The Thirty-Third International
FLAIRS Conference (FLAIRS-33)

Node Similarity for Anomaly Detection in Attributed Graphs

Prajjwal Kandel,[1] **William Eberle**[2]
Tennessee Technological University, Cookeville, TN
prkandel42@students.tntech.edu[1], weberle@tntech.edu[2]

Abstract

Most graph-based anomaly detection work uses structural graph connectivity or node information for discovering anomalies in a graph. Approaches solely relying on node information for detecting anomalies do not exploit the structural information, and approaches relying on just the structural connectivity information do not exploit node label values, or attribute information. In this work, in order to preserve the closeness information of numeric node attributes, we consider the similarities in node values using not only single attributes, but also multiple attributes. In order to discover the similarity between the attribute values, we use a discretization method, distance-based similarity measures, and a k-means clustering approach. After discovering nodes with similar label values, we use revised labels together with structural properties for discovering anomalies in a graph. Our hypothesis is that if we use node label similarity information together with structural properties of the graph, we can detect anomalies which would be missed by approaches only relying on either structural connectivity or node attribute information. Experimental results on real world as well as synthetic datasets show that the proposed approach is able to detect both structural and attribute anomalies. We also compare the results of the proposed approach with an existing structural anomaly detection tool and show that the proposed approach can detect anomalies where traditional structural techniques cannot.

Introduction

Most existing graph-based anomaly detection work either uses the structural graph properties or the node information. The structural approaches focus on the connectivity of the graph and explores frequent substructures to discover deviations from normal substructures. While these approaches use the relationships between entities for structural oddities, they do not incorporate the entities' attributes information, which is also an important part of the data. The approaches which use node information for anomaly detection tend to form communities in the graph based on the node attributes. Then they identify anomalies as a by-product of community formation. There has been some recent research in the area of unifying structural information with node attribute information (Yang et al. 2009; Muller et al. 2013), however, they mostly use the information for forming communities and thus finding community outliers while still missing the structural anomalies.

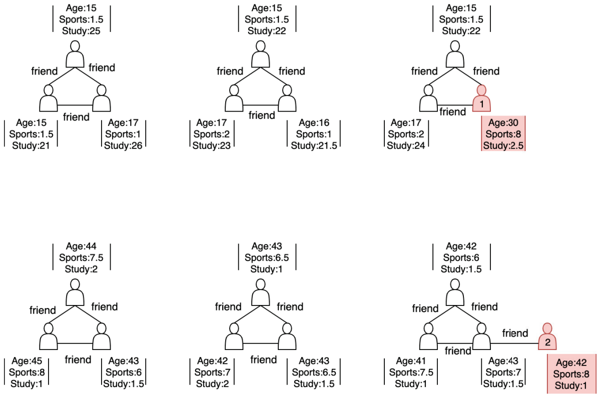

Figure 1: An example graph of members of a fitness center

Figure 1 demonstrates an example graph of groups of people that are members of a school fitness center. The attributes *age*, *sports*, and *study* are the age of the person, average weekly hours spent by the person in sports, and average weekly hours spent by the person in study (as if they are a student) respectively. First, notice the anomaly labelled with a 1. The attribute values of this node are not unusual themselves, but they are different from it's neighbor nodes. Anomaly detection approaches which use node attribute information are able to detect anomaly 1 because it's attribute values deviate from it's neighbor nodes. Approaches using structural approaches are not able to discover this anomaly because they only use structural connectivity information and do not exploit node attribute values. Anomaly 1 is not structurally unusual, and thus not detected by structural approaches. Second, notice the structural anomaly labelled with a 2. The attribute values of this anomalous node is similar to that of it's connected node. Thus, approaches relying on solely node attribute information are not able to discover this structural anomaly (i.e., the presence of an extra node).

In this work, we focus on discovering anomalies in *attributed* graphs, combining node attribute information with the structure of the graph. Existing structure based methods attempt to discover substructures in the graph which are connectivity-wise rare. While doing this, they treat all attribute values as discrete, and thus, when the attribute are numeric values, they lose their measure of similarity, or closeness (Akoglu, Tong, and Koutra 2015). Previous work has attempted to preserve the closeness information (Eberle and Holder 2009; Davis et al. 2011), but they do not take into account *multiple attributes*, and the attribute values which individually perhaps are normal, but when taken in combination are in fact rare.

In order to achieve this, the first step is to discover the similarity between attribute values of nodes in the graph. Then, we label nodes whose attributes are discovered as similar with identical labels. This new label is then representative of all the original attributes present in the graph's nodes. Finally, we create a new graph using these revised node labels from the original graph, allowing us to then apply structural anomaly detection on the new graph.

Related Work

In literature, most of the work in anomaly detection on attributed graphs are based on community detection. Community based methods tend to detect communities in graphs and detect anomalies as a by-product. (Gao et al. 2010) partition graph nodes into clusters and then use the object information to discover outliers. (Muller et al. 2013) propose an approach which ranks graph nodes according to their deviation in both graph and attribute properties. However, this work only considers outlier nodes, and does not consider anomalous edges, or irregular subgraphs. ConSub (Sánchez et al. 2013) and ConOut (Sánchez et al. 2014) extract important attributes from the graph for detecting community outliers. (Boden et al. 2012) and (Shah et al. 2016) use edge attribute information instead of node attribute information for detecting anomalous nodes.

There are some structural anomaly detection techniques which exploit subgraph patterns to spot anomalies. (Noble and Cook 2003) use SUBDUE for anomaly detection whose main intuition is to look for the structures that occur infrequently in the graph. (Eberle and Holder 2007) hypothesize anomalies as an unexpected deviation to a normative pattern. A normative pattern is the substructure that compresses the graph the best. They formulate three types of anomalies based on modifications, insertions, and deletions to the graph. Structural anomaly detection approaches treat numeric attributes as discrete values. (Eberle and Holder 2009) extend their approach to detect anomalies in graphs with continuous labels by including the numeric values into a probability calculation. However, this work does not handle anomalies involving multiple numeric attributes which individually are not anomalous, but together are rare. (Romero, Gonzalez, and Holder 2010) propose a numerical range generation algorithm based on frequency histograms. After finding the suitable ranges for numeric labels, they use SUBDUE for the task of anomaly detection. (Davis et al. 2011) propose discretizing the numerical attributes before running SUBDUE for the task of anomaly detection. However, they experiment with graph data containing only a single numeric attribute. (Ramesh Paudel and Talbert 2017; Ramesh Paudel and Holder 2018) are other works which use discretization of numeric attributes as a pre-processing step before using a graph based anomaly detection tool.

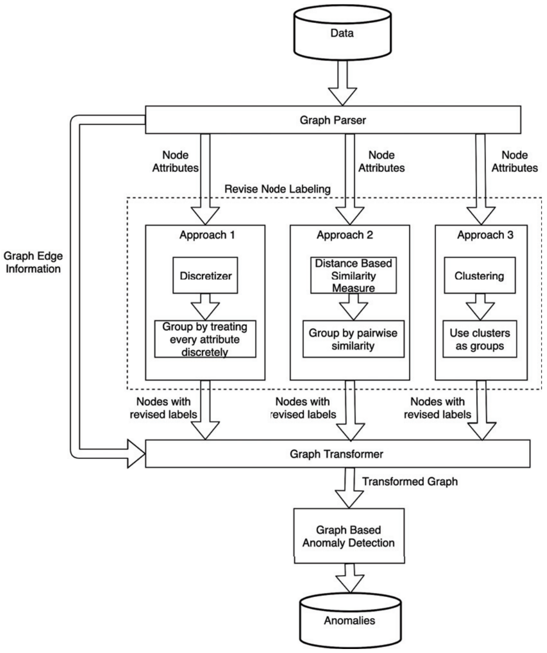

Figure 2: Proposed Architecture

Methodology

The architecture of our proposed approach is shown in Figure 2. First, the input dataset is parsed into graph objects. Second, we discover the similarity between attribute values of nodes in the graph. Third, we label nodes whose attributes are discovered as similar with identical labels. This new label is representative of all the original attributes present in the nodes. Then, we create a new graph using just the newly added labels in lieu of the original node labels, and edge information from the original graph. Finally, we run an anomaly detection tool on this transformed graph. To discover the similarity between attribute values of nodes, we implement three different approaches. Each of the approaches is described in Node Labeling Reviser subsections.

Node Labeling Reviser

We attempt to discover the similarity between graph nodes based on their attribute values so that nodes with similar attribute values are treated as similar when discovering normative patterns and anomalous graph substructures. The closer the attribute values between two nodes, the more similar the nodes. As is the case in the example graph shown in Figure 1, often times, there are multiple attributes associated with a node. Thus, we need to handle *multiple* attributes to discover the similarity between node objects.

We implement three different approaches for determining the similarity of nodes, and then label the nodes that are deemed to be similar with identical labels.

Discretization We *discretize* all the numeric attributes of nodes using the unsupervised discretization technique described in (Biba et al. 2007). This approach uses non-parametric density estimators to automatically adapt sub-interval dimensions to the data. The algorithm searches for the next two sub-intervals, evaluating the best cut-point on the basis of the density induced in the sub-intervals by the current cut and the density given by the kernel density estimator for each sub-interval. After discretizing numeric attributes, we can then compare each attribute. While comparing two nodes, if all the discrete attribute values of both nodes are the same, we consider such nodes as similar, and label them with single, identical labels. For example, say a node A has attribute values $x1$, $y1$, and $z1$, and another node B has attribute values $x2$, $y2$, and $z2$. After discretizing each attribute, say the discrete labels for node A are x_a, y_a, and z_a, and the discrete labels for node B are x_b, y_b, and z_b. We can now make a one-on-one comparison between attributes, and both nodes A and B will get the same single label if $x_a = x_b$, $y_a = y_b$, and $z_a = z_b$.

Distance Based Similarity Measure This approach uses a distance-based similarity measure to discover the similarity between nodes with numeric attributes. We use a normalized euclidean distance for finding the similarity score between node objects. Since the higher the distance the more dissimilar the objects, we use the inverse of the distance as the similarity score between the objects.

$$Similarity\ Score = \frac{1}{Normalized\ Euclidean\ Distance}$$

Our goal is to find similar entities based on the similarity score between each pair of nodes. We use the grouping algorithm defined by (Caceres 2013) for this purpose. In this algorithm, we sort the entity pairs according to the similarity score from greatest to least. We then use this sorted list of entity pairs for forming suitable groups of entities.

If the comparison score between two entities is within a given threshold, a group containing the two entities is created. Afterwards, for each pair, if the comparison score is within the given threshold, and both entities are not present in any existing group, a new group is formed containing the two entities. If the entities are in different groups, it is checked if the similarity score for all pairs for both groups is within the threshold. If all pairs for both groups are within the threshold, the two groups are merged into a single group. Similarly, if only one of the entities in the pair is in a group, then it is checked whether the similarity score between the ungrouped entity and all the entities existing in the group is within the given threshold. If that is the case, then the ungrouped entity is added to the group. In the end, a singleton group is created for each ungrouped entity.

After we form groups, we label all the nodes falling within a group with an identical label. This new label is used by the next component (i.e., the Graph Transformer) to transform the graph. While we used normalized euclidean distance as similarity metric, the approach does not prohibit other approaches, like cosine similarity, from being used.

Clustering In order to discover similar nodes based on the attribute information, we use an unsupervised, k-means clustering algorithm, with the number of clusters based upon the visual elbow method. The idea is to start with k=2, and keep increasing the value of k by 1, discovering the clusters and the sum of squared errors. At some value of k, the sum of squared error drops significantly, and reaches a plateau by increasing it further, leaving us with a chosen value for k. Once we have clusters, we label all the nodes falling within a cluster with an identical label.

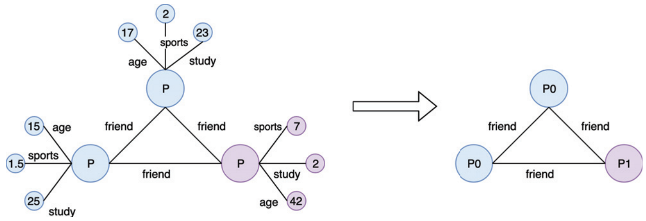

Figure 3: Graph Transformation Example

Graph Transformer

Finally, we create a new graph with revised, single node labels, and the edge labels from the original graph.

Figure 3 shows an example of a transformed graph. The nodes with similar attribute values are given the same label from the Node Label Reviser component. The graph transformer then creates a new graph with the revised node label and edges from the original graph. The anomaly detection tool operates on the transformed graphs. Since the graph transformer creates a new graph with single node labels, some node attribute information from the original graph is lost. However, the label helps to identify similar nodes in the graph.

Graph-Based Anomaly Detection

An advantage of graph-based anomaly detection is that it is able to identify structural oddities by analyzing relationships between entities. To analyze relationships between entities, structural graph based anomaly detection methods examine the graph structure and exploit patterns to discover anomalies. The purpose of a structural anomaly detection approach is to identify infrequent substructures in the graph or find substructures which are deviating from a frequent substructure in the graph. For more information about graph-based anomaly detection, the reader can refer to Akoglu et al. (Akoglu, Tong, and Koutra 2015).

In order to test our approach, we will use the publicly available GBAD test suite[1]. It should be noted that while GBAD is used for evaluation, the proposed approach has been designed to work with any other graph based anomaly detection tool. The design of the proposed approach does not limit it to be just used with GBAD.

Data

In order to evaluate the performance of the proposed approach, we experiment with three synthetic datasets varying in size, and two real world datasets from different domains.

[1] http://www.gbad.info/

Synthetic Dataset

For our synthetically generated dataset, we use an artificial graph generation tool subgen[2]. Through the subgen tool, we specify the number of vertices and edges, and a normative substructure of the graph to be generated. We experiment with three different synthetic datasets varying in size. The normative substructure for these synthetic graphs contain 20 nodes and 20 edges. The three datasets consist of 5,020 nodes and 5,020 edges, 100,400 nodes and 100,400 edges, and, 401,600 nodes and 401,600 edges respectively. For each of the datasets, we inject different types of anomalies like random node insertions, the random removal of nodes, and the random modification of attribute values. A visualization of a normative substructure in the graph is shown in Figure 4.

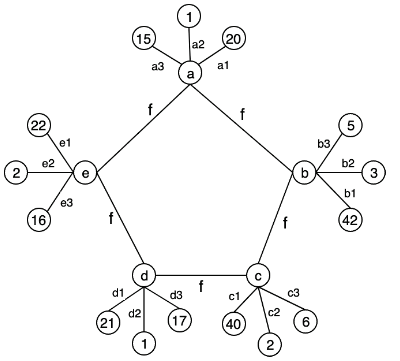

Figure 4: Basic Substructure of Synthetic Graph

Amazon Co-purchase Dataset

The Amazon Co-purchase Network (Muller et al. 2013) is a publicly available benchmark dataset for community outlier detection. This dataset is a co-purchase network of 124 Disney movies available on Amazon. This dataset consists of 124 nodes and 335 edges, where each node contains 30 attributes like price, review, rating, etc. The anomalies in this dataset are products with high prices, unusual reviews, unusual ratings, etc. This benchmark dataset consists of six known anomalies. Among the six anomalies, three of them are anomalies because of their high prices. Two of them are anomalies because of unusual reviews and poor ratings, and one is an anomaly because of low price.

Enron Email Dataset

The Enron Corpus Dataset (Klimt and Yang 2004) consists of over 500,000 emails generated by 150 employees of the Enron Corporation. We use a MySQl database dump [3] of the Enron Corpus Dataset for our experiments. While graph based approaches have known complexities in terms of time and memory, the purpose of this work is to use multiple node attribute information together with structural properties of the graph to detect anomalies. Thus, we pulled a random sample of about 17,000 emails from the MySQL dump to evaluate our approach. We then parsed the email bodies in order to find the number of attachments in each email. Finally, we created a graph whose (partial) structure is shown in Figure 5. The graph contains 98,740 vertices, and 93,239 edges.

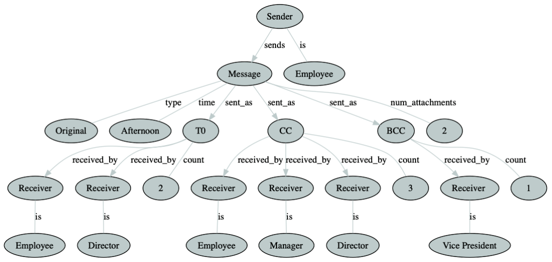

Figure 5: Graph Representation of Enron Email

Results and Evaluation

In this section, we discuss the results obtained from our experiments and evaluate the performance of the proposed approach.

Results

Synthetic Dataset We experimented with three different synthetic datasets varying in size but having the same basic substructure. We ran 20 experiments for each approach on all three datasets. On average, the discretization based approach was able to detect 5 out of 10 injected attribute anomalies. Similarly, the distance-based similarity measure approach was able to detect 8 out of 10 injected attribute anomalies, and the clustering-based approach was able to detect 7 out of 10 injected attribute anomalies. All three approaches were able to detect all three types of injected structural anomalies.

Amazon Co-purchase Dataset In this dataset, we used price, number of reviews, and average rating of the Disney movies as attributes. While we did not run any experiments for feature selection, the choice of the attributes are based on the features highlighted in literature (Sánchez et al. 2014).

By running GBAD on the transformed graph obtained by using the distance-based similarity measure, we are able to discover anomalies whose price deviate significantly from the co-purchased products. While we are able to detect 4 out of 6 known outliers, 4 products are also reported as false positives. The products obtained as false positives are because these products had higher prices than their co-purchased products but are not as expensive as the ones in the given outliers. The normative pattern and anomaly detected are shown in Figure 6. The attribute values shown in the Figure 6 are the average values of attributes for the revised labels. Both discretization- and clustering-based approaches, however, are only able to detect 2 outliers and report 6 false positives.

Enron Email Dataset In this dataset, all three approaches reported the same structural anomalies as well as the anoma-

[2] http://ailab.wsu.edu/subdue/datasets/subgen.tar.gz

[3] http://www.ahschulz.de/enron-email-data/

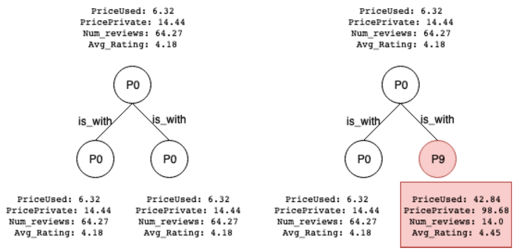

Figure 6: Normative pattern(left) and Anomalous Instance(right) in Amazon Co-purchase Dataset

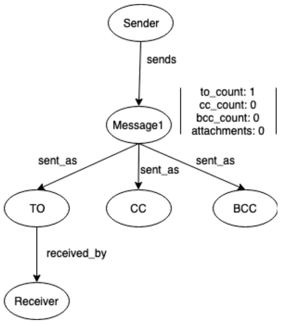

Figure 7: Normative Pattern of Enron Graph

lies based on attribute values. The normative pattern for the Enron email graph is shown in Figure 7.

By using the normative pattern as shown in Figure 7, the two types of anomalies detected by all three proposed approaches are shown in Figure 8. Both anomalies obtained are unusual communication behaviours. The one on the left of Figure 8, shows an email with an attribute anomaly of an unusually high number of attachments. Since we parsed the email bodies to count the number of attachments, we also discovered that those 57 attachments were all ".exe" files. Email communication with such a high number of ".exe" files are unusual in nature.

Similarly, the one on the right of the Figure 8 shows an email communication between the employees at an unusual time. It shows an email being sent by the "Vice President" to the "In House Lawyer" late at night, around three in the morning. That is an unusual time to send a professional email, and given what we know happened with Enron, that is certainly suspicious.

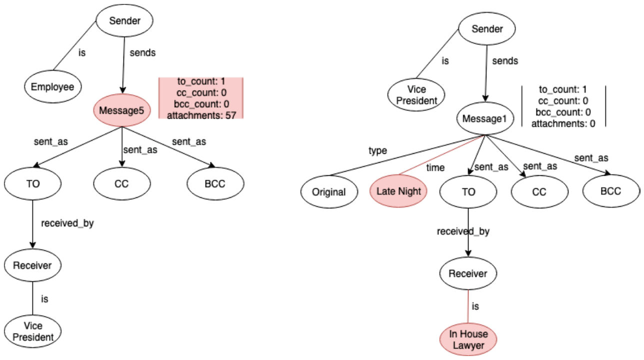

Figure 8: Attribute Anomaly (left) and Structural Anomaly (right) in Email Communication

Evaluation

Comparison with GBAD We compare the results obtained from the proposed approach with the results obtained by running the publicly available GBAD tool (Eberle and Holder 2007) on the original graph.

On all of the three datasets, the results obtained from the proposed approach was better than running GBAD on the original graph both in terms of recall and precision. In the synthetic datasets and Amazon Co-purchase network dataset, GBAD was not able to detect the attribute anomalies. Since GBAD treats all attribute values as discrete, it loses the measure of similarity between values, thus cannot detect an attribute anomaly. In the Enron dataset, GBAD was able to detect the same outliers that our proposed approach detected. In summary, the proposed approach outperforms GBAD on two datasets, while GBAD provides similar results on the Enron e-mail dataset, where structure was the primary anomaly.

Discussion Even though we also use GBAD as the integrated anomaly detection tool, the proposed approach performs better than the standalone GBAD tool because GBAD loses the measure of similarity when the attributes are numeric. Particularly, in the Amazon co-purchase network dataset, GBAD can not perform well because of the numeric attributes. However, the proposed approach is able to identify 4 out of 6 anomalies. The proposed approach adds the measure of similarity between the attributes when the attributes are numeric, precisely what the existing structural anomaly detection approaches like GBAD lack. Thus, structural anomaly detection techniques can benefit from the proposed approach.

All three approaches perform almost identically on the Enron e-mail dataset. Clustering- and distance-based method performed similar on the synthetic dataset by discovering, on average, 7 out of 10 and 8 out of 10 anomalies respectively, while the discretization-based approach is able to discover 5 out of 10 anomalies on average. However, the performance of the distance-based similarity measure is better in the case of the Amazon co-purchase network.

Among the three approaches used, the discretization-based approach could not perform on par with the other two approaches. Particularly when the values of features are skewed, the discretization approach produces too many unique labels causing issues with the anomaly detection tool. The distance-based approach allows the user to input a continuous threshold between a value of 0 and 1, which gives flexibility to the user to tune the performance of the approach. However, the clustering-based approach requires the user to specify the number of clusters, which is difficult to determine. So, among the three approaches used in our experiments, the distance-based approach is effective as well as easy to tune.

Conclusion and Future Work

In this work, we proposed an approach which is able to identify structural anomalies as well as anomalies in attribute values. We compared the performance of the proposed approach with an existing structural anomaly detection tool and showed that the proposed technique can detect anomalies where the traditional structural techniques cannot. We conclude that using node label similarity information together with structural properties of the graph, we can detect both structural anomalies and attributed anomalies in a graph.

Although our research shows some good results, because of the known complexities of the graph-based anomaly detection approaches, we were limited to small-to-medium sized graphs. In the future, we will test the scalability of the proposed approach on larger, ”big data” datasets. Also, we only experimented with sparse synthetic graphs. Since, we only use the attribute value information for discovering similarity between nodes, the labelling algorithms used are not affected by the topology of the graph. However, in future, we would like test our approach against denser graphs.

Similarly in this work, we only focused on the numerical attributes and left the categorical attributes as distinct. However, in some real world scenarios, categorical attributes could also be similar. For example, the proposed approach treats ’river’ and ’stream’ as completely different, but in general they could be considered as similar. One potential way to figure out similarity between categorical attributes is by using an ontology. In the previous example, ’river’ and ’stream’ both belong to the type ’body of water’. So, we could use ’body of water’ for both ’river’ and ’stream’. Also, extracting an ontology for entities that share common structure provides added knowledge about those entities (Paudel, Kandel, and Eberle 2019). This would allow us to extend our work to categorical attributes, resulting in the discovery of more interesting anomalies.

References

Akoglu, L.; Tong, H.; and Koutra, D. 2015. Graph based anomaly detection and description: a survey. *Data mining and knowledge discovery* 29(3):626–688.

Biba, M.; Esposito, F.; Ferilli, S.; Di Mauro, N.; and Basile, T. M. A. 2007. Unsupervised discretization using kernel density estimation. In *IJCAI*, 696–701.

Boden, B.; Günnemann, S.; Hoffmann, H.; and Seidl, T. 2012. Mining coherent subgraphs in multi-layer graphs with edge labels. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1258–1266. ACM.

Caceres, B. M. 2013. Grouping similar values for a specific attribute type of an entity to determine relevance and best values. US Patent 8,615,516.

Davis, M.; Liu, W.; Miller, P.; and Redpath, G. 2011. Detecting anomalies in graphs with numeric labels. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, 1197–1202. ACM.

Eberle, W., and Holder, L. 2007. Discovering structural anomalies in graph-based data. In *Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on*, 393–398. IEEE.

Eberle, W., and Holder, L. B. 2009. Discovering anomalies to multiple normative patterns in structural and numeric data. In *FLAIRS Conference*.

Gao, J.; Liang, F.; Fan, W.; Wang, C.; Sun, Y.; and Han, J. 2010. On community outliers and their efficient detection in information networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 813–822. ACM.

Klimt, B., and Yang, Y. 2004. The enron corpus: A new dataset for email classification research. In *European Conference on Machine Learning*, 217–226. Springer.

Muller, E.; Sánchez, P. I.; Mulle, Y.; and Bohm, K. 2013. Ranking outlier nodes in subspaces of attributed graphs. In *Data Engineering Workshops (ICDEW), 2013 IEEE 29th International Conference on*, 216–222. IEEE.

Noble, C. C., and Cook, D. J. 2003. Graph-based anomaly detection. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 631–636. ACM.

Paudel, R.; Kandel, P.; and Eberle, W. 2019. Detecting spam tweets in trending topics using graph-based approach. In *Proceedings of the Future Technologies Conference*, 526–546. Springer.

Ramesh Paudel, W. E., and Holder, L. B. 2018. Anomaly detection of elderly patient activities in smart homes using a graph-based approach. In *Proceedings of the 2018 International Conference on Data Science*, 163—-169. CSREA.

Ramesh Paudel, W. E., and Talbert, D. 2017. Activity in diabetic patients using graph-based approach. In *FLAIRS Conference*, 423—-428.

Romero, O. E.; Gonzalez, J. A.; and Holder, L. B. 2010. Handling of numeric ranges for graph-based knowledge discovery. In *FLAIRS Conference*.

Sánchez, P. I.; Muller, E.; Laforet, F.; Keller, F.; and Bohm, K. 2013. Statistical selection of congruent subspaces for mining attributed graphs. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, 647–656. IEEE.

Sánchez, P. I.; Müller, E.; Irmler, O.; and Böhm, K. 2014. Local context selection for outlier ranking in graphs with multiple numeric node attributes. In *Proceedings of the 26th International Conference on Scientific and Statistical Database Management*, 16. ACM.

Shah, N.; Beutel, A.; Hooi, B.; Akoglu, L.; Gunnemann, S.; Makhija, D.; Kumar, M.; and Faloutsos, C. 2016. Edgecentric: Anomaly detection in edge-attributed networks. In *Data Mining Workshops (ICDMW), 2016 IEEE 16th International Conference on*, 327–334. IEEE.

Yang, T.; Jin, R.; Chi, Y.; and Zhu, S. 2009. Combining link and content for community detection: a discriminative approach. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 927–936. ACM.