the transactions not linked to ransom payments. They found that Cerber and Locky ransomware families generated most income and over the course of 22 months, they were able to track $16K USD in approximately 19,000 suspected victim ransom payments for five ransomware variants.

Cuneyt G. Akcora in their paper [13] used Topological Data Analysis (TDA) for detection of ransomware transaction patterns on the Bitcoin blockchain. Bitcoin transaction graph was created using the bitcoin ledger, with nodes as addresses and edges connecting the address and transaction nodes. Co-spending heuristic [14] was used as one of the baseline methods for detecting ransomware addresses. Compared to conventional detection methods like DBSCAN clustering algorithm they achieved an accuracy of 69% with TDA, with overall gain of 213.8% and compared to pairwise Cosine similarity algorithm they were able to achieve an accuracy of 78% with a gain of 2.9%.

Authors of [15] used a pre-encryption detection algorithm (PEDA) which detects the ransomware before it encrypts the user files. Compared to machine learning algorithms, random forest, naive bayes and ensemble. PEDA was shown to perform the best, with an AUC of 0.9930 and a test error of 0.0295. Their approach has few shortcomings, it may achieve a high false positive rate and only known crypto ransomwares were detected using this approach.

Kirat Jadhav in his paper [5] showcases the impact of several supervised machine learning techniques in the detection of Bitcoin payments for Ransomware attackers. The Gradient Boosting and XGBoost algorithms successfully recognised more attack types with an accuracy of 99% and average F-Measure of 0.98. However we believe that the results obtained were incorrect because the data set in question is very imbalanced and no preprocessing setps were taken in converting skewed to normal distribution, hence the models had a bias towards the valid Bitcoin addresses than the ransomware bitcoin addresses.

Authors of [7] used a Recurrent Neural Network model for identifying crypto ransomware attacks. The opcodes from Windows applications' are analysed using the RNN. Several models were trained and evaluated using cross validation with 10 folds and the best configuration achieves about 98% detection accuracy.

Authors of [16] used a Software Defined Networking (SDN) approach that utilises the network communication. Attacks were detected by analysing HTTP communication patterns and their related content sizes. Communications of two ransomware variants, namely crypto-wall and locky were analysed. With 1–2 percent or 4–5 percent false positives, they were able to attain detection rates of 97–98 percent. The results from the experiment show that the proposed method is both practicable and efficient.

By analyzing the power usage of Android devices, authors of [12] presented a machine learning-based strategy for detecting ransomware outbreaks. To distinguish ransomware from non-malicious apps, the suggested technique tracks the energy utilization trends of several operations. Four ML algorithms, namely KNN, SVM, random forest and Neural Networks were used. These conventional models weren't as promising and hence a method is proposed where power consumption samples are separated into subsets before various classification approaches are used to overcome the large distribution of attributes. This new technique produces much better results and achieves accuracy of 95.65% and a precision of of 89.19%.

IV. Implementation

Bitcoin Heist Ransomware Address dataset was used to perform bitcoin ransomware detection. The dataset was obtained by parsing the bitcoin transaction graph. The graph consists of daily Bitcoin transactions from 2009 to 2018. Since ransom amounts are rarely below 0.3 Bitcoins, the network edges that transferred less than the threshold were removed. The dataset consists of 10 attributes in which the last attribute is the target label of the bitcoin addresses.

The dataset attributes are described below [13] :

TABLE I: Dataset Attribute Details

Attribute	Description
address	unique Bitcoin address
year	year of the transaction.
day	day of the year. 1 denotes 1st day, similarly 365 is the last day of the year.
weight	sum of the coins that come from a beginning transaction.
count	number of beginning transactions which the acyclic directed path connects the address node of interest to a number of beginning transactions..
looped	number of starter transactions connected to the address node of interest by more than one directed path.
neighbors	number of transactions with the output address of the address node of interest.
income	total quantity of coins produced to the address node of interest .
label	name of the ransomware variant (e.g., Cryptxxx, CryptoLocker etc) or white (i.e., not a ransomware).

The dataset imposes few challenges like skewed distribution of the class labels i.e the dataset is imbalanced which might affect the quality of the classifier models, its performance metrics, and its generalization to real world ransomware address detection. Also the results obtained from the classifier model cannot be validated on more such datasets because the data that contains the ransomware addresses with proper target labels is relatively smaller than the non-ransomware addresses. Since the data is heavily skewed the model gives more bias towards the majority classes and does not take into account the minority classes which results in overfitting of the data.

A. Data Preprocessing

On inspecting the dataset, we found there is no values in any of the feature. Irrelevant features like year,day,Bitcoin address were dropped since year and day does not add any value to the classification and each of the Bitcoin addresses are unique so its of no use in the task of detecting bitcoin ransomware addresses. On performing co-relation analysis we found the features are not co-related so we cannot apply any dimensionality reduction techniques like PCA.

On inspecting the distribution of each feature, all of the features under consideration are highly right skewed as shown

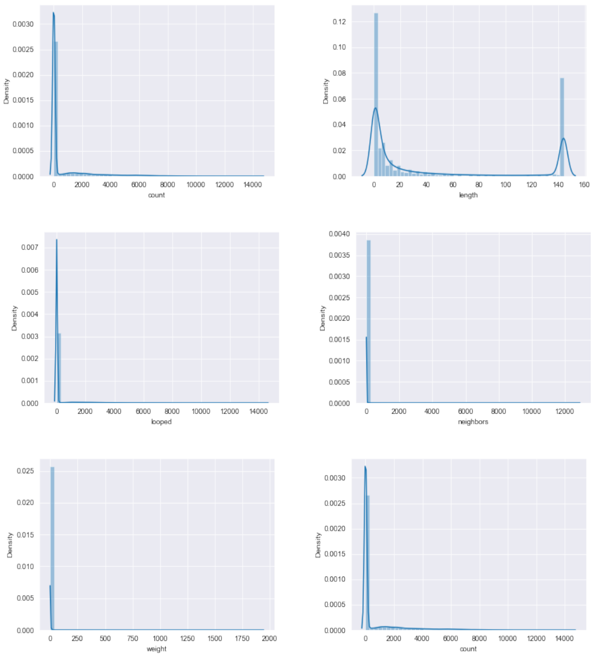

Fig. 1: Distribution of each feature showing they are highly skewed

in Fig. 1, so log transformations were applied to convert them into normal distributions. On plotting the box plots there were no outliers in the dataset. The label feature which shows if the address belongs to ransomware or not has two types of label categories. white label which indicates the address is not a ransomware address and rest of the label categories which can be grouped under black label which indicates it is a belongs to ransomware addresses. The label feature is imbalanced since the proportion of white label is more than 98.5% as shown in Fig. 2 and TABLE II.

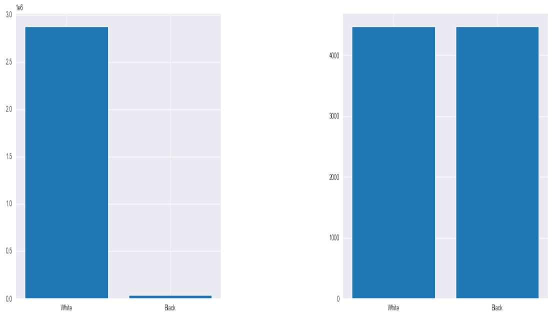

Fig. 2: Target Label imblanced (left) balanced (right)

TABLE II: Dataset Attribute Details

Label	**Count**
White	2875284
PaduaCryptoWall	12390
MontrealCryptoLocker	9315
PrincetonCerber	9223
PrincetonLocky	6625
MontrealCryptXXX	2419
MontrealNoobCrypt	483
MontrealDMALockerv3	354
MontrealDMALocker	251
MontrealSamSam	62
MontrealCryptoTorLocker2015	55
MontrealGlobeImposter	55
MontrealGlobev3	34
MontrealGlobe	32
MontrealWannaCry	28
MontrealRazy	13
MontrealAPT	11
MaduaKeRanger	10
MontrealFlyper	9
MontrealXTPLocker	8
MontrealXLockerv5.0	7
MontrealVenusLocker	7
MontrealCryptConsole	7
MontrealEDA2	6
MontrealJigSaw	4
PaduaJigsaw	2
MontrealXLocker	1
MontrealSam	1
MontrealComradeCircle	1

Binary label encoding was performed on target label, neighbours, length, count, looped columns. In target label column white label was encoded as 1 and rest of the labels grouped as black label was encoded as 0. In neighbours column those values which was greater than 2 was encoded 0 else 1, in length column those values which was greater than 8 was encoded as 0 else 1, in count as well as looped columns those values which was greater than 1 was encoded as 0 else 1. Encoding as 1 indicates its a non-ransomware address whereas encoding 0 indicates its a ransomware address. The reason for this is those addresses which have more than 2 neighbours, more than 8 units of length, count and looped greater than 1 belong to ransomware addresses. Next step was to balance the labels since 98.5% belonged to white label, so count of black labels was considered to sample the same number of white labels so the model would have same number of white as well as black labels and hence no bias will be introduced towards one category.

B. *Model Building*

The dataset was divided into 80% train and 20% test splits. Various scaling techniques like StandardScaler, MinMaxScaler, RobustScaler was applied to both the train and test data to scale them down appropriately. Various supervised classification machine learning models was applied like logistic regression, KNN, SVM, Decision Tree, Random Forest, AdaBoost, XGBoost, Neural Networks. To evaluate our models we have used metrics like accuracy, precision, recall, F1 score and ROC.

V. Results

Logistic Regression was applied on our cleaned dataset. However, the model did not perform well on the training set. The model achieved 69% accuracy on the test set and had low precision, recall and F1 scores. Logistic Regression assumes that the data is linear and tries to fit linear decision boundaries on the training data to perform classification. K-nearest Neighbour (KNN) was trained on the dataset. Since KNN is a lazy learning algorithm that performs classification of new example based on the classes of k nearest neighbours. KNN does not perform well when the data is large. KNN is also sensitive to nosiy data and is not robust to outliers. Because of which, KNN does not perfom well on our test dataset resulting in 72% accuracy.

Support Vector Machines (SVM) was the next model that we tried. SVMs work well with both linear and non-linear data. SVMs use different kernels that transform the input features into a much higher dimensions and make predictions on them. SVM achieved an accuracy of 72% on our test data. However, it did not improve in precision, recall and F1 scores. Decision Trees were used for ransomware classification. In our experiments, we noticed that the decision trees performs the worst compared to all the other algorithms we used. This maybe due to the fact that a small change in the data causes the whole structure of the decision tree to change. This leads to instability as the tree structure keeps changing during training. Decision Trees obtained only 67% accuracy on our test set. RandomForest models were also tested on our test splits. The model achieves about 72% accuracy on our test set and there is a slight increase in precison, recall and F1 scores.

AdaBoost and XGBoost were two boosting algorithms that we also tried for detecting ransomware addresses. Both of the models showed a decent improvements in all the metrics that we used to evaluate the model. Finally, we tried applying a deep neural network on the problem and checked if it performs well compared to other algorithms. In our experiments, we found out that neural networks works well with our dataset and scores the highest in all the metrics.

TABLE III: Model Comparison Study

Model	**Acuracy**	**Recall**	**precision**	**F1 Score**	**ROC**
Logistic Regression	0.69	0.62	0.73	0.67	0.69
KNN	0.72	0.69	0.73	0.71	0.71
SVM	0.72	0.61	0.79	0.69	0.72
Decision Tree	0.67	0.68	0.67	0.68	0.67
Random Forest	0.72	0.70	0.73	0.71	0.72
AdaBoost	0.73	0.64	0.78	0.70	0.72
XGBoost	0.74	0.65	0.79	0.72	0.73
Neural Network	0.74	0.69	0.79	0.72	0.74

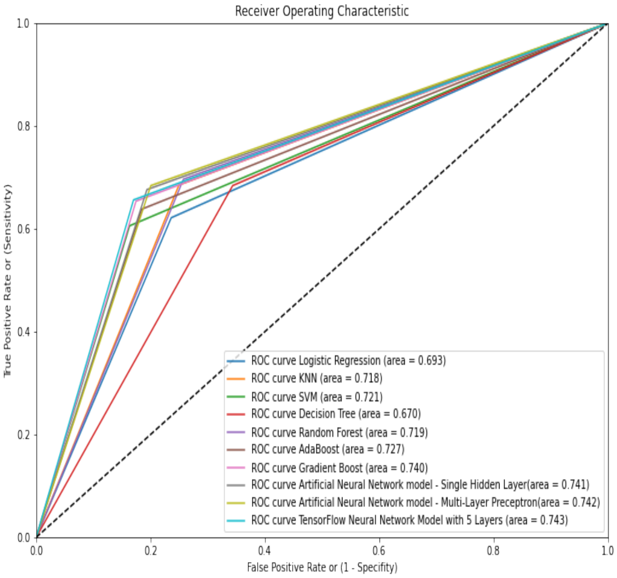

Fig. 3: ROC curves of all models

Follow our work on Github : https://github.com/iVishalr/Bitcoin-Ransomware-Detection

VI. Conclusion

Because neural networks have the capacity to learn on their own and produce output which is not limited by inputs, they can learn from past events and apply what they have learned whenever a similar circumstance arises, enabling them to cope with real-time problems, therefore it performs better than conventional machine learning algorithms. It would have performed much more significantly if the proportion of white to black labels was more, if the data had not been skewed and even with log transformations the data is still not normal enough to show good metric results.

REFERENCES

[1] A. O. Almashhadani, M. Kaiiali, S. Sezer, and P. O'Kane, "A multi-classifier network-based crypto ransomware detection system: A case study of locky ransomware," *Ieee Access*, vol. 7, pp. 47 053–47 067, 2019.

[2] R. Richardson and M. M. North, "Ransomware: Evolution, mitigation and prevention," *International Management Review*, vol. 13, no. 1, p. 10, 2017.

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized Business Review*, p. 21260, 2008.

[4] D. Y. Huang, M. M. Aliapoulios, V. G. Li, L. Invernizzi, E. Bursztein, K. McRoberts, J. Levin, K. Levchenko, A. C. Snoeren, and D. McCoy, "Tracking ransomware end-to-end," in *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018, pp. 618–631.

[5] K. Jadhav, "Investigating machine learning approaches for bitcoin ransomware payment detection systems."

[6] S. Kok, A. Azween, and N. Jhanjhi, "Evaluation metric for crypto-ransomware detection using machine learning," *Journal of Information Security and Applications*, vol. 55, p. 102646, 2020.

[7] A. Yazdinejad, H. HaddadPajouh, A. Dehghantanha, R. M. Parizi, G. Srivastava, and M.-Y. Chen, "Cryptocurrency malware hunting: A deep recurrent neural network approach," *Applied Soft Computing*, vol. 96, p. 106630, 2020.

[8] M. Paquet-Clouston, B. Haslhofer, and B. Dupont, "Ransomware payments in the bitcoin ecosystem," *Journal of Cybersecurity*, vol. 5, no. 1, p. tyz003, 2019.

[9] A. Liska and T. Gallo, *Ransomware: Defending against digital extortion*. " O'Reilly Media, Inc.", 2016.

[10] M. M. Ahmadian, H. R. Shahriari, and S. M. Ghaffarian, "Connection-monitor & connection-breaker: A novel approach for prevention and detection of high survivable ransomwares," in *2015 12th International Iranian Society of Cryptology Conference on Information Security and Cryptology (ISCISC)*. IEEE, 2015, pp. 79–84.

[11] F. Reid and M. Harrigan, "An analysis of anonymity in the bitcoin system," in *Security and privacy in social networks*. Springer, 2013, pp. 197–223.

[12] A. Azmoodeh, A. Dehghantanha, M. Conti, and K.-K. R. Choo, "Detecting crypto-ransomware in iot networks based on energy consumption footprint," *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 4, pp. 1141–1152, 2018.

[13] C. G. Akcora, Y. Li, Y. R. Gel, and M. Kantarcioglu, "Bitcoinheist: Topological data analysis for ransomware prediction on the bitcoin blockchain," in *Proceedings of the twenty-ninth international joint conference on artificial intelligence*, 2020.

[14] S. Meiklejohn, M. Pomarole, G. Jordan, K. Levchenko, D. McCoy, G. M. Voelker, and S. Savage, "A fistful of bitcoins: characterizing payments among men with no names," in *Proceedings of the 2013 conference on Internet measurement conference*, 2013, pp. 127–140.

[15] S. Kok, A. Abdullah, N. Jhanjhi, and M. Supramaniam, "Prevention of crypto-ransomware using a pre-encryption detection algorithm," *Computers*, vol. 8, no. 4, p. 79, 2019.

[16] K. Cabaj, M. Gregorczyk, and W. Mazurczyk, "Software-defined networking-based crypto ransomware detection using http traffic characteristics," *Computers & Electrical Engineering*, vol. 66, pp. 353–368, 2018.