Learning Correlation Graph and Anomalous Employee Behavior for Insider Threat Detection

Pratibha, Junshan Wang, Saurabh Aggarwal*, Feng Ji, and Wee Peng Tay
School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore 639798
Email: (pratibha001, wangjs2, saggarwal, jifeng, wptay)@ntu.edu.sg

***Abstract*—Insider attacks can result in significant costs to an organization. There is an urgent need for an automatic insider threat detector with good accuracy and low false alarms. In this work, we propose a graph based insider threat detector to identify potential insider attackers based on identifying not only self-anomalous behaviors of an employee but also anomalies relative to other employees with similar job roles. A machine learning approach is developed to first infer the correlation graph among the organization's employees. Then, a graph signal processing method is designed to identify the potential insiders with detection and false positive rates better than performing detection independently on each employee. Our approach demonstrates that the correlated behaviors of an organization's employees should be exploited for a better detection of suspicious behaviors.**

I. Introduction

Insider threat detection is a challenging and urgent problem for many organizations. An insider is a malicious employee who abuses his access rights to benefit himself or accidentally leaks some sensitive corporate data to outsiders, which harms the organization. As an insider is aware of the company policies and physical facilities, his anomalous activities may potentially cause greater damages, and also more difficult to be detected compared with external cyber attacks. In this paper, we propose a graph based unsupervised anomaly detection algorithm for insider threat detection. We detect changes in the comparative behavior among the peers, in addition to the activities of individuals, to improve detection and reduce false positives.

In recent literature, some works have focused on defining the insider threat problem and proposing frameworks for insider threat detection [1], [2]. In [3], the authors created activity trees to represent the daily activities of an employee, considering his job role and the range of malicious activities that could indicate suspicious behavior. For each newly observed activity branch, its similarity was computed with the activity tree. It was classified as insider behavior on failing a given acceptance criteria. Similarly, in [2], the authors considered tree structures to keep track of user's day-to-day activities. They also defined the normal behavior based on the past behavior of the employee and his peers who have similar role in the organization. Anomalous behavior was detected based on his deviation from the predefined normal behavior in terms of different attributes, such as login, USB usage, and emails sent. In [4], the authors proposed a scalable approach which utilized role-based analysis to detect anomalous behavior of an employee. The behavior of the employee was compared with an activity pattern baseline which was maintained by monitoring an employee's behavior based on his job role, different job tasks, and the frequency ranges of tasks. Researchers in [5] proposed fusion algorithms to combine anomaly scores corresponding to different categories of data to detect malicious insiders, which may not be anomalous in any single category. The employees with same job roles were expected to exhibit similar clustering behavior in each category, over time.

Recently, graph signal processing techniques have been used in inference tasks by utilizing the contextual information or inherent relationships among different entities in the data [6], [7]. A system was developed for proactive detection of insider threats using computer usage data [8]. Graph analysis based methods and multiple other anomaly detection algorithms were used to overcome the challenges of weak signals corresponding to continuously evolving scenarios of malicious insiders. In another work, a graph based approach was used to detect anomalous structural patterns in malicious insider activities which appeared similar to the normal user activities [9]. A proactive insider threat detection method was proposed which combined two aspects: structural anomaly detection (SAD), and psychological profiling (PP) based anomaly detection [10]. SAD used graph analysis and machine learning techniques to sense structural anomalies in social networks data. Under PP, dynamic psychological profiles were built to represent the behavioral patterns of individuals. PP provided the psychological semantics for SAD to focus on the relevant data, and helped reduce the false alarm rate. In [11], an integrated system was proposed for dynamic graphs by including three types of anomalies: node level, community level, and evolutionary path level. A set of graph features was learned to detect node-level outliers. On the other hand, average node-level graph features were used to detect anomaly in behavior of communities, which were evolving over time.

In our current work, we present a graph based approach, which considers the correlation in activities of the employees to improve insider detection and reduce false positives. Unlike networked data, in many application scenarios, a graph may

The authors are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. (*now with Boeing Research and Technology, India)

not be readily available to represent the data. For insider threat detection problem, we propose an algorithm to learn a correlation graph among the employees. A graph Laplacian matrix [6] is used to obtain the comparative behavior and activity pattern among the peers. Furthermore, for each employee, a time series prediction model is used to learn and represent the normal individual and group activity pattern in a dynamic way. By comparing the observed activity with the expected activity based on past behavior pattern, we can detect anomalous behaviors such as: (1) an employee acting differently from his past behavior, whilst his peers in the same group are not exhibiting similar changes; (2) an employee acting in the same way as his past behavior, but his peers in the same group are moving to a different activity pattern. Our proposed graph based anomaly detection algorithm can be easily applied to various situations that utilize different signals to represent the employee's behaviors, e.g., daily activities of the employees saved in the organizational logs or any other behavioral indicators.

The rest of the paper is organized as follows. In Section II, we propose algorithms for computing the similarity values to infer correlation graph among the employees. Section III presents an integrated anomaly detection system using the correlation graph for insider threat detection. Section IV discusses the dataset used for our experiments and the obtained results. We conclude with Section V.

II. Employee Activity Based Correlation Graph Inference

We intend to learn the correlation graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of vertices, which represents the set of N employees, and $\mathcal{E}$ is a set of edges denoting the similarity in their behavior. The adjacency matrix $\mathbf{W} \in \mathbb{R}^{N \times N}$ contains weight value $0 \leq \mathbf{W}_{i,j} \leq 1$ for the edge connecting the i^{th} and the j^{th} employees. The degree matrix is denoted by $\mathbf{D} = \text{diag}([\sum_{j=1}^{N} \mathbf{W}_{i,j}]_{i=1}^{N})$. As emphasized in [2], [8], anomalous behavior detection can be built on the data collected by monitoring the activity patterns, and social and psychological indicators, over a long period. In the current work, we utilize the historical daily activity of the employees saved in the organizational logs, for example, email, file activities, logon, web browsing sessions, etc. We further derive various attributes from each activity log, for example, frequency of emails sent, frequency of emails read, and so forth. We use these indicators at the employee nodes to capture the similarity based contextual relationships for our application.

To extract the desired attributes, we parse the organizational logs for each employee. We obtain a multivariate graph signal, which consists of the slot-wise time-series corresponding to each of the attributes. We use $\mathcal{A}$ to denoted the set of attributes, then for any attribute $a \in \mathcal{A}$ and any employee u_i, we use $[X_i^a(t)]_{t=1}^T$ to denote the time series of the attribute values, computed using the activities occurring between the slots $t-1$ and t, upto $t = T$. Such kind of vector space representation has been used by researchers to study social media usage patterns and insider threats in [4], [8].

For each attribute $a \in \mathcal{A}$, the normalized similarity measure between two employees, u_i and u_j, is computed as follows:

$$S_{i,j}^a = \begin{cases} 1, & \text{if for any } t, \quad X_i^a(t) = X_j^a(t) = 0; \\ \dfrac{\max_\tau R_{X_i^a, X_j^a}(\tau)}{\max\left(R_{X_i^a, X_i^a}(0), R_{X_j^a, X_j^a}(0)\right)}, & \text{otherwise,} \end{cases} \tag{1}$$

where $R_{X_i^a, X_j^a}(\tau)$ is the cross correlation between the two time series $[X_i^a(t)]_{t=1}^T$ and $[X_j^a(t)]_{t=1}^T$ defined in [12]. Compared with other similarity measures, such as Jaccard Index, Min/Max similarity, which consider direct intersection of the two series [13], we adopt the correlation method as it can detect the time-shifted similar activity patterns in the employee activities.

In order to find the overall similarity score between the two employees, the similarity values obtained for different attributes have to be aggregated. We need to define the concept of similarity which is relevant to our scenario. The correlation measure for some attributes may be emphasized, while understating the dissimilarity in other attributes, to quantify the similarity in the activities of any two employees.

A. Correlation Similarity Measure (CSM)

We propose the correlation similarity measure (CSM) given in Algorithm 1. Under CSM, we compute the similarity measure among the employees of an organization using the correlation among their attribute time-series. To define the pairwise similarity measure for a set of employees $\mathcal{U}_f$, we choose a set of reference employees $\mathcal{U}_r$. $\mathcal{U}_r$ consists of K mutually exclusive subsets of employees G_k, such that $\mathcal{U}_r = \bigcup_{k=1}^K G_k$ and $G_{k_1} \bigcap G_{k_2} = \emptyset$, for any $k_1, k_2 \in \{1, 2, \cdots, K\}$. For each attribute $a \in \mathcal{A}$, we compute the similarity score between each employee $u_i \in \mathcal{U}_f$, and each employee $u_j \in \mathcal{U}_r$, which is denoted by $S_{i,j}^a$. For any employee $u_i \in \mathcal{U}_f$, its mean similarity score with the subset of reference employees G_k, for any attribute $a \in \mathcal{A}$ and any $k \in \{1, 2, ..K\}$, is given by

$$S_{i,G_k}^a = \frac{1}{|G_k|} \sum_{u_j \in G_k} S_{i,j}^a, \tag{2}$$

where $|\mathcal{X}|$ denotes the cardinality of set $\mathcal{X}$. The final feature vector for any employee $u_i \in \mathcal{U}_f$ is given by $S_G^i = \left[\left[S_{i,G_k}^a\right]_{a \in \mathcal{A}}\right]_{k=1}^K$, which is a $K * |\mathcal{A}|$-dimensional feature vector. Lastly, the similarity between any two employees u_{i_1} and u_{i_2} is defined based on the pattern of their similarity scores with the subsets of reference employees belonging to different groups G_k, for all attributes. It is based on the assumption that the employees in the same group G_k exhibit high correlation in their similarity patterns, given by S_{i,G_k}^a, with their own groups and with other groups. For example, the reference groups can be defined based on the job roles in the organization, as it is expected that the employees in similar job roles have highly correlated activity patterns. Hence, the CSM score between

any two employees $u_{i_1}, u_{i_2} \in \mathcal{U}_f$ is given by

$$\mathbf{W}_{i_1,i_2} = \cos\left(S_G^{i_1}, S_G^{i_2}\right) = \frac{S_G^{i_1} \cdot S_G^{i_2}}{\|S_G^{i_1}\| \|S_G^{i_2}\|}. \tag{3}$$

Algorithm 1 Correlation Similarity Measure (CSM)

1: obtain slot wise activity record of employees $X_i^a(t)$, for all $u_i \in \mathcal{U}_r \bigcup \mathcal{U}_f$, $a \in \mathcal{A}$, and $t \in [0, T]$.
2: **for** each attribute $a \in \mathcal{A}$ **do**
3: for each employee $u_i \in \mathcal{U}_f$ and each employee $u_j \in \mathcal{U}_r$, find activity similarity score $S_{i,j}^a$ defined in Eq. (1).
4: **end for**
5: **for** $u_i \in \mathcal{U}_f$ **do**
6: **for** each attribute $a \in \mathcal{A}$ and each $k \in \{1, 2, ..K\}$ **do**
7: calculate

$$S_{i,G_k}^a = \frac{1}{|G_k|} \sum_{u_j \in G_k} S_{i,j}^a.$$

8: obtain $S_G^i = \left[\left[S_{i,G_k}^a\right]_{a \in \mathcal{A}}\right]_{k=1}^K$.
9: **end for**
10: **end for**
11: **for** each $u_{i_1}, u_{i_2} \in \mathcal{U}_f$ **do**
12: calculate

$$\mathbf{W}_{i_1,i_2} = \cos\left(S_G^{i_1}, S_G^{i_2}\right) = \frac{S_G^{i_1} \cdot S_G^{i_2}}{\|S_G^{i_1}\| \|S_G^{i_2}\|}.$$

13: **end for**
14: obtain

$$\mathbf{W} = \left[\mathbf{W}_{i_1,i_2}\right]_{u_{i_1}, u_{i_2} \in \mathcal{U}_f},$$

$\mathbf{W}$ acts as adjacency matrix of similarity graph among employees in $\mathcal{U}_f$.

CSM provides similarity measure between two employees by considering all the feature values. However, for better performance, the different features should be taken into account based on the application of the similarity measure. For application specific similarity measures, we further extend CSM to group propensity similarity measure (GPSM) presented in the following section.

B. Group Propensity Similarity Measure (GPSM)

We propose group propensity similarity measure (GPSM) given in Algorithm 2. Under GPSM, the similarity measure among the employees of an organization is computed based on their tendency to belong to different groups. Similar to CSM, we consider K different mutually exclusive groups that can be formed based on the parameters of interest, for example, job role, insider behavior, etc. The set $\mathcal{U}_r = \bigcup_{k=1}^K G_k$ is referred to as the training set of employees. For any employee $u_i \in \mathcal{U}_r$, its mean similarity score with the subset of training employees $\{G_k \setminus u_i\}$, for any attribute $a \in \mathcal{A}$ and any $k \in \{1, 2, ..K\}$, is given by

$$S_{i,G_k}^a = \frac{1}{|\{G_k \setminus u_i\}|} \sum_{u_j \in \{G_k \setminus u_i\}} S_{i,j}^a. \tag{4}$$

The feature vector for any employee $u_i \in \mathcal{U}_r$ is

$$S_G^i = \left[\left[S_{i,G_k}^a\right]_{a \in \mathcal{A}}\right]_{k=1}^K. \tag{5}$$

Additionaly, a group label vector Z^i is associated to each employee $u_i \in \mathcal{U}_r$, which is given as follows:

$$Z^i = \left[Z_k^i\right]_{k=1}^K = \left[\mathbb{I}_{u_i \in G_k}\right]_{k=1}^K, \tag{6}$$

where $\mathbb{I}_x$ is the indicator function, which attains a unit value when x is true, and zero otherwise. We train K classifier functions for classifying the employees based on their mean similarity measure with the reference employees of different groups. For each group G_k, we train the classifier function $\mathcal{C}_{G_k}$ given by

$$\mathcal{C}_{G_k} = \text{classifierTraining}\left(\left[S_G^i\right]_{u_i \in \mathcal{U}_r}, \left[Z_k^i\right]_{u_i \in \mathcal{U}_r}\right). \tag{7}$$

Using the classifier function, we obtain the probabilities of each employee $u_i \in \mathcal{U}_f$ belonging to the different groups G_k, for $k \in \{1, 2, ..K\}$, given by

$$\mathbb{P}_G^i = \left[\Pr\left(\mathcal{C}_{G_k}\left(S_G^i\right) = 1\right)\right]_{k=1}^K. \tag{8}$$

The GPSM score between employees $u_{i_1}, u_{i_2} \in \mathcal{U}_f$ is given by

$$\mathbf{W}_{i_1,i_2} = \cos\left(\mathbb{P}_G^{i_1}, \mathbb{P}_G^{i_2}\right) = \frac{\mathbb{P}_G^{i_1} \cdot \mathbb{P}_G^{i_2}}{\|\mathbb{P}_G^{i_1}\| \|\mathbb{P}_G^{i_2}\|}. \tag{9}$$

Unlike CSM, where the mean similarity feature vectors are used to compute similarity between two employees, in GPSM, these feature vectors are used to learn the employee's behavioral similarity to different reference groups, which is then used to infer the similarity scores between any two employees.

III. Insider Threat Detection Framework

In the following, we present an integrated anomaly detection algorithm that considers both the self and group behavior of the employees. By utilizing the correlation graph among the employees of an organization, we can include the analysis of behavior of an employee in comparison to his peers. With this group behavior analysis, it is expected that the performance of insider threat detection can be improved.

A. Group Comparative Behavior Representation

To include the group behavior information, we analyze the weighted sum of the difference of the employee attribute signals from other employees in the organization. For any attribute $a \in \mathcal{A}$, the group comparative time series for employee u_i is denoted by $[Y_i^a(t)]_{t=1}^T$. The difference signals for all employees, for any attribute $a \in \mathcal{A}$ and any slot $t \in [1, T]$, is computed as follows:

$$[Y_i^a(t)]_{i=1}^N = \mathbf{L}[X_i^a(t)]_{i=1}^N, \tag{10}$$

where the difference operator $\mathbf{L}$ is the graph Laplacian matrix [6] corresponding to the correlation graph among the employees inferred by using Algorithm 2. The Laplacian matrix can be expressed as

$$\mathbf{L} = \mathbf{D} - \mathbf{W}. \tag{11}$$

Algorithm 2 Group Propensity Similarity Measure (GPSM)

obtain slot wise activity record of employees $X_i^a(t)$, for all $u_i \in \mathcal{U}_r \bigcup \mathcal{U}_f$, $a \in \mathcal{A}$, and $t \in [0, T]$.
for each attribute $a \in \mathcal{A}$ **do**
  for each employee $u_i \in \mathcal{U}_f$ and each employee $u_j \in \mathcal{U}_r$, find activity similarity score $S_{i,j}^a$ defined in Eq. (1).
end for
for $u_i \in \mathcal{U}_r \bigcup \mathcal{U}_f$ **do**
  for each attribute $a \in \mathcal{A}$ and each $k \in \{1, 2, ..K\}$ **do**
    calculate

$$S_{i,G_k}^a = \frac{1}{|\{G_k \setminus u_i\}|} \sum_{u_j \in \{G_k \setminus u_i\}} S_{i,j}^a.$$

    obtain $S_G^i = \left[\left[S_{i,G_k}^a \right]_{a \in \mathcal{A}} \right]_{k=1}^K$.
  end for
  if $u_i \in \mathcal{U}_r$ **then**
    obtain $Z^i = \left[Z_k^i \right]_{k=1}^K = [\mathbb{I}_{u_i \in G_k}]_{k=1}^K$.
  end if
end for
for $k \in \{1, 2 \cdots, K\}$ **do**
  train the classifier function for G_k,

$$\mathcal{C}_{G_k} = \text{classifierTraining}\left(\left[S_G^i \right]_{u_i \in \mathcal{U}_r}, \left[Z_k^i \right]_{u_i \in \mathcal{U}_r} \right).$$

end for
for each $u_i \in \mathcal{U}_f$ **do**
  calculate $\mathbb{P}_G^i = \left[\Pr\left(\mathcal{C}_{G_k}\left(S_G^i \right) = 1 \right) \right]_{k=1}^K$.
end for
for each $u_{i_1}, u_{i_2} \in \mathcal{U}_f$ **do**
  calculate

$$\mathbf{W}_{i_1,i_2} = \cos\left(\mathbb{P}_G^{i_1}, \mathbb{P}_G^{i_2} \right) = \frac{\mathbb{P}_G^{i_1} \cdot \mathbb{P}_G^{i_2}}{\|\mathbb{P}_G^{i_1}\| \|\mathbb{P}_G^{i_2}\|}.$$

end for
obtain

$$\mathbf{W} = \left[\mathbf{W}_{i_1,i_2} \right]_{u_{i_1}, u_{i_2} \in \mathcal{U}_f},$$

$\mathbf{W}$ acts as adjacency matrix of similarity graph among employees in $\mathcal{U}_f$.

The group comparative activity series $[Y_i^a(t)]_{t=1}^T$ can measure the difference of activity between employee u_i and his peers over time. Thus it is a good representative of the behavior of employee u_i from the group point of view.

B. Graph Based Anomaly Detection

For the employee u_i and the attribute $a \in \mathcal{A}$, we use a state space model [14] for time series prediction modeling. It is used to learn and represent the behavior histories of the self and group activity series given by $[X_i^a(t)]_{t=1}^T$ and $[Y_i^a(t)]_{t=1}^T$, respectively, in a dynamic way. At any time t, we define a history based anomaly score for the employee u_i as follows:

$$\sum_{a \in \mathcal{A}} \left[X_i^a(t) - \hat{X}_i^a(t) \right]^2, \tag{12}$$

where $\hat{X}_i^a(t)$ is the predicted value obtained from his trained time-series prediction model for each attribute $a \in \mathcal{A}$. The prediction model is learned based on the past observations for a predefined training period of M weeks.

Furthermore, to eliminate the influence from different magnitude of observations corresponding to different attributes, we standardize the prediction error. Standardization is performed by taking a difference of the prediction error from its mean, and then dividing by its standard deviation. Hence, we obtain the self-anomaly score $A_i^s(t)$ for the employee u_i at time t, given by

$$A_i^s(t) = \sum_{a \in \mathcal{A}} \left\{ \frac{(X_i^a(t) - \hat{X}_i^a(t)) - \frac{1}{t-1} \sum_{k=1}^{t-1} (X_i^a(k) - \hat{X}_i^a(k))}{\text{sd}([X_i^a(k) - \hat{X}_i^a(k)]_{k=1}^{t-1})} \right\}^2, \tag{13}$$

where $\text{sd}(\cdot)$ denotes the empirical standard deviation of its arguement, which is a sequence.

Similarly, a state space model is learned to represent the behavior history of $[Y_i^a(t)]_{t=1}^T$. For each time t, we obtain a group anomaly score $A_i^g(t)$ as follows:

$$A_i^g(t) = \sum_{a \in \mathcal{A}} \left\{ \frac{(Y_i^a(t) - \hat{Y}_i^a(t)) - \frac{1}{t-1} \sum_{k=1}^{t-1} (Y_i^a(k) - \hat{Y}_i^a(k))}{\text{sd}([Y_i^a(k) - \hat{Y}_i^a(k)]_{k=1}^{t-1})} \right\}^2, \tag{14}$$

We can detect insiders with a reduced number of false alarms by leveraging the self-anomaly and group-anomaly scores: $A_i^s(t)$ and $A_i^g(t)$. The proposed anomaly detection framework is described in Algorithm 3. When it is found that an employee does not follow his own historical behavior pattern, that is, the self-anomaly score is above a predefined threshold λ, his behavior is further examined among his peers. If his group comparative behavior is observed to be normal, that is, the group-anomaly score is below the predefined threshold λ, we consider the employee as non-anomalous. Thus, by considering the group behavior, in our proposed framework, we obtain a reduced false alarm rate.

IV. Numerical Results and Discussions

To demonstrate the algorithm and understand its applicability, we developed a proof-of-concept software tool for assessing the behavioral similarity among the employees of an organization. We also present the ability and performance of the integrated anomaly detection algorithm. The system has been developed using Python and R programming languages.

A. Dataset Description

Our experiment uses the publicly available dataset provided by Carnegie Mellon University's insider threat program [15]. We have specifically used the R6.2 version of the dataset. The dataset provides answer key file that contains details about the malicious activities included in the dataset and the employees involved [15]. This dataset represents a synthetic organization consisting of 4000 employees, each with a defined job role. Various activities of the employees, such as, system login and logout, sending or viewing emails, email attachments, file access information, website access information, and use of USB devices have been provided in the dataset, which

Algorithm 3 Integrated Anomaly Detection Framework

obtain slot wise activity record of employees $[X_i^a(t)]_{i=i}^N$, for all $a \in \mathcal{A}$, and $t \in [0, T]$.
at time t**:**
obtain Laplacian matrix $\mathbf{L}$ using the graph correlation matrix for N employees learned using Algorithm 2 and generate observations $[Y_i^a(t)]_{i=i}^N = \mathbf{L}[X_i^a(t)]_{i=i}^N$.
for each employee u_i **do**
  for each attribute $a \in \mathcal{A}$ **do**
    compute the predicted value $\hat{X}_i^a(t)$ based on the prediction model learned from the past M weeks observations.
  end for
  calculate the self-anomaly score $A_i^s(t)$ defined in Eq. (13).
  if $A_i^s(t) \geq \lambda$ **then**
    for each attribute $a \in \mathcal{A}$ **do**
      calculate the predicted value $\hat{Y}_i^a(t)$ based on the prediction model learned from the past M weeks observations.
    end for
    calculate the group-anomaly score $A_i^g(t)$ defined in Eq. (14).
    if $A_i^g(t) \geq \lambda$ **then**
      return employee u_i as anomalous user at time t.
    else
      return employee u_i as normal user at time t.
    end if
  else
    return employee u_i as normal user at time t.
  end if
end for

are parsed to obtain time-series information on the different attribute values listed in Table I. These attributes from the employee activity profiles are used to define the similarity scores among the employees as descibed in Section II.

B. Correlation Graph

Based on job roles in the organization, we choose four different role based groups, namely, salesman (G_1), IT admin (G_2), computer scientist (G_3), and electrical engineer (G_4). We consider a training/reference set $\mathcal{U}_r$ consisting of 280 employees. The similarity measures are computed for an observation period of 1 month with slot durations of 6 hours. We obtain the similarity scores for $N = 80$ employees in set $\mathcal{U}_f$. For the ease of visibility, we present the adjacency matrix for only 20 employees in Figures 1 and 2, which are based on the CSM and GPSM methods, respectively.

For the results in Figure 1, we present the similarity score matrix obtained by using CSM over observations during the time period of June 15, 2010 to July 15, 2010. The similarity scores have been color-coded depending on their values. The visible block diagonal structure validates our hypothesis that the employees in same role depict high correlation in their activity pattern. In addition, it is observed that the employees in groups G_3 and G_4 have high similarity measure with each other.

TABLE I: Set of attributes for characterizing employee profiles.

Logon:	L1: Number of logons L2: Number of Logoffs
Email:	E1: Number of e-mails sent E2: Number of e-mails read E3: Number of attachments E4: Average Content Word Length E5: Average size of sent emails
Http:	H1: Number of files uploaded H2: Number of files downloaded H3: Number of URLs visited
File:	F1: Number of files opened up F2: Number of copies made F3: Number of files copied to/from USB
Device:	D1: Number of devices connected D2: Number of devices disconnected

In Figure 2, we present the similarity score matrix computed using GPSM over observations during the time period from June 15, 2010 to July 15, 2010. We have used a support vector machine (SVM) classifier with Gaussian kernel to obtain Platt scaling based group probabilities [16], [17]. Similar to the observation in Figure 1, it depicts that the employees in the same role exhibit high correlation among themselves. Employees in group G_2 have very low similarity scores with those in G_1. The employees in group G_1 have moderate similarity scores with those in G_3 and G_4 whereas the employees in G_3 and G_4 are highly correlated. The obtained similarity values vary from those in Figure 1, as in GPSM, the similarity is based on the group affinity values predicted using the mean similarity feature vector. Hence, the similarity scores will depend on the similarity definition given as per our objective. The similarity score matrix shown in Figure 2 represents the adjacency matrix of the inferred correlation graph.

C. Group Anomaly Detection

For time-series prediction modeling, we use a state space model referred to as ETS model (Error, Trend, and Seasonal) [14]. There are several possibilities for each of the states: Error , Trend and Seasonality. For our current implementation, we use the software R, where the most appropriate state space model is selected automatically by using the Akaike information criterion (AIC).

We consider the set $\mathcal{U}_f$ with $N = 80$ employees, for an observation period of 10 months, with slot durations of 2 hours. We present the results for data corresponding to the time period of Jan 1, 2010 to Oct 15, 2010, which contains 3 insider instances spread over different time periods as shown in Table II.

For each employee, we use the first $M = 12$ weeks of observations to initially train the ETS prediction models for self and group activity series, and detect the abnormal activity

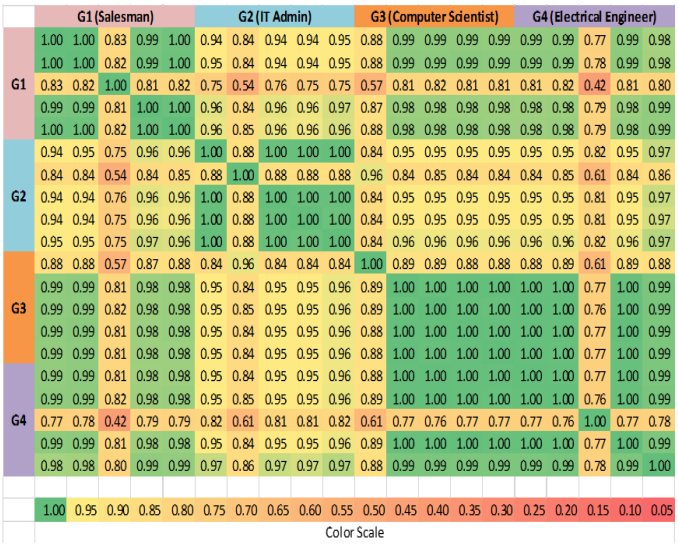

Fig. 1: Pairwise similarity scores obtained using CSM in Algorithm 1.

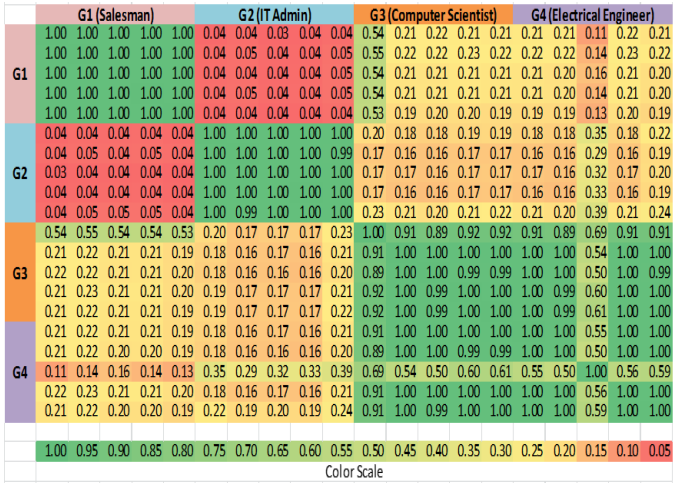

Fig. 2: Pairwise similarity scores obtained using GPSM in Algorithm 2.

for the time period of April 2, 2010 to Oct 15, 2010, while dynamically training the prediction models. For this time period, there are 3 insiders, employees 19, 40 and 80. On the other hand, we use the employee's 2-hourly activity series from June 15, 2010 to July 15, 2010 to learn the correlation graph, which captures the similarity relationships for normal behavior period.

We plot Figure 3 to show the effect of graph based group anomaly detection in reducing the false alarm rate. We plot the number of detected anomalous employees in each day using both the self-anomaly detection and the integrated framework proposed in Algorithm 3, for a predefined threshold $\lambda = 400$. It is observed that after considering the group activity information, the number of normal employees which are detected to be anomalous reduces significantly.

Furthermore, in Figure 4, we depict the anomaly scores

TABLE II: Assumption on the ground truth of insider activity.

Date	**Aug** 12	**Aug** 18 **- Aug** 24	**Oct** 12
Insider	**Employee** 40	**Employee** 19	**Employee** 80

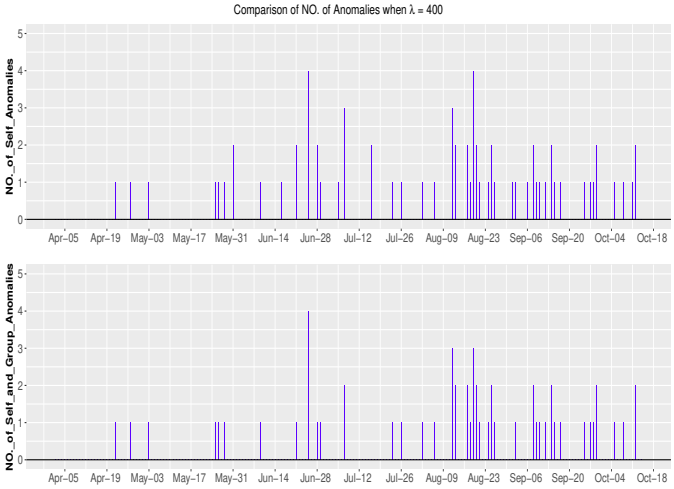

Fig. 3: Comparison of the daily count of anomalous employees before and after group anomaly based reduction.

obtained using the proposed method, for the employees detected to be anomalous in the dates when there were actual insider activities. As mentioned before, we use the ground truth information from the dataset, that is, the actual insiders are employee 40 on Aug 12, employee 19 from Aug 18 to Aug 24, and employee 80 on Oct 12, as listed in Table II. Figure 4 can be regarded as a close explanation to a snapshot of the plot in Figure 3. In this figure, red dots, blue rectangles and green stars represent malicious insiders $E19$, $E40$, and $E80$ (employees 19, 40, and 80) at their insider dates, respectively. Black dots represent the normal employees detected to be anomalous on these days. It can be observed that the insiders generally have higher anomaly scores than the other normal employees who are detected to be anomalous. Our integrated

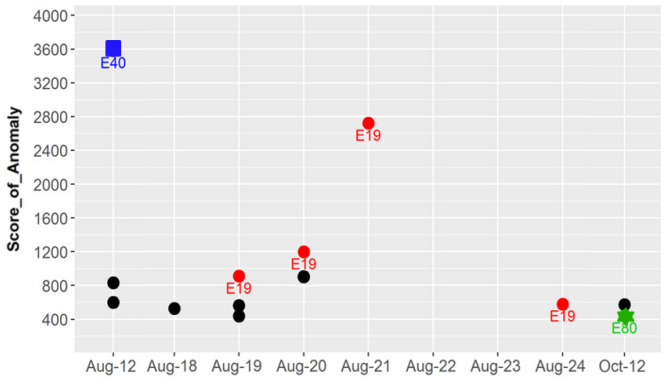

Fig. 4: Anomaly scores for the detected anomalous employees in the actual dates when there are insider activities.

anomaly detection system could successfully detect all of the 3 insiders in their insider periods. However, the algorithm failed to detect insider 19 on Aug 18, Aug 22, and Aug 23. The possible reason is that the assumption that employee 19 is a malicious insider on each of the days from Aug 18 to Aug 24 might be too conservative; the employee may not be anomalous on each of these days. Furthermore, as observed in Figure 3, our system raises only few false alarms on normal days, which can be manually investigated. Hence, it enables us to trigger further investigation to avoid or reduce potential losses.

V. Conclusion

The problem studied in this paper has been motivated by the insider threat in organizational scenarios. To detect the insider behavior of the employees, the self anomaly based on employee's own historical activities and group anomaly based on the historical comparative behavior with respect to the peers have been combined to enhance the insider threat detection.

We have proposed an algorithm to infer a graph that provides the contextual information for our application. We have defined correlation similarity measure that computes similarity among the employees based on their mean similarity patterns with a set of reference employees belonging to different role-based groups. Furthermore, we have proposed group propensity similarity measure, for application specific needs, which learns to identify their group affinities correctly based on the mean similarity measures obtained for various attributes. Results depict that both the proposed similarity measures are able to validate our assumption that employees with similar job roles exhibit high correlation in their activity patterns. The similarity between employees belonging to different groups can also be learned.

An integrated anomaly detection algorithm has been proposed, and the experimental results have shown its capability to detect all of the 3 insider instances and on most of the insider dates. Most importantly, it could reduce the false alarm rate significantly on comparing with self anomaly detection only.

Acknowledgement

We thank Liu Kai, Ph.D. student at Nanyang Technological University, Singapore, for his technical assistance in parsing the log data.

References

[1] J. R. C. Nurse, O. Buckley, P. A. Legg, M. Goldsmith, S. Creese, G. R. T. Wright, and M. Whitty, "Understanding insider threat: A framework for characterising attacks," in *2014 IEEE Security and Privacy Workshops*, May 2014, pp. 214–228.

[2] 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, June 2017.

[3] I. Agrafiotis, P. Legg, M. Goldsmith, and S. Creese, "Towards a user and role-based sequential behavioural analysis tool for insider threat detection," vol. 4, no. 4, pp. 127–137, 01 2014.

[4] J. Park and J. Giordano, "Role-based profile analysis for scalable and accurate insider-anomaly detection," in *IEEE International Performance Computing and Communications Conference*, 2006.

[5] H. Eldardiry, E. Bart, J. Liu, J. Hanley, B. Price, and O. Brdiczka, "Multi-domain information fusion for insider threat detection," in *IEEE Security and Privacy Workshops*, May 2013, pp. 45–51.

[6] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, Apr. 2013.

[7] J. Mei and J. M. F. Moura, "Signal processing on graphs: Causal modeling of unstructured data," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 2077–2092, April 2017.

[8] T. E. Senator, H. G. Goldberg, A. Memory, W. T. Young, and et al., "Detecting insider threats in a real corporate database of computer usage activity," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2013, pp. 1393–1401.

[9] W. Eberle, J. Graves, and L. Holder, "Insider threat detection using a graph-based approach," *Journal of Applied Security Research*, vol. 6, no. 1, pp. 32–81, Dec. 2010.

[10] O. Brdiczka, J. Liu, B. Price, J. Shen, A. Patil, R. Chow, E. Bart, and N. Ducheneaut, "Proactive insider threat detection through graph learning and psychological context," in *IEEE Symposium on Security and Privacy Workshops*, May 2012, pp. 142–149.

[11] T. Wang, C. V. Fang, C.-M. Lai, and S. F. Wu, "Triaging anomalies in dynamic graphs: Towards reducing false positives," in *IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, Dec 2015.

[12] S. Haykin, *An Introduction to Analog and Digital Communications*. New York, NY, USA: John Wiley & Sons, Inc., 1989.

[13] M. S. Charikar, "Similarity estimation techniques from rounding algorithms," in *Proceedings of the Thiry-fourth Annual ACM Symposium on Theory of Computing*, New York, NY, USA, 2002, pp. 380–388.

[14] R. J. Hyndman and G. Athanasopoulos, *Forecasting : principles and practice / Rob J Hyndman and George Athanasopoulos*, print edition. ed. OTexts.com [Heathmont,Victoria], 2014 2014.

[15] "Carnegie Mellon University," https://www.cert.org/insider-threat/tools/, accessed: October 2016.

[16] "User Guide Documentation for Sci-kit Learn SVM," http://scikit-learn.org/stable/modules/svm.html.

[17] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *ADVANCES IN LARGE MARGIN CLASSIFIERS*. MIT Press, 1999, pp. 61–74.