Detection of Interdomain Routing Anomalies Based on Higher-Order Path Analysis

Murat Can Ganiz, Sudhan Kanitkar and
Mooi Choo Chuah
Lehigh University CSE Department
Bethlehem, PA 18015, USA
{mug3, sgk205, mcc7}@lehigh.edu

William M. Pottenger
Rutgers University DIMACS Lab
Piscataway, NJ 08854, USA
billp@dimacs.rutgers.edu

Abstract

Internet routing dynamics have been extensively studied in the past few years. However, dynamics such as interdomain Border Gateway Protocol (BGP) behavior are still poorly understood. Anomalous BGP events including misconfigurations, attacks and large-scale power failures often affect the global routing infrastructure. Thus, the ability to detect and categorize such events is extremely useful. In this article we present a novel anomaly detection technique for BGP that distinguishes between different anomalies in BGP traffic. This technique is termed Higher Order Path Analysis (HOPA) and focuses on the discovery of patterns in higher order paths in supervised learning datasets. Our results demonstrate that not only worm events but also different types of worms as well as blackout events are cleanly separable and can be classified in real time based on our incremental approach. This novel approach to supervised learning has potential applications in cybersecurity/forensics and text/data mining in general.

1. Introduction

Border Gateway Protocol (BGP) is the de facto interdomain routing protocol. BGP is responsible for the discovery and maintenance of paths between autonomous systems (ASes) in the Internet. The Internet is made of thousands of ASes, which are loosely defined as a connected set of IP prefixes under a single administration [18]. BGP provides reachability information to ASes and distributes external reachability internally within an AS. With the exponential growth of ASes, BGP has become one of the most critical components of the Internet's infrastructure. Within the last few years, internet routing dynamics have been extensively studied [1], [2], [3], [4]. However, interdomain routing dynamics such as BGP activities are still poorly understood. Abnormal BGP events including misconfigurations [5], attacks [6], and large-scale power failures [7] often affect global routing infrastructure. For example, in January 2003, the Slammer worm caused a surge of BGP updates [8]. In August 2003, the East Coast electricity blackout affected 3175 networks and many BGP routers were shut down [9]. Since anomalous BGP events often cause major disruptions in the Internet, the ability to detect and categorize BGP events is extremely useful. Different abnormal BGP events can have different effects on the performance of the Internet and may warrant different corrective actions. For example, some worm events may cause a surge in BGP traffic on the control plane but may not cause noticeable degradation in the packet delivery performance on the Internet's data plane [23]. Different worm quarantine mechanisms may be needed for different attack types.

In this article we propose a novel data mining approach termed Higher Order Path Analysis (HOPA) that focuses on distinguishing different anomalous events in BGP traffic. To our knowledge this is the first such work to successfully distinguish between Slammer, Witty and Blackout events. Our work with the Witty worm is especially notable because the Witty worm event poses challenges to detection algorithms and has not been widely studied. Although in this work our main focus is interdomain routing, our supervised HOPA learning algorithm is not limited to this application, but rather can be applied in other learning domains as well. HOPA discovers higher-order link patterns in data based on relationships between entities. In this context, a higher-order link can be represented as a chain of co-occurrences of entities in different records as seen in figure 1.

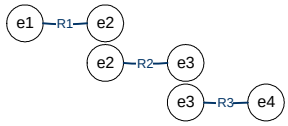

Figure 1. Higher-order path as a chain of co-occurrences

We also refer to such a link as a higher-order path. Given a supervised learning dataset (i.e., labeled training data), HOPA discovers patterns in sets of higher-order links that distinguish between the classes in the data. (Note that in this paper records correspond to what are usually referred to as instances in machine learning and our use of entities corresponds to attribute-value pairs. We use these pairs of terms interchangeably.) Our results are based on statistical BGP data extracted from the RouteViews archive [10]. Our target is to characterize and distinguish different anomalous BGP events such as worm attacks (e.g., Slammer, Witty) and power failures using our HOPA learning algorithm. In this paper we also

present and report results for an incremental HOPA algorithm for online (real time) detection and characterization of different abnormal BGP events. We tested our algorithm on BGP data from the Slammer worm attack, the Witty worm attack and the 2003 East Coast Blackout event.

The rest of the article is organized as follows: in Section 2 we briefly review related work. In Section 3, we present our approach followed by results in Section 4 and discussion in Section 5. Section 6 outlines some interesting research issues that we wish to explore in future work, and our conclusions are drawn in Section 7.

2. Related Work

In [11] Li et al. use attributes derived from BGP traffic to detect internet routing anomalies. They employ data mining techniques, in particular a decision tree machine learning algorithm, to train a model using labeled data. The authors use the counts of different types of BGP messages divided into one minute bins. Their model consists of the rules learned, and is used to detect occurrences of abnormal events. Basically their system can distinguish between two classes – event and normal – but cannot differentiate between different types of events. Thus one important drawback in their approach is that it cannot distinguish between different anomalous events and worms. In fact, in her public review, Dina Katabi from MIT points out the importance of identifying whether an abnormal event is caused by a worm, blackout, or misconfiguration [11].

Several other efforts have been undertaken in [18], [19], [20] and [21] of a similar nature. Zhang et al. [20] proposes two approaches, signature based and statistics-based detection. Zhang et al. [21] employs wavelets and k-means clustering to build an instance-learning framework that identifies anomalies for a given prefix as well as across prefixes. Most of these efforts follow the same basic steps: first the system is trained using labeled training data, and then the system examines test data and flags anomalies. Our approach differs in the sense that we characterize anomalous events and use models of these anomalous events to classify test data. In summary, it is important to note that to the best of our knowledge none of this prior work has successfully distinguished between different worm types.

Higher order co-occurrence is closely related to our HOPA technique. In our previous work in [12], we proved mathematically that Latent Semantic Indexing (LSI), a well-known approach to information retrieval, implicitly depends on higher-order co-occurrences. We also demonstrated empirically that higher-order co-occurrences play a key role in the effectiveness of systems based on LSI. LSI can reveal hidden or latent relationships among terms, as terms semantically similar lie closer to each other in the LSI vector space. In our prior work in [17] we analyzed a machine learning dataset from the UCI repository and concluded that the classes of instances in labeled training data may be separable using the characteristics of higher-order paths. In this preliminary work we developed both theoretical and algorithmic approaches to enumerating and characterizing higher-order paths between attribute-value pairs. Based on statistical comparisons of distributions of higher-order path itemset frequencies, we discovered evidence that classes of instances in a labeled dataset containing nominal attributes may be separable based on the characteristics of higher-order paths.

One of the challenges facing us in this work is the complexity of enumerating the various higher-order paths. We used graph representations of the data and graph algorithms to enumerate higher-order paths. In this area too, fortunately, there has been prior work on which we can build. In [13], Uno proposes efficient algorithms for enumerating chordless paths and cycles of a given graph G = (V, E). Given two vertices, this algorithm takes $O(|V||E|)$ time for each path connecting them. Additionally, in related work, Uno [14] presents enumerating algorithms for perfect, maximum and maximal matchings in a bipartite graph $G_b = (V_1 \cup V_2, E)$. A matching is a set of edges that have no vertices in common between the two sets of vertices in G_b. On the other hand, maximum matchings are defined as matchings whose cardinalities are maximum among all matchings. An algorithm that has a time complexity of $O(|V_1 \cup V_2|)$ per matching is proposed for maximum matchings in bipartite graphs.

3. Approach

We focus on discovering higher-order link patterns in BGP traffic based on higher-order associations between elements of data termed entities. In this context, entities can be aggregate counts of announce or withdraw BGP updates, and a higher-order link is represented as a chain of co-occurrences of such entities in different snapshots of BGP traffic taken over time. As noted we also refer to such a link as a higher-order path. Given a supervised learning dataset (i.e., labeled training data), we attempt to discover patterns in sets of higher-order links that distinguish between the classes in the labeled data. As such, our approach is a supervised learning technique.

Our definition of a higher-order path is similar to that found in graph theory, which states that given a non-empty graph G = (V, E) of the form $V = \{x_0, x_1, \ldots, x_k\}$, $E = \{ x_0x_1, x_1x_2, \ldots, x_{k-1}x_k \}$ with nodes x_i distinct, two vertices x_i and x_k are linked by a path P where the number of edges in P is its length [15]. Our definition of a higher-order path differs from this in a couple of respects. First,

vertices V = {e_0, e_1, …, e_k} represent entities, and edges E = {r_0, r_1, …, r_m} represents records, documents or instances. Several edges may exist between given entities. Finally and most importantly, in a higher-order path both vertices and edges must be distinct. We are interested in enumerating all such paths.

In order to use conventional graph structures and algorithms, we divided the above representation into two structures. First, we form a co-occurrence graph G_c = (V, E) in which the vertices are the entities and there is an edge between two entities if they co-occur in one or more records. A path (length ≥ 2) extracted from G_c satisfies the first requirement of our higher-order path definition since the vertices in this path are distinct. The second requirement entails that records on a path must be distinct, and another data structure that contains lists of records for each edge is needed. We term this structure a *path group*.

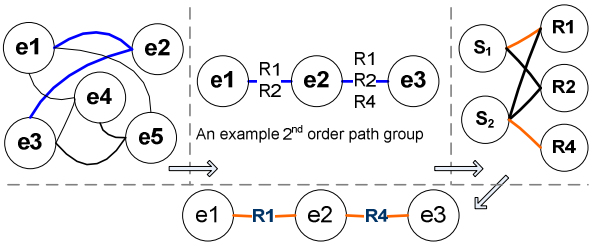

Figure 2. Extracting/enumerating higher-order paths from co-occurrence graph

Using the path group representation, we need to identify the system of distinct representatives (SDR) of the record sets. Each distinct representative in the path group satisfies the second requirement and corresponds to a higher order path. In order to enumerate all the distinct representatives in a given path group, a bipartite graph G_b = ($V_1 \cup V_2$, E) is formed such that V_1 is the sets of records (S_1, S_2,…) in a given path group and V_2 is the records themselves. A maximum matching with cardinality $|V_1|$ in this bipartite graph yields the SDR for the higher order path. This process is summarized in figure 2. In figure 2 we can see an example 2nd order path group (e_1-{1,2}-e_2-{1,2,4}-e_3) that is extracted from the co-occurrence graph G_c. This particular 2nd order path group includes two sets of records: S_1={1,2} and S_2={1,2,4}. S_1 corresponds to the records in which e_1 and e_2 co-occur, and S_2 is the set of records in which e_2 and e_3 co-occur. A bipartite graph G_b = ($V_1 \cup V_2$, E) is formed where V_1 is the two sets of records and V_2 is the all records in these sets. Enumerating all maximum matchings in this graph yields all higher-order paths in the path group. The fourth diagram (depicted in figure 2) shows an example of the many paths in this path group. In this higher-order path, edge labels R_1 and R_4 are records in S_1 and S_2 and the path corresponds to the orange-colored maximum matching in the bipartite graph.

Motivated by the fact that the order of the entities in a higher-order path encapsulates information about the relations between records/instances as well as for performance reasons, we implemented our own method to discover frequent itemsets in the higher-order paths. Thus, our definition of frequent itemsets is a bit different from the standard definition used in association rule mining (ARM). Itemsets in our framework are ordered, and thus must appear in order in a given supporting path. Additionally, the items (entities) in an itemset must be adjacent in the higher-order path. During computational enumeration of the paths, statistics are gathered. When dealing with labeled training data used in supervised machine learning, we divide the instances by class and then characterize the resulting sets by higher-order itemset frequencies. The end result is a distribution of itemset frequencies for a given class. Actually we compute two distributions. The first is the frequencies of higher-order itemsets for particular order paths (e.g., 3-itemsets from 4th order paths). These frequencies are similar to the support metric in Apriori, a well-known ARM algorithm [22]. However, instead of counting the number of records containing a given k-itemset, we count the number of higher order paths containing a given higher-order k-itemset. The second distribution is the counts of same-frequency itemsets. Either of these distributions can be compared for different classes using simple statistical measures such as the t-test. If two distributions are statistically significantly different, we conclude that the higher-order path patterns (i.e., itemset frequencies) separate the classes.

4. Incremental HOPA Algorithm

We developed an incremental HOPA algorithm for use in online BGP anomaly detection as well as other similar applications. In our algorithm the main data structure is the path group, which consists of entities and sets of records. There are several path groups formed from the dataset. In the incremental algorithm, a sliding window moves forward in time and new records are added and old records deleted as appropriate. The size of the sliding window is a parameter of the algorithm, and in the results reported herein we employed windows composed of 120 instances, with three seconds per instance (i.e., time bin). This approach results in changes in some of the path groups. Specifically for BGP data, however, we have observed that only a small fraction of these path groups need to be updated. There are two update operations. First we check to see if any new entities are introduced by new records from the next window or whether any existing entities are no longer referenced after deletion of old

records from the previous window. If changes occur, we need to first update the entity co-occurrence graph G_c and then the corresponding bipartite graph G_b of each path group impacted by a change. Recall from the Approach in section 3 that extracting path groups from G_c is quite fast. For deleted records we need only modify path groups which include deleted records. Following this the maximum matchings in G_b of the path group are re-enumerated. Once this step is complete, either the itemset counts or the counts of same-frequency itemsets can be updated and a statistical test performed to ascertain significance. The algorithm is given below.

Algorithm 1. (INCREMENTAL HOPA)

E_W: the set of distinct entities in the window
E_D: the set of distinct entities in deleted records
E_A: the set of distinct entities in added records
R_D: deleted records
R_A: added records
G_b: bipartite graph of a single path group
G_c: entity co-occurrence graph

For window x
1. delete records R_D from window (x-1)
2. update E_W
3. **if** E_D-E_W≠{} /* *some entities are no longer referenced after deletion of* R_D */
   a. delete path groups which has entities from E_D-E_W
   b. update G_c by deleting vertices E_D-E_W
   c. enumerate path groups in G_c
4. **for each** path group containing entities from E_D
   a. delete nodes R_D from G_b
   b. enumerate maximum matchings in G_b
5. add records R_A to window (x-1)
6. **if** E_A-E_W≠{}/**new entities are introduced from* R_A*/
   a. update G_c by adding vertices E_A-E_W
   b. enumerate path groups in G_c
   c. **for each** path group containing entities from E_A-E_W
      i. form G_b
      ii. enumerate maximum matchings in G_b
7. **else** /* *no new entities are introduced from* R_A */
   a. **for each** path group that has entities from E_A
      i. add appropriate records to G_b
      ii. enumerate maximum matchings in G_b
8. delete n-itemsets supported only by deleted path groups
9. update n-itemset frequencies supported by changed path groups
10. form n-itemsets from new path groups

5. Results

As noted previously, the implementation of our algorithm is based on the TMI [16] and thus implemented in C++. We performed the experiments to discover the higher-order path statistics on the National Center for Supercomputing Applications (NCSA) Tungsten Supercluster (Xeon Linux) as well as on Windows-based PCs in our lab. We performed experiments to discover the higher order path statistics in sets of BGP data extracted from the RouteViews archive [10]. The anomalous events we experimented with were a Slammer worm attack, a Witty worm attack and a Blackout (i.e., power failure). On January 25, 2003 the Slammer worm infected between 75,000 and 100,000 computers and caused network outages[1]. The Witty worm infected only about 12,000 hosts on March 19, 2004[2]. Finally, the 2003 USA east coast power blackout occurred on August 14, 2003 [9].

The data was collected and divided into three-second bins. Each bin became a single instance in our training data and was labeled with the appropriate event (Slammer, Witty or Blackout) or non-event class. We employed the first six attributes used in [11] since they are indicators of the routing dynamics and were straightforward to extract.

Empirically we found that 240 instances (i.e., 720 second windows) are sufficient to characterize a particular abnormal event. After applying HOPA on this training set, the same-frequency 3-itemset counts from 4th order paths is the *model* learned for each event. Given a sample of BGP traffic, the probability associated with the Student's t-test is used to assess whether the sample is *statistically* significantly different from the event models. Table 1 depicts the result of comparing the event models with one another. These t-test results show <5% two-tail probability, meaning that with a high degree of confidence (greater than 95%) our event models are statistically significantly different from one another. We also selected non-Slammer, non-Witty and non-Blackout data samples and tested them against our event models. These also resulted in statistically significant differences.

Real Time Classification of Abnormal Events

Given our success in developing models of abnormal events, our next task was to develop an approach to classify events in real time. To do so, as noted we developed an incremental algorithm and employed sliding windows to allow events to be recognized in real time. Specifically, sliding windows samples of 120 instances (360 seconds) were extracted every 30 seconds. If a given sliding window matched one of our learned models (Slammer, Witty, or Blackout), we detected the corresponding event. In order to ascertain a match, as with the aforementioned model comparisons, we computed the two-tail probability using the t-test. This approach provides a robust mechanism for detecting anomalous

[1] http://www.caida.org/publications/papers/2003/sapphire/
[2] http://www.caida.org/analysis/security/witty/

events because it does not rely on only a single comparison, but rather numerous comparisons are made.

The sliding window samples include instances from given event periods starting with the 13th window. By the time the 25th window is reached, the sample overlaps the first 360 seconds of the given abnormal event model completely. This point in time is indicated by a vertical dotted line in the figures below. From this point forward, the window continues sliding through the given event period. Figures 4, 5 and 6 below depict the change of the probability associated with the t-test as the samples slide with time. The significance threshold of 5% is indicated by the red horizontal line at 0.05 on the y-axis. A probability below the red line indicates that the particular sliding window is statistically significantly different from the given abnormal event model. In this sense, what we expect is for the probability to increase from near zero in window number one to a value greater than 0.05 as the windows slide through time into the event period.

Table 1. Event vs. Event comparison

Event 1	Event 2	t-test results
Slammer	Witty	0.00023
Blackout	Witty	0.00016
Slammer	Blackout	0.018

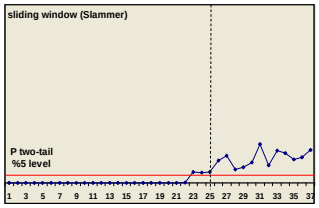

Figure 3. Probabilities associated with t-tests from sliding windows of Slammer

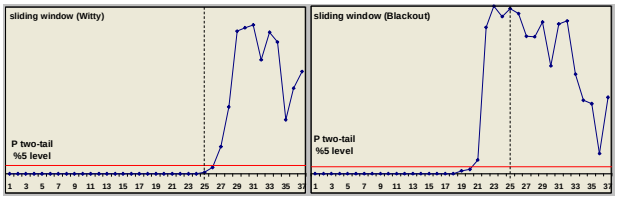

Figure 4. Probabilities associated with t-tests from sliding windows of Witty (left) and Blackout (right)

For the Slammer worm and Blackout events, as can be seen from figures 3 and 4, the t-test probability starts increasing as the sliding window approaches the 25th window. When the number of abnormal event instances inside the current window exceeds a certain threshold (around the 21st – 23rd window), we observe a sharp increase in the probability. After the 25th window, the probability stays above 5%, revealing that we are in the event period and have detected and distinguished both the Slammer and Blackout events using their respective event models. These results confirm that the HOPA technique is able to detect and distinguish these events in 360 seconds or less. Results are similar for the Witty worm event in figure 4, although the detection takes slightly longer.

In contrast to existing supervised learning algorithms, however, HOPA is not performing single-instance prediction. Rather, HOPA classifies a set of instances. This approach is an especially good fit for this domain since prediction of an abnormal event from a single instance is not feasible. By using a window of instances, HOPA also exploits additional information from the relations/links between entities and instances, thereby giving a better characterization of the overall event.

This approach to event detection and classification can be used as part of an online classification and mitigation system. Since anomalous BGP events often cause major disruptions in the Internet, the ability to detect and categorize BGP events is extremely useful.

6. Discussion

The really interesting question here is "Why do patterns in higher-order paths correlate with the class?" In a sense it hearkens back to our prior work with Latent Semantic Indexing (LSI) [12] – in that work, as noted, we determined that the 'Latent' aspects of term similarity that LSI reveals are dependent on the higher-order paths between terms. Likewise, in real-world supervised machine learning datasets, the goal is to learn the relation between the attributes and the class. It is noteworthy that attributes are certainly not equally important. In addition, neither attributes nor instances are independent of one another, given the class. As we found with LSI, it is our contention that the 'latent semantics', if you will, of attribute-attribute relations also depend on the higher-order paths linking attribute-value pairs. By taking attribute-value pairs as our base unit of 'semantics' and linking them via higher-order co-occurrence relations, we reveal these latent semantics, or patterns, that distinguish instances of different classes. These results are extremely interesting given that we have uncovered evidence of separability based on the higher order path patterns alone. We consider this achievement significant, and something that can be exploited in many different domains with different data as long as there is a meaningful context of entities that allows us to leverage co-occurrence relations.

7. Conclusions and Future Work

In this work we analyzed higher-order path patterns in data generated during interdomain routing. We represent the data as a machine learning dataset composed of instances that correspond to three second samples of Border Gateway Protocol (BGP) traffic. Using these three second bins, we successfully modeled anomalous BGP events caused by power failures and worm attacks including the Slammer worm attack, the Witty worm attack and the 2003 USA East Coast blackout. Based on

these results, in order to build a real time classification system, we proposed an incremental algorithm for HOPA. To evaluate this approach we employed a sliding window that starts in a non-event period and slides through an event period. Our results show that the HOPA technique is able to detect abnormal events using an incremental algorithm. To the best of our knowledge, this is the first work to accomplish both the detection and classification of these anomalous events with a high degree of confidence. Since abnormal BGP events can cause major disruptions in Internet, the ability to detect and categorize BGP events is extremely useful.

Our higher-order path analysis technique has applications in text mining as well. For instance, by considering a document or paragraph as an instance, we may determine higher order path characteristics that aid in classifying text. We plan to explore this approach further since it likely has important applications in security, counterterrorism and law enforcement.

Acknowledgments: This work was supported in part by USA NSF grant number 0534276 and USA National Institute of Justice, Department of Justice grant numbers 2003-IJ-CX-K003, 2005-93045-PA-IJ and 2005-93046-PA-IJ. Points of view in this document are those of the authors and do not necessarily represent the official position or policies of the Department of Justice or the NSF. Co-author William M. Pottenger wishes to thank His Lord and Savior Yeshua (Jesus) the Messiah for His grace and truth. Amen.

References

[1] Griffin, T. What is the Sound of One Route Flapping?, IPAM 2002.

[2] Caesar, M., Subramanian, L., and Katz, R.H. Route Cause Analysis of Internet Routing Dynamics. Tech Report, UCB/CSD-04-1302, 2003.

[3] Lad, M., Nanavati, A., and Massey, D. An Algorithmic Approach to Identifying Link Failures. Proceedings of Pacific Rim Dependable Computing Symposium, March, 2004.

[4] Mao, Z.M., Bush, R., Griffin, T.G., and Roughan, M. BGP Beacons. Proceedings of ACM IMC, 2003.

[5] Mahajan, R., Wetherall, D., and Anderson, T. Understanding BGP Misconfigurations. Proceedings of ACM Sigcomm, Aug, 2002.

[6] Wang, L., Zhao, X., Pei, D., Bush, R., Massey, D., and Mankin, A. Observation and Analysis Of BGP Behavior Under Stress. Proceedings of Internet Measurement Workshop, Nov, 2002.

[7] Wu, Z., Purous, E. S., and Li, J. BGP Behavior Analysis during the August 2003 Blackout. In International Symposium on Integrated Network Management, 2005.

[8] Lad, M., Zhao, X., Zhang, B., Massey, D., and Zhang, L. An Analysis of BGP Updates Surge during Slammer Attack. Proceedings of IWDC, 2003.

[9] Cowie, J., Ogielski, A., Premore, B., Smith, E., and Underwood, T. Impact of 2003 blackouts on Internet Communications., Tech Report, Renesys, Nov, 2003.

[10] University of Oregon Route Views Project. http://antc.uoregon.edu/route-views/.

[11] Li, J., Dou, D., Wu, Z., Kim, S. An Internet Routing Forensics Framework for Discovering Rules of Abnormal BGP Events. ACM Sigcomm, (35)5:55-66, Oct, 2005.

[12] Kontostathis, A., and Pottenger, W.M. A Framework For Understanding LSI Performance. Information Processing & Management, 42(1), 2006, pp. 56-73.

[13] Uno, T. An Output Linear Time Algorithm for Enumerating Chordless Cycles. 92nd SIGAL of Information Processing Society Japan, 47-53, 2003.

[14] Uno, T. Algorithms for Enumerating All Perfect, Maximum and Maximal Matchings in Bipartite Graphs. Lecture Notes in Computer Science, Vol. 1350. Proceedings of the 8th International Symposium on Algorithms and Computation, 1997, pp. 92 – 101, ISBN: 3-540-63890-3, Springer-Verlag, London, UK

[15] Diestel, R. Graph Theory. Springer Press, 2000, ISBN 0-387-95014-1

[16] Holzman, L.E., Fisher, T.A., Galitsky, L.M., Kontostathis, A., and Pottenger, W.M. A Software Infrastructure for Research in Textual Data Mining. The International Journal on Artificial Intelligence Tools, 14 (4), 2004, pp. 829-849.

[17] Ganiz, M.C., Pottenger, W.M., and Yang, X. Link Analysis of Higher-Order Paths in Supervised Learning Datasets. In the Proceedings of the Workshop on Link Analysis, Counterterrorism and Security, 2006 SIAM Conference on Data Mining. Bethesda, MD, April 2006.

[18] Zhao, X., Pei, D., Wang, L., Massey, D., Mankin, A., Wu, S., and Zhang, L. Detection of Invalid Routing Announcement in the Internet. Proceedings of Dependable Systems and Networks, 2002

[19] Kruegel, C., Mutz, D., Robertson, W., and Valeur, F. Topology-based detection of anomalous BGP messages. Proceedings of ACM Symposium on Recent Advances in Intrusion Detection, (28)20:17-35, Sept, 2003.

[20] Zhang, K., Yen, A., Zhao, X., Massey, D., Wu, S.F., and Zhang, L. On Detection of Anomalous Routing Dynamics in BGP. Networking 2004, 3042, pp. 259 - 270

[21] Zhang, J., Rexford, J., and Feigenbaum, J. Learning-Based Anomaly Detection in BGP Updates. Proceeding of the 2005 ACM SIGCOMM Workshop on Mining Network Data. 219 - 220, 2005

[22] Agrawal, R., Imielinski, T., and Swami, A.N. Mining Association Rules Between Sets of Items in Large Databases. In P. Buneman and S. Jajodia, editors, Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pages 207–216, Washington, D.C., 26–28 1993.

[23] Roughan, M., Li, J., Bush, R., Mao, Z., Griffin, T., Is BGP update storm a sign of trouble: observing the internet control and data planes during internet worms, Proceedings of Spects, 2006.