Realtime BGP Anomaly Detection using Graph Centrality Features

Janel Huang, Murugaraj Odiathevar, Alvin Valera, Jyoti Sahni, Marcus Frean, and Winston K.G. Seah

Abstract Border Gateway Protocol (BGP) anomalies, such as hijacking, is currently growing in trend due to limited detection capabilities. BGP hijacking maliciously reroutes Internet traffic, causing Denial of Service (DoS) to major Internet Service Providers (ISPs) or redirection attacks to Internet users. While it has been shown that BGP anomalies can be detected using machine learning (ML) methods, the features used to train these ML models are not comprehensive. This is because node level features, such as the number of BGP announcements, average Autonomous System (AS) path length and average edit distance do not consider the structure or relationships present in the network graph. In this paper, an approach to extract information from BGP updates to build a network graph is proposed. Then, centrality information is used as features to model the graphical structure of the network to build an early detection tool for BGP anomalies using ML. The proposed method has been validated on real world data from the CenturyLink outage and shows promising results for anomaly detection (as early as one hour before the event was reported) in both individual and a defined group of networks. Furthermore, the anomaly source can be determined using the proposed method.

Keywords: Anomaly detection, Border Gateway Protocol, Graph Centrality, Autoencoders, Gaussian Mixture Model.

1 Introduction

Operating as the backbone of the Internet, Border Gateway Protocol (BGP) is the routing protocol used for transferring information across different networks or Au-

Janel Huang, Alvin Valera, Jyoti Sahni, Marcus Frean and Winston K.G. Seah
School of Engineering and Computer Science, Victoria University of Wellington, New Zealand. e-mail: {firstname.lastname}@ecs.vuw.ac.nz

Murugaraj Odiathevar
Research & Business Foundation, Sungkyunkwan University, South Korea. e-mail: muru.raj@g.skku.edu

tonomous Systems (ASes) on the Internet. The presence of BGP routers are known through BGP update messages transmitted from router to router across the Internet. Over the years, there have been many incidents caused by anomalous BGP updates. BGP anomalies are caused by events such as hijacking or misconfigurations which have been shown to cause severe outages and redirection attacks [8], proving to be an important consideration for securing Internet traffic. The disconnectivity of panix.com in United States of America (USA) is an example showcasing the intentional redirection of BGP routes [23]. Another BGP anomaly example is the global BGP CenturyLink Outage that was a consequence of the misconfiguration of BGP routes [19]. BGP anomalies have caused severe outages for Internet users and revenue loss for many businesses across the globe, making the detection of BGP anomalies very crucial.

All BGP anomalies historically have shown changes in the network structure and thus provides the ability to capture the BGP anomalies. This has motivated the use of graphical network features that are fed to graph neural networks (GNN) [10] and shown to be highly effective [11]. However, the process requires the collection of datasets, which can be tedious and resource intensive [9], and more importantly, only suitable for offline detection like most ML-based anomaly detection methods. Methods that are designed for realtime detection (e.g. [7], [17], [20]) have been shown to detect the anomalies as they occur which leaves little to no time for network operators to react.

The main contribution of this paper is a technique to identify (even predict) network anomalies in realtime (as they develop, even before these anomalies become full-blown events) and determine the source of the anomaly by extracting information from BGP update messages; hereafter, also referred to simply as “BGP updates”. Instead of using node features, a network structure is built using BGP updates. From the graph generated, centrality features are extracted to model the structure of the network. This information is then passed into two machine learning (ML) algorithms to detect anomalies, viz., an autoencoder to detect anomalies in the entire network, and a Gaussian Mixture Model (GMM) to detect anomalies in individual networks.

The rest of this paper is structured as follows. Section 2 describes the current methods used to detect BGP anomalies and Section 3 outlines the design of the proposed anomaly detection method. Section 4 then discusses the validation and experimental results. Finally, Section 5 ends with concluding remarks and research directions recommended to be examined in the future.

2 Related Work

Current BGP anomaly detection methods include time series, statistical pattern recognition, using historical data, reachability check and ML [2]. In this section, we review representative methods for BGP anomaly detection.

The earliest method used to detect BGP anomalies is *time series* [18, 1] which gained popularity initially as it could find characteristics of abnormal behaviour in a set of BGP updates collected within a time period. However, only a limited number of incidents can be detected in data collected over two years using statistical features such as the number of announcements and message volume. Such features have a distinct behaviour for a specific type of anomaly, thus limiting the ability of using statistical features to detect a wide range and new types of BGP anomalies.

Building upon the time series method, *statistical pattern recognition* has been shown to be successful in determining existing BGP anomalies as it can find relationships amongst BGP updates, e.g. [26], using features such as AS-path and edit distance [4] to determine the behaviour of the network topology. On the contrary, new types of BGP anomalies are still undetected because the features do not consider the entire network topology as they are examined independently in an instance. Furthermore, without constructing of a topology, relationships amongst AS-path and edit distance features cannot be accurately described. Thus, this motivates the use of network topology built using BGP updates to accurately describe the inherent relationships.

To overcome the limitation of identifying new BGP anomalies, a whitelisting approach that uses *historical BGP data* to determine the abnormality of new BGP updates has been proposed [8, 22]. This method utilises a BGP attribute of prefix origin change but only specific types of anomalies, such as prefix hijacking, can be detected. Other anomalies such as link failures and sub-prefix hijacks remain undetected as the feature used is not comprehensive to reflect all changes in the network topology. Thus, reinforcing those features which model the network topology must be utilised.

In contrast to methods discussed above, the *reachability check* method gained popularity as it is less computationally expensive as only a single hop count calculation is required for each BGP update [25, 29]. But such a method only considers reachability changes and does not identify relationships amongst multiple reachability checks. Thus, identification of all and new BGP anomalies is not possible [25, 29], thereby corroborating that a network topology must be built to extract a more comprehensive set of features.

All methods presented above have not proven to allow the capability for a method to automatically learn from experience and improve its performance over time. To counter this limitation, *machine learning (ML)* methods can be used where an ML model is trained using existing BGP updates to detect anomalies within a network [27, 3, 15]. This method is sought-after as the objective function for modelling abnormal and normal behaviour can be found and optimized automatically. However, ML methods typically rely on statistical features such as the number of announcements, withdrawals or the average AS path length which are not sufficiently comprehensive to model the entire network [17]. Only direct and indirect anomalies are detectable while other or new (not previously known) types of anomalies cannot be detected [27, 3]. This reinforces the notion that graph-level features must be extracted to capture all and new BGP anomalies [10, 24] as well as the potential of graph neural networks [12, 11].

While training ML models with graphical network features and the use of GNN have been shown to be effective, they require the collection and analysis of large datasets, which are time and resource intensive, making them primarily suitable for offline detection and forensic analysis. Hence, this research proposes a method to use BGP updates to build the network structure and extract graph-level features for determining anomalies in realtime. ML will be used as a detection tool to enable automatic identification of normal and abnormal features. As the network topology is complex and high dimensional, a method like ML that can automatically detect correlations in such data is suitable in time and accuracy.

3 Design

The design of the anomaly detection method focuses on graphical feature extraction from BGP updates. This process includes the construction of the network graph and the selection of graph-level features, as shown in Figure 1. Subsequently, this information is used to train an ML model to detect anomalies.

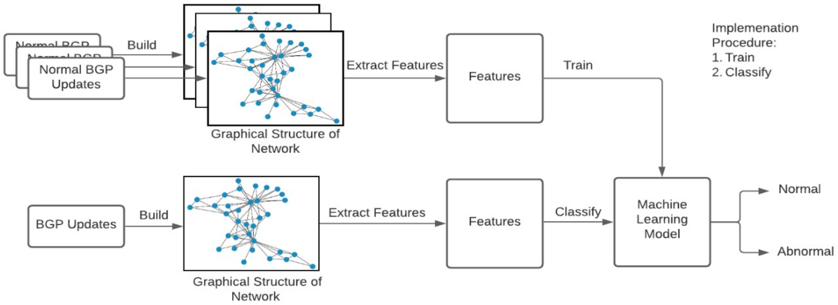
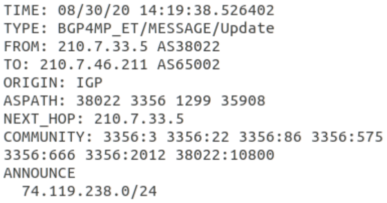


Fig. 1: Workflow of Proposed Anomaly Detection Method

3.1 BGP Updates



```
TIME: 08/30/20 14:19:38.526402
TYPE: BGP4MP_ET/MESSAGE/Update
FROM: 210.7.33.5 AS38022
TO: 210.7.46.211 AS65002
ORIGIN: IGP
ASPATH: 38022 3356 1299 35908
NEXT_HOP: 210.7.33.5
COMMUNITY: 3356:3 3356:22 3356:86 3356:575
3356:666 3356:2012 38022:10800
ANNOUNCE
  74.119.238.0/24
```

Fig. 2: BGP Update Message Example

To formulate the graphical view of the network, appropriate BGP update attributes must be selected. BGP updates are used to inform BGP routers of paths to ASes in the Internet. This enables BGP routers to select the best path when forwarding traffic to destinations. An example of a BGP update is shown in Figure 2. According to RFC4271 [21], the AS_PATH attribute can be used where each node present in the path represents a direct connection to the next node in the path. The source (FROM) and destination (TO) nodes within each update can also be included as a node in the graph. Each node can have connections added or removed within each BGP update. Therefore, the ANNOUNCEMENT and WITHDRAWAL attributes should be used to ensure that connections are added or removed appropriately from the corresponding nodes. Therefore, these five attributes are used to construct the network graph.

3.2 Data Structures

After constructing the network, the graph must then be represented in a form suitable for training an ML model. The representation must model the graph structure and the node relationships to capture a graph-like form of the network. Hence, adjacency lists are used to store a list of nodes in the network and their connected immediate neighbours.

During announcements and withdrawals of BGP routes, a BGP router will add or remove routes respectively. Therefore, the IP addresses of an AS must be present in each node to find withdrawn or announced routes. Each node stores a dictionary of IP addresses as an AS may compose of multiple IP addresses. Our system uses a trie structure to store all IP addresses with their associated nodes. E.g., when storing an IP address of 172.168.2.1, 173.168.2.1 and 172.168.3.1, the trie structure will compute the tree as shown in Figure 3. Using a trie structure yields better efficiency, $O(1)$. The pseudo code for building the graph using BGP updates is shown in Algorithm 1.

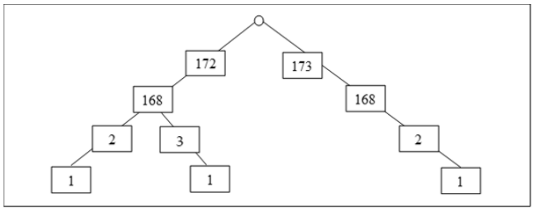


Fig. 3: Example of a Computed Trie Structure for 172.168.2.1, 173.168.2.1 and 172.168.3.1

Algorithm 1 BGP Update Graph Creation

```
$f$ = Read BGP Update
$l$ = Read line in $f$
$a$                                   ▷ Announcing Node

while $l$ is not $null$ do
    if $l$ == “ASPATH” then
        Add AS Path using $l$
    else if $l$ starts with “TO”
            or $l$ starts with “FROM” then
        $n$ = add ip to node using $l$
        if $l$ starts with “TO” then
            $a = n$
        end if
    else if $l$ starts with “ANNOUNCE” then
        Add path, $l$ to $a$
    else if $l$ starts with “WITHDRAW” then
        Remove path, $l$ from $a$
    end if
    $l$ = Read next line in $f$
end while
```

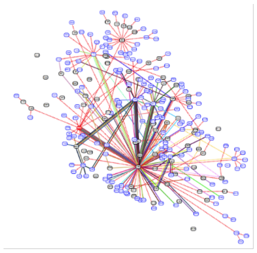


Fig. 4: Graph on 2020-08-30 at 9:04 (before anomaly event)

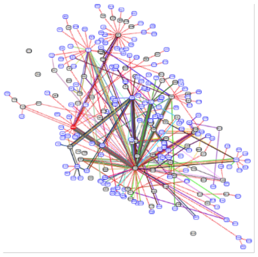


Fig. 5: Graph on 2020-08-30 at at 10:05 (during anomaly event)

3.3 Features

Centrality features are extracted from the constructed network graph and used in the ML algorithm to determine whether the BGP updates are abnormal. In this paper, we propose using the features of node centrality as it is computable and can reflect a node's presence in relation to the entire network. Other features such as storing the connectivity of nodes are not feasible due to over 60,000 nodes in the network. The feature of clustering coefficients can also be used, but individual network information is lost as aggregations of neighbourhoods is required.

Node centrality measures how central a node's position is within a network. BGPlay visualisations [6] of the CenturyLink outage in Figures 4 and 5 show that the node centralities changed significantly. This is because a large portion of the traffic was rerouted, leading to several nodes having more or fewer paths routing through them, thereby changing its centrality. Therefore, node centrality can be used as a feature in the ML algorithm to detect anomalies because a large difference in the centrality of a node can indicate abnormal behaviour [5]. We used centrality metrics that are feasible to compute, namely:

(i) Degree Centrality (DC) – Number of immediate neighbours of a node and is calculated, as follows [5]:

$$\mathbf{DC}(u) = \frac{n}{N-1} \tag{1}$$

(ii) Closeness Centrality (CC) – Inverse distance to all the reachable neighbours of a node; the CC formula proposed by Wasserman *et al.* [28], as shown in Eqn. (2), is used as it scales each node's CC separately by the size of the corresponding node and its neighbours,

$$\mathbf{CC}(u) = \left\{ n-1/\sum_{v=1}^{n-1} distance(u,v) \right\} \frac{n-1}{N-1} \tag{2}$$

where n denotes the number of reachable neighbours of a node u and N denotes the total number of nodes in the graph [28].

The pseudo code for extracting centrality features is shown in Algorithm 2. DC is inexpensive to compute as it only requires enumerating the number of immediate neighbours of each node. CC is also computationally inexpensive as enumerating the distance to all reachable neighbours of a node is only required. The distance to all reachable neighbours is calculated using Dijkstra's algorithm to allow efficient computation of distances. Although the A* algorithm can instead be used for greater efficiency [13], the generation of a heuristic is dependent on the geolocation of the nodes, which is unobtainable due to insufficient information in BGP updates.

Other centralities such as betweenness and eigenvector [5] are not used as they are computationally infeasible for realtime detection. Betweenness centrality requires all possible paths to be computed, which is infeasible due to the presence of over 60,000 nodes in the network. Calculating the eigenvector centrality requires the computation of the adjacency matrix which is infeasible due to the computational resource constraints, viz., extensive memory needed to create a large matrix size for over 60,000 nodes present in the BGP updates.

3.4 Machine Learning Models

The ML model must identify the main patterns in normal data (extracted from the normal non-anomalous BGP updates) as such patterns can reveal outliers or anomalies. ML models as suggested by literature [2] such as autoencoder and GMM are selected. To detect anomalies in the entire network, an autoencoder is used as it can find the complex relationships amongst normal centralities and thereby determine the anomalous centralities. An antoencoder is a NN with an encoder, hidden and decoder layers. Each input in the encoder layer is reduced in dimensionality within the hidden layer to $1+\sqrt{n}$, where n represents the number of nodes in the network graph, using the Relu activation function [16]. This removes the noise in the data and preserves the important variations. Subsequently, the features are reconstructed in the decoder layer which enables the model to identify complex relationships amongst the inputs and reconstruct the input. By training the weights within each layer using centralities during normal operation, this enables the model to learn the definition of normal centralities. Hence, anomalous centralities would return a large reconstruction error as the model cannot reconstruct the centrality using the learnt normal relationships.

The main disadvantage of Autoencoders is that the detection of anomalies for a specific AS is not addressed. Identification of problematic ASes is useful for network administrators to avoid routing to such ASes in an abnormal event. To detect individual network anomalies, a white-box method such as GMM is used [16]. GMM uses normal centralities to compute a Gaussian distribution for each AS. Each new centrality has a probability fitting into the trained distribution. Hence, it is assumed that a low probability of a centrality value fitting into the trained normal distribution would indicate anomalies for the AS. The training parameter indicating the number of components within each GMM is 1, as there was one cluster present in the normal distribution of every AS, essentially a univariate Gaussian (UG).

To train the ML models, BGP updates that are gathered during known normal network operations, such as two weeks before the anomaly incident (excluding two

Algorithm 2 Centrality Feature Extraction

```
$f$ = Read BGP Update
$g$ = Get graph from $f$                    ▷ Contains each node with its immediate neighbours
$N$ = Enumerate number of nodes in $g$

for $n = 1, \ldots, N$ do
    $node$ = get $n$ in $graph$
    Calculate $paths$ as the shortest path length from each $node$ to its reachable neighbours
    Calculate CC as in Eqn. (2) using number of $paths$, $N$ and total length of $paths$
    Calculate DC as in Eqn. (1) using number of immediate neighbours of $node$ in $g$ and $N$
end for
```

days before the anomaly incident.) Using less than 2 weeks of data is inadequate, as there is insufficient training data provided for the ML models to find correlations or define a correct distribution for detecting anomalies.

4 Evaluation

Our proposed method is evaluated against the detection of abnormal behaviour during the BGP CenturyLink outage from 08-30-2020 10:04 (UTC). This helps to determine whether the learnt model can correctly classify anomaly incidents. The BGP CenturyLink outage was detected by ISPs through Twitter feeds by CenturyLink [19]. This reinforces the fact that an earlier and more reliable detection method is required to alert BGP anomalies. Anomaly detection for the entire network and specific ASes from the New Zealand (NZ), Japan (WIDE) and Serbia (SOXRS) core routers on the day of the BGP CenturyLink outage will be evaluated using autoencoders and GMM respectively. Experiments included an evaluation of the CC and DC features used for determining the anomaly score. The detection of an anomaly incident is based on the autoencoder reconstruction error (anomaly score) breaching a “threshold” that is determined (viz., maximum reconstruction error) using a validation set which consists of a small subset of normal data, two days before the anomaly incident.

4.1 Entire Network Detection

Determination of whether an entire network is anomalous allows ISPs to have a generalised view of a network’s stability. This allows faster determination of anomalous behaviour in comparison to monitoring multiple individual networks. Hence, an experiment using the degree and closeness centralities from the BGP updates of the NZ and WIDE core routers are used to determine the anomaly score in the BGP CenturyLink Outage. For the NZ core router, both the closeness and degree centrality (Figures 6 and 7) show an increase in the anomaly score before the estimated time breach.

We observe a similar increase in anomaly score for the WIDE core router, as shown in Figures 8 and 9. This indicates that the proposed method can detect the anomaly incident earlier, thus increasing the time for remediating the incident. A rise in the anomaly score is observed as the BGP CenturyLink Outage caused network instability where many nodes in the network needed to redirect traffic to compensate the loss of a Tier 1 ISP. This caused the number of connected neighbours for each node to change significantly. E.g., AS174 (Cogent Communications) had an increase of 200 immediate neighbours [19], further validating that our method is able to detect the anomalous activity in the network during the BGP CenturyLink Outage.

However, the detection time of anomalies can be significantly impacted by the geographic position of the observing router. E.g., a 3-hour delay of detection for the BGP CenturyLink Outage is observed for the SOXRS router as shown in Figures 10 and 11. BGP updates are transferred router to router and routers like SOXRS which are located further away from the source of the incident will receive the anomalous

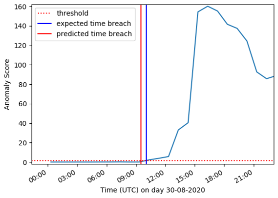


Fig. 6: NZ CC

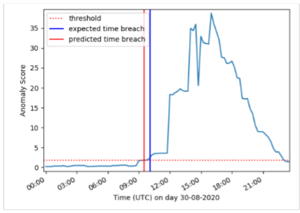


Fig. 8: WIDE CC

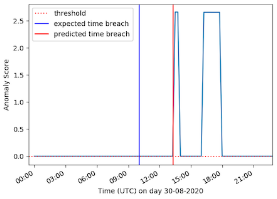


Fig. 10: SOXRS CC

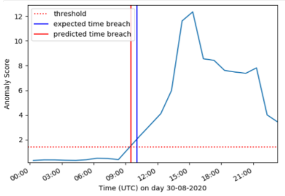


Fig. 7: NZ DC

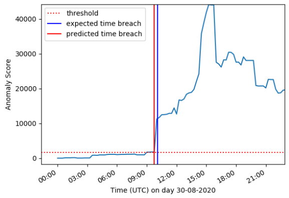


Fig. 9: WIDE DC

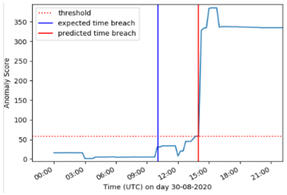


Fig. 11: SOXRS DC

updates later. This suggests that multiple core routers located at different positions around the world should be used to monitor anomalies.

4.2 Individual Network Detection

If the entire network is deemed anomalous, specific ASes should be checked to determine whether they are affected or is the source of the BGP incident. This helps to ensure that ISPs can update their routing tables to prevent traffic from being routed to such ASes, thus minimizing the chance of Denial of Service (DoS). The evaluations of individual networks will focus on AS3561 and AS38022 that are involved in the BGP CenturyLink Outage. AS3561 is the BGP CenturyLink AS which caused the outage through a misconfiguration and AS38022 (Research and Education Advanced Network New Zealand) is a peer of AS3561 that had to redirect its traffic to compensate for the disconnectivity of AS3561.

Using UG to compute (cf: Section 3.4) the anomaly score of an AS (e.g. AS3561) from the BGP updates received by the other (NZ, WIDE and SOXRS) ASes' core routers can be used to determine the anomaly source. Depending on the source of the anomaly, different core routers will detect anomalies at different times. E.g., NZ detected abnormal behaviour at least 1 hour before the anomaly event because AS3561 is an immediate neighbour of NZ, hence it can detect abnormal behaviour before WIDE (further away) and SOXRS (furthest away.) As SOXRS is geographically further away from AS3561 than WIDE, the detection of the anomalous activity is 2 hours after the anomaly event.

Figures 12 and 13 show the CC and DC anomaly scores of AS38022, respectively, from the NZ core router (itself) while Figures 14 and 15 show the CC and DC anomaly scores of AS38022, respectively from the WIDE core router. Figures 13, 14 and 15 show a rise in the anomaly score before the estimated time breach as the network was unstable. A later time breach is predicted for NZ (by itself) as shown in Figure 12 as the anomaly event did not originate from AS38022. The dis-

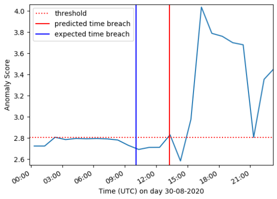


Fig. 12: NZ AS38022 CC

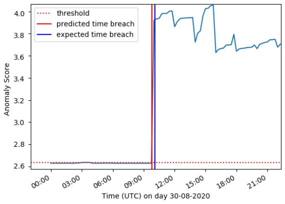


Fig. 14: WIDE AS38022 CC

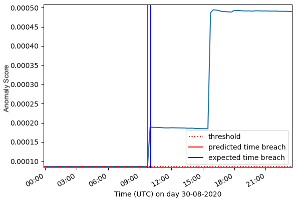


Fig. 15: WIDE AS38022 DC

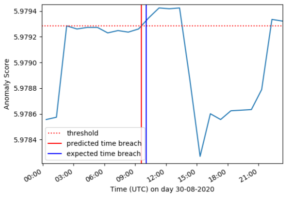


Fig. 13: NZ AS38022 DC

Network	Earliest Detection Time (UTC)
Entire Network	09:30
AS3561 Network	08:30 (detected by NZ)
AS38022 Network	10:00 (detected by WIDE)

Table 1: Earliest Anomaly Detection Time during BGP CenturyLink Outage on 08-30-2020 with an estimated time breach at **10:04 UTC** based on Twitter feeds.

tance from AS38022 to its reachable neighbours did not change until the error from the source of the anomaly, AS3561, propagated through the Internet. As AS3561 is a trusted network peer of AS38022, the error that is propagated by AS3561 is deemed normal when transferred to AS38022. However, an earlier time breach is predicted from the WIDE core router as it can view the anomalous activity between AS38022 and AS3561 from an outsider's point of view. The earliest detection times of the anomalies occurring in the entire network as well as individual networks are summarized in Table 1. The NZ router detected an anomaly in AS3561 about 1hr 30min before it was reported while the WIDE router detected an anomaly in AS38022 (NZ) a few minutes before the event was report.

No anomaly score is generated from SOXRS for AS38022 as it does not have AS38022 within its routing table during the detection period. This is because SOXRS is geographically further away from AS38022 and did not have any traffic that travelled to AS38022 within the detection period. This suggests that multiple core routers should be used for detection to allow anomalies to be discovered throughout the Internet.

4.3 Results Analysis and Findings

The key objective of our approach is early detection of anomalous events using the BGP update messages. This is achieved by comparing the anomaly scores (reconstruction errors) during anomalous events with those during normal operations. Our aim is not the analysis of ML models through typical performance metrics of accuracy, precision, recall, F1, etc. [14, 17].

Both the closeness and degree centralities show similar results, with a similar time of anomaly detection. However, DC is cheaper to compute than CC as it only requires calculation of nodes' immediate neighbours instead of reachable neighbours. DC yields a complexity of $O(n)$ and CC yields a complexity of $O(ne)$ where n

represents the number of nodes and e represents the number of edges in the network. This suggests that DC should be used to alert abnormalities in the entire network. CC, however, performs better than DC in determining the severity of an incident where a large number of ASes can be indicated as anomalous as demonstrated in the BGP CenturyLink Outage. This is because CC takes into account the reachable neighbours of each node, meaning a major ISP outage will be reflected in more ASes in comparison to DC which only considers the immediate neighbours of nodes.

5 Conclusion and Future Work

In this paper, we have proposed a network anomaly detection method that extracts graph-like features from BGP updates on-the-fly for realtime anomaly detection. Construction of the network graph using BGP update attributes is conducted, and degree and closeness centrality features are extracted for anomaly detection using ML models, viz., autoencoders and GMM. The detection method is successful in detecting anomalous BGP incidents such as misconfiguration events.

An immediate future work item would be the study of different ML models and their effectiveness, as well as, other network centrality measures. Other future work items include distributed processing to enhance the computational speed of feature extraction and network neighbourhood aggregation to enable Internet-wide anomaly detection. Using traffic link analysis to increase the confidence of the proposed method also needs further study.

References

1. Al-Musawi, B., Branch, P., Armitage, G.: Detecting BGP instability using Recurrence Quantification Analysis (RQA). In: Proc. of the IEEE 34th Int'l Performance Computing and Communications Conf. (IPCCC). Nanjing, China (Dec 2015)
2. Al-Musawi, B., Branch, P., Armitage, G.: BGP Anomaly Detection Techniques: A Survey. IEEE Communications Surveys & Tutorials **19**(1), 377–396 (2017)
3. Al-Rousan, N.M., Trajković, L.: Machine learning models for classification of BGP anomalies. In: Proc. of the IEEE 13th Int'l Conf. on High Performance Switching and Routing. pp. 103–108. Belgrade, Serbia (24-27 Jun 2012)
4. Blazakis, D., Karir, M., Baras, J.S.: Analyzing BGP ASPATH behavior in the Internet. In: Proc. of the 9th IEEE Global Internet Symposium (2006)
5. Chen, H., Yin, H., Chen, T., Nguyen, Q.V.H., Peng, W.C., Li, X.: Exploiting Centrality Information with Graph Convolutions for Network Representation Learning. In: Proc. of the IEEE 35th Int'l Conf. on Data Engineering (ICDE). pp. 590–601. Macao, China (8-11 Apr 2019)
6. Di Battista, G., Mariani, F., Patrignani, M., Pizzonia, M.: BGPlay: A System for Visualizing the Interdomain Routing Evolution. In: Proc. of the Int'l Symposium on Graph Drawing. pp. 295–306. Perugia, Italy (21-24 Sep 2003), http://bgplay.routeviews.org/
7. Fezeu, R.A.K., Zhang, Z.L.: Anomalous Model-Driven-Telemetry Network-Stream BGP Detection. In: Proc. of the IEEE 28th International Conference on Network Protocols (ICNP). pp. 1–6 (2020)
8. Haeberlen, A., Avramopoulos, I., Rexford, J., Druschel, P.: NetReview: Detecting When Interdomain Routing Goes Wrong. In: Proc. of the 6th USENIX Symposium on Networked Systems Design and Implementation. pp. 437–452. Boston, MA, USA (22-24 Apr 2009)

9. Hoarau, K., Tournoux, P.U., Razafindralambo, T.: BML: An Efficient and Versatile Tool for BGP Dataset Collection. In: Proc. of the IEEE Int'l Conf. on Communications Workshops (ICC Workshops). pp. 1–6 (2021)
10. Hoarau, K., Tournoux, P.U., Razafindralambo, T.: Suitability of Graph Representation for BGP Anomaly Detection. In: Proc. of the IEEE 46th Conf. on Local Computer Networks (LCN). pp. 305–310. Edmonton, AB, Canada (04-07 Oct 2021)
11. Hoarau, K., Tournoux, P.U., Razafindralambo, T.: BGNN: Detection of BGP Anomalies Using Graph Neural Networks. In: Proc. of the IEEE Symposium on Computers and Communications (ISCC). Rhodes Island, Greece (Jun 2022)
12. Latif, H., Paillissé, J., Yang, J., Cabellos-Aparicio, A., Barlet-Ros, P.: Unveiling the Potential of Graph Neural Networks for BGP Anomaly Detection. In: Proc. of the 1st Int'l Workshop on Graph Neural Networking (GNNet). pp. 7–12. Rome, Italy (Dec 2022)
13. Li, F., Yang, E., Ma, A., Dong, R.: Optimal Representation of Large-Scale Graph Data Based on Grid Clustering and K^2-Tree. Mathematical Problems in Engineering **2020**, 1–8 (Jan 2020)
14. Li, Z., Rios, A.L.G., Trajković, L.: Machine Learning for Detecting Anomalies and Intrusions in Communication Networks. IEEE Journal on Selected Areas in Communications **39**(7), 2254–2264 (2021)
15. Lutu, A., Bagnulo, M., Cid-Sueiro, J., Maennel, O.: Separating wheat from chaff: Winnowing unintended prefixes using machine learning. In: Proc. of the IEEE Conf. on Computer Communications (INFOCOM). pp. 943–951. Toronto, ON, Canada (27 Apr - 02 May 2014)
16. Odiathevar, M., Cameron, D., Seah, W.K.G., Frean, M., Valera, A.: Humans Learning from Machines: Data Science Meets Network Management. In: Proc. of the Int'l Conf. on COMmunication Systems NETworkS (COMSNETS). pp. 421–428. Bengaluru, India (5-9 Jan 2021)
17. Peng, S., et al.: A multi-view framework for BGP anomaly detection via graph attention network. Computer Networks **214**, 109129 (2022)
18. Prakash, B., Valler, N., Andersen, D., Faloutsos, M., Faloutsos, C.: BGP-lens: Patterns and anomalies in internet routing updates. In: Proc. of the ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. pp. 1315–1324. Paris, France (28 Jun - 1 Jul 2009)
19. Prince, M.: Aug 30th 2020: Analysis of century-link/level(3) outage, https://blog.cloudflare.com/analysis-of-todays-centurylink-level-3-outage/
20. Putina, A., et al.: Unsupervised real-time detection of bgp anomalies leveraging high-rate and fine-grained telemetry data. In: Proc. of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). pp. 1–2 (2018)
21. Rekhter, Y., Hares, S., Li, T.: A Border Gateway Protocol 4 (BGP-4). RFC 4271 (Jan 2006), https://rfc-editor.org/rfc/rfc4271.txt
22. Shi, X., et al.: Detecting Prefix Hijackings in the Internet with Argus. In: Proc. of the Internet Measurement Conf. (IMC). pp. 15–28. Boston, MA, USA (Nov 2012)
23. Simon, T.L.: oof. panix sidelined by incompetence... again, https://www.mail-archive.com/nanog@merit.edu/msg40003.html
24. Sun, J., et al.: An Efficient BGP Anomaly Detection Scheme with Hybrid Graph Features. In: Proc. of the Int'l Conf. on Emerging Networking Architecture and Technologies (ICENAT). pp. 494–506. Shenzhen, China (15-17 Oct 2022)
25. Tahara, M., Tateishi, N., Oimatsu, T., Majima, S.: A Method to Detect Prefix Hijacking by Using Ping Tests. In: Proc. of the 11th Asia-Pacific Network Operations and Management Symposium (APNOMS). pp. 390–398. Beijing, China (22-24 Oct 2008)
26. Theodoridis, G., Tsigkas, O., Tzovaras, D.: A novel unsupervised method for securing bgp against routing hijacks. In: Computer and Information Sciences III, pp. 21–29. Springer (2013)
27. de Urbina Cazenave, I.O., Köşlük, E., Ganiz, M.C.: An anomaly detection framework for BGP. In: Proc. of the Int'l Symposium on Innovations in Intelligent Systems and Applications. pp. 107–111. Istanbul, Turkey (15-18 Jun 2011)
28. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications. Structural Analysis in the Social Sciences, Cambridge University Press (1994)
29. Zhang, Z., Zhang, Y., Hu, Y.C., Mao, Z.M., Bush, R.: iSPY: Detecting IP Prefix Hijacking on My Own. IEEE/ACM Trans. on Networking **18**(6), 1815–1828 (2010)