Anomaly Detection with Graph Convolutional Networks for Insider Threat and Fraud Detection

Jianguo Jiang[†‡], Jiuming Chen[†‡§], Tianbo Gu[§], Kim-Kwang Raymond Choo[¶], Chao Liu[†‡], Min Yu[†‡*], Weiqing Huang[†‡], and Prasant Mohapatra[§]
[†] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[‡] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[§] Department of Computer Science, University of California, Davis, CA, USA
[¶] Department of Information Systems and Cyber Security, University of Texas at San Antonio, San Antonio, TX 78249, USA
Email: {jiangjianguo,chenjiuming,liuchao, yumin, huangweiqing}@iie.ac.cn, tbgu,pmohapatra@ucdavis.edu, raymond.choo@fulbrightmail.org
[*] Corresponding Author

***Abstract*—Anomaly detection generally involves the extraction of features from entities' or users' properties, and the design of anomaly detection models using machine learning or deep learning algorithms. However, only considering entities' property information could lead to high false positives. We posit the importance of also considering connections or relationships between entities in the detecting of anomalous behaviors and associated threat groups. Therefore, in this paper, we design a GCN (graph convolutional networks) based anomaly detection model to detect anomalous behaviors of users and malicious threat groups. The GCN model could characterize entities' properties and structural information between them into graphs. This allows the GCN based anomaly detection model to detect both anomalous behaviors of individuals and associated anomalous groups. We then evaluate the proposed model using a real-world insider threat data set. The results show that the proposed model outperforms several state-of-art baseline methods (i.e., random forest, logistic regression, SVM, and CNN). Moreover, the proposed model can also be applied to other anomaly detection applications.**

***Index Terms*—Anomaly Detection, Graph Convolutional Networks, Insider Threat Detection, Fraud Detection**

I. Introduction

Anomaly detection typically involves the locating of patterns that do not conform to (or deviate from) expected behaviors [1], and has applications in domains such as intrusion detection, financial crime investigation (e.g., money laundering and fraud detection), and data mining. In this paper, we focus on anomaly detection for insider threat detection and fraud detection. This is partly because of the increasing number of high profile data leakage [2] and fraud incidents [3], and the risk of organizations and individuals been subject to cyber security fatigue (in the sense that the community becomes 'numb' to such breaches and eventually accepting these incidents to be the norm).

In order to mitigate insider threats and frauds, many approaches focus on detecting anomalous user behaviors. Techniques proposed in the literature include those designed to establish baseline behavior profiles to classify normal users from anomalous users [4], [5], using graphs [6], game theory [7], machine learning [8], deep learning (e.g., CNN, RNN, and LSTM), etc. There are several limitations in these existing approaches, and we will briefly discuss two of the challenges which we attempt to address in this paper.

- Lacking enough labeled samples; The number of malicious insiders and activities within a given organization is usually minimal. Moreover, most of the collected data is unlabeled, which makes it challenging to train a robust anomaly detection model with high accuracy.
- Ignoring structural information between entities; The users' relationship can present essential information useful to detect users or groups. However, existing machine learning or deep learning methods generally use users' individual properties, rather than including the relationship information between users. In other words, essential information that can be useful for detection may be lost.

To overcome the above problems, we design a GCN (graph convolutional networks) [9] based anomaly detection scheme for insider threat detection and fraud detection. GCN is the extension version of CNN (convolutional neural networks) in the graph domain, which is designed to learn the end-to-end models for nodes feature information and structural information. The structural information or connection information between entities contains essential information for training a robust anomaly detection model. However, previous models (i.e., CNN) could only learn the features for entities' individual property information. Hence, GCN models have been proposed to deal with the learning tasks for some non-euclidean data (node classification, edge prediction, etc.). Besides, the GCN model does not need to label all the nodes in the graph for training, which could overcome the limitation for only collecting partly labeled samples.

In this paper, we characterize users' behaviors and their connection relationships into a graph and then train the robust anomaly detection model for insider threat and fraud detection using the GCN algorithm. However, there are many isolated nodes when converting the network into a graph as many users seldom communicate with others. Using users' direct connection to build the adjacency matrix for the GCN model does not allow us to characterize the structural information of the network. Therefore, we design a weighted function, which leverages users' connection relationships and the similarity of their behaviors to quantify the structural information of the network. The result shows that the updated adjacency matrix

input could primarily improve the detection accuracy for insider threat detection. Therefore, we consider our contributions in this paper to be two-fold, as described below.

- As we know, this is the first work to use the GCN model for anomaly detection applications (and in our context, insider threat detection and fraud detection), which achieves improved accuracy for detecting anomalous users and malicious groups.
- We also design a weighted function to quantify the structural information between users, which allows us to overcome challenges due to the isolated nodes in the network.
- We provide a general framework of an anomaly detection system based on the GCN algorithm, which could be easily implemented and extended to other anomaly detection issues with high detection accuracy.

The rest of the paper is organized as follows. We will briefly summarize the related literature in the next section. In Section III, we will present the proposed model, followed by its evaluation setup and findings in Section IV. Finally, we present the discussion and conclusion in Section V.

II. Related Works

In this section, we will briefly summarize the existing representative research work about anomaly detection algorithms and anomaly detection models, designed for insider threat detection and fraud detection.

A. Anomaly Detection

Anomalies or outliers in data always represent significant, and often critical, actionable information in a wide variety of application domains, which makes anomaly detection crucial for many areas such as network security, fraud detection, etc. Anomalies could be classified into two main categories as follows:

- Point anomalies, which aims to detect an individual anomalous data sample respect to other data samples.
- Contextual or collective anomalies, which aims to detect some related or conditional set of anomalous data samples respect to the entire data set.

Machine learning and deep learning methods have been utilized to automatically extract anomaly detection rules to detect point anomalies, such as CNN, RNN and LSTM [10], etc. These models generally automatically or manually build a feature set from the dataset, and then train the learning based models to detect anomalies [11]. Such models could achieve high accuracy in detecting point anomalies on structural data such as images and videos. However, many real-world scenarios could not directly be characterized and quantified by structural data, such as social networks and knowledge graphs.

B. Anomaly Detection Models on Insider Threat and Fraud

Fraud detection [12] refers to the detection of criminal activities occurring in public and private sector organizations such as banks, credit card companies, insurance agencies, and government agencies. Insider threat [13] refers to the detection of users within organizations undertaking malicious activities, which could take severely impact on the organizations (e.g. information theft, sabotage, fraud, and data exfiltration). A number of models and systems have been proposed in the literature, designed to characterize and detect insider threats. These detection models generally build users' profile or nodes' profile, based on their activities such login/logon, file access, email, instant message and web-browsing, where machine learning [14] and deep learning models [15] could be used to train the classification models for anomaly detection. Existing insider threat detection and fraud detection models focus on detecting point anomalous users or point anomalous nodes, and there are few schemes which could predict or detect some complex insider threat scenario or fraud scenarios with collective users or collective nodes.

C. Application of GCN Model

Some traditional deep learning algorithms such as CNN could have good results when processing euclidean data, which is regular spatial structure, such as image and voice, etc. However, most of the graphs in the real world could not be represented by euclidean data, such as social network, knowledge, graph, market graph, these data are spatial structure without rules. Therefore, GCN (graph convolutional network) model is proposed to overcome the limitation of the previous models. With GCN models, the researches could represent data from the spatial domain to graph domain, which could overcome the limitation that CNN meets when processing non-euclidean data. For example, the GCN model could outperform other state-of-art machine learning and deep learning models on many domains such as opinion inference [16], social networks [9]. When representing users' behaviors into graphs, the learning based models could include not only individual properties but also the collective information between nodes in the graph. Therefore, we propose the first GCN based model on anomaly detection for insider threat detection and fraud detection. In the next section, we will describe the detail of the anomaly detection framework using graph convolutional networks.

III. Framework

In this section, we provide the details of the anomaly detection framework for insider threat detection and fraud detection. We firstly give a short background description of GCN. Then we describe the main modules of the proposed anomaly detection model.

A. Graph Convolutional Networks

As described before, the GCN model is an extended version of CNN, which could overcome CNN's limitation when processing some domain that could not directly be represented into euclidean or structural data. The graphs of the GCN model have two essential properties to represent the domain information. First, each node of the graphs has its features, such as IP, port, etc. Then each node contains structural information such as communication patterns between nodes.

After transforming the spatial domain into graph domain, the GCN model could learn both the feature information and structural information of some graphs such as social network, communication network.

1) Definition of GCN: The definition of the graph used in GCN is as formula 1, V represents the nodes set in the graph, and E represents edges set. The input of the GCN model is a graph which contains many nodes with features, and connections with nodes as edges of the graph. For the detection models using GCN, the goal is to learn a function of features on the graph to classify the input nodes.

$$G = (V,\ E) \tag{1}$$

- Input for GCN. As described before, the input of the GCN model could be divided into two parts. One is the feature set for nodes, which could represent as a $N * D$ matrix F (N is the number of nodes on the graph, D is the number of features each node). The other part of the input graph is an adjacency matrix A ($N * N$) represents the structural information for the graph.
- Output of the GCN model O represents $N * M$ matrix, where M is the number of output classification categories per node, N is the number of nodes in the graph.

Similar to the CNN model, the input matrix needs to be transformed by multiple hidden convolutional layers before output. The process of the hidden layer showed as formula 2, and we use H(0), and H(l) represents the input and output of the GCN model as formula 2. In each hidden layer, the input matrix is computed as the above formula 3, h_i^{l+1} and h_i^l represent the feature set for node i in layer l+1 and layer l, N_i represents set of node i's neighbors(including itself), $W_{R_j}^l$ represents weight parameter to transform information from node j, $\frac{1}{c_{ij}}$ represents normalization factor, such as degree of node i. Moreover, $\sigma()$ represents a non-linear activation function like the ReLU. The multiple hidden layers of the GCN model could process the input matrix as the following three aspects.

- Send; means that each node would send their feature information after transform to their neighbor node.
- Receive; means that each node would receive all the feature information from their neighbors.
- Transform; collects the above feature information and structural information and then transform them using a non-linear function, to improve the expressive ability of models.

$$H(l+1) = f(H(l), A) \text{ } \textit{(H(0)=X, H(l)=Z)} \tag{2}$$

$$h_i^{l+1} = \sigma(\sum_{j \in N_i} \frac{1}{c_{ij}} h_j^l W_{R_j}^l) \tag{3}$$

After converting the user nodes and their connections into graphs, the GCN framework could make the classification model more accurate by combining individual information with structural information. In the next section, we will detail the main modules of the GCN based anomaly detection model for insider threat detection and fraud detection.

B. Anomaly Detection Framework

As described before, traditional insider threat and fraud detection models always extract features singly from users' property information, ignoring users' communication such as email communication, role-based relationships, etc., which is essential for associated threat group detection. In order to combine users' properties and connection information between users in insider threat and fraud scenarios, we convert the communication network and users' behaviors into a graph. Then we design a GCN based anomaly detection model for insider threat detection and fraud detection using the graph as input. The framework is shown in Figure 1.

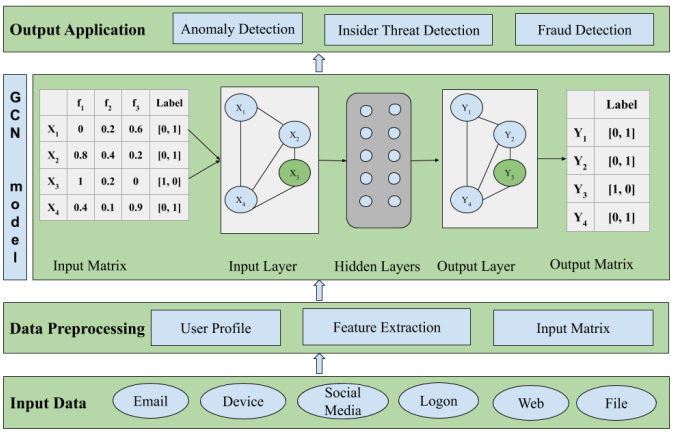


Fig. 1. Graph Convolutional Networks based Anomaly Detection Framework

In shortly, the system firstly initials a graph characterizing users or entities within the network. The nodes of the graph represent users' or other entities, and the edges between nodes represent the structural information of these users or entities. In order to fulfill nodes' properties and structural information of the graph, the system then extracts behavioral features of the users and entities to get nodes' properties information. For the structural information of these nodes, instead of using the direct communication relationship, we design a weighted function combining the connection relationship and similarity of users' behaviors as the adjacency matrix to improve the detection accuracy. After building the input matrix, the GCN model could generate the classification of each node about whether they are normal or anomalous. In this paper, we use insider threat detection as a domain application example to show the detail to build a GCN model based anomaly detection model. The progress of building an anomaly detection model for many other domains such as fraud detection is similar to the example, only need to take some change in the feature extraction module. Otherwise, the anomaly detection framework could be easily transformed into other application domains.

As shown in the Figure 1, the GCN based anomaly detection framework for insider threat detection could be divided into three modules.

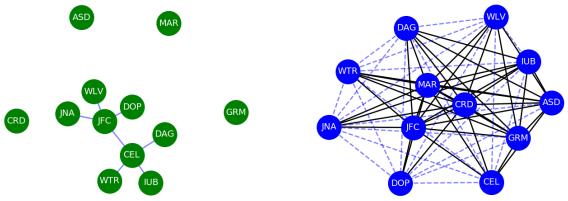
1) *Feature extraction module F*, this module builds a $N * D$ feature matrix for the graph, N represents the number of users within the network, D represents the number of features extracted by this module for each user. In the insider threat field, this module extracts users' features and indicators for insider threat scenarios. We collected and designed 31 features to build a behavior profile of users for each user within the network; the list of features extracted by this module is shown below. In this module, we not only extract users behavioral features from users activities in email, web, file, logon, device, etc., and also extract content-based features of users using natural language processing [17]. The combination of behavioral features and content based features provide a solid basis for characterizing nodes' property information.

- Logon/Logoff features, Daily Logon/Logoff Times, Off-Work Hours Logon/Logoff Times, numbers of PC for Logon/Logoff.
- Device features, Daily number of device connection, off-work hours device connection, PC for device connection.
- File features, Daily number of different files, total files, files in off-works hours, exe files, PC for files.
- Email features, Daily number of sent emails, out organization sent emails, in-organization sent emails, average email size, receivers, topic-related emails, sentiment-related emails.
- Web features, Daily number of web pages browsed, Wikileaks related web pages, sentiment-related web pages, topic-related web pages, key-logger related web pages.

2) *Graph matrix module*, this module builds a $N * N$ adjacency matrix to characterize users' connections between the users. Traditionally, the adjacency matrix for the GCN model is fulfilled as a 0-1 matrix, where $A(i, j) = 0$ if there is no connection from node i to node j, else $A(i, j) = 1$. Different from the areas in social networks and knowledge graphs, there are many isolated users within the network graph in insider threat and fraud detection areas. Using the direct connections between users, such as email communications may ignore essential structural information between users. Therefore, in this paper, we design a comprehensive function to generate the adjacency matrix, which leverages the direct connections between users and the relevance of their behaviors. The formula to quantify the relationship between users as formula 4. We use a parameter ω (0 1) to balance the connections and similarities between users. In detail, C_{ij} (0, 1) represents whether there is a direct connection between node i and node j, and we use *cos* formula to compute the similarity of two users.

$$A(i, j) = \omega * cos(F_i,\ F_j) + (1 - \omega) * C_{ij} \tag{4}$$

In Figure 2(a) and Figure 2(b), we display the difference of building adjacency matrix using direct connections between users and using the weighted function as formula 4. The graphs show parts of the connections network in our evaluation data set. For Figure 2(a), we connect the two users when there exist email communications between them. However,



(a) Direct Connections (b) Weighted Function

Fig. 2. Graph of the Inside Network Characterizing by Direct Connections and Weighted Function for Building Adjacency Matrix

there are many isolated nodes such as users with ID "ASD," "MAR" and "CRM," which may decrease the learning ability to train a robust anomaly detection model for the GCN model. Therefore, we use the weighted function computed by formula 4 to connect the users of the network, shown in Figure 2(b). The solid edge between two nodes indicates that the adjacency matrix between these two nodes is $A(i, j) > 0.5$. The dotted edge between two nodes represents that the adjacency matrix is $A(i, j) < 0.5$. Based on our experiment, the detection accuracy could be largely improved after updating the adjacency matrix for the input graph.

3) *GCN Design module*, after transferring the inside networks' users and their behaviors into graph matrix, we design a 2-layers GCN model to train the anomaly classification model. The information flow of the designed GCN model is shown in Figure 3, which could also be characterized by formula 5. For the information flow, W^0 represents input to the hidden layer weight matrix for a hidden layer with H feature maps. W^1 represents a hidden-to-output weight matrix. To compute the classification output of each node, we use a softmax activation function which shows in formula 6. To train the parameters of the GCN models W_0 and W_1, we perform batch gradient descent, and use the cross-entropy error showed as formula 7 over all labeled examples to evaluate during training step. In the next section, we will show the experiment to evaluate the effectiveness using the designed GCN model for detecting anomalous insider behaviors.

$$Z = f(X, A) = softmax(A\ ReLU(AXW^0)\ W^1) \tag{5}$$

$$softmax(x_i) = \frac{1}{\sum_i exp(x_i)} exp(x_i) \tag{6}$$

$$e = -\sum_{l \in y_L} \sum_{f=1}^{F} Y_{lf} ln Z_{lf} \tag{7}$$

IV. Evaluation

In this paper, we use a public insider threat detection data set named CMU CERT v4.2 [18] to evaluate the performance of the designed GCN model. In this section, we first introduce the background information of the data set and the preprocessing methods to build the input matrix. Then we give the detail

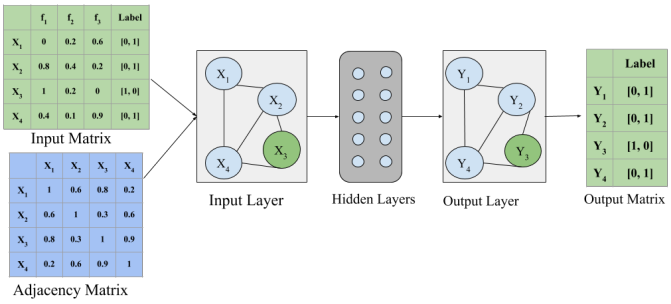


Fig. 3. Information Flow of Graph Convolutional Networks Framework

parameters of the designed GCN model and the detection accuracy using the developed GCN model. Finally, we will discuss the classification result and compare it with some state-of-art methods.

A. Data Set and Data Preprocessing

The data set provided by CMU CERT consists of 1000 users in a simulated network and their activities from 01/02/2010 to 05/16/2011. These activities are mainly logs on web browsing, email communication, file operation, device usage, and logon. The CMU team collected these 1000 users' activities as normal activities, and they also developed three insider threat scenarios which consist of the labeled malicious insiders and their anomalous activities. Therefore, we use the tagged user and labeled anomalous activities as input data to train the anomaly detection model.

To build the input matrix for training GCN model, we convert the 1000-user network into a graph where the nodes represent the users and edges represent the relationships between users quantified by formula 4. The following table I shows the number of edges when using the direct email connection and weighted function as formula 4 to quantify the adjacency matrix of the network. The original number of edges is only 3556, which contains many isolated nodes. Moreover, the updated adjacency matrix consists of 1000,000 non-zero values which could characterize much comprehensive information for training a robust model. Then we process the logs of users and extract daily features of them as described in section III, and builds an input feature matrix as $F(User, feature) = \{M, S\}$, where M denotes the set of users in the organization, and S represents the extracted features. In this experiment, the size of M equals to 1000, and the size of S equals 31. Then we use the ground truth provided by the data set about the labeled users and their anomalous activities to build the output matrix as $O(User, Label) = \{M, C\}$ where M denotes the user set in the network, and C represents the classification label for each node. The following table I displays the summary of the data for evaluation, such as the number of nodes and edges, train set, test set, etc.

B. GCN model

To implement the GCN model, we use TensorFlow as the deep learning platform to implement the GCN model. As described before, the framework of the designed GCN model is shown as Formula 1. The parameter of the designed GCN model is shown in table II. For training a robust GCN model and avoiding over-fit, we add a drop-out function before the output layer where the dropout rate is 0.5. As described before, the designed GCN model consists of two hidden graph convolutional layers. The learning rate is initialized as 0.01. Moreover, we set the epoch for training as 50 and 16 units in hidden layer 1. In the next section, we will show the accuracy result using the GCN model for detecting anomalous activities of insiders, and compare the results with other state-of-art methods.

TABLE I
SUMMARY OF THE DATA SET

Category	*Property*	*Value*
Dataset	Number of Nodes	1000
	Number of Features	31
	Number of Edges Using Direct Connections	3556
	Number of Edges Using Weighted Function	1000*1000
Input Matrix	Feature Matrix	1000*31
	Adjacency Matrix	1000*1000
	Label Matrix	1000*2
Train Set	Number of Normal Nodes	160
	Number of Anomalous Nodes	40
Test Set	Number of Normal Nodes	170
	Number of Anomalous Nodes	30

TABLE II
PARAMETER FOR TRAINING THE GCN MODEL

Parameter	*Value*
Initial Learning Rate	0.01
Weight parameter in formula 4 ω	0.8
Number of epochs to train	50
Number of units in hidden layer	16
Dropout Rate	0.5

C. Result Discussion and Comparison

In this paper, we compare the designed GCN model with four widely used machine learning and deep learning algorithms such as Random Forest, Logistic Regression, SVM, and CNN. We use SK-learn tools to implement these four algorithms. The Figure 4 shows the detection accuracy for anomaly activities using the GCN model and four comparison algorithms. To evaluate the effectiveness of using the weighted function to quantify the network's structural information, we also test the result of using the direct connection between users and using the weighted function as formula 4.

Figure 4 shows the detection performance of the revised GCN in comparison with four other machine learning and deep learning algorithms. We observe that the revised GCN based model outperforms the other four competing algorithms which ignore the structural information between users. For the four competing models, the CNN model achieves the highest detection accuracy (i.e., 93%), and the two-deep learning models (CNN and GCN) outperform the other three

machine learning based methods whose detection accuracy approximately 85%. We also compared the detection accuracy using a direct connection or the weighted function to build the adjacency matrix for the GCN model, the detection accuracy could be significantly improved (and in the context of our experiment, from 85.5% to 94.5%). Moreover, the recall rate of the revised GCN based algorithm is about 83.3%, while the other algorithms only achieve the recall rate at nearly 70%, which shows that GCN based model could largely reduce the number of undetected anomalous behaviors. Therefore, when converting the network and users' behaviors into graph structural data, the classification accuracy and recall could be primarily improved instead of only using single points' feature information. The result shows the effectiveness of using the graph to characterize the property information and structural information for anomaly detection.

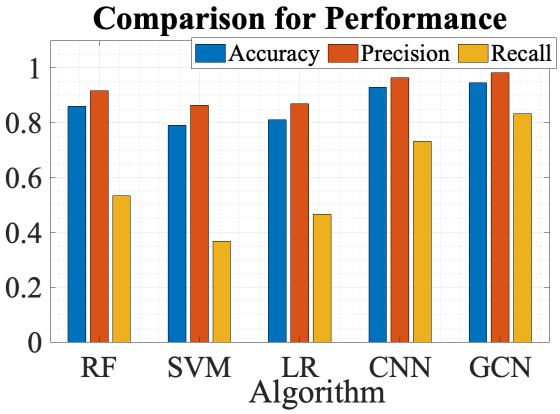


Fig. 4. Detection Accuracy on Insider Threat Using GCN model

From the result, we could see that characterizing the network and users' behaviors into a graph, could improve the anomaly detection accuracy. Using direct communications between nodes for the adjacency matrix could get excellent performance in many other areas such as social networks, and traffic prediction, etc. However, for insider threat and fraud detection issues, merely using the connection relationships between users to quantify the network's structural information, may not make the best of the GCN model. Therefore, we update the process to generate the adjacency matrix as formula 4, and the experiment showed the effectiveness of the updated adjacency matrix. Moreover, the designed GCN model could also take effect when detecting malicious groups with correlated anomalous events and groups.

V. Conclusion

In this paper, we designed a GCN based anomaly detection system for insider threat detection and fraud detection. To characterize and quantify the structural information between users in insider threat detection and fraud detection problems, we designed a weighted function which leverages the direct connection between users and their similarity in activities. The designed GCN model outperforms four other state-of-art machine learning and deep learning algorithms, in detecting anomalous activities of malicious insiders or frauds and correlated threat groups. In the future, we will implement the GCN based models in real-world applications to further evaluate its utility and scalability.

Acknowledgement

The support provided by China Scholarship Council (CSC) for Jiuming Chen's stay at UC Davis is acknowledged. We also thank Chenggang Jia for his contribution in the processing of the raw data from CMU dataset.

References

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[2] H. Schulze, "Insider threat report: 2018." https://www.ca.com/content/dam/ca/us/files/ebook/insider-threat-report.pdf.

[3] Experian, "2019 global identity and fraud report." https://www.experian.com/decision-analytics/global-fraud-report.html.

[4] P. A. Legg, O. Buckley, M. Goldsmith, and S. Creese, "Automated insider threat detection system using user and role-based profile assessment," *IEEE Systems Journal*, vol. 11, no. 2, pp. 503–512, 2017.

[5] G. Gavai, K. Sricharan, D. Gunning, J. Hanley, M. Singhal, and R. Rolleston, "Supervised and unsupervised methods to detect insider threat from enterprise social and online activity data.," *JoWUA*, vol. 6, no. 4, pp. 47–63, 2015.

[6] F. Toffalini, I. Homoliak, A. Harilal, A. Binder, and M. Ochoa, "Detection of masqueraders based on graph partitioning of file system access events," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 217–227, IEEE, 2018.

[7] X. Feng, Z. Zheng, D. Cansever, A. Swami, and P. Mohapatra, "Stealthy attacks with insider information: A game theoretic model with asymmetric feedback," in *Military Communications Conference, MILCOM 2016-2016 IEEE*, pp. 277–282, IEEE, 2016.

[8] B. A. Alahmadi, P. A. Legg, and J. R. Nurse, "Using internet activity profiling for insider-threat detection.," in *ICEIS (2)*, pp. 709–720, 2015.

[9] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[10] M. Du, F. Li, G. Zheng, and V. Srikumar, "Deeplog: Anomaly detection and diagnosis from system logs through deep learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1285–1298, ACM, 2017.

[11] T. Gu and P. Mohapatra, "Bf-iot: Securing the iot networks via fingerprinting-based device authentication," in *2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pp. 254–262, Oct 2018.

[12] A. Abdallah, M. A. Maarof, and A. Zainal, "Fraud detection system: A survey," *Journal of Network and Computer Applications*, vol. 68, pp. 90–113, 2016.

[13] F. Greitzer, J. Purl, Y. M. Leong, and D. S. Becker, "Sofit: Sociotechnical and organizational factors for insider threat," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 197–206, IEEE, 2018.

[14] P. Dutta, G. Ryan, A. Zieba, and S. Stolfo, "Simulated user bots: Real time testing of insider threat detection systems," in *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 228–236, IEEE, 2018.

[15] A. Tuor, S. Kaplan, B. Hutchinson, N. Nichols, and S. Robinson, "Deep learning for unsupervised insider threat detection in structured cybersecurity data streams," in *Workshops at the Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[16] X. Zhao, F. Chen, and J.-H. Cho, "Deep learning for predicting dynamic uncertain opinions in network data," in *2018 IEEE International Conference on Big Data (Big Data)*, pp. 1150–1155, IEEE, 2018.

[17] J. Jiang, J. Chen, K.-K. R. Choo, K. Liu, C. Liu, M. Yu, and P. Mohapatra, "Prediction and detection of malicious insiders' motivation based on sentiment profile on webpages and emails," in *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*, pp. 1–6, IEEE, 2018.

[18] J. Glasser and B. Lindauer, "Bridging the gap: A pragmatic approach to generating insider threat data," in *Security and Privacy Workshops (SPW), 2013 IEEE*, pp. 98–104, IEEE, 2013.