CONTEXT-AWARE GRAPH-BASED ANALYSIS FOR DETECTING ANOMALOUS ACTIVITIES

Sreyasee Das Bhattacharjee *, Junsong Yuan, Zhang Jiaqi, Yap-Peng Tan

School of Electrical & Electronic Engg.,
Nanyang Technological University, Singapore
Email: sreya.iitm@gmail.com; jsyuan, jqzhang, eyptan@ntu.edu.sg

ABSTRACT
This paper proposes a context-aware, graph-based approach for identifying anomalous user activities via user profile analysis, which obtains a group of users maximally similar among themselves as well as to the query during test time. The main challenges for the anomaly detection task are: (1) rare occurrences of anomalies making it difficult for exhaustive identification with reasonable false-alarm rate, and (2) continuously evolving new context-dependent anomaly types making it difficult to synthesize the activities apriori. Our proposed query-adaptive graph-based optimization approach, solvable using maximum flow algorithm, is designed to fully utilize both mutual similarities among the user models and their respective similarities with the query to shortlist the user profiles for a more reliable aggregated detection. Each user activity is represented using inputs from several multi-modal resources, which helps to localize anomalies from time-dependent data efficiently. Experiments on public datasets of insider threats and gesture recognition show impressive results.

Index Terms— anomaly detection, graph analysis, insider threat, activity recognition

1. INTRODUCTION
The task for the anomaly detection is to identify anomalous behavior, events or items based on its deflection from the usual patterns in a normal scenario. Although the problem has been explored by the research community for several application scenarios such as cyber activity monitoring, intrusion detection, several aspects of fraud detection or crime prediction etc., the ever increasing problem complexity keeps it persistently challenging. Consider an insider threat scenario, where a disgruntled employee is trying to perform IP theft, place false information in the system or alter the classified information by executing a sequence of events. The organization should be sufficiently equipped to identify such attempts /motifs well in advance, to prevent some potential damage. However, at the same point of time, it is important to remember that having the trusted employees involved, raising false allegations can have a significant impact on the employee-organization relationships. Therefore, the need of the hour is for an insightful detection method, that can capture multiple aspects of data patterns typically available from several heterogeneous resources and their mutual dependence to shortlist the genuine anomalies with better precisions while remaining sufficiently generic to handle different cross-modal applications.

The entire spectrum of anomaly detection algorithms can be broadly classified into two classes, based on their adopted learning schemes: supervised [1, 2] and unsupervised [3, 4]. Supervised methods need a large amount of training data and an intensive time consuming learning session to build an effective model, which is not very practical always. On the other hand, with the increasing problem complexity and rare occurrences of anomalies unsupervised solutions also leave rooms for improvement.

In this paper, we propose a graph-based anomaly detection framework that can investigate a test activity capturing context-rich information simultaneously from two complementary perspectives: (1) deviation with respect to the usual normal activities of each user, as observed in a prolonged period of time and (2) mutual similarity among the user models. Each activity feature represents a user action detail obtained via multiple cross-modal resources in terms of a comprehensive descriptor. An alert is defined as the result of detecting a user incited event which is aberrant for a given user specific model based on certain criteria on the training activity patterns. While each user model can provide an initial feedback about the anomalous nature of the given test activity pattern, it is important to note that a user profile usually undergoes some continuous changes over time. Therefore, decision relying on any single individual response is often prone to raise many false alerts. For example, a user may perform an action not aligned with the organization’s rule, just to expedite the task to meet an approaching deadline and not with any malicious intent. This being significantly deviant from his usual working pattern, the individual level evaluation is prone to raise a large number of false alarms. In a large organization like setup, manual sorting of this huge amount of false
alarms is nearly infeasible. In order to automate the shortlisting process and generate more genuine alarms, a group-level feedback capturing more discriminative context information is found to be more reliable. Given a test activity, the proposed graph-based subset optimization task is designed to jointly optimize a smaller number of close neighbors (e.g. colleagues with similar roles/profiles), which are also sufficiently similar among themselves and jointly provides an optimized detection performance for the test activity. Lone feedbacks from each of these shortlisted members are aggregated to achieve a more salient anomaly decision.

The rest of the paper is organized as follows: Section 2 briefly describes related works. The proposed anomaly detection strategy is explained in Section 3. Section 4 and 5 respectively present the experimental results and conclusion.

2. RELATED WORK

Anomaly detection has been an important research problem for several security-based applications [5, 6]. Several techniques use domain knowledge [7, 8] or data-mining and machine learning algorithms [2, 4] to propose the application specific solutions.

While unsupervised methods fit better in this problem scenario, advanced methods use a good amount of normal class samples to learn the models, which can reject samples significantly deviant from what has been learnt. In his seminal work, Parzen [9] fits a non-parametric model to the normal samples and identifies the outliers using a statistical hypothesis test. In a similar line, Breunig et al.[10] evaluate a test pattern by comparing its local density to an average local density of its neighbors. However, with \( O(N^2) \) complexity where \( N \) is the number of samples, these methods are computationally expensive and do not fit well for a large database. One-class Support Vector Machine (SVM) [11] have been effective for various applications of anomaly detection. However, it does not take into consideration of the mutual dependence within the data. Yu and Joachims [1] have extended the standard SVM to incorporate the data dependence information in a multi-class problem but does not fit well into a one-class scenario.

Graph-based methods [12, 13, 14] have also been proposed in this problem layout. Static Graph based approaches [12, 13] represent the threat and the normal data as a graph and apply unsupervised learning to identify anomalies. Parveen et al. [14] extends to support dynamic, evolving streams. Many authors [15, 16] propose methods based on user access patterns for identifying threats. Eberle et al. [15] have used the social graph for detecting insiders, based on modifications, insertions, and deletions of activities from the graph. Eldardiry et al. [16] first select the peer groups of users and models user behavior with respect to its respective peer groups and subsequently identifies anomalous users as the ones who deviate from their peers with respect to the user behavior models. However, these methods focus on some specific threat scenarios and thus remain unable to handle previously unknown anomalies.

Several methods [17, 18] for sequential anomaly detection have also been proposed to localize the anomalies in a time-series discrete data. Song et al. [4] has proposed a log linear model to deal with the continuous attributes. Tan et al. [19] have used an ensemble of half-space trees with a sliding window to detect anomalies in the streaming data. Compared to the existing methods, our approach focuses on improving the baseline performance by exploiting the more insightful contextual evidence via a graph based subset optimization technique. As proved with the experimental evaluations, the proposed method handling features extracted from multiple input resources is sufficiently generic and can be seamlessly adopted in several cross-domain application scenarios.

3. PROPOSED METHOD

Given a set of users \( \mathcal{U} = [U_1, ... U_n] \) and an anomaly scoring function \( f \), the task is to identify an optimized subset \( S_c \) of \( \mathcal{U} \) that finds a probabilistic measure to quantify the anomalous nature of \( a \). \( f(U_i, a) \) evaluates the prior anomaly probability of a test activity \( a \) based on the learnt model for the user \( U_i \). The resulting score can, in turn, be thresholded to obtain the binarized output. Figure 1 illustrates the overall framework of the proposed method.

Each user activity is represented in terms of a compact descriptor which is typically a concatenation of several inputs details obtained from a collection of heterogeneous resources. As will be shown with the experiments on both gesture and insider threat datasets, feature descriptors typically capture action details from multiple perspectives.

3.1. Prior Score

The first task is to select a function \( f \) that can offer a reliable prior, based on the individual activity patterns for each user in \( \mathcal{U} \). Given a user \( U_i(\in \mathcal{U}) \), we consider the unsupervised learning of a classifier \( c_i : \mathcal{X}_i \rightarrow \mathcal{Y} \) from the on-class dataset \( \mathcal{D}_i = \{ (x^i_j, y^i_j) | x^i_j \in R^{(d \times t_i)}, y^i_j = +1 \}^{N_i}_{j=1} \). Therefore, all the training data are from the ‘normal’ class, i.e. \( \mathcal{X}_i \) consists of the \( d \) dimensional multivariate time-dependent data representing \( t_i \) activity samples of \( U_i \) (with \( y^i_j = +1, \forall j \) ) and \( \text{Card}(\mathcal{X}_i) = t_i \). Given a test activity \( a \), the anomaly scoring function \( f \) is defined as:

\[
 f(U_i, a) = \frac{1}{1 + e^{-\lambda c_i(a)}}
\]

(1)

where \( \lambda \) controls the penalty level and \( f(U_i, a) \) computes the prior score of \( a \) specific to \( U_i \), based on the given \( c_i \).

While several classifiers like one-class conditional random field [4], active outlier [3], local outlier factor [10], hidden Markov model [20], one-class support vector machine
(SVM) [11] have been used in the literature, we used one-class SVM as an example to model each $c_i$. The task of one-class SVM is to estimate a subset of the input space such that a test sample lies outside the subset equals a pre-specified parameter $\rho \in [0, 1)$. Following [11] we choose a Gaussian kernel $K(x, x') = e^{-\gamma \|x - x'\|^2}$, which ensures the data separability from the origin in the feature space. Important to note, the proposed method is not dependent on the choice of this basic scoring function and users can choose their own favorites for this purpose. The primary goal of this paper is to prove the effectiveness of the proposed framework which utilizes the context information by defining a graph-based user network to improve the baseline detection performance.

3.2. Context-Aware Graph Based Subset Optimization

Given a test activity $a$, the proposed graph regularization jointly optimizes to a smaller subgroup of users (showing a similar activity pattern as $a$) as well as their mutual agreement amongst themselves to obtain a comprehensive anomaly score for $a$. As will be discussed more in section 4, the proposed subset optimization performs considerably better than the usual clustering based methods like k-mean clustering where grouping is performed only using the mutual affinity among the candidate members. This improved behavior can be primarily attributed to the fact that the proposed optimization scheme is designed to exploit both query level context information as well as the mutual dependence of the data samples within an integrated query adaptive framework. Assuming the mutual independence of each $U_i \in \mathcal{U}$, the average estimate for the pairwise pattern agreement is defined as:

$$K(U_i, U_j) = \frac{Acc(U_i, D_j) + Acc(U_j, D_i)}{2}$$

where $Acc(U_i, D_j)$ computes the accuracy (proportion of the true detection in the total population) achieved by $c_i$ having operated on the training activity samples in $D_j$.

In a graphical platform this problem scenario is then adopted by defining an undirected weighted graph $G = (\mathcal{U}, E, K)$. Each graph node represents a user in $\mathcal{U}$ and $|\mathcal{U}| = n$. The connecting edge $e(U_i, U_j) \in E$ between the two nodes evaluates their mutual proximity and the assigned weight is defined as follows:

$$e(U_i, U_j) = \begin{cases} K(U_i, U_j) & \text{if } st(i,j) > Th \\ 0 & \text{otherwise} \end{cases}$$

where $st(i,j) = \max(Acc(U_i, D_j), Acc(U_j, D_i))$. This ensures retaining only significant connections. In our experiments, we used $Th = 0.4$. Given this $K$ and the test activity $a$, the proposed objective function is defined as follows:

$$d_a = \arg \max_{d \in \{0, 1\}^n} \{L^T d - \lambda d^T W d - \eta ||d||_0 \}$$

where $L = [(1 - f(U_1, a)), \cdots , (1 - f(U_n, a))]^T, W = D - e$ with $D \in R^{n \times n}$ a diagonal matrix with $D(i,i)$ representing the degree of the node $U_i$ in $G$ and $d \in \{0, 1\}^n$ is an indicator vector specifying the inclusion/exclusion of a node in the resulting subgraph. The parameter $\eta$ controls the sparsity factor.

In our implementation, we use the Boykov Kolmogorov algorithm [21] for solving (4). As the number of users in a typical real-life database is considerably smaller compared to the number of activity samples reported, with an order $O(cn^2|E|)$, where $c$ is the size of the minimum cut, the proposed method is significantly faster. As defined by the indicator vector $d_a$, the resulting maximum flow identifies an optimized subgraph $S_a$ of $\mathcal{U}$ in context to $a$. The resulting aggregated anomaly score for $a$ is then computed as:

$$Acc_{avg}(a) = 1 - \frac{||S_a||}{||\mathcal{U}||}$$

Intuitively, for a normal activity test pattern $a$, $||S_a||$ should be large and thus the resulting $Acc_{avg}(a)$ is small. On the other hand, a less crowd in $S_a$ influences a large anomaly score $Acc_{avg}(a)$ for $a$.

4. EXPERIMENTS

The proposed method is evaluated in several datasets, which includes our own in-house data collections as well as the state-of-the-art datasets. Due to space constraints and easy reproducibility of the results, in this paper, we report the results on two public datasets describing the problem scenarios in completely different settings. Promising results on both of these datasets prove the generic nature of the algorithm.

4.1. Dataset

We have used CMU-CERT insider threat dataset [22] and the NATOPS aircraft signal handling dataset [23] for evaluations.

**CMU-CERT Insider Threat Dataset:** The dataset captures the 17 months of activity logs of the 1000 users (with only 70 insiders) in an organization (12 months of 2010 and 5
months of 2011), that consists of information on five different activities: login, usb device, email, web and file access. Each record is parsed to obtain details like a timestamp, user ID, device ID, action details etc. This being a synthetic dataset, email and web details are not much useful and in our experiments, we have used only three types of activities (login, usb device, and file access) to build 1000 user specific profiles in terms of the compact feature descriptors with 23 different feature attributes obtained from cross-modal resources:

1. **Timestamp:** The month is indexed in \([1, \ldots, 17]\), the date in \([1, \ldots, 31]\) and time in \([0, 1, \ldots, 23]\). Therefore, the quantization for time is only in terms of its hour stamp, while ignoring the ‘minute’ details. Each timestamp, therefore, represents a 3 dimensional observation.

2. **User ID:** Indexed in \([1, \ldots, 1000]\). The 1 dimensional ID stamp is represented by the integer datatype. This information for every activity facilitates some baseline experiments, but not used by the proposed method in this work.

3. **Logon/Logoff Activity:** On a given day, the logon/logoff activity in a specific day is described in terms of the following 3 observations (attributes) below:
   - **First Login Time**
   - **Last Login Time**
   - **Last Logoff Time**

   Each of these three attributes provides an 1 dimensional hour stamp. The datatype is integer.

   - **Average Session Duration:** In a specific day, a session is defined in terms of a pair of consecutive logon and logoff. It defines 1 dimension of the proposed descriptor, with a double datatype.
   - **Maximum Session Duration:** On a specific date, defines 1 dimension of the proposed descriptor. The datatype is integer.
   - **Number of Sessions**
   - **Number of Computers Accessed**

   Each of these two attributes provides an 1 dimensional observation with integer datatype.

4. **File Activity:** On a given day, the file access activity details is captured in terms of the following 10 observations (attributes) below:
   - **First Copying Activity**
   - **Last Copying Activity**

   Each of these two defines an 1 dimensional hour stamp. The datatype is integer.

   - **Number of Computers Accessed for the Copying Activities:** Defines 1 dimension of the proposed descriptor. The datatype is integer.
   - **Counts of the different types of files copied:** 6 types of files (pdf, word, doc, jpg, txt, zip, exe) have been accessed/copied by the users in this database. These attributes, therefore, create 6 dimensions of the proposed feature descriptor. The datatype is integer.
   - **Total number of files copied:** 1 dimensional observation. The datatype is integer.

5. **Device Activity:** On a given day, the file access activity details is captured in terms of the following 3 observations below (each is 1 dimensional integer datatype):
   - **Earliest (and last) time in a day an external device used**
   - **Number of times external devices used**

The entire processed dataset has around 450K activity instances (each is a 23 dimensional feature descriptor) among which only 2993 instances represent the anomalous (insider) activities.

**NATOPS Gesture Dataset:** This dataset consists of 6 classes of aircraft handling gesture signals used by the US Navy in aircraft handling aboard aircraft carriers following the Naval Air Training and Operating Procedures Standardization (NATOPS). Body features include 3 dimensional velocities of four body joints (left/right elbows and wrists) which result in a 12 dimensional feature vector. Hand features include probability estimates of four predefined hand shapes (opened/closed palm, and thumb up/down), creating a 8 dimensional feature vector. In total the feature descriptor is 20 dimensional. The dataset consists of samples from 20 participants performing each gesture 20 times (400 samples per gesture), resulting in 2400 samples compiled in total. Each gesture sample in this dataset is represented in terms of a sequence which is a \(20 \times T_i\) matrix, where \(T_i\) is the length of the \(i^{th}\) gesture sequence (i.e., each column of the matrix represents the observation at each time frame). The average length of each sequence in the dataset is 36. Thus, there is an average of 700 sample points representing a given gesture for each user. We repeated the experiments using NATOPS dataset for 6 times. At every iteration, only 1 of the 6 classes is used as the normal class and the instances from all the other 5 classes are taken to be anomalous.

**Evaluation Metrics:** Given the fact that the primary challenge in this problem scenario is the unbalanced data, we use Receiver Operator Characteristic (ROC) curves [24] and Area-Under-Curve (AUC) measure computed from these ROC curves for evaluating the performance.
4.2. Results

**CMU-CERT Insider Threat Dataset**: There are 1000 users in this dataset. 300 randomly chosen normal activity samples of each user $U_i$ are used to train its one-class SVM classifier $c_i$. We have used three types of baseline scenarios for performing the comparative study in this dataset: *Single Model One-Class SVM, Individual Profile Analysis, k-User Clustering*.

- In *Single Model One-Class SVM*, we use some randomly chosen 2 months of normal activity samples gathered from all users to train a single SVM model for the whole dataset.

- In *Individual Profile Analysis*, given a test activity $a$, we first identify the user $U_k$ who performed this activity and then evaluate the anomalous score of $a$ only against that specific one-class SVM classifier $c_k$, i.e. the anomalous score $f(U_k, a)$ as computed in (1) defines the anomaly score for $a$. However such information about the actor ID (who performed the test action) is not required for the proposed method.

- In *k-User Clustering*, we cluster users showing similar activity patterns together using k-means clustering. The subset of activities from a given user in the training data collection is max-pooled (or mean-pooled) to obtain a single aggregated descriptor for the user. The entire collection of 1000 such aggregated user representative descriptors are then clustered using k-means clustering, we set $k = 100$. Given a cluster $C_i \in \{C_i\}_{i=1}^{k}$, the entire sub-collection of training activities performed by any of the users from $C$ is used to learn a single one-class SVM $f_{C_i}^{last}$ resulting in $k$ such one-class SVMs each representing a cluster. Given a test activity $a$, we find its nearest neighbor $C_a$ from $\{C_i\}_{i=1}^{k}$ and evaluate it using $f_{C_a}^{last}$. While the proposed optimization scheme also attempts to shortlist some closely similar users, a favorable comparison against this linear, offline clustering method as baseline, proves its superiority.

All the 2293 anomalous activity instances and the normal activity samples which were never used for learning create the test data collection. At any given iteration of experiments, all the anomalous samples and 20000 randomly chosen normal samples from the test collection are used for testing. The same experiments are repeated for 10 times. Figure 2 displays the result of one iteration. As can be seen in the figure, the proposed graph based subset optimization offers an improvement of around 6% over the best performing baselines. Additionally, although each individual profile classifier $c_i$ is learnt using a much lesser amount of training data compared to that used for training the single model one-class SVM, both these baseline models perform nearly equivalently. An average performance computed for the entire 10 iterations is described in the Table 1. Except mean-pool k-user clustering, all the other 3 baselines show nearly similar average performance, our proposed graph based subset optimization shows around 5 – 7% improvement over these basic settings. We have also compared the performance against Maximum Clique (MC) based subset selection process as baseline. Given the top $N$ matched nodes to the query, constituting a subgraph of $G$, we find the MC within it. The selected nodes in MC defines the subgraph $S_a$ as in Eqn 5. As shown in the table, the proposed method shows around 5% improved performance over MC.

**NATOPS Gesture Dataset**: There are 20 users in this dataset. For any given gesture type $j \in [0, 1, \cdots, 5]$, a randomly chosen 60% of the sample points obtained from a user $U_i$ are used to train its one-class SVM classifier $c_i$. The rests are left for testing. The test collection includes all these left over sample points for class $j$ as well as all sample points from the other 5 gesture classes. Table 2 explains the comparative performance details of the proposed subset optimization compared to a similar set of baseline methods Individual Profile Analysis and Single Model One-Class SVM as described earlier for the CMU-CERT insider Threat database. As seen in the table that the proposed method achieves an impressive 14% performance improvement on this multiclass dataset over the Single Model One-Class SVM model and nearly 10% improvement over the average performance reported for the Individual Profile Analysis. Improved performance of Individual Profile Analysis compared to Single Model One-Class SVM is attributed to the fact that for performing this individual level analysis, required is an extra piece of information about the performer identity of the test activity. In a scenario, where this information is not available or the test activity is performed by an outsider, such user specific analysis is not possible. We also report a more detailed class specific performance comparison between Single Model One-Class SVM and the proposed method in the figure 3.

5. CONCLUSION

This paper addresses the problem of anomaly detection for a diverse range of application scenarios. Using multimodal
Table 1. Summary of the experimental results in the CMU-CERT Insider Threat Dataset using Area-Under-Curve (AUC) measures.

<table>
<thead>
<tr>
<th>Method</th>
<th>Single Model HMM</th>
<th>Single Model One-Class SVM</th>
<th>Individual Profile Analysis</th>
<th>Max-pool k-user Clustering</th>
<th>Mean-pool k-user Clustering</th>
<th>Maximum Clique subset Section</th>
<th>Proposed Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC</td>
<td>0.6135</td>
<td>0.8917</td>
<td>0.8875</td>
<td>0.9005</td>
<td>0.8170</td>
<td>0.9185</td>
<td>0.9520</td>
</tr>
</tbody>
</table>

Table 2. Summary of the experimental results in the NATOPS gesture Dataset using Area-Under-Curve (AUC) measures.

<table>
<thead>
<tr>
<th>Method</th>
<th>Single Model HMM</th>
<th>Single Model One-Class SVM</th>
<th>Individual Profile Analysis</th>
<th>Proposed Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC</td>
<td>0.4570</td>
<td>0.5772</td>
<td>0.6113</td>
<td>0.7196</td>
</tr>
</tbody>
</table>

data inputs within a compact activity descriptor, the proposed method can efficiently detect the anomalous activities by jointly optimizing the query-sample similarities as well as the mutual similarities within the samples in collaboration. Showing a generic nature, the framework can be adopted over any baseline scoring function to improve the detection performance. By compartmentalization of a time-dependent data in terms of a sequence of activities, this method can efficiently perform the task of localization. We plan to apply this method to various more complicated real-life anomaly detection scenarios in future.

Fig. 3. Comparative Study: Results are shown using the AUC scores. Given the normal class index represented by the horizontal axis, the bar pairs display the results of the 6 stage experiments of the proposed method against the baseline.

6. REFERENCES


