Volume: 08, Issue: 05 Pages: 128-130 (2017) Special Issue

A Study of Anomaly Detection in Bipartite Graph

K. Punitha
Department of Computer Science, Government Arts College for Women, Ramanathapuram
Email: punimsc@gmail.com

ABSTRACT

Many real applications can be modeled using bipartite graphs, such as users vs. files, traders vs. stocks, conferences vs. authors, and so on. Bipartite graph perform the operation finding similar nodes (Neighborhood Formation) and abnormal nodes (Anomaly detection). To propose algorithms to compute graph partitioning and also propose algorithms to identify abnormal nodes, using normality scores (ns) based on relevance scores (rs). Evaluate the quality of the datasets, and also measure the performance of the anomaly detection algorithm with manually injected anomalies. Effectiveness and efficiency of the method are confirmed by experiments on several real datasets.

Keywords – **Neighborhood Formation, Anomaly Detection, Bipartite Graph, Matrix Representation, Relevance Scores (rs), Normality Scores (ns).**

1. INTRODUCTION

Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. These nonconforming patterns are often referred to as anomalies, outliers, discordant observations, exceptions, aberrations, surprises, peculiarities, or contaminants in different application domains. Of these, anomalies and outliers are two terms used most commonly in the context of anomaly detection; sometimes interchangeably. Anomaly detection finds extensive use in a wide variety of applications such as fraud detection for credit cards, insurance, or health care, intrusion detection for cyber-security, fault detection in safety critical systems, and military surveillance for enemy activities.

1.1. TYPES OF ANOMALY

1.1.1. Point Anomalies: If an individual data instance can be considered as anomalous with respect to the rest of data, then the instance is termed a point anomaly.

1.1.2. Contextual Anomalies: If a data instance is anomalous in a specific context, but not otherwise, then it is termed a contextual anomaly also referred to as conditional anomaly.

1.1.3. Collective Anomalies: If a collection of related data instances is anomalous with respect to the entire data set, it is termed a collective anomaly.

2. BIPARTITE GRAPH REPRESENTATION

A bipartite graph is a graph where nodes can be divided into two groups V1 and V2 such that no edge connects the vertices in the same group. More formally, a bipartite graph. G is defined as G = (V,E), where V = V1 U V2, V1 = $\{a_i | 1 \le i \le n\}$ and V1 = $\{b_j | 1 \le j \le k\}$, E = V1 × V2 as shown in Figure 1.

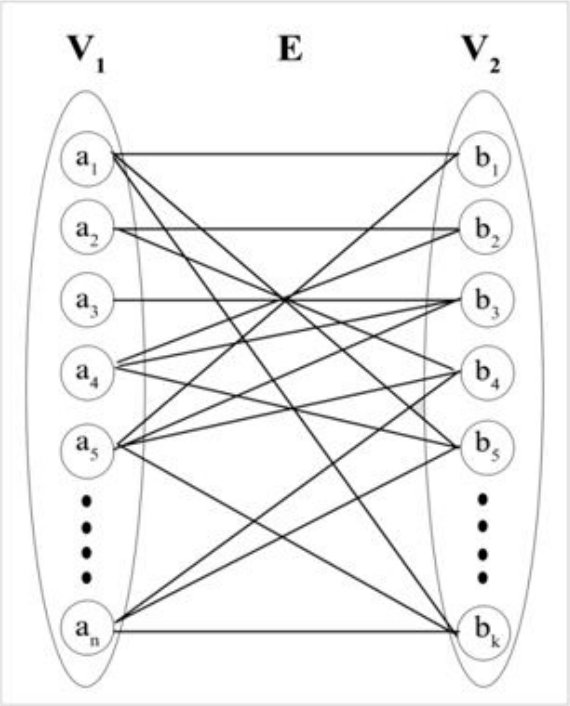

Figure 1. Bipartite Graph

2.1. NEIGHBORHOOD FORMATION (NF)

Given a query node 'a' in V1, NF computes the relevance scores of all the nodes in V1 to 'a'. The ones with higher relevance are the "neighbors" of 'a'.

2.1.1. TYPES OF NEIGHBORHOODS

(i) Hard Neighborhood: select a set of nodes as the neighbors and the other nodes are not the neighbors.

(ii)Soft Neighborhood: assign a relevance score to every node where "closer" nodes have high scores, and no hard boundary exists (Soft Neighborhood).

2.2. ANOMALY DETECTION (AD)

Given a query node 'a' in V1, AD computes the normality scores for nodes in V2 that link to 'a'. A node with a low normality score is an anomaly to a. More intuitively, they are the persons who published in different fields.

Nodes that belong to the same group (V1 or V2) have the same type; it is the connections between the two types of objects that hold the key to mining the bipartite graph. Given the natural inter-group connections (betweenV1 andV2), our objective is to discover the intra-group relationships, such as the clusters and outlier within the group. For example, in the research publication bipartite graph, we have two natural groups of entities: conferences and authors. The relationship between these two groups is reflected by the edges. The bipartite graph symbols are represented in Table 1.

Symbol	Description
V_1	set of n row nodes
V_2	set of k column nodes
M	the n-by-k bipartite matrix
M^T	transpose of m
M_A	the (k+n)-by-(k+n) adjacent matrix
P_A	the (k+n)-by-(k+n) transition matrix
rs(a)	1-by-k relevance score vector for a belongs to V1
ns(t)	the normality score of the column node t belongs to V2
St	the set of row nodes linking to t

Table 1. Symbol Table

3. MATRIX REPRESENTATION

Bipartite graph. G is defined as G = (V,E), where V = V1 U V2, V1 = $\{a_i | 1 \le i \le n\}$ and V1 = $\{b_j | 1 \le j \le k\}$, E = V1 × V2. The graph G is conceptually stored in a n-by-k matrix M, where M(i, j) is the weight of the edge.

$$M_{nxk} = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ . \\ . \\ . \\ a_n \end{array} \begin{array}{c} \begin{array}{ccccccccc} b_1 & b_2 & b_3 & b_4 & b_5 & ..... & b_k \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & ..... & 1 \\ 0 & 1 & 0 & 0 & 1 & ..... & 0 \\ 0 & 0 & 1 & 0 & 0 & ..... & 0 \\ 1 & 0 & 1 & 0 & 1 & ..... & 0 \\ 1 & 0 & 1 & 1 & 0 & ..... & 1 \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & 1 & 1 & ..... & 1 \end{pmatrix} \end{array}$$

4. RELEVANCE SCORES (rs)

Compute the relevance scores use a query node 'r' in V1. Nearest node of 'r' has secure the higher relevance score. Nodes that are far away from 'r' have almost 'zero' relevance score. Relevance Score represented in Figure 2.

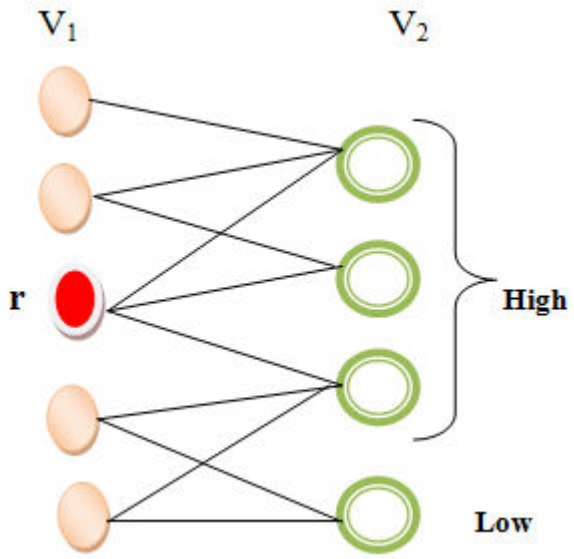

Figure 2. Relevance Scores (rs)

From figure 2 the relevance score based on high probability which means nearest and visited repeated the same node.

5. NORMALITY SCORES (NS)

Compute the normality scores use a query node 'a' in V1 that connect to the node V2. Normality Scores represented in Figure 3.

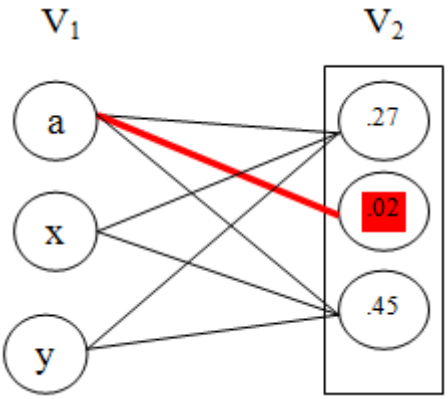

Figure 3. Normality Scores (ns)

From figure 3 V2 nodes are all contain normality scores. Node with low normality score is detect as an anomaly (it is highlighted).

A column node 't' link to the row nodes i.e. 't' belongs to V2. High normality and low normality clearly represented in Figure 4 and Figure 5.

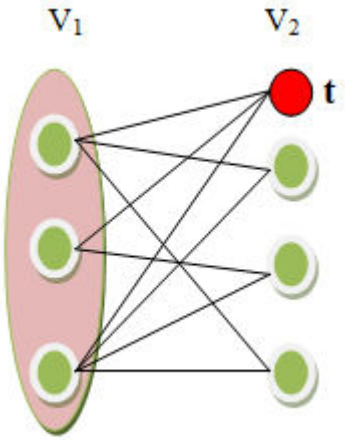

Figure 4. High Normality

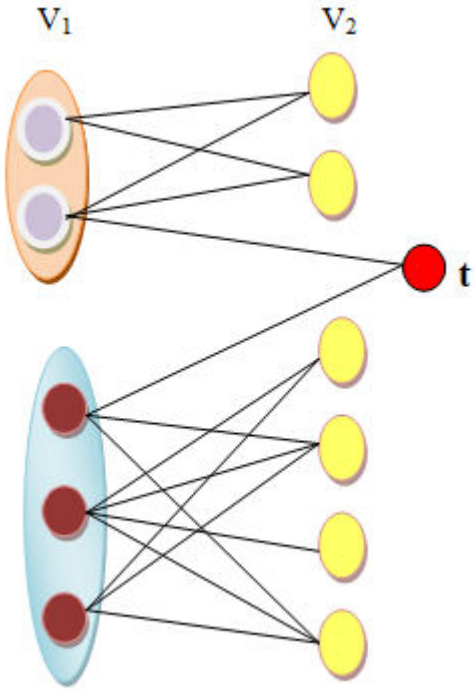

Figure 5. Low Normality

From figure 4 column node 't'(anomaly node) link with all other row nodes(maximally connected) is referred as High Normality. In figure 5 the row nodes are split as two parts column node 't'(anomaly node) connect as minimally(only one link from two parts of row nodes) is referred as Low Normality.

6. ALGORITHM FOR ANOMALY DETECTION

The relevance scores for the nodes are very skewed, with most nodes have almost zero relevance scores, and only a few nodes having high scores. This suggests that we can possibly filter out many "irrelevant" nodes before applying the NF computation.

6.1 Algorithm(Neighborhood Formation)

Input: the bipartite graph G, the number of partitions k, input node 'r'

Procedure for Computation :

Step 1: divide G into k partitions G_1, G_2 ,…., G_k.

Step 2: find the partition G_i containing 'r'.(i=1,2,…,k).

Step 3: construct bipartite matrix.

Step 4: apply neighborhood formation on 'r' in bipartite matrix.

Step 5: set zero relevance scores for the nodes that are not in G_i.

Based on the relevance score compute the normality scores for the nodes in V2. A node with a low normality score is an anomaly. Given a column node t belongs to V2, find the set of row nodes (St) to link with n node. St = {a |< a, t > belongs to E}(In figure 1 t= bj, (j = 1,2,3,....,k)).

6.2 Algorithm(Anomaly Detection)

Input: input node t belongs to V_2.

Procedure for Computation :

Step 1: find the set St = {a_1, a_2 ,……}. St means set of row nodes belongs to V_1 that link with t belongs to V_2.

Step 2: construct the relevance score based on NF.

Step 3: construct the normality score(ns(t)) based on relevance scores.

Step 4: return ns(t).

Step 5: finally detect the anomaly based on normality score ns(t).

7. CONCLUSION

In Bipartite Graph an efficient method NF using relevance score (rs) for graph partitioning techniques to spot anomalies (AD) using Normality Scores (ns). The main properties of the methods are Fast convergence, Scalability to large graphs, Simplicity of implementation and Results that are easily interpreted. Both NF and AD are efficient as well as the effectiveness of the proposed methods.

REFERENCES

[1] J. Sun, H. Qu, D. Chakrabarti, and C. Faloutsos, "Neighborhood formation and anomaly detection in bipartite graphs," in ICDM Conf., 2005.

[2] S.d. Lin and H. Chalupsky, "Discovering and explaining abnormal nodes in semantic graphs," IEEE Trans. on Knowl. and Data Eng., vol. 20(8), 2008.

[3] G. Wang, S. Xie, B. Liu, and P. S. Yu, "Review graph based online store review spammer detection," in ICDM Conf., 2011.

[4] C. C. Noble and D. J. Cook, "Graph-based anomaly detection," in KDD Conf., 2003.

[5] C. Aggarwal and P. Yu. Outlier detection for high dimensional data. In *SIGMOD*, pages 37–46, 2001.

[6] C. D. Meyer, Matrix Analysis and Applied Linear Algebra Book and Solutions Manual. SIAM: Society for Industrial and Applied Mathematics, 2001.

[7] Deepayan Chakrabarti. Autopart: Parameter-free graph partitioning and outlier detection. In *PKDD*,pages 112–124, 2004.