MULTI-CENTRALITY GRAPH SPECTRAL DECOMPOSITIONS AND THEIR APPLICATION TO CYBER INTRUSION DETECTION

Pin-Yu Chen[⋆] *Sutanay Choudhury*[†] *Alfred O. Hero III*[⋆]*, Fellow, IEEE*

[⋆]Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, USA
{pinyu, hero}@umich.edu
[†]Pacific Northwest National Laboratory
Sutanay.Choudhury@pnnl.gov

ABSTRACT

Many modern datasets can be represented as graphs and hence spectral decompositions such as graph principal component analysis (PCA) can be useful. Distinct from previous graph decomposition approaches based on subspace projection of a single topological feature, e.g., the Fiedler vector of centered graph adjacency matrix (graph Laplacian), we propose spectral decomposition approaches to graph PCA and graph dictionary learning that integrate multiple features, including graph walk statistics, centrality measures and graph distances to reference nodes. In this paper we propose a new PCA method for single graph analysis, called multi-centrality graph PCA (MC-GPCA), and a new dictionary learning method for ensembles of graphs, called multi-centrality graph dictionary learning (MC-GDL), both based on spectral decomposition of multi-centrality matrices. As an application to cyber intrusion detection, MC-GPCA can be an effective indicator of anomalous connectivity pattern and MC-GDL can provide discriminative basis for attack classification.

1. INTRODUCTION

Many real-world data ranging from physical systems, social interactions, network flows, knowledge graphs to biological and chemical reactions are often represented as graphs, especially for anomaly or community detection [1–9] and graph signal processing [10–14]. Dimensionality reduction methods on graphs allow one to decompose a graph into principal components using a spectral decomposition of the graph adjacency or graph Laplacian matrix. In this paper we propose a general framework to dimensionality reduction based on spectral decomposition of a matrix composed of many different graph centrality statistics. This general framework leads to a single-graph decomposition method that extends graph principal components analysis (PCA) and a graph-ensemble decomposition method that extends dictionary learning. These methods are applicable to both directed and undirected graphs with edge weights and are based on a spectral decomposition, specifically the singular value decomposition (SVD), of a matrix composed of multiple graph centrality statistics. The proposed methods are denoted multi-centrality graph PCA (MC-GPCA) and multi-centrality graph dictionary learning (MC-GDL), respectively. By integrating multiple descriptions of graph centrality, the proposed methods provide graph community detection and graph structure learning that are significantly more robust to noise and variation affecting graph connectivity structures.

In [15], a kind of graph PCA is performed on the distance matrix of average commute time between nodes. In [16], PCA can also be performed on the graph Laplacian matrix of nodal similarities. In [17], the graph Laplacian matrix is used as a smooth regularization function for robust PCA on graphs. In [18, 19], PCA is performed on the matrix of origin-destination traffics. In [20–22], dictionary learning methods for graph signals are proposed based on the graph Laplacian matrix. Dictionary learning, also known as sparse coding, linear unmixing and matrix factorization, has been applied to collections of images, audio, and graph signals to learn low dimensional representations that give a sparse approximation to the entire collection. Dictionary learning finds a low rank factored-matrix approximation to the observation matrix, whose columns span this collection. Many different methods for this approximation problem have been proposed [23, 24]. Among the simplest methods is the K-SVD approach [25] which uses a spectral decomposition to determine the best low rank approximation to the observed matrix. For the purposes of illustration, in this paper we adopt this latter spectral approach for learning a dictionary spanning an ensemble of graphs.

More often graph PCA and graph dictionary learning approaches start with a set of raw multivariate data samples, create a similarity (or dissimilarity) graph of the data samples, and aim to learn a low-dimensional or sparse representation of the original multivariate dataset. When applied to graph data, these methods are often limited to graphs that are weighted, undirected and connected, which may not be feasible for applications such as cyber network data analysis. Furthermore, these methods often accomplish graph decomposition based on a single measure of centrality, e.g., betweeness centrality [26], closeness centrality [27], ego centrality [28], or eigenvector centrality [1]. In this paper we introduce graph spectral decomposition methods that combine multiple centrality measures such as graph walk statistics and graph distances as structural features and apply them to different graph types including weighted, directed and disconnected graphs. The proposed MC-GPCA method decomposes a single graph utilizing multiple centrality features, achieving dimensionality reduction and feature decorrelation of the graph. The proposed MC-GDL performs dictionary learning across a population of graphs using multiple centrality features to learn the atoms of the dictionary and the corresponding coefficients to represent each individual graph in terms of its projection onto the dictionary. Applying our approach to cyber intrusion detection, we use MC-GPCA to define a structural difference score (SDS) that reflects structural variations within a graph and we use MC-GDL to learn discriminative structural atoms for classifying the presence of cyber attacks.

This work was partially supported by the Consortium for Verification Technology under Department of Energy National Nuclear Security Administration award number DE-NA0002534 and by the Asymmetric Resilient Cyber Security initiative at Pacific Northwest National Laboratory, which is operated by Battelle Memorial Institute.

2. STRUCTURAL FEATURE EXTRACTION ON GRAPHS

Here we describe three categories of generic structural features that can be extracted from a graph, namely graph walk statistics, centrality measures and internode distances. The utility of the introduced features with respect to different graph types, including weighted, directed and disconnected graphs, is summarized in Table 1. While not investigated in this paper, application-specific features such as website hit rates, social interaction frequency, source-destination traffics can also be leveraged as structural features. Without loss of generality a graph $G = (\mathcal{V}, \mathcal{E})$ can be characterized by two $n \times n$ matrices $\mathbf{A}$ and $\mathbf{W}$ representing the adjacency and weight matrix, respectively, where $\mathcal{V}$ ($\mathcal{E}$) is the set of nodes (edges), and n is the total number of nodes (i.e., graph size). $\mathbf{A}$ is a binary matrix such that its entry $[\mathbf{A}]_{ij} = 1$ if there is an edge connecting from node i to node j, and $[\mathbf{A}]_{ij} = 0$ otherwise. Throughout this paper we consider graphs with nonnegative edge weights such that $\mathbf{W}$ is a nonnegative matrix, where its entry $[\mathbf{W}]_{ij} \geq 0$ if $[\mathbf{A}]_{ij} = 1$, and $[\mathbf{W}]_{ij} = 0$ otherwise.

2.1. Graph walk statistics

Graph walk statistics include commute time and cover time [29], graph diffusion [30], hitting times [31], and hop walks. In this paper we focus on hop walk statistics. An h-hop walk of a node on a graph is a path starting from the node and traversing through (possibly repeated) h edges. An h-hop walk weight is defined as the sum of edge weights of the corresponding path. We consider the number and total weight of h-hop walks of each node as features since they entail the structural information of nodal reachability relative to its h-hop vicinity. In principle one should extract graph walk statistics from $h = 1$ to at least $h =$ *graph diameter* hops as structural features, where graph diameter is the largest shortest path hop count between any node pairs in all connected components of a graph. We propose an efficient iterative computation method to incrementally computes these two structural features with respect to the hop count number h:

• Iterative computation of number of h-hop walks

Let $\mathbf{A}^h$ denote the matrix product of h copies of $\mathbf{A}$. Observe that the entry of $\mathbf{A}^2$, $[\mathbf{A}^2]_{ij} = \sum_k [\mathbf{A}]_{ik}[\mathbf{A}]_{kj}$, is the number of 2-hop walks from i to j. Extending this result to $\mathbf{A}^h$ we have $[\mathbf{A}^h]_{ij}$ being the number of h-hop walks from i to j. Let $\mathbf{a}^{(h)} = \mathbf{A}^h \mathbf{1}_n$ be a column vector where its entry $[\mathbf{a}^{(h)}]_i$ is the number of h-hop walks starting from i and $\mathbf{1}_n$ denotes the $n \times 1$ column vector of ones. Then $\mathbf{a}^{(h+1)}$ can be computed by the matrix-vector product iteration

$$\mathbf{a}^{(h+1)} = \mathbf{A}^{h+1}\mathbf{1}_n = \mathbf{A} \cdot \mathbf{A}^h \mathbf{1}_n = \mathbf{A}\mathbf{a}^{(h)}. \tag{1}$$

• Iterative computation of total h-hop walk weight

Let $\mathbf{W}^{(h)}$ be an $n \times n$ matrix such that its entry $[\mathbf{W}^{(h)}]_{ij}$ is the sum of all h-hop walk weights from node i to node j. Then we have

$$\begin{aligned}
[\mathbf{W}^{(h+1)}]_{ij} &= \sum_{k \in \mathcal{V}} \left([\mathbf{W}]_{ik} \cdot [\mathbf{A}^h]_{kj} + [\mathbf{W}^{(h)}]_{kj} \right) \cdot \mathbf{A}_{ik} \\
&= \sum_{k \in \mathcal{V}} [\mathbf{W}]_{ik} \cdot [\mathbf{A}^h]_{kj} + \sum_{k \in \mathcal{V}} [\mathbf{W}^{(h)}]_{kj} \cdot \mathbf{A}_{ik} \\
&= [\mathbf{W}\mathbf{A}^h + \mathbf{A}\mathbf{W}^{(h)}]_{ij},
\end{aligned} \tag{2}$$

where we use $[\mathbf{W}]_{ik} \cdot [\mathbf{A}]_{ik} = [\mathbf{W}]_{ik}$. Let $\mathbf{w}^{(h)} = \mathbf{W}^{(h)}\mathbf{1}_n$ denote a column vector such that its entry $[\mathbf{w}^{(h)}]_i$ is the total h-hop walk weight starting from node i. Then $\mathbf{w}^{(h+1)}$ can be computed by

$$\mathbf{w}^{(h+1)} = \left[\mathbf{W}\mathbf{A}^h + \mathbf{A}\mathbf{W}^{(k)} \right] \mathbf{1}_n = \mathbf{W}\mathbf{a}^{(h)} + \mathbf{A}\mathbf{w}^{(h)}. \tag{3}$$

Table 1: Utility of the introduced structural features.

Feature / Graph Type	Weighted	Directed	Disconnected
# of h-hop graph walks	✓	✓	✓
total h-hop walk weight	✓	✓	✓
degree	✓	✓	✓
betweenness	✓	✓	
closeness	✓	✓	
eigenvector centrality	✓	✓	✓
ego	✓	✓	✓
LFVC	✓		✓
graph distance	✓	✓	

2.2. Centrality measures

A centrality measure is a quantity that evaluates the level of importance or influence of a node in a graph and it reflects certain topological characteristics. Here we introduce several centrality measures, which will be used in the sequel to define feature sets associated with a graph or a set of graphs.

• Degree. Degree is defined as the number of edges associated with a node. It can be extended to directed graphs by considering the number of edges connecting to (from) a node as in-degree (out-degree).

• Betweenness [26]. Betweenness is the fraction of shortest paths passing through a node relative to the total number of shortest paths in the graph. It is infeasible for disconnected graphs since it is based on shortest path distance. The betweenness of node i is defined as

$$\text{betweenness}(i) = \sum_{k \in \mathcal{V}, k \neq i} \sum_{j \in \mathcal{V}, j \neq i, j > k} \frac{\sigma_{kj}(i)}{\sigma_{kj}}, \tag{4}$$

where σ_{kj} is the total number of shortest paths from k to j and $\sigma_{kj}(i)$ is the number of such shortest paths passing through i.

• Closeness [27]. Closeness is associated with the shortest path distances of a node to all other nodes. Let $\rho(i, j)$ denote the shortest path distance between node i and node j in a connected graph. Then

$$\text{closeness}(i) = \frac{1}{\sum_{j \in \mathcal{V}, j \neq i} \rho(i, j)}. \tag{5}$$

• Eigenvector centrality [1]. Eigenvector centrality of node i is the i-th entry of the eigenvector associated with the largest eigenvalue of the weight matrix $\mathbf{W}$. It is defined as

$$\text{eigenvector centrality}(i) = \lambda_{\max}^{-1} \sum_{j \in \mathcal{V}} [\mathbf{W}]_{ij} [\boldsymbol{\xi}]_j, \tag{6}$$

where $(\lambda_{\max}, \boldsymbol{\xi})$ is the largest eigenpair of $\mathbf{W}$.

• Ego centrality [28]. Ego centrality can be viewed as a local version of betweenness that computes the shortest paths between its neighboring nodes. Let d_i denote the degree of node i, $\mathbf{W}(i)$ denote the $(d_i + 1) \times (d_i + 1)$ local weight matrix of node i, $\mathbf{I}$ be the identity matrix, and let $\circ$ denote entrywise matrix product. Ego centrality is defined as

$$\text{ego}(i) = \sum_{k \in \mathcal{V}} \sum_{j \in \mathcal{V}, j > k} \frac{1}{[\mathbf{W}^2(i) \circ (\mathbf{I} - \mathbf{W}(i))]_{kj}}. \tag{7}$$

• Local Fiedler Vector Centrality (LFVC) [32]. LFVC is a centrality measure that evaluates the structural importance of a node regarding graph connectivity. Let $\mathbf{y}$ denote the eigenvector associated

with the smallest nonzero eigenvalue of the graph Laplacian matrix. LFVC is defined as

$$\text{LFVC}(i) = \sum_{j \in \mathcal{N}_i} ([\mathbf{y}]_i - [\mathbf{y}]_j)^2, \tag{8}$$

where $\mathcal{N}_i$ is the set of nodes connecting to or from i (i.e., neighbors).

2.3. Graph distances to a set of reference nodes

We propose to use graph distances of each node to a set of reference nodes as structural features that compensate the insufficiency of graph walk statistics and centrality measures when one performs MC-GPCA on graphs with high structural symmetry. For example, consider a star-like graph where the central node is a singleton and each leaf node is an identical clique (i.e., a complete graph). All edges in the graph are undirected and have identical weight. Therefore this graph has high structural symmetry and apparently the nodes of identical structural property (e.g., connected to the central node or not) have the same graph walk statistics and centrality measures. To resolve the ambiguity of graph walk statistics and centrality measures due to high structural symmetry in graphs we use the shortest path distance of each node to the selected r reference nodes as the r additional structural features. In the example of the star-like graph with high structural symmetry, if $r = 1$ then selecting any but the central node as a reference node can yield distinguishable structural features due to difference in shortest path distance to the reference node. The reference nodes are selected according to a user specified criterion, e.g. the nodes of maximal degrees.

3. METHODOLOGY

The extracted centrality features introduced in Sec. 2 can be represented as an $n \times p$ matrix $\mathbf{X}$, where n is the graph size, p is the number of extracted features and each column of $\mathbf{X}$ corresponds to a particular centrality feature that is normalized to have unit norm. The multi-centrality feature matrix $\mathbf{X}$ is then centered by subtracting the row-wise empirical average from each row.

3.1. Multi-centrality graph PCA (MC-GPCA)

In analogy to standard graph PCA, which is applied to the graph Laplacian matrix, MC-GPCA is PCA applied to $\mathbf{X}$. PCA can be formulated as finding an orthonormal transformation $\mathbf{Q}$ on $\mathbf{X}$ such that after transformation the multi-centrality feature matrix $\mathbf{X}$ is represented by an $n \times q$ ($q \leq p$) matrix $\mathbf{Y} = \mathbf{XQ}$ that maximally preserves the total data variance $\text{trace}(\mathbf{Y}^T\mathbf{Y})/n$, where $\text{trace}(\cdot)$ denotes the sum of diagonal entries of a matrix and $\mathbf{Q}$ is a $p \times q$ matrix such that $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$. Such a matrix $\mathbf{Q}$ can be obtained by solving the q right singular vectors associated with the q largest singular values of $\mathbf{X}$, which is denoted by a $p \times q$ matrix $\mathbf{V}_q$. Moreover, the total variance of $\mathbf{Y}$ is equivalent to the sum of the q squared largest singular values of $\mathbf{X}$ divided by n. Therefore using MC-GPCA we obtain n q-dimensional coordinates representing structural scores with respect to the q principal components (i.e., columns of $\mathbf{V}_q$). The algorithm for MC-GPCA is summarized in **Algorithm** 1.

3.2. Structural difference score (SDS)

We use these structural coordinates (i.e., each row of $\mathbf{Y} = \mathbf{XV}_q$) to define a structural difference score (SDS) for each node in a graph. The SDS of node i is associated with the total squared Euclidean distance to its neighboring nodes $\mathcal{N}_i$ and its number of edges (i.e., degree d_i), which is defined as

$$\text{SDS}(i) = \frac{\sum_{j \in \mathcal{N}_i} \|\text{row}_i(\mathbf{Y}) - \text{row}_j(\mathbf{Y})\|^2}{d_i + 1}, \tag{9}$$

where $\text{row}_i(\mathbf{Y})$ denotes the i-th row of $\mathbf{Y}$, $\|\cdot\|$ denotes Euclidean distance, and the denominator $d_i + 1$ is such that the SDS of a singleton node is well-defined.

Algorithm 1 Multi-centrality graph PCA (MC-GPCA)

Input: A graph $G = (\mathcal{V}, \mathcal{E})$, desired dimension q
Output: n structural coordinates $\mathbf{Y}$ for each node in G
1. Extract p structural vectors $\mathbf{X}$ from G
2. Normalize each column of $\mathbf{X}$ to have unit norm
3. Subtract row-wise empirical average from $\mathbf{X}$
4. Solve the right singular vectors $\mathbf{V}_q$ of $\mathbf{X}$
5. $\mathbf{Y} = \mathbf{XV}_q$

Algorithm 2 Multi-centrality graph dictionary learning (MC-GDL)

Input: A set of graphs $\{G_\ell\}_{\ell=1}^g$, number of atoms K, sparsity constraint S, number of highest SDS feature z
Output: graph structure dictionary $\mathbf{D}$, coefficient matrix $\mathbf{C}$
1. Obtain z highest SDS from (9) for each graph as columns of $\mathbf{Z}$
2. Subtract column-wise empirical average from $\mathbf{Z}$
3. Perform K-SVD on $\mathbf{Z}$ to obtain $\mathbf{D}$ and $\mathbf{C}$

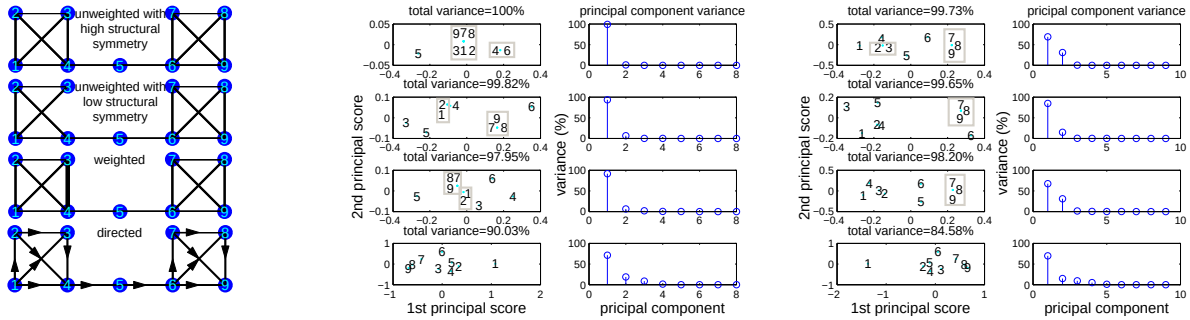
3.3. Multi-centrality graph dictionary learning (MC-GDL)

Consider the case where a set of graphs $\{G_\ell\}_{\ell=1}^g$ is available, each possibly being of different graph size and connectivity pattern, e.g., data from a cyber network at different time instances. Multiple-centrality graph dictionary learning (MC-GDL) is proposed to learn a sparse structure representation of $\{G_\ell\}_{\ell=1}^g$ by finding a dictionary $\mathbf{D}$ consisting of K atoms (columns of $\mathbf{D}$) and an associated sparse coefficient matrix $\mathbf{C} \in \mathbb{R}^{K \times g}$ such that the representation error $\|\mathbf{Z} - \mathbf{DC}\|_F$ is minimized while satisfying the column-wise sparsity constraints on $\mathbf{C}$ that the number of nonzero entries of each column can not exceed a specified value S, where the columns in $\mathbf{Z}$ are structural features of $\{G_\ell\}_{\ell=1}^g$ and $\|\cdot\|_F$ denotes the Frobenius norm. Many different methods exist for solving the dictionary learning problem of estimating $\mathbf{D}$ and $\mathbf{C}$, often called the sparse coding problem [23, 24]. In this paper, we focus on a spectral method (K-SVD) of dictionary learning introduced in [25]. The proposed MC-GDL selects the z highest SDS from each graph as one column of $\mathbf{Z}$ and applies K-SVD to find the dictionary and the corresponding coefficient matrix. The algorithm is summarized in **Algorithm** 2.

4. EXPERIMENTS AND CYBER INTRUSION DETECTION

4.1. Illustration of sensitivity to structural changes on graphs

Here we consider four similar graphs with different structural characteristics as displayed in Fig. 1 (a). From top to bottom, these four graphs represent high structural symmetry, reduced structural symmetry due to edge removal, increase of the weight of edge (3,4), and change in edge direction. The extracted multi-centrality features are 1) graph walk statistics from 1 to 4 hops, and 2) the graph distance to node 1 (the reference node). It can be observed from Fig. 1 (b)



(a) Four example graphs (b) MC-GPCA without reference nodes (c) MC-GPCA with one reference node

Fig. 1: Illustration of sensitivity of proposed MC-GPCA algorithm to structural perturbations. Each node on a graph is represented by a 2-dimensional structural coordinate. Nodes marked by a gray box have identical MC-GPCA score.

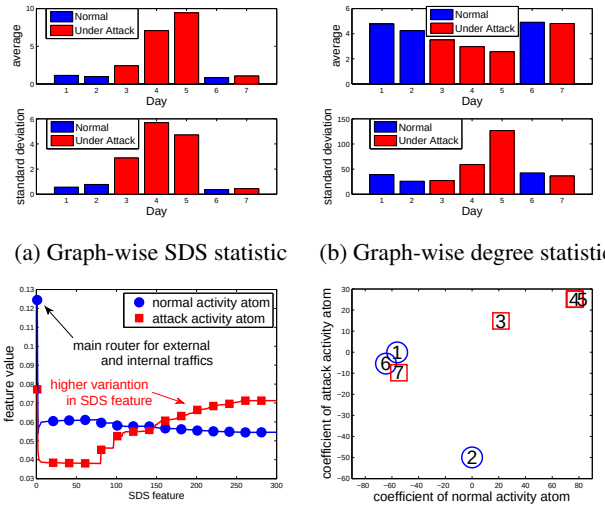
Table 2: Description of the University of New Brunswick (UNB) Intrusion Detection Evaluation Dataset [33]

Dataset	# nodes	# edges	Description
Day 1	5357	12887	Normal activity
Day 2	2631	5614	Normal activity
Day 3	3052	5406	Infiltrating attack and normal activity
Day 4	8221	12594	HTTP denial of service attack and normal activity
Day 5	24062	32848	Distributed denial of service attack using Botnet
Day 6	5638	13958	Normal activity
Day 7	4738	11492	Brute force SSH attack and normal activity

that MC-GPCA can reflect structural perturbations, and total data variance is explained by one or two principal components. Moreover, the first principal component is shown to completer describe the network flow pattern for the directed example graph. Fig. 1 (c) shows that the graph distance feature adds discrimination power as the MC-GPCA scores are better differentiated.

4.2. Cyber intrusion detection

The UNB intrusion detection evaluation dataset [33] described in Table 2 is a collection of directed cyber network graphs where each node is a host (machine) in a cyber system and an edge indicates the existence of communication between hosts. No information beyond graph topology is used for analysis. The extracted multi-centrality features are 1) graph walk statistics from 1 to 20 hops, 2) all centrality measures introduced in Sec. 2.2 (edge directions are omitted for computing LFVC), and 3) graph distances to 10 reference nodes of highest degree, resulting in $p = 56$ features (columns of $\mathbf{X}$). Fig. 2 (a) shows that the proposed SDS statistic (Eqn. (9)) with $q = 2$ principal components from MC-GPCA. The SDS statistics are similar over days without attacks, whereas they are significantly higher in days under attacks that induce anomalous connectivity patterns (i.e. Days 3, 4 and 5). On the other hand degree statistic (Fig. 2 (b)) fails to be a valid indicator of cyber attacks. The SDS statistic fails to detect the SSH attack (Day 7) since it is a password attack that takes place only between a single host and a single server.



(a) Graph-wise SDS statistic (b) Graph-wise degree statistic

(c) Dictionary from MC-GDL (d) Coefficients from MC-GDL

Fig. 2: Cyber intrusion detection on the UNB dataset. MC-GPCA and MC-GDL are shown to be effective indicators of cyber attacks.

We applied MC-GDL to the entire UNB database of graphs to learn a dictionary that spans the dataset. For this implementation of MC-GDL we select $K = 2$ atoms, $z = 300$ SDS features and $S = 2$ sparsity level. The two learned structural atoms in Fig. 2 (c) can be interpreted as a normal activity atom consisting of identical SDS features except for one spike accounting for the main router and an attack activity atom of higher variance in SDS features. The corresponding coefficients in Fig. 2 (d) reflect the mixture portion of these atoms and they can be used for attack classification. For instance, K-means clustering with 2 clusters identifies Days 3, 4 and 5 as being anomalous and thus under attack.

5. CONCLUSION

This paper proposes PCA and dictionary learning graph decomposition methods that are based on multi-centrality features of the graph. The proposed methods can reflect structural perturbations in graph symmetry, edge weight and edge direction. When applied to cyber intrusion detection, our experiments show that MC-GPCA and MC-GDL can effectively detect attacks on the network.

6. REFERENCES

[1] M. E. J. Newman, *Networks: An Introduction*. Oxford University Press, Inc., 2010.

[2] C. C. Noble and D. J. Cook, "Graph-based anomaly detection," in *ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003, pp. 631–636.

[3] E. Hogan, P. Hui, S. Choudhury, M. Halappanavar, K. Oler, and C. Joslyn, "Towards a multiscale approach to cybersecurity modeling," in *IEEE International Conference on Technologies for Homeland Security (HST)*, 2013, pp. 80–85.

[4] P.-Y. Chen and A. O. Hero, "Assessing and safeguarding network resilience to nodal attacks," *IEEE Commun. Mag.*, vol. 52, no. 11, pp. 138–143, Nov. 2014.

[5] C. Joslyn, S. Choudhury, D. Haglin, B. Howe, B. Nickless, and B. Olsen, "Massive scale cyber traffic analysis: A driver for graph database research," in *International Workshop on Graph Data Management Experiences and Systems (GRADES)*, 2013, pp. 3:1–3:6.

[6] P.-Y. Chen and A. Hero, "Phase transitions in spectral community detection," *IEEE Trans. Signal Process.*, vol. 63, no. 16, pp. 4339–4347, Aug 2015.

[7] L. Akoglu, H. Tong, and D. Koutra, "Graph based anomaly detection and description: a survey," *Data Mining and Knowledge Discovery*, vol. 29, no. 3, pp. 626–688, 2015.

[8] B. Miller, M. Beard, P. Wolfe, and N. Bliss, "A spectral framework for anomalous subgraph detection," *IEEE Trans. Signal Process.*, vol. 63, no. 16, pp. 4191–4206, Aug. 2015.

[9] K. Oler and S. Choudhury, "Graph based role mining techniques for cyber security," in *FloCon*, 2015.

[10] D. Shuman, S. 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 Process. Mag.*, vol. 30, no. 3, pp. 83–98, 2013.

[11] A. Bertrand and M. Moonen, "Seeing the bigger picture: How nodes can learn their place within a complex ad hoc network topology," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 71–82, 2013.

[12] A. Anis, A. Gadde, and A. Ortega, "Towards a sampling theorem for signals on arbitrary graphs," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 3864–3868.

[13] X. Wang, P. Liu, and Y. Gu, "Local-set-based graph signal reconstruction," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2432–2444, May 2015.

[14] S. Chen, A. Sandryhaila, J. Moura, and J. Kovacevic, "Signal recovery on graphs: Variation minimization," *IEEE Trans. Signal Process.*, vol. 63, no. 17, pp. 4609–4624, Sept. 2015.

[15] M. Saerens, F. Fouss, L. Yen, and P. Dupont, "The principal components analysis of a graph, and its relationships to spectral clustering," in *Machine Learning: ECML*. Springer, 2004, pp. 371–383.

[16] B. Jiang, C. Ding, B. Luo, and J. Tang, "Graph-laplacian PCA: Closed-form solution and robustness," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 3492–3498.

[17] N. Shahid, V. Kalofolias, X. Bresson, M. M. Bronstein, and P. Vandergheynst, "Robust principal component analysis on graphs," *CoRR*, vol. abs/1504.06151, 2015. [Online]. Available: http://arxiv.org/abs/1504.06151

[18] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," in *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4, 2004, pp. 219–230.

[19] J. Terrell, K. Jeffay, F. D. Smith, L. Zhang, H. Shen, Z. Zhu, and A. Nobel, "Multivariate SVD analyses for network anomaly detection," in *ACM SIGCOMM Conference Poster Session*, 2005.

[20] D. Thanou, D. Shuman, and P. Frossard, "Learning parametric dictionaries for signals on graphs," *IEEE Trans. Signal Process.*, vol. 62, no. 15, pp. 3849–3862, Aug. 2014.

[21] X. Zhang, X. Dong, and P. Frossard, "Learning of structured graph dictionaries," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2012, pp. 3373–3376.

[22] D. Thanou and P. Frossard, "Multi-graph learning of spectral graph dictionaries," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015, pp. 3397–3401.

[23] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Advances in neural information processing systems (NIPS)*, 2006, pp. 801–808.

[24] R. Jenatton, J. Mairal, F. R. Bach, and G. R. Obozinski, "Proximal methods for sparse hierarchical dictionar learning," in *International Conference on Machine Learning (ICML)*, 2010, pp. 487–494.

[25] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[26] L. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, vol. 40, pp. 35–41, 1977.

[27] G. Sabidussi, "The centrality index of a graph," *Psychometrika*, vol. 31, no. 4, pp. 581–603, 1966.

[28] M. Everett and S. P. Borgatti, "Ego network betweenness," *Social Networks*, vol. 27, no. 1, pp. 31–38, 2005.

[29] L. Lovász, "Random walks on graphs: A survey," *Combinatorics, Paul erdos is eighty*, vol. 2, no. 1, pp. 1–46, 1993.

[30] M. Gomez Rodriguez, J. Leskovec, and A. Krause, "Inferring networks of diffusion and influence," in *ACM SIGKDD international conference on Knowledge discovery and data mining*, 2010, pp. 1019–1028.

[31] L. Galluccio, O. Michel, P. Comon, M. Kliger, and A. O. Hero, "Clustering with a new distance measure based on a dual-rooted tree," *Information Sciences*, vol. 251, pp. 96–113, 2013.

[32] P.-Y. Chen and A. O. Hero, "Local Fiedler vector centrality for detection of deep and overlapping communities in networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1120–1124.

[33] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, "Toward developing a systematic approach to generate benchmark datasets for intrusion detection," *Computers & Security*, vol. 31, no. 3, pp. 357–374, 2012.