BGP ANOMALY DETECTION USING ASSOCIATION RULE MINING ALGORITHMS

2023
MASTER THESIS
COMPUTER ENGINEERING

Mubaarak ABDULLAH ALTAMIMI

Thesis Advisor
Assist. Prof. Dr. Zafer ALBAYRAK

BGP ANOMALY DETECTION USING ASSOCIATION RULE MINING ALGORITHMS

Mubaarak ABDULLAH ALTAMIMI

Thesis Advisor
Assist. Prof. Dr. Zafer ALBAYRAK

T.C
Karabuk University
Institute of Graduate Programs
Department of Computer Engineering

Prepared as
Master Thesis

KARABUK
Januaray 2023

I certify that in my opinion the thesis submitted by Mubaarak Abdullah ALTAMIMI titled “BGP ANOMALY DETECTION USING ASSOCIATION RULE MINING ALGORITHMS ” is fully adequate in scope and in quality as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Zafer ALBAYRAK .....
Thesis Advisor, Department of Computer Engineering

This thesis is accepted by the examining committee with a unanimous vote in the Department of Computer Engineering as a Master of Science thesis. Jan 26, 2023

Examining Committee Members (Institutions)	Signature
Chairman : Prof. Dr. Adib HABBAL (KBÜ)	.....
Member : Assist. Prof. Dr. Fatih VARÇIN (SUBU)	.....
Member : Assist. Prof. Dr. Zafer ALBAYRAK (SUBU)	.....

The degree of Master of Science by the thesis submitted is approved by the Administrative Board of the Institute of Graduate Programs, Karabuk University.

Prof. Dr. Müslüm KUZU .....
Director of the Institute of Graduate Programs

"I declare that all the information within this thesis has been gathered and presented in accordance with academic regulations and ethical principles and I have according to the requirements of these regulations and principles cited all those which do not originate in this work as well."

Mubaarak ABDULLAH ALTAMIM

ABSTRACT

M. Sc. Thesis

BGP ANOMALY DETECTION USING ASSOCIATION RULE MINING ALGORITHMS

Mubaarak ABDULLAH ALTAMIMI

Karabük University
Institute of Graduate Programs
The Department of Computer Engineering

Thesis Advisor:
Assist. Prof. Dr. Zafer ALBAYRAK
January 2023, 84 pages

Border Gateway Protocol (BGP) is the most common gateway protocol for the communication between autonomous systems to share routing and reachability information. Anomalous behavior of protocol attributes could occur due to a variety of factors, including inadequate provisioning, malicious attacks, traffic or equipment issues, and network operator mistakes. A rule-based machine learning method was proposed to detect anomalies behavior features of BGP protocol according to the training attributes dataset model. Depending on the association rule, the dataset is analyzed and divided into sub-frequent pattern datasets. One anomalous frequent itemset is chosen from each pattern dataset to compare them and determine the itemset that has the highest value of anomaly. Finally, the association rules are built between these anomalous itemsets. The most utilized rule-based machine learning algorithms in anomaly detection are Frequent Pattern (FP) growth and apriori algorithms. However, no comparative evaluation studies have been conducted between these

algorithms in the literature. In this context, this study aims to employ a feature selection approach by association rule unsupervised algorithms to detect BGP anomalies and evaluate the performance of these algorithms in terms of values of support, confidence, and accuracy. Moreover, the anomaly detection findings were visualized using an effective and customized tool and framework to clarify the performance in a more accurate and detailed manner.

Key Words: BGP anomalies detection, Datamining, Association rules, Dataset, Apriori & FP growth algorithm.

Science Code : 92403

ÖZET

Yüksek Lisans Tezi

İLİŞKİLENDİRME KURALI MADENCİLİĞİ ALGORİTMASINI KULLANARAK BGP ANOMALİ TESPİTİ

Mubaarak ABDULLAH ALTAMIMI

Karabük Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı:
Dr. Öğr. Üyesi Zafer ALBAYRAK
Ocak 2023, 84 sayfa

Sınır Ağ Geçidi Protokolü (BGP), yönlendirme ve erişilebilirlik bilgilerini paylaşmak için otonom sistemler arasındaki iletişim için en yaygın ağ geçidi protokolüdür. Yetersiz provizyon, kötü niyetli saldırılar, trafik veya ekipman sorunları ve ağ operatörü hataları gibi çeşitli faktörler nedeniyle protokol özniteliklerinde anormal davranış meydana gelebilir. Eğitim nitelikleri veri seti modeline göre BGP protokolünün anormal davranış özelliklerini tespit etmek için kural tabanlı bir makine öğrenme yöntemi önerilmiştir. İlişkilendirme kuralına bağlı olarak, veri seti analiz edilir ve alt frekanslı model veri setlerine bölünür. Bunları karşılaştırmak ve en yüksek anormallik değerine sahip öğe kümesini belirlemek için her örüntü veri kümesinden bir anormal sık öğe kümesi seçilir. Son olarak, bu anormal öğe kümeleri arasında birliktelik kuralları oluşturulur. Anormallik tespitinde en çok kullanılan kural tabanlı makine öğrenimi algoritmaları, Sık Model (FP) büyümesi ve apriori algoritmalarıdır. Ancak literatürde bu algoritmalar arasında herhangi bir karşılaştırmalı değerlendirme

çalışması yapılmamıştır. Bu bağlamda, bu çalışma, BGP anormalliklerini tespit etmek ve bu algoritmaların performansını destek, güven ve doğruluk değerleri açısından değerlendirmek için birliktelik kuralı denetimsiz algoritmalarla bir özellik seçimi yaklaşımı kullanmayı amaçlamaktadır. Ayrıca, performansı daha doğru ve ayrıntılı bir şekilde netleştirmek için anomali tespit bulguları etkili ve özelleştirilmiş bir araç ve çerçeve kullanılarak görselleştirildi.

Anahtar Kelimeler : BGP anormallik tespiti, Veri madenciliği, Birliktelik kuralları, Veri kümesi, Apriori & FP algoritması.

Bilim Kodu: 92403

ACKNOWLEDGEMENT

First of all, I would like to give thanks to my advisor, Assist. Prof. Dr. Zafer ALBAYRAK, for his great interest and assistance in preparation of this thesis.

CONTENT

Page

Page

Page

LIST OF FIGURES

Page

Page

LIST OF TABLES

Page

SYMBOLS AND ABBREVITIONS INDEX

AS	: Autonomus System
ASes	: Autonomus Systems
BGP	: Border Gateway Protocol
IGP	: Interior Gateway Protocol
OSPF	: Open Shortest Path First
EGP	: Exterior Gateway Protocol
ISPs	: Internet Service Providers
RIP	: Routing Information Protocol
ASBR	: Autonomous System Boundary Router
IBGP	: Interior Border Gateway Protocol
EBGP	: Exterior Border Gateway Protocol
IXP	: Internet Exchange Point
RFC	: Request For Comments
IDS	: Internet Detection System
IP	:Internet Provider
TCP	: Transmission Control Protocol
FSM	: Finite State Machine
NLRI	: Network Layer Reachability Information
AS-PATH	: Autonomus System Path
CLI	: Command Line Interface
SRD	: System Reference Documment
MED	: Multi Exit Discriminator
RA	: Route Aggregation
OSPF	: Open Shortest Path First
FPM	: Frequent Pattern Mining
GUI	: Graphical User Interface
FP-GRWOTH	: Frequent Pattern grwoth

NCC	: Network Coordination Centre
SPSS	: Statistical Package for Social Sciences
ARFF	: Attribute Relation File Format

PART 1

INTRODUCTION

1.1. OVERVIEW

The Internet is a decentralized network that spans the whole globe and is composed of tens of thousands of autonomous systems (ASs). An AS is a group of routers that work together under the same technical management. They connect to each other with an Interior Gateway Protocol (IGP) (e.g., Open Shortest Path First (OSPF)) and to other ASs with an Exterior Gateway Protocol (EGP) (e.g., Border Gateway Protocol (BGP)) [1]. The routers send updates only upon the occurrence of route changes. Hence, it is possible that the internet may one day reach a steady state in which receiving fresh update notifications is unnecessary. However, BGP Routing (BGPR) seems far from stable due to the changes that may occur for a variety of reasons [2]. An active BGP session between two routers is required for the exchange of update messages. Each router uses local policies to determine which route is the best for each prefix and whether to broadcast it to the neighbor. A single event, such as a failed connection, might trigger a series of changes as routers seek other routes. Alterations to the BGP routing might potentially result in performance issues in which packets destined for the target prefix might be stuck in forwarding loops [3-4].

Internet Service Providers (ISP) execute their relationships via routing policies. ISP may employ traffic engineering to regulate traffic direction and routing protocols through route prepending. Based on their algorithms, routing protocols are divided into three categories: link-state, such as OSPF, Distance vector (i.e. Routing Information Protocol (RIP)) and path vector (i.e. BGP) are types of vectors [5]. By sending path-vector signals, Autonomous System Boundary Routers (ASBR) use BGP to let people know which networks can be reached [6]. Border gateway protocol is divided into two types: Internal Border Gateway Protocol (IBGP), which connects BGP routers inside an AS to External Border Gateway Protocol (EBGP), which runs between BGP routers across ASs.

BGP anomalies can result in anything from a single abnormal BGP update to hundreds more anomalous BGP updates. Update is considered anomalous if it includes an inaccurate AS number, invalid or reserved IP prefixes, a prefix published by unlawful AS, AS-PATH without a physical counterpart [16]. The rule-based unsupervised machine learning algorithms technique use to build a traditional anomaly detection systems based on the available dataset model, Additionally, rules are generated from the regular features and labels that have been used to filter the anomalies value and calculate the average of the anomalies and positives value using association rule apriori and fp growth algorithms [25].

1.2. PROBLEM STATEMENT

Border gateway protocols may be affected by disorder and abnormal behavior caused by various factors, including malicious attacks, hijacking, misconfiguration, DoS &DDOS attacks, malicious attacks, data corruption, and network operator mistakes and traffic or equipment issues. Whereas, Cyberattacks on the BGP routing absolutely result in performance issues and threaten networks in internal networks BGP backbones as layers. Previous studies used classification and clustering and neural network approaches for BGP anomaly detection Without explaining the mount of association rules between anomalous features and how similar their frequency. Moreover, correlating more data sources and employing various techniques, such unsupervised learning, do not result in model improvements that reduce false positives. In conclusion, the following issues have been noted and will be dealt with in the thesis:

1. The F-score and SVM 2 models, that used to detect an anomaly in the Border gateway protocol only over time and identify the flows connected with the event(s) that caused the anomaly which is based on Code Red I and Slammer datasets, produce results with lower accuracy. Moreover, in previous studies, in classification and clustering methodologies, there is no clear framework that visualized and shows the lift thresholds of algorithms [5].
2. The paucity of using a rule-based machine learning methodology for BGP anomaly detection to detect its anomalous behavior features according to

training network traffic, and how that approaches apply for extracting unusual association rules from the network packet dataset was put forth.

3. Lack of explanation of association rules between BGP anomalous features and their frequency similarity, and representation to confirm accuracy using an effective framework to make sure the accuracy of the results of the algorithms.

1.3. AIM AND OBJECTIVES

The association rule analysis technique use to build a traditional BGP anomaly detection based on the available dataset model. Its used to extract appropriate features from raw data including rules and recurring patterns. moreover, apriori and fp growth algorithms used to generate rules from among the regular features and labels that have been used to filter the anomalies value and find the average of the anomalies and positives value. Consequently, three specific objectives that this thesis aims to achieve are listed below :

1. To design classifying method of Extracting a features that using the dataset to detect anomalies using association rule mining Unsupervisor algorithms.
2. To provide Comprehensive Evolution the results of the two apriori and FP growth algorithms and the rules associated between them to take the best of by evaluating.
3. To develop a professional visualization process to represent the results of the study to confirm its accuracy by using an effective and customized tool and framework.

1.4. THESIS CONTRIBUTION

These are the main contributions of this study:

1. Generate a features of anomalous rules by using association rule mining Unsupervisor algorithms and candidates detectors technique.
2. Detect anomalous Frequent Items by analyzing and dividing a dataset into sub-frequent pattern datasets, and perform Comprehensive testing of frequent Item Sets candidates to extract with the highest value of anomaly.

3. This study focuses on the value of support, Confidence, and accuracy for each algorithm and selects the most accurate results, an represent it to confirm its accuracy by using an effective framework.

1.5. LITERATURE REVIEW

Several methods of machine learning techniques that have been used in the detection of BGP anomalies, such as calsifications and clustring algorithms ,association rule minig, artificial nural network, decision tree. Author examine the results In [2], were derived on a comprehensive analysis of BGP update messages as well as Low-level traffic measurements obtained from AT&Ts internet protocol backbone. Examine the reliability of the routing to destination prefixes that match the list of popular websites maintained by netrating while using the messages that have been acquired by the routeviews or RIPE-NCC servers. This illustrates that the BGP routes for the very few well-known destinations that are responsible for the vast majority of internet traffic are quite dependable. A small handful of unpopular destinations are to blame for the great bulk of BGP instability.

In [11], the researcher followed the machine learning methodology to detect the anomalies of the border gateway protocol and the use of unsupervisor technique and the clustering model and carried out the experiment through the program K-means to get the results. The clustering procedure and how it is used to identify anomalies depend on the dataset, according to the researcher's in-depth description. The border gateway protocol BGP should be explained in great depth, making clear how it functions and key terms in the protocol.

According to [13], This paper focuses on identified a number of vulnerabilities and risks in the design of the BGP routing protocol, and suggested changes to the way things are done that will reduce or get rid of the most serious risks. It also discusses the use of this precursor information to verify the path of the chosen route, as well as the protection of information provided in the AS-PATH characteristics about the second-to-last hop by digital signatures.

In [19], the researcher relied on the statistic pattern analysis technique to detect anomalies in the border gateway protocol, Where the patterns of the border gateway protocol features are analyzed, the internet data is monitored in the autonomous systems and the focus on changing the traffic values in the network. By changing the values of the features of the border gateway protocol, it is determined whether the instability in those values represents an anomaly for the BGP by an algorithm dedicated to it. In the intrusion detection system, a novel practical association rule mining approach for anomaly detection was proposed IDS. Define a quantitative association rule in a relational database, suggest a realistic technique for mining uncommon association rules from a network packet database, and demonstrate the benefits with an example. The technique can be used in fields where rare hidden patterns must be mined [21].

In [22], researchers using the apriori and FP growth algorithms. presenting a theoretical overview of association rules for various current algorithms, the concepts behind association rules are presented. A reliable increased security In order to overcome the Modbus TCP protocol's security flaw, which is a widespread problem in ICSs, the Modbus TCP protocol was developed. In order to guarantee dependability, the industrial control system uses the SM4 algorithm and the HMAC method to provide secrecy and integrity for communication data.

In [24], detecting suspicious flow anomalies using a histogram-based detector, then utilizing Association rule mining to locate and summarize anomalous flows.

In [23], the authors aimed to find a technique for detecting malware and Malicious applications on android platforms, where the apriori association rule mining algorithm has been improved with the Particle swarm based on Novell discovery dataset model, where the apriori association rule is used to create rules from the candidate detectors produced by swarm optimization. The extraction algorithm and these rule models are combined to categorize and identify harmful android applications.

In [25], authors used the association rules mining algorithm, where the apriori algorithm was used to extract appropriate features from raw data including rules and

recurring patterns. After that, the extracted features were used to classify the data and detect anomalies in telecom networks.

In [26], authors simulate the results obtained from the comparison between the several Real-time frequent itemsets on dataset and the normal frequent itemsets in the control systems, where the reliability parameter describes the abnormal behavior of the control systems based on the quantitative analysis. The different procedures involved in data mining using the Apriori algorithm and the realization mode are illustrated in great detail. The similarity factor on the health condition of the control systems is created using a comparison technique based on real-time frequent itemsets and typical frequent itemsets.

In [27], authors used neural network and the association rule analysis technique to build a traditional anomaly detection system based on the available dataset model. Where they used the apriori algorithm to generate rules from among the regular features and labels that were used to filter the anomalies value and find the average of the positive values and reduce the ratio of false positives.

In [29], the researcher presented a proposal for the apriori parallel algorithm and confirmed through the results of his study that the proposed apriori parallel algorithm processes arabic texts of large size with acceptable efficiency. The researcher conducted the experiment on the mapreduce model and noticed an improvement in performance as well as the strong association rules.

In [31], the researcher used the association rule mining technique to detect anomalies depends on network traffic dataset with its three algorithms, which are apriori, fuzzy-apriori and FP algorithms, to obtain results for that experiment. The researcher obtained the results that confirm that the fuzzy-apriori algorithm provides the best quality, and that the FP algorithm is the fastest in the timing of the search for the solution or the goal. Table 2.1 offers a comprehensive summary of the aforementioned resource schedulers.

Table 1.1. Comparison of several works on bgp anomaly detections summarized

	Main aspects	Enhancements (Strengths)	Limitations (weakness).
[3] (2019)	suggests a technique for detecting anomalous behavior and assessing reliability in control systems based on a quantitative examination of a reliability parameter.	The appropriate generating procedure and illustrative values of change behaviors for processing parameters are thoroughly shown.	-Lack of concentrate on determining the site of anomalous action. -Lack in system reliability analysis, the discrepancy problem, or increasing measure accuracy
[5,11] (2019,2019)	Clustering k-means and DBSCAN Unsupervised machine learning techniques algorithms are used to offer strategies for anomaly identification Were successfully deployed and demonstrated the ability to discover known abnormalities in historical events.	The upper box comprises the offline prediction and parameterization procedure. They test the optimum model(s) by applying them to known historical occurrences.	-Modeling may be improved to decrease false positives by including additional data sources to correlate. - Router syslogs including BGP session establishment signals might significantly increase the number of possibly triggered alerts.
[14,15] (2019,2017)	Identify the pattern of hijacking incidents in Samarinda by comparing apriori and hash-based data. According to the findings of this study,the pattern is frequently created by as many asfour5-itemset combinations,but the hashbased acquired pattern is frequently generated by three 3-itemset combinations.	The researchers look at an analysis based on a case study of a hijacking in Samarinda that used the Association rules technique with an apriori algorithm and a hash-based methodology. Apriori procedure that has been used to gather the itemsets commonly happened in order to assemble Association Rule	- Lack of clarity about the methodology used in the study. -Lack in the show of the results of the apriori algorithm and the promise of limiting minimum support and confidenceas threshould of it.
[27] (2020)	In terms of anomaly detection, deep neural network (DNN) technology was used. and creates a two-	Apriori association algorithm used to mine the association rules here between	-The associated rules that had resulted were not content of lift that

	tiered anomaly detection system using deep neural networks and association analysis. They conducted a thorough evaluation of experiments involving DNNs and other neural networks using publicly available datasets.	discretized features and the hidden normal label in the dataset, the mining association rules are then used to match the classified malicious traffic set and sieve out the miscategorized normal traffic.	measure the rate connection between them and conviction. -The deep neural network framework still has a lot of room for improvement.
[39] (2022)	Used a novel framework that combines numerous machine learning and data mining methodologies to automatically learn variational rules that are satisfied consistently in a given dataset when it comes to AUC and partial AUC.	-They present a novel method for learning invariant rules that are consistently satisfied in a given dataset that combines several machine learning and data mining techniques, including decision tree learning and association rule mining. -Experiment findings in many application areas demonstrate that technique may achieve equivalent or even superior performance under Receiver Operating.	-The proposed approach has several limitations. Where they leave the expansion of the procedure for other data types because it only works for tabular data. -Their method works only for datasets with a considerable number of features. -Lack of performance of their methods, where reduced if there are not enough features.
[41] (2021)	Compute the association discrepancy, they propose the anomaly transformer in conjunction with a novel anomaly-attention method. To increase the normal-abnormal they hypothesized of the association discrepancy, a minimax method is designed.	They propose the anomaly amplifier with an anomaly-attention methodology based on the essential discovery of association discrepancy, which can model the past association and series discrepancy. -Present minimax technique to increase the normal-abnormal and others of the association discrepancy or further construct a novel association-based detection criteria.	-They did not offer a theoretical examination of the anomaly amplifier in light of traditional analyses for autoregressive and state space models. -Only with the reconstructive loss do the normal and abnormal time points perform similarly in the association weights to neighboring time points, resulting in a contrast value close to one. - The statistical findings of nearby.

1.6. ORGANIZATION OF THESIS

This thesis used association rule mining algorithms based-weka to detect an anomaly in the border gateway protocol (BGP) over time and identify the flows connected with the event(s) that caused the anomaly. This thesis consists of seven parts, and the rest of the thesis is organized as follows: Part 2, an explanation of border gateway protocol and its mechanism. Part 3 discuss the BGP anomalies and its categories and anomaly detection techniques. Part4 focused on an association rule mining technique algorithms and method of improvement apriori algorithm. Part5 includes the proposed methodology for the work of this thesis, which also contained a a detail study of dataset. Part 6 deals with the results achieved and discuss them. Part 7 includes both the conclusion and future work.

PART 2

BORDER GATEWAY PROTOCOL(BGP)

2.1. OVERVIEW

BGP is divided into two types: Internal BGP (IBGP), which connects BGP routers inside an AS to external BGP routers (EBGP),which runs between BGP routers across ASs. On the other hand, are connected by a dedicated link between peers or a third party like the Internet Exchange Point (IXP). Over the years, BGP has experienced multiple revisions and modifications. Version 4 of BGP is now in use, as documented in RFC4271[5]. figure2.1 shows an infrastructure of autonomous systems in a network.

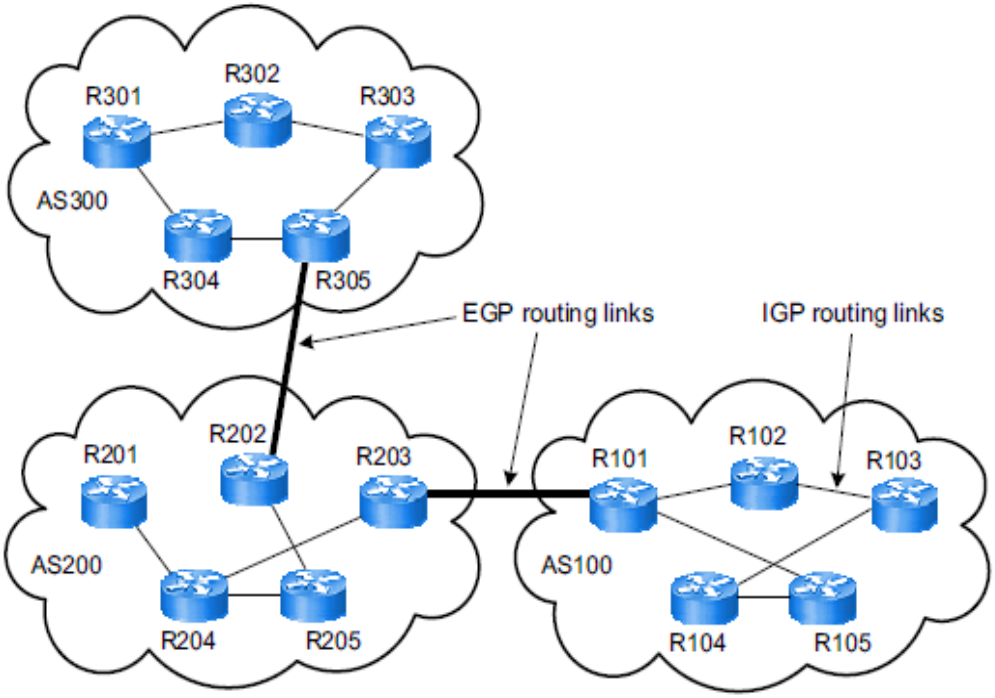


Figure 2.1. BGPs Autonomus systems [5].

2.2. BGP CONFIGURATION AND CONNECTION

The BGP routing process uses BGP policy configuration to filter routes from incoming and outgoing ads and to manage prefix processing. The number of prefixes that will be accepted by the routing mechanism, BGP timing settings, BGP prefix dampening settings, and how BGP uses path attributes can all be changed to regulate prefix processing [6]. Table 2.1. shown the main configration of BGP with autonomous system with routers that connected.

Table 2.2. BGP configuration [7].

Router	Autonomous system(AS)	BGP Configuration
R1	12	router BGP 12 neighbor 192.0.2.1 remote-as 12 neighbor 192.0.2.1 update-source Loopback0
R2	12	router BGP 12 neighbor 192.0.3.1 remote-as 34 neighbor 192.0.4.1 remote-as 34 neighbor 192.0.5.1 remote-as 12 neighbor 192.0.5.1 update-source Loopback0
R3	34	router bgp 34 neighbor 192.0.4.2 remote-as 12 neighbor 192.0.7.1 remote-as 56 neighbor 192.0.8.1 remote-as 34 neighbor 192.0.8.1 update-source Loopback0
R4	34	router bgp 34 neighbor 192.0.3.2 remote-as 12 neighbor 192.0.9.1 remote-as 56 neighbor 192.0.10.1 remote-as 34 neighbor 192.0.10.1 update-source Loopback0

Autonomous-system-path access lists, route maps, filter lists, IP prefix lists, and IP policy lists are all examples of IP routing information, and distribution lists are used to filter prefixes in inbound and outgoing ads. Both outbound and inbound refer to traffic or anything that leaves or enters my domain. The processing order of the BGP policy filters is displayed in the Table 2.2. below:

Table 2.3. Order of BGP policy processing [6].

Inbound	**Outbound**
Route map	Distribute list
AS-path access list, Filter list, or IP policy	IP prefix-list
IP prefix-list	Filter list, AS-path access list, or IP policy
Distribute list	Route map

2.3. BGP NEIGHBOR RELATIONSHIP

Before exchanging routing information, BGP routers must first establish a neighbor adjacency. In contrast to other routing systems that employ multicast or broadcast to find their neighbors, BGP neighbors are specified explicitly [6]. BGP establishes a BGP session through a TCP connection on TCP port 179 with its peers or neighboring routers. A BGP Finite State Machine (FSM) is being used to maintain the BGP table, which includes the peers and operational status. To initiate a BGP session, a BGP FSM must lead the routers through the various BGP states [6]. Figure 2.2. shows a topology of border gateway protocol that has been used in networking with multiple family addresses.

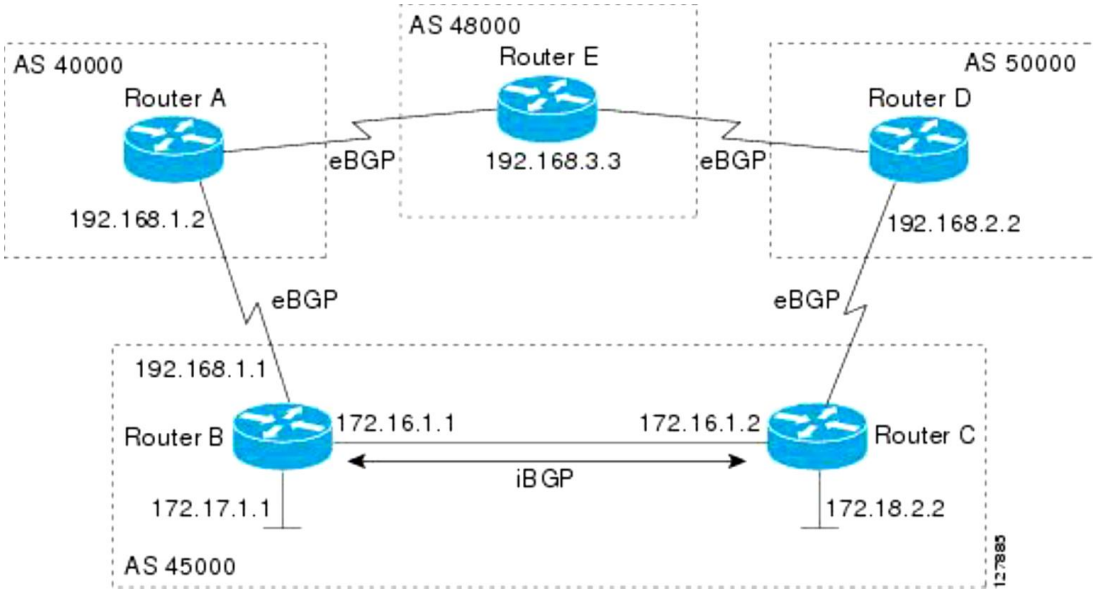


Figure 2.1. Topology of BGP network for multiple family addresses [6].

To discous the essence of the BGP, we have summarized the main stations and foundations of the BGP to understand its mechanism of action and its most contents, which will be addressed in detail. Figure 2.3. Shows the important concepts in BGP.

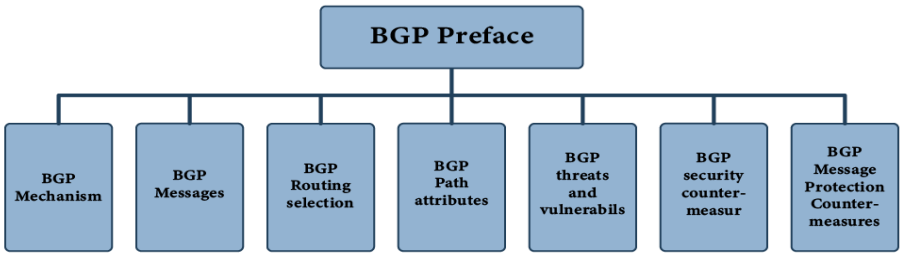


Figure 2.2. BGP preface Flowchart

2.4. BGP MESSAGES

The border gateway protocols two basic actions are route announcement and withdrawal. Update messages are used to exchange routing info. Although BGP includes three more communication types, none are directly related to routing. A route is made up of a set of IP prefixes and attributes.

2.4.1. Transmission of BGP Messages:

2.4.1.1. Speaker

Any router that transmits BGP messages is referred to as a BGP speaker. The speaker obtains or develops fresh route information and then broadcasts it to other BGP speakers. A BGP speaker compares a route received from another AS with its own local routes. If the route is superior than its local routes or brand-new, the speaker announces it to all other BGP speakers.

2.4.1.2. Peer

Peers are BGP speakers who send and receive messages [8]. We will show some wireshark captures to illustrate the BGP messages.

For this, we will utilize the topology shown below:

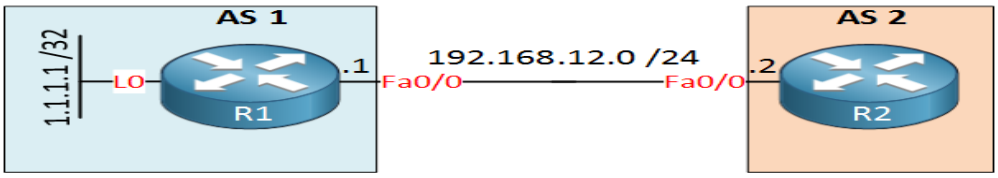


Figure 2.3. Topology of BGP message connection [8].

Figure 2.4. shows how a mechanism of peer in BGP to processes of sending or receiving the messages. Regarding the BGP messages, open, notification, update, keepalive, and route-refresh are the four types of messages sent by BGP as we show it below:

2.4.1.2.1. Open

After a TCP connection is established, the first communication transmitted is an open message, which is used to establish BGP peer connections. After receiving an open message and successfully completing, a peer sends a keepalive message to affirm and preserve the peer connection during peer negotiation. Then, peers may send each other updates, notifications, keepalive, and route-refresh messages.

2.4.1.2.2. Update

BGP peers exchange routes using this form of communication:

1. Multiple reachable routes with the same properties can be advertised in an update message. These route attributes are applicable to all destination addresses (as defined by IP prefixes) in the update messages Network Layer Reachability Information (NLRI) field.

2. Multiple unreachable routes can be removed using an update message. Each route is identifiable by its destination address (as determined by the IP prefix), which identifies routes that were previously broadcast between BGP speakers.

3. The only permitted use of an update message is to remove a route. It is not necessary to include the route characteristics or the NLRI in this scenario. Furthermore, update message may only be used to promote accessible routes,

in this case, it is unnecessary to transmit information regarding eliminated routes.

2.4.1.2.3. Notification

BGP sends a notification message to its peer when it finds a problem. The BGP connection then is immediately terminated.

2.4.1.2.4. Keepalive

BGP sends keepalive signals to peers on a regular basis to sustain peer connections.

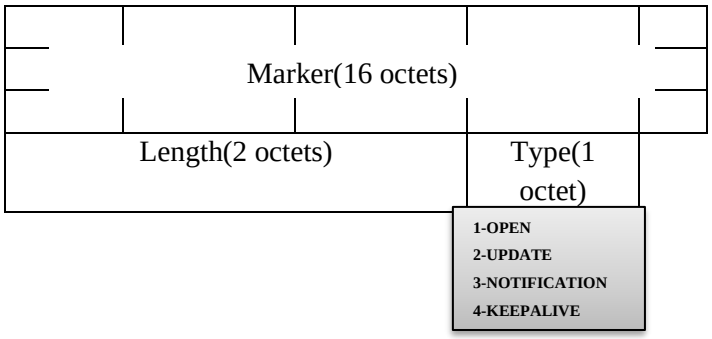


Figure 2.4. Common message of BGP header format [1].

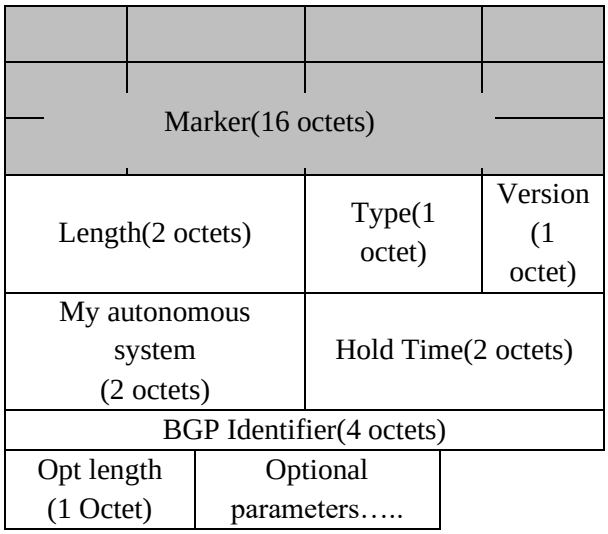


Figure 2.5. BGP open message format [1].

Figure 2.6. shown if all BGP routers are configured with the route-refresh capability and the BGP import policy changes, in this case the local BGP router sends its peers a route-refresh message. The peers transmit their routing information to the local BGP router after receiving a route-refresh message. In this approach, BGP routing tables were dynamically updated, and new routing rules were deployed without interrupting BGP connections [9].

2.5. BGP ROUTING SELECTION

BGP uses a lot of attributes to transmit different types of information. For instance, the as-path element is a list of ASes that the routing message traverses before arriving at the current AS. BGP, as a route vector protocol representation, makes routing decisions based on the attribute. Each AS may utilize these attributes to construct its policies, such as import policies, optimum route selection, and export policies. Furthermore, each AS can have an external impact on BGP by specifying some specific attributes and custom exports and imports policies [10]. The following is a list of the most important BGP attributes pertinent to this study:

NLRI : stands for network layer shortest path information, which is defined as the destination IP prefix.

NEXT HOP: a next-hop routers IP address.

AS-PATH: an organized list of ASes crossed by the announcement of a route.

LOCAL PREF: local selection (a local route ranking value that may be provided and passed inside an AS).

Even while routers are linked to the internet through a vast number of physical connections, each ASes routing policy limits the available routes [10]. Different import and export rules can be customized by network operators based on their economic and security demands. Before issuing route announcements to its neighbors, each AS will examine its export policies [10]. Only routes that fulfill the policies requirements may be propagated. When such notification reaches an AS, it may also be filtered according to the ASes import tariffs.

2.5.1. BGP Path Attributes

BGP develops routing rules based on the attributes associated with each route, which are used to identify the shortest path across many autonomous systems [8]. Additionally, one or more routing rules govern it [7]. By default, BGP selects a single route as the optimal path to a target host or network. The best-path selection algorithm analyzes path attributes to decide which BGP routing table route should be implemented as the best path. BGP best-path analysis considers a variety of attributes for each path [7].

By changing these attributes via the command-line interface, cisco IOS software can influence BGP path selection CLI. Standard BGP policy setup can also influence BGP path selection. BGP finds a group of equally good routes using the best-path selection technique. These are the multipath that might exist. In cisco IOS Release 12.2(33) SRD and later, when there are more equally good multipath available than the maximum permitted number, the oldest paths are selected as multipath [6].

In update messages, BGP can contain path attribute information. Its attributes specify a routes characteristics, and the program utilizes these to make judgments as to which routes to broadcast. A BGP-speaking network connection can configure some of these attributes. Some properties must always be included with the updating message and others are optional. BGP path attributes are classified into four categories well-known necessary and well-known discretionary, optional transitive, and optional non-transitive [8]. The majority of these attributes must be provided in each update message and are required by all BGP implementations, while some are optional. Optional characteristics are qualities that one or more parties may add to a route, however, all BGP implementations are not required to include all optional attributes. The origin of any BGP speaker in the route may attach additional optional transitive characteristics to the path; the conditions for adding further optional non-transitive attributes are specified by the type of the particular attribute. As stated in Table 2.3, several BGP attributes are managed by the local AS administrator or the administrator of the neighboring AS [7].

Table 2.4. Attributes of BGP [7].

Attribute	Category	Controlled by Local AS / Neighbor AS
AS_Path	Well-known mandatory	Local AS
Origin	Well-known mandatory	Neither
Next_Hop	Well-known mandatory	Local AS
Atomic aggregate	Well-known discretionary	Neighbor AS
Local_Pref	Well-known discretionary	Local AS
Multi_Exit_Discriminator(MED)	Optional non-transitive	Neighbor AS
Aggregator	Optional transitive	Local AS
Community	Optional transitive	Local AS

As we see in Table 2.3. that lists the BGP attributes and associated categories which we will discuss in detail.

2.5.1.1. As_path

This property holds a collection or list of numbers of an autonomous system through which routing data has passed. When forwarding the update message to external peers, The BGP speaker includes its own autonomous system in the list.

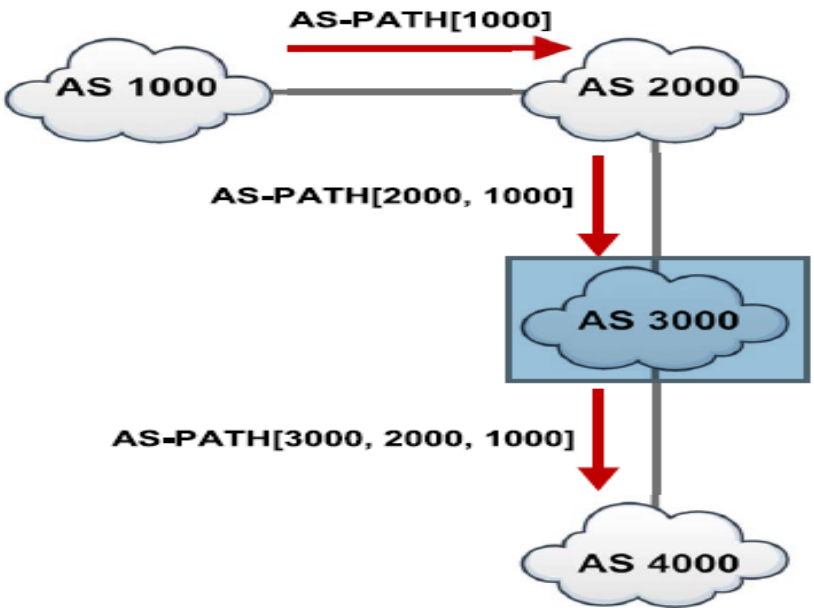


Figure 2.7. Example of BGP AS_PATH attribute [1].

2.5.1.2. Origin

This property describes how well the route was added to the BGP routing table. An interior gateway protocol origin code is assigned to a route formed using the BGP network command in cisco software. Exterior Gateway Protocol routes are coded with an EGP origin, whereas routes redistributed from those other protocols are labeled Incomplete. For origin, the BGP decision plan favors IGP placed above a white EGP, followed by EGP over Incomplete [6].

2.5.1.3. Next_hop

The next_hop element specifies the IP address that will be used as the BGP next hop for the destination. The router does a recursive query in the routing table to determine the BGP next hop. In external BGP (EBGP), the next hop is the IP address of the peer that provided the update. Internal BGP (IBGP) allocates the next_hop address to a peers IP address, which broadcasts the prefix for the route that originated locally [7]. The next_hop attribute remains unchanged when any routes that did learn from EBGP are advertised to IBGP. In addition for a router to use a BGP routing, its next_hop IP address must be reachable.

2.5.1.4. Atomic Aggregate

BGP peers are informed by the atomic aggregate characteristic that the local router is utilizing a less specified (aggregated) route to a destination. When a BGP speaker selects a less specific route if a more specific route is available, it is required to propagate the route with the atomic aggregate characteristic. The atomic aggregate property informs BGP neighbors that the BGP speaker utilized a route that was aggregated. The aggregator attribute includes the AS number and IP address of the router that created the aggregated route. RID of the router that conducts route aggregation is the IP address in cisco routers.

2.5.1.5. Local_Pref

Within an autonomous system, the local_pref property will be included in each update message between BGP peers. If many paths go to the same destination, the local preference property with the highest value indicates the preferred departing route from the local autonomous system. Internal peers are advised of the route with the greatest rating. Local Prefs value is not sent to exterior peers [6].

2.5.1.6. Multi_Exit_Discriminator

The MED attribute indicates a favored path into such an autonomous system (to an external peer). If an autonomous system has many entry points, the MED might be used to convince another autonomous system to choose a certain entry point. A metric is assigned, with the program preferring a lower MED measure more than a higher MED metric. Between autonomous systems, the MED metric is exchanged. However, the MED metric is reset to 0 once a MED is passed into an autonomous system. When an update is sent to an internal BGP (iBGP) peer, the MED remains unaltered, enabling all peers in the same autonomous system to pick the same route [6].

2.5.1.7. Aggregator

Route Aggregation(RA) attribute, also known as BGP Route summarization, is a technique for reducing the routing tables size. Non-aggregation routing, in which individual sub-prefixes of address block are published to BGP peers, is the polar opposite of RA shrinks the global routing database, lightens the strain on routers, and saves network traffic.

2.5.1.8. Community

Irrespective of the network, autonomous system, or physical borders, the BGP community is used to organize networking objects that have similar attributes. In big networks, using prefix lists or access lists to impose a common routing strategy necessitates separate peer declarations on each networking device BGP neighbors with common routing rules may utilize the BGP community attribute to build inbound or

outbound route filters based on the community tag, as opposed to reading vast lists of individual permit or deny statements [6].

2.6. ESTABLISHMENT OF BGP SESSION

Establishing a BGP session differs from establishing IGPs such as EIRP and OSPF. IGPs adapt to the topology by bootstrapping and finding neighbors dynamically. BGP speakers must be explicitly established to test peering sessions with another BGP speaker. In addition, BGP must wait until a stable connection is established before continuing with the remainder of the session setup. The justification for this need is the enhancements made to BGP to solve issues with its predecessor, EGP as the extent of the internet grew, so did the size of EGPs update messages, to the point that numerous IP pieces were required to broadcast the changes. Some of these parts were lost in transit, resulting in significant internet routing flaws. To avoid a scenario like this with BGP, a trustworthy mechanism for delivering BGP messages was required [11]. The developers may have built a new transport protocol for reliable exchange, or they may have used an existing transport protocol for reliable delivery. Instead of recreating the wheel, the creators of BGP opted to use TCPs established solid dependability mechanisms. BGP session setup is split into two steps as a result of this interaction with TCP.

2.6.1. TCP Connection Establishment.

Throughout the two phases of session establishment, BGP employs a finite state machine (FSM) to keep track of session establishment with an intended BGP peer. A finite state machine is a device that allows an entity in this example, a machine–to function in a limited number of states. Each state has a distinct mission and set of operations. At any one time, the machine is only in one of these states. Input events cause a state change to occur. The FSM of BGP contains a total of six states [11]. The TCP connection establishing phase is represented by the following three states of BGPs FSM:

- Idle
- Connect

- Active

2.6.2. BGP Session Establishment.

TCP messages are sent in these stages to establish the TCP connection necessary for reliable BGP message delivery. BGP enters the following three stages of the BGP FSM, which correspond to the BGP session establishment phase, after the TCP connection establishment phase:

- Opensent
- Open confirm
- Established

BGP exchanges communications related to the BGP session in these stages. The Opensent and Open Confirm statuses describe how BGP session characteristics are exchanged between BGP speakers. The established state denotes that the peer is stable and capable of accepting BGP routing updates [11]. Those six states make up the complete BGP and FSM when combined. Each intended peer has its own FSM, which BGP keeps track of. The reception of input events causes a peer to move between these states. The BGP speaker, for example, transmits an open message and enters the opensent state when a connection of TCP is successfully originating in the connect or active stages. An error that occurs in any state, however, might force the peer to enter the IDLE state [12].

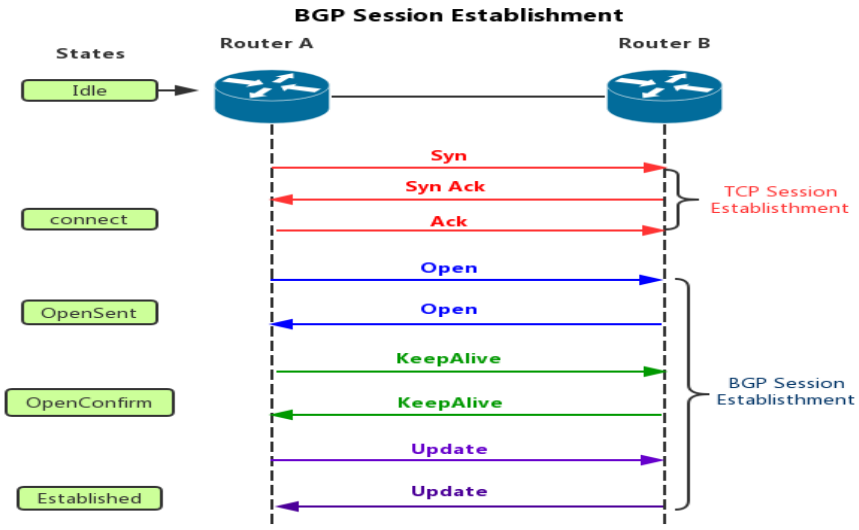


Figure 2.8. Establishment of BGP session [11].

2.7. BGP THREATS AND VULNERABILITIES

We have identified the attacks that BGP is vulnerable to, as well as the weaknesses that these threats exploit [14]. Threats to the flow of data traffic including parts of the routing infrastructure are considered independently from risks to the flow of routing traffic. When we talk about attacks, there are two different kinds of internet nodes that we talk about: authorized BGP speakers and intruders. Nodes that the authoritative network administrator has chosen to perform the functions of BGP speakers are referred to as authorized BGP speakers [13].

2.7.1. Intruders

We assume that an intruder can be identified at every point in the network through which all relevant communication flows and that this intruder can generate, replay, monitor, change, or remove anything like this traffic. Based on this description, we may be able to recognize the four distinct categories of attackers in a BGP environment:

2.7.1.1. Subverted BGP Speaker

When an authorized BGP speaker is compelled to violate BGP protocols or improperly assert control over network resources, this is known as a subverted BGP speaker. This is often caused by defects in the BGP software, errors in the speakers configuration, or by compelling a BGP speaker to load unapproved software or configuration information, which may be achieved in several ways based on the BGP speakers architecture and setting [13].

2.7.1.2. Unauthorized BGP Speaker

When a node that is not allowed to be a BGP speaker circumvents access control systems and establishes a BGP connection with an authorized BGP speaker, it really is considered to as an unauthorized BGP speaker. How this is achieved will be determined by the design and configuration of current control mechanisms.

2.7.1.3. Masquerading BGP Speaker

A masquerading BGP speaker is a node that successfully assumes the identity of an authorized BGP speaker. To accomplish this, IP spoofing and source routing attacks can be used.

2.7.1.4. Subverted Link

This type can take several different shapes. One must have access to the actual media so that the channel may be managed. Furthermore, a link may be subverted by exploiting the links lower layer protocols in such a manner that the channel can be controlled. The TCP session hijacking attack is an example of this attack [14].

2.7.2. Threats to Routing Information

An intruder may create, modify, replicate, or eliminate routing traffic under the correct conditions. With these skills, an attacker may compromise the network in a number of ways. The alteration or fabrication of routing updates enables an attacker to alter the logical routing structure of the internet, potentially resulting in-network service denial, network traffic disclosure, and false network resource accounting. Replaying or deleting routing updates either stops the creation of logical routing structure subsets (in response to structural or policy modifications) or resets it to its previous state with the same results. These attacks exploit the absence of access control, authentication, and integrity of BGP message contents. Furthermore, gaining access to traffic routing is quite simple for an intruder [15]. Among the facts available from this traffic are the next hop required to reach a destination and the route traveled by traffic to targeted destinations. The next_hop information may be retrieved from several sources, including monitoring authorized traffic to the desired destination for the next_hop it uses. Therefore, the next_hop information cannot be protected just by traffic routing methods. In some instances, the route traveled to reach targeted locations may be considered secret. The weaknesses exploited by these attacks include the lack of secrecy of peer connections and the degree of confidence put in BGP speakers [15].

2.8. BGP SECURITY COUNTERMEASURES

There are two different forms of communication used in BGP, as well as in routing protocols in general: communication between neighbors and communication among a particular speaker as well as a completely random collection of distant speakers that are produced in real-time by routing selections. The communication between neighboring speakers is made up of the routing updates for destinations that the sender has judged are suitable to convey to the receiver. The sender has also concluded that it is acceptable to communicate these updates to the receiver [15]. Communication between a speaker and distant speakers is comprised of fields of routing updates that define a particular destination. As a result, we propose the two types of counter-measures below:-

2.8.1. BGP Message Protection Countermeasures

These countermeasures are designed to guarantee that routing communications between BGP peers, the first kind of communication described above, are authenticated, secret, and secure. Particularly, the message encryption and message sequence number offer methods for corrupt detection, sequence, confirmation, and repetition. This processes are redundant with those provided by TCP but are necessary owing to the TCP mechanism vulnerability [16-17]. Ideal implementation of these countermeasures would occur at the network or transport layer. Using a secure network or transport protocol would render these BGP defenses unnecessary.

The following two steps demonstrate how to encrypt BGP messages:

1. Encrypt all BGP messages sent by neighbors using session keys that were exchanged at the establishment of the BGP connection. This encryption assures the authenticity and integrity of all route attributes with a maximum validity of one AS hop, as well as the privacy of all routed exchanges.
2. To prevent messages from being replayed or deleted, provide a message sequence number [13].

PART 3

BGP ANOMALIES

BGP anomalies can result in anything from a single abnormal BGP update to hundreds more anomalous BGP updates. A BGP update is considered anomalous if it includes an inaccurate AS number, invalid or reserved IP prefixes, a prefix published by an unlawful AS, an as-path without a physical counterpart, or if it does not adhere to a common routing principle [18]. If the amount of BGP updates fluctuates rapidly, or if the behavior of total BGP traffic changes over time, a set of BGP updates might be characterized as an anomaly [19].

3.1. CATEGORIES OF BGP ANOMALIES

We create a taxonomy of BGP anomalies with the following four primary categories:

3.1.1. Direct Unintended Anomaly

This sort of anomaly pertains to BGP configuration errors made by BGP router operators [1]. Origin misconfiguration and export misconfiguration are two categories of BGP misconfiguration Origin, misconfiguration happens when an operator inadvertently declares prefixes they do not possess or fails to filter private ASes. Export configuration error happens when operators in advertently define BGP policies, by blocking certain approved routes, for instance, and generating DOS to the blocked prefixes [1].

3.1.2. Directly Intended Anomaly

This sort of anomaly relates to all forms of BGP hijacking that may occur in a variety of circumstances.

3.1.2.1. Prefix Hijack

In this sort of hijack, an attacker configures their BGP router to advertise a prefix from a different AS.

3.1.2.2. Prefix and AS Hijack

In this situation, an attacker broadcasts a direct connection between its AS and a victims AS, forcing the victims AS to divert its routes to the attacker.

3.1.2.3. Sub-Prefix Hijack

In this situation, the attacker publishes a sub-prefix belonging to the victims autonomous system.

3.1.2.4. Sub-Prefix and Its AS Hijack

In this instance, the attacker broadcasts a spoofed route to a subnet inside the target prefix.

3.1.2.5. Hijack Legitimate Path

This form of hijacking does not need the attacker to make any notifications.

3.1.3. Indirect Anomaly

This sort of anomaly relates to hostile attacks on internet components like web servers. BGP is a routing protocol used to manage internet reachability information between autonomous systems (ASes), however, it was unstable during the Nimda, Code Red I, and Slammer worm assaults.

3.1.4. Link Failure

ASes are peering with one another either privately, using a secure connection between peers, or publicly, through a third party such as an IXP [1]. Figure 3.1. shows all the main categories of BGP anomalies.

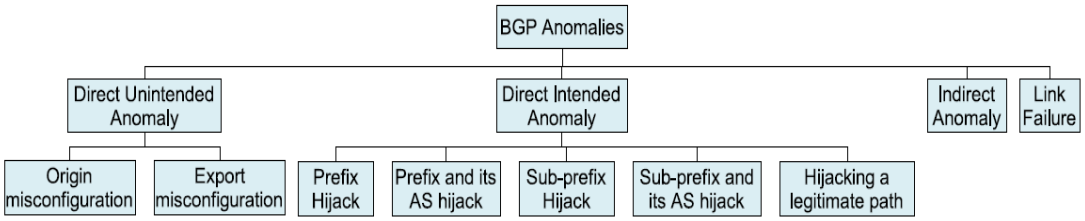


Figure 3.1. A main categories of BGP anomalies.

3.2. ROUTING ANOMALIES

There are anomalies in the Internet routing planes regularly and can last anywhere from a few seconds to several months [18]. This section will offer a summary of how anomalies arise and how to react to them. BGP protocol focused on sending and receiving messages. ASes send signals to their physically connected neighbors in other ASes, notifying them of their own IP prefix x and inductively accessible IP prefixes of many other ASes. These messages include the registered AS number of the peering AS, the AS Number of the originating AS, i.e. the AS which own the announced prefix, and the AS numbers of all ASs between the receiving and originating AS, i.e. the AS route [18].

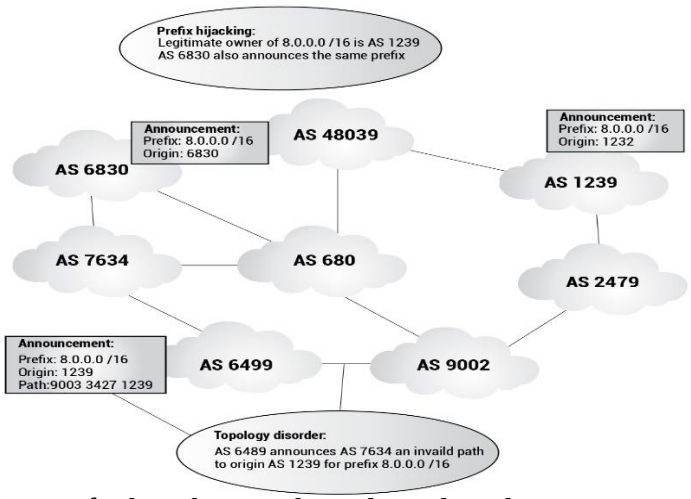


Figure 3.2. Prefix hijacking and topology disorder routing anomalies [18].

3.3. ANOMALY DETECTION

Anomalies are outlier data points in a dataset that contradict the data typical trend. These data points or data deviate from the typical activity patterns of the dataset in real-world datasets, anomalies and outliers are frequent. They may be caused by data corruption, failed experiments, or human error. Because the existence of anomalies might affect the models performance, the dataset should be devoid of an anomaly in order to train a robust data science model [20]. Anomaly detection is a method of processing data that does include an unsupervision data process that may be used to identify anomalies in a dataset.

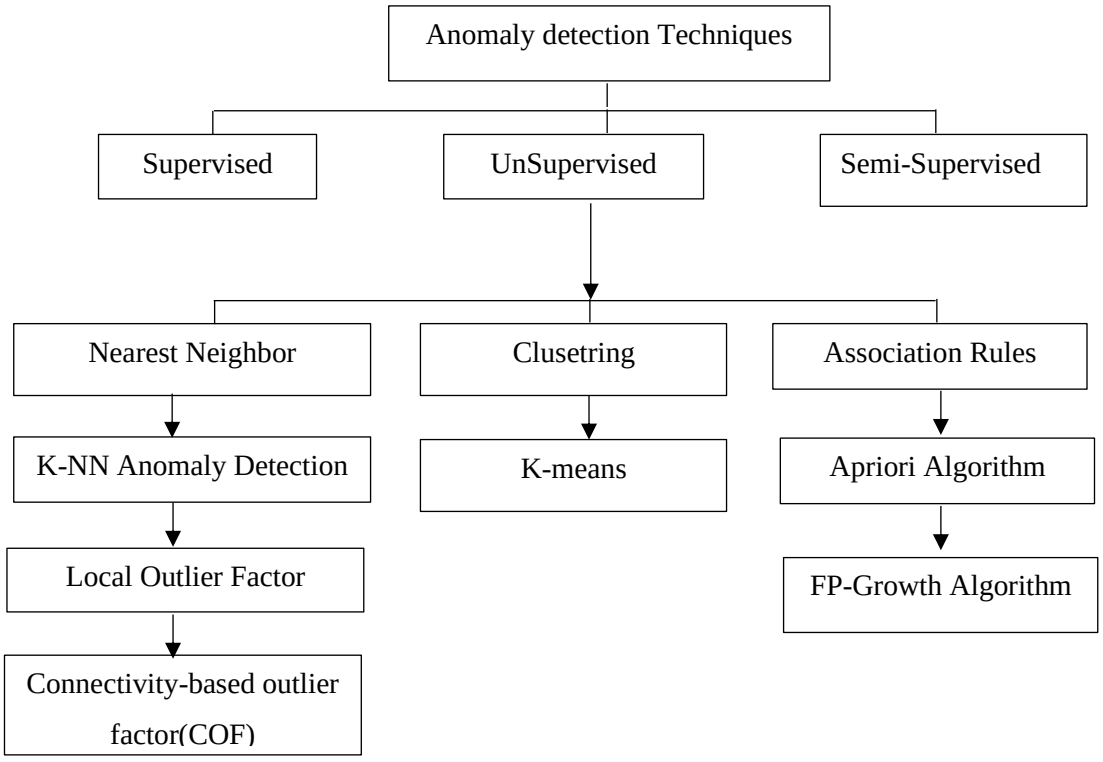


Figure 3.3. System of unsupervised anomaly detection [20].

3.4. BGP ANOMALY DETECTION TECHNIQUES

Detecting BGP anomalies helps network operators to protect their network against the most severe repercussions of anomalous behavior [1]. Numerous techniques, systems, and approaches for detecting BGP anomalies have been published. We explore BGP anomaly detection approaches based on three factors:

1. Used BGP data sources.
2. Observed and focused on BGP features.
3. Other approaches that are used for detection.

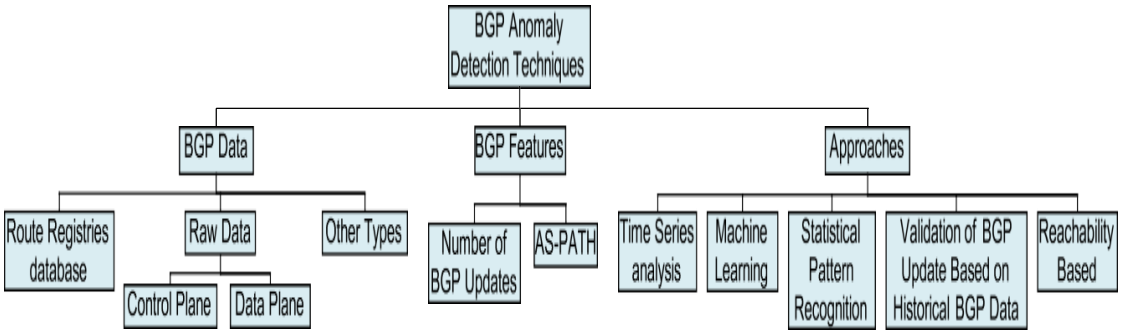


Figure 3.4. A techniquies of BGP anomaly detection

PART 4

DATA MINING AND ASSOCIATION RULE TECHNIQUE ALGORITHMS

Data mining is the process of extracting meaningful information from vast quantities of data stored in databases, data warehouses, and other datasets [21]. Patterns, correlations, alterations, anomalies, and notable structures are examples of knowledge that is beneficial. Specifically, over the last decade, studies on association rule mining have already been widely conducted in a variety of application fields. Association rules it is if-then statements that aid in identifying associations between apparently unrelated facts in a relational dataset or similar repository of information. Market basket analysis, which tries to uncover relationships between itemsets in a transaction database, is a similar example of association rule mining [22]. The frequent itemsets in the transaction database are used to find relationships. As the market basket analysis is the origin of the association rule mining issue, typical association rule mining algorithms, such as apriori and FP growth, are centered on effectively locating frequent itemsets. In a database for market basket analysis, there are many more infrequent itemsets than frequent ones, and these methods can be enhanced by dealing with such infrequent itemsets as little as feasible [22].

4.1. AN ITEMSET

An itemset is a collection of related items. A k-itemset is any item set that has k items. A set consists of two or more objects a frequent itemset is a group of items that occur often. As a result, frequent itemset mining is a data mining approach for identifying items that frequently appear together [23].

4.2. A FREQUENT ITEMSET

A set is deemed to be frequent if it satisfies a minimal threshold value for support and confidence. Support shows transactions in which many goods were bought in a single purchase. One-by-one goods purchases are represented by the term confidence. We only analyze transactions that satisfy the approachs minimal threshold support and confidence constraints. These mining algorithms give a multitude of advantages, including cost reductions and competitive advantages. For frequent data mining, there is indeed a tradeoff between data mining time and data volume. A frequent mining technique is an efficient method for detecting hidden patterns within item sets quickly and with low memory use [23].

4.3. FREQUENT ITEMSET MINING

Frequent itemset or pattern mining is extensively used because of its many applications in mining association rules, correlations, and graphing pattern restrictions based on frequent patterns, sequential patterns, and other data mining tasks.

4.4. FREQUENT PATTERN MINING (FPM)

The frequent pattern mining method is one of the most significant data mining techniques for detecting relationships between diverse components in a collection. These ties are represented by association rules. It facilitates the identification of data abnormalities. FPM has several applications in areas such as data analysis, software problems, cross-marketing, sale campaign analysis, and market basket analysis, among others. Frequent item sets found a priori may be used in a range of data mining tasks. Finding exciting patterns within the database, defining sequences…etc, are all necessary jobs. The most essential of these is association rule mining. The frequency with which the things are purchased is described by the association rules [23].

4.5. ASSOCIATION RULES MINING

Association rules are mined out after frequent itemsets in a big dataset are found. Mining methods like apriori and FP growth are used to identify these datasets. Using support and confidence measures, frequent itemset mining mines data [4]. Mining ssociation rule is defined as Let *I= { I1, I2, ..., Im}* a set of *'m'* with different attributes that are called items. Let *D= { ....}* A set of transactions is known as a dataset. With different transactions Ts. Each transaction in D includes a subset of the items in I and has a unique transaction identifier. The rule is described as an insinuation of form *X->Y* where *X, Y⊂ I* are collections of objects called itemsets and *X∩Y=∅*. X and Y are both referred to as antecedents. The rule states that X implies Y [25]. Association rule learning is used to determine the associations between attributes in huge datasets. For a group of transactions, an association rule X=> Y will take the form [24]. Support(s) and confidence(c) are two important basic indicators of association rules. Users are primarily concerned with the commonly purchased things because the database is so large. Users may pre-define support and confidence thresholds to avoid unnecessary rules. The two thresholds are Minimum support and Confidence [25]. The following is an instance of support and confidence:

$$Bread => butter\ [support = 2\%, confidence - 60\%] \quad (4.1)$$

An association rule like the above statement. This indicates that 2% of consumers purchased daily bread together, whereas 60% of customers purchased bread and butter individually [23].

The formulas show that there is support and confidence in Itemsets X and Y:

$$Support\ (X) = \frac{transaction\ Number\ \ in\ which\ X\ appears}{Total\ transactions\ number}\ , support(X \cup Y) = \frac{support\ sum\ of\ X \cup Y}{overall\ records\ in\ dataset} \quad (4.2)$$

$$Confidence\ (X \to Y) = \frac{Support(XUY)}{Support\ (X)} \quad (4.3)$$

Where the X, Y is the item or tranzaction or value that done in the process of dataset, $Support$ is the minimum value that configure to the number of repeating time of X or Y and a $Confidence$ is the range of validate the $(X \rightarrow Y)$ process.

The mining of association rules is done in two stages:

1. Find frequent itemsets.
2. Generate rules associated with frequent itemsets.

Figure 4.1. shown the process of Data miming with association rule and the main ideas about Association rule technique over dataset.

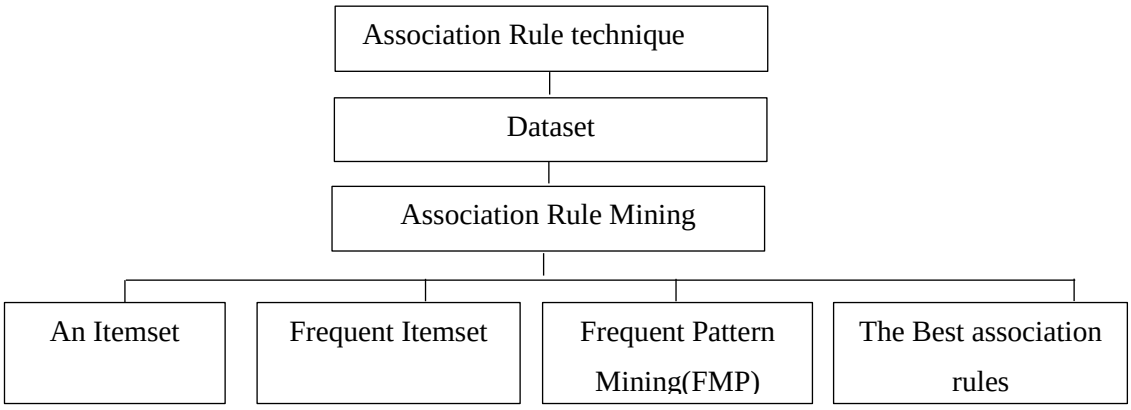


Figure 4.2. Data miming with association rule concepts flowchart

4.6. ASSOCIATION RULE TECHNIQUE AND WEKA

Weka is a Java-based package of cutting-edge machine learning algorithms and preprocessing data tools created at the new zealand university of waikato. It is GNU General Public License-licensed free software that runs on practically every platform. It may be used in many different data mining strategies. It offers comprehensive assistance for full experimental data mining operations, including data preprocessing, evaluation of learning schemes statistically, and the visualization of the data input and the results of the learning [26]. Weka workbench includes algorithms for regression, classification, clustering, association rule mining, and attribute selection. It can be used in any of the two connections below:–

1. The Command Line Interface(CLI)

2. The Graphical User Interface (GUI)

To find frequent subsets and compare them, use the modified apriori method and the filtred associator algorithm. We must first preprocess a dataset before applying association rule mining to it. The weka program was utilized to complete the task as well as all association rule algorithm operations on the dataset [26].

Weka supports different algorithms for data mining and has different algorithms for association rule modeling such as apriori, filtered- associator, FP growth,… etc. But KNIME does not provide a special algorithm.

Table 4.1. compares between WEKA and KNIME with its difference in the main features.

Table 4.1. Comparasion a features of rule mining in WEKA and KNIME

FEATURES	WEKA	KNIME
PreProcessing	Yes	yes
Rule generation Count	Yes	--
Support Count	Yes	yes
ItemSet	Yes	yes
Confidence	Yes	yes

Table 4.1., which compares the applications of Weka and knime, and through the features for each of them, we notice that the two applications are common to most of the features and have the same value, except that the difference of the Rule generation Count feature in Knime is not supported. As we There are several approaches to association rules. In this study, we used the apriori algorithm and FP growth approaches, which are often used in NIDS.

4.6.1. Apriori Algorithm

For frequent itemset mining, the apriori technique was the first algorithm. Later, it was improved by R agarwal and R srikant, and it was given the name apriori. This strategy includes two phases to reduce the search space join and prune. It is an iterative technique for determining the most prevalent itemsets [23]. In apriori assuming that item I does not frequent often, the probability is if:

- *P(I)* < minimum support threshold, then *I* is not frequent.

- *P (I+A)* < minimum support threshold, then *I+A* is not frequent, where *A* also belongs to itemset.

If the value of an itemset is less than the minimal support, its supersets will likewise be less than the minimum support and may thus be disregarded. The apriori algorithm for data mining includes the following steps:

4.6.1.1. Join Step

This process generates (K+1) itemsets from K-itemsets by connecting each item to itself.

4.6.1.2. Prune Step

This step counts each database item. If a candidate item does not meet the minimal support standards, it is categorized as infrequent and subsequently eliminated. This step is intended to reduce the size of candidate item sets.

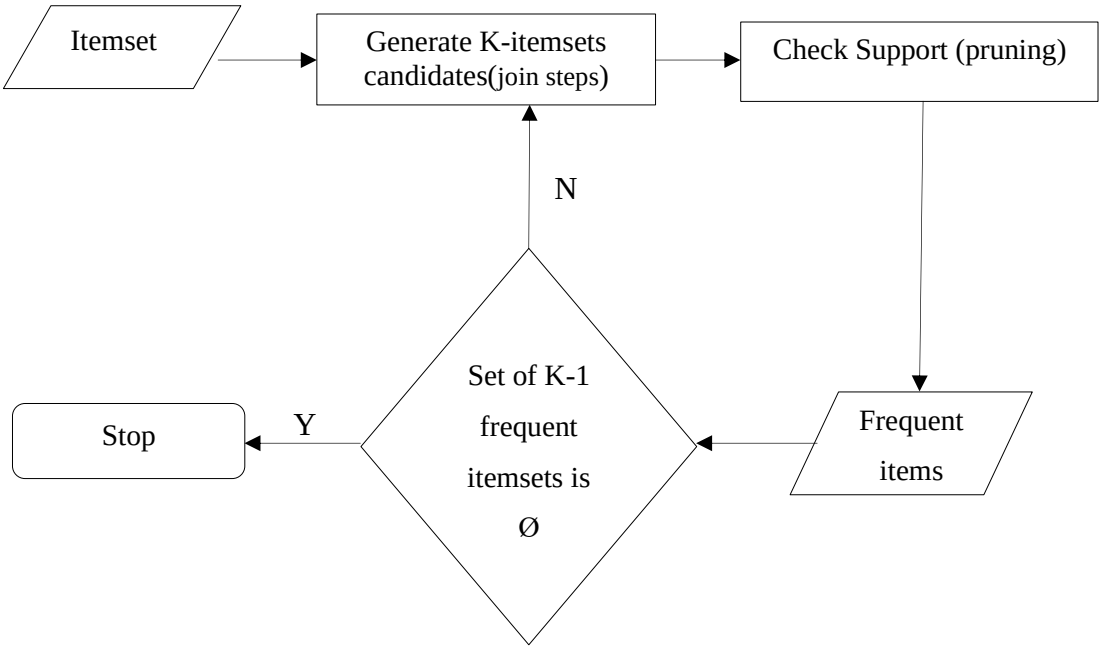


Figure 4.3. Flowchart the steps of apriori algorithm [22].

4.6.2. A Steps In Apriori Algorithm

The apriori algorithm determines a datasets most-used itemset. The join and prune steps are repeated until the most common itemset is discovered. The issue assumes minimal support [24]. The main step in the apriori algorithm is:

1. In the first algorithm iteration, each item is regarded as a candidate for 1-itemsets. The algorithm will tally each items occurrences.
2. Minimum support (e.g. 2). Set of 1-itemsets whose occurrence passes minimal sup criterion. Only candidates with a score above min sup progress to the next round.
3. Then, 2-itemset frequent things are identified using min sup. In the join step, the 2-itemset is created by producing a group of two items by combining them together.
4. Using the min-sup threshold value, the 2-itemset candidates are reduced. The table will now include two –itemsets, one including min-sup and the other containing simply min-sup.
5. The next iteration creates three –itemsets using join and prune. This iteration uses the antimonotone property, so the 2-itemset subsets of each group fall in min sup. If all 2-itemset subsets are frequent, the superset will be frequent.
6. Next, a 4-itemset will be created by merging a 3-itemset with itself cutting if its subset does not satisfy the min sup condition [23]. When the most common collection of items is achieved, the algorithm ends.

4.6.3. Hash-Based Apriori Algorithm

The use of a hash-based technique to determine the frequency of potential itemsets is aviable option. At first, this approach lowers the number of k-itemset candidates. An itemsets number created may be reduced to improve the efficiency of database scanning to identify the next itemsets [27]. The following are the steps of the hash-based apriori algorithm:

1- Create a Table 4.2 with the itemsets ordered where $I_1,I_2,I_3,\ldots\ldots\ldots,I_{j-1},I_j$ is an Itemsets with order 1,2,3,……J-1,J.

Table 4.2. Order of Itemset table [27].

Itemset	Order
I_1	1
I_2	2
I_3	3
…	…
I_{j-1}	J-1
I_j	J

2- Points to 1-itemset (Set k = 1)
3- Calculate all values.
4- Reation of hash table

The hash table will utilized to filter item sets for the subsequent iteration. As we see in Table 4.3. that is composed of a hash addresses, item sets, linkages, support values, confidence values, and bit vector values.

Table 4.3. Table of hash k-Itemset [27].

Address	Itemset	Link	Support	confidence
0	I_{10}	L_{10}	S_{10}	C_{10}
1	I_5	L_5	S_5	C_5
2	I_3	L_3	S_3	C_3
…	…	…	…	…
h(j)	I_1	L_1	S_1	C_1

As we have seen in table 4.3., the link that indicates a set of large itemsets in one cycle of mining data to an algorithm takes L_{10},……,L_1 with order of itemsets and the value of minimum support that prepared in all one of mining process and results a confidence of all process with C_{10},……..,C_1.

5- Create frequent table

Table 4.4. Frequent Itemset table

Itemset	Support
I_{10}	1
I_5	2
I_3	3
…	…
I_{j-1}	J-1
I_j	J

Table 4.4. displays the process of finding a frequency of a transactions itemsets with a value of support as we prepared, where in this case minimum supprot indicates to number transaction of itemset in the dataset, for example. In this table 4.4, an a number of tarnsaction to I10 itemset is 1 time, and a number of I5 itemset transaction is 2 time,…….etc.

6- Determination of combination

7- Steps 2–5 should be repeated until all bit vectors are zero [27].

The following table explains how the mechanism of the apriori algorithm and illustrates frequent itemsets mining done:

Table 4.5(a). A firest transaction of apriori algorithm sample

T_ID	Itemset
T_1	{B,C,D,E}
T_2	{B, C, D}
T_3	{A, B, D}
T_4	{A, B, C, D, E}
T_5	{A, B, C}
T_6	{B, E}

Itemsets	T_ID
{A}	T_3,T_4,T_5
{B}	T_1,T_2,T_3,T_4,T_5,T_6
{C}	T_1, T_2, T_4, T_5
{D}	T_1, T_2, T_3, T_4
{E}	T_1, T_4, T_6

Table 4.5(a). clarifies the first step of the apriori algorithm when the value of the minimum support = 2, and as T_ID indicates the order of transaction in the process, starting from T_1 to T_6, and we note the support for each element shown in the table, for example, the element {A} has support in T_3, T_4, T_5, and the {B} has support in T_1, T_2, T_3, T_4, T_5, T_6 .....etc.

Table 4.5(b). A second transaction of apriori algorithm sample

Candidate 1-itemsets	Support
{A}	3
{B}	6
{C}	4
{D}	4
{E}	3
{A}	4

1-itemsets	Support
{A}	3
{B}	6
{C}	4
{D}	4
{E}	3
{A}	4

1-itemsets	Support
{A}	3
{B}	6
{C}	4
{D}	4
{E}	3
{A}	4

Table 4.5(b). shows the operation of candidates after prepared Support = 2 and Candidate 1-itemsets, Each candidate 1-itemset is taken if the Support value is > 2. Where 2 is a value of minimum support.

Table 4.5(c). A third transaction of apriori algorithm sample

Candidate 2-itemsets	Support
{A, B}	3
{A, C}	2
{A, D}	2
{A, E}	1
{B, C}	4
{B, D}	4
{B, E}	3
{C, D}	3
{C, E}	2

2-itemsets	Support
{A, B}	3
{B, C}	4
{B, D}	4
{B, E}	3
{C, D}	3

2-itemsets	Support
{A, B}	3
{B, C}	4
{B, D}	4
{B, E}	3
{C, D}	3

Table 4.5(c). shows the operation of candidates and itemsets after prepared support = 2 and candidate 2-itemsets. All itemsets that belong to candidate 2-itemsets and have minimum support less than 2 value , it will be ignored as we sew before in Table 4.5(c) operations.

Table 4.5(d). A fourth transaction of apriori algorithm sample.

Candidate 3-itemsets	Support
{A,B,C}	2
{A,B,D}	1
{A,B,E}	1
{B,C,D}	3
{B,C,E}	2
{C,D,E}	2

3-itemsets	Support
{B,C,D}	3

In Table 4.5(d). that shows the operation of candidate C3 and itemsets after prepared support = 2 and candidate 3-itemsets, each candidate 3-itemset is taken if the support value is > 2.

Table 4.5(e). A fifth transaction of apriori algorithm sample.

Candidate 4-itemsets	Support
{ }	

Table 4.5(e). shows the operation of candidate C4 and itemsets after prepared support = 2 and, when candidate =4 , candidate 4-itemsets will be Ø and in this case a supprot is null because a candidate 4-itemsets is Ø.

4.6.4. Mathematical Model of Apriori Algorithm

The apriori algorithm finds the optimum association rules from a dataset using three matrices, making it an effective dataset approach. The following are some of the matrices:

Support: It determines the frequency with which a certain item or combination of objects occurs in a dataset. The following is the mathematical equation for support:

$$Support(I) = frac\{transaction\ containing(I)\}\{total.transactions\}\$\$ \quad (4.4)$$

Where $$ is a particular item in an items dataset.

Confidence: It computes the probability that the consumer would drink $$I 2$$ after consuming $$I 1$$.The formula is used to compute it:

$$Confidence(I_{1\rightarrow I_2}) = frac\{transaction\ containing(I_{1and}I_2)\}\{transactions\ containing(I\_1)\}\$\$ \quad (4.5)$$

Lift: The strength of the connection between both the best rules is assessed by a statistic called a lift [30]. It has the following mathematical formula:

$$Lift(I_{1\rightarrow}I_2) = frac\{Confidence(I_{1\rightarrow I_2})\}\{Support(I_2)\}\$\$ \quad (4.6)$$

Figure 4.4. shown in easly a mathimatical code of apriori algorithm and its main parameters and how the calculations and operations performs.

Input: *dataset D, Minimum Support ϵ , Minimum Confidence ε*
Output: *Rt All association rules*
Method:

1- *L1 = large 1-itemsets;*
2- *for(**k**=**2**; $\mathbf{L}_{k-1} \neq \emptyset$; **k**++)* ***do begin***
3- $\mathbf{Ck}$ *=apriori-gen(*$\mathbf{L}_{k-1}$*); //generate new candidates from* $\mathbf{L}_{k-1}$
4- *for all transactions* $\mathbf{T} \in \mathbf{D}$ ***do begin***
5- $\mathbf{Ct}$*=subset(*$\mathbf{C}_k$*,**T**); //candidates contained in **T**.*
6- *for all candidates* $\mathbf{C} \in \mathbf{C}_t$ ***do***
7- *Count(**C**)=Count(**C**)+**1**; // increase support count of **C** by **1***
8- *End*
9- $\mathbf{L}_k = \{\mathbf{C} \in \mathbf{C}_t \mid Count(\mathbf{C}) \geq \epsilon \times |\mathbf{D}|\}$
10- *End*
11- $\mathbf{L}_f = \cup_k \mathbf{L}_k$
12- $\mathbf{R}_t$*=GenerateRules(*$\mathbf{L}_f$ *, ε)*

Figure 4.4. Pseudo code of apriori algorithm [29].

Where D is a dataset, ϵ is minimum support , and ε is minimum confidence , T is transaction of itemset in one process, RT is association rules that simple output of algorithm, $L1$ is the itemset that had been appeared in level of transaction to take candidates itemsets, C_i is the itemset candidate, C as a count of itemset.

4.6.5. An Improved Apriori Algorithm for Anomaly Rule Detection

In addition to being the first association rule mining technique deployed, the apriori methodology is also the most popular because it searches datasets for common word sets and intriguing correlations. According to research, the traditional apriori algorithms have two main bottlenecks:

1. Needs a lot of dataset scans, which results.
2. A ton of candidate sets

In light of the inherent weaknesses in the apriori algorithms, certain pertinent improvements are made:

1. Avoid repetitive dataset scanning by using novel dataset mapping techniques.
2. Reducing candidate itemsets and frequent itemsets, even more, to improve joining efficiency.
3. Increasing effectiveness by counting support with the overlap method [30].

The findings show that, as compared to earlier improved approaches, the augmented Apriori algorithm improves operating efficiency under the same conditions [30]. The results demonstrate that the suggested enhanced apriori algorithm enhances operational efficiency under the same settings when compared to earlier improved methods [30]. These are this improvement study main phases:

1. Advising a fresh search strategy to quicken the search procedure.
2. Using a compressed vector architecture to reduce the cost of storage.

4.6.5.1. Hash-Based Techniques

1. Hash table is used as data structure
2. First iteration is needed to count itemset support.
3. At second iteration on, efforts are being made to improve apriori execution using the hash table concept.
4. In the second iteration, i.e., *2-itemset* generation, W maps each combination of two items into different buckets of the hash table structure and increments the correspon- ding value, the bucket number.
5. If the bucket count is less than the minimum support count, they are removed from the candidate sets.

Table 4.6(a,b,c). Transaction tables of improved apriori algorithm [30].

a

TID	List of items
T1	I1,I2,I5
T2	I2,I4
T3	I2,I3
T4	I1,I2,I4
T5	I1,I3
T6	I2,I3
T7	I1,I3
T8	I1,I2,I3,I5
T9	I1,I2,I3

b

Itemset	Support count
I1	6
I2	7
I3	6
I4	2
I5	2

c

Itemset	COUNT	Hash function
I1,I2	4	(1*10+2)mod 7=5
I1,I3	4	(1*10+3)mod 7=6
I1,I4	1	(1*10+4)mod 7=0
I1,I5	2	(1*10+5)mod 7=1
I2,I3	4	(2*10+3)mod 7=2
I2,I4	2	(2*10+4)mod 7=3
I2,I5	2	(2*10+5)mod 7=4
I3,I4	0	
I3,I5	1	(3*10+5)mod 7=0
I4,I5	0	

Table 4.6(a,b,c). shows the transaction steps of an improved apriori algorithm where that transactions discuse in following steps syntax of mathimatics inistructions where minsupport count and oreder of items I_1, I_2, I_3, I_4, I_5 :

$$Min\ Support\ Count = 3$$

$$Order\ Of\ Items\ I1 = 1, I2 = 2, I3 = 3, I4 = 4, I5 = 5$$

$$H(x, Y) = ((order\ Of\ First) * 10 + (order\ Of\ Second))\ mod\ 7 \quad (4.7)$$

Where arrangement of elements $I1 = 1, I2 = 2, I3 = 3, I4 = 4, I5 = 5$, and the result of the algorithm $H(x, Y) =$ (the order of the first item in the itemset column), multiplied *by 10*, (plus the order of the second item in the same itemset column), *mod by 7*.

4.6.5.2. Hash Table Structure to Generate L2

There are a two main advantages of hash table structure technique that summeried as follow:

1. Decrease the number of scans
2. Eliminate the large candidates responsible for high output.

Table 4.7. Hash table structure of improved apriori algorithm [30].

address of Bucket	0	1	2	3	4	5	6
count of Bucket	1	2	4	2	2	4	4
Contents of Bucket	{I1-I4}-1 {I3-I5}-1	{I1-I5}-2	{I2-I3}-4	{I2-I4}-2	{I2-I5}-2	{I1-I2}-4	{I1-I3}-4
L2	NO	NO	YES	NO	NO	YES	YES

Table 4.5. explains the operations in the previous tables, where the first row represents the result of the remainder of the division by 7(mod 7), the second row is the count, the third row represents the content of the itemset, and the fourth row represents the itemset that will be taken to the level 2 cycle, where every item set will be taken if its count greater than 2.

4.6.5.3. Partitioning

Any itemset that may be common in D.B must be common in at least one of its partitions (2 D.B. Scan)

1. Three partitions make up the database as shown in Figure 4.5.
2. Two transactions per each, with a 20 percent support [30].

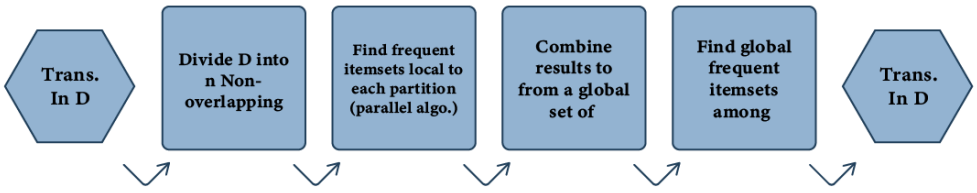


Figure 4.5. Partitioning of improved apriori algorithm [30].

Table 4.8. Partitioning table of improved apriori algorithm [30].

Transaction	itemset	First scan	Second scan	shortlisted
		Support=20%Min. sup=1	Support=20%Min. sup=2	
T1	I1, I5	I1-1,I2-1,I4-1,I5-1 {I1,I5}-1, {I2,I4}-1	I1-1,I2-3,I3-2,I4-3,I5-3,{I1,I5}-1, {I4,I5}-1,{I2,I4}-2,{I4,I5}-1, {I2,I3}-2, {I3,I4}-1,{I2, I3,I4}-1	I2-3,I3-2,I4-3,I5-3, {I2,I4}-2 {I2,I3}-2
T2	I2, I4			
T3	I4, I5	I2-1,I3-1,I4-1,I5-1 {I4,I5}-1,{I2-I3}-1		
T4	I2, I3			
T5	I5	I2-1,I3-1,I4-1,I5-1 {I2,I3}-1,{I2,I4}-1		
T6	I2,I3,I4			

In Table 4.6. the partitioning of improved apriori algorithm, it is done according to what we have observed from the first scan and the second scan, where I1, I5 appeared in the T1 and in the first scan when the support = 20% and the values of I1-1, I2-1, I4-1, I5-1,{I1, I5}-1, {I2, I4}-1 It relied on the contents of bucket as shown in Table 4. 5 before. Similarly, in T2, the itemset I2 and I4 appeared with its contents of bucket. similarly, with all the transactions shown in the table, and according to the percentage of the support, the contents of the bucket had a group of items attached to it, as shown in the table 4.6.

4.7. DIFFERENCE BETWEEN APRIORI AND FP GROWTH ALGORITHM

There are diffrence between two apriori and FP growth algorithms depends on many criteria like task, aproach of search, and the creation of candidate itemsets as will have been showning below as points:

4.7.1. Apriori Algorithm

1. The algorithm is array-based.
2. It employs the join and prune method.
3. Apriori performs breadth-first searches.
4. Using a level-wise approach, the apriori algorithm generates patterns with one item, two things, three items, and so on.

5. The creation of candidates is a long process. Runtime increases exponentially as the number of distinct items rises.
6. Generation of candidates may be readily parallelized.
7. It requires a large amount of memory since it creates many candidates.
8. It performs repeated database scans to find candidate sets.

4.7.2. FP Growth Algorithm

1. It is a tree-based algorithm to start.
2. It generates a conditional frequent pattern tree and a conditional pattern base from a database meeting the criteria for minimum support.
3. FP growth employs a depth-first approach.
4. FP growth employs a pattern-growth technique, which means it only considers patterns that are present in the dataset.
5. Runtime grows linearly as the number of transactions and commodities increases.
6. Data are highly interconnected; each node requires the root.
7. Due to its tiny size and absence of candidate creation, it utilizes less ram.
8. To construct the frequent pattern tree, the database is just examined twice.

Procedure fp growth*(T)

Input: *A conditional FP-tree T*

Output: *The complete set of all FTPs corresponding to T.*

Method:

1- **If** *T only contains a single branch B*
2- **For each** *subnet Y of the set of items in B*
3- *Output itemset Y ∪ T.base with count =smallest count of nodes in Y;*
4- **Else for *each*** *i in T.header* **do begin.**
5- *Output Y = T.base ∪ {i} with i.count;*
6- **If** *T.FP-array is defined*
7- *Construct a new headrer table for Y's FP-tree from T.FP-array*
8- **Else** *Construct a new headrer table from T;*
9- *Construct Y's conditional FP-tree TY and possibly its FP-array Ay;*
10- **If** $T_y \neq \emptyset$
11- *Call FP growth*(TY)*
12- End

Figure 4.6. Pseudo code of FP algorithm [29].

The FP-growth algorithm goes through a process of cluster preparation, computation of the number of k-clusters, and centroid computation. Different data groupings are grouped based on the grouping criteria, and the association rule is used to create models. The databases association rules are used to identify the intrusion scenario.

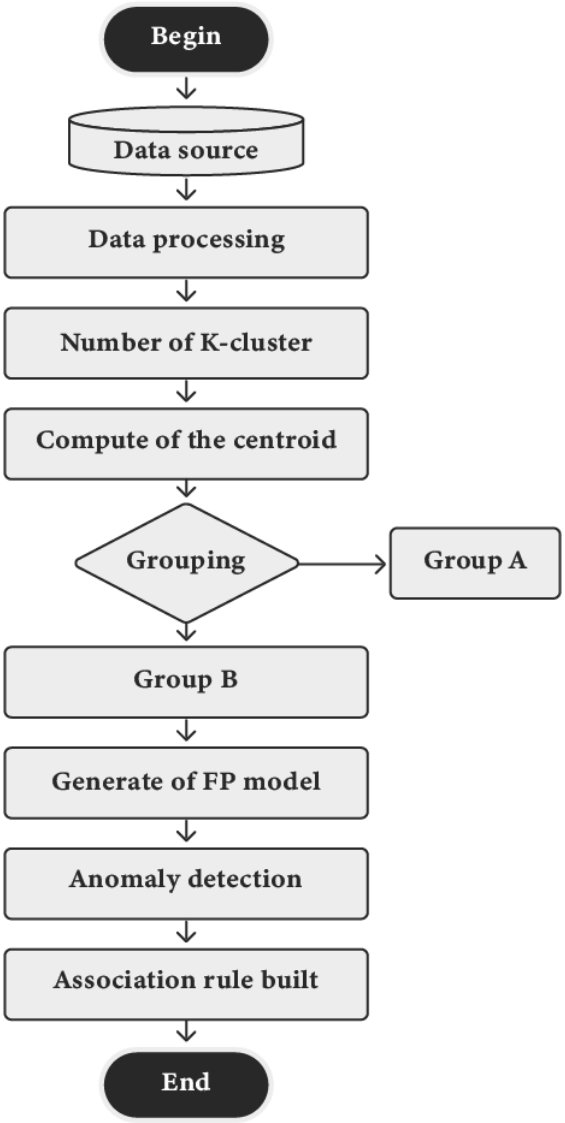


Figure 4.7. Flowchart of FP algorithm [29].

PART 5

ASSOCIATION RULE FOR NETWORK INTRUSION DETECTION

The IDS based association rules attempt to categorize every traffic as either regular or malicious [31]. Within the domain of network anomaly detection, the table below includes the most prominent strategies for the association rule method. Table 5.1 shows association rule mining algorithms and type of data on datasets of NID that interact with them, also declare a method of finding association rule to detect intrusion and anomalies.

Table 5.1. An association rule mining algorithms for NID [31].

Data mining Algorithms	**Type of data**	**Finding**
Apriori, FUZZY Apriori	D.A.R.A.P. 99	The association rule, which manages to combine association with a supplementary decision tree method, can be utilized for intrusion detection.
Apriori, FP-growth	Wired and wireless network data	The most current advancements in association rule mining techniques for identifying different networks are reviewed.
Apriory,FUZZY apriory	Real-life database	They introduce FARM, a linguistic terms-based algorithm. FARM offers special capabilities for detecting positive and negative associations.
Apriori, FUZZY apriori, classification algorithm.	KDD99 dataset	A novel classification approach leverages fuzz association rulesets as description models for different categories.
Apriori, Fuzzy Association Rule-based on prefix trees.	D.A.R.A.P. 99	To optimize the fuzzy association rules, a genetic algorithm was enhanced.
Apriori, Fuzzy Apriori, Genetic Algorithms.	Network traffic data	The abstract correlation between multiple security aspects may be discovered using fuzzy association algorithm.
Apriori,Fuzzy Association Rule	Network traffic data	Using the Fuzzy Association Rule to detect intrusion is beneficial, and the accuracy has increased.

5.1. ASSOCIATION RULE FOR BGP ANOMALY DETECTION

There are other approaches that identify anomalies of BGP like the time series analysis method that used wavelet transform to detect anomalies. A new framework was introduced called BALet which analyzes BGP features based on extracteding BGP message volume [5]. a machine learning approach that had been used supervised and unsupervised algorithms depends on classification and clustering, association rule techniques to detect BGP anomalies.

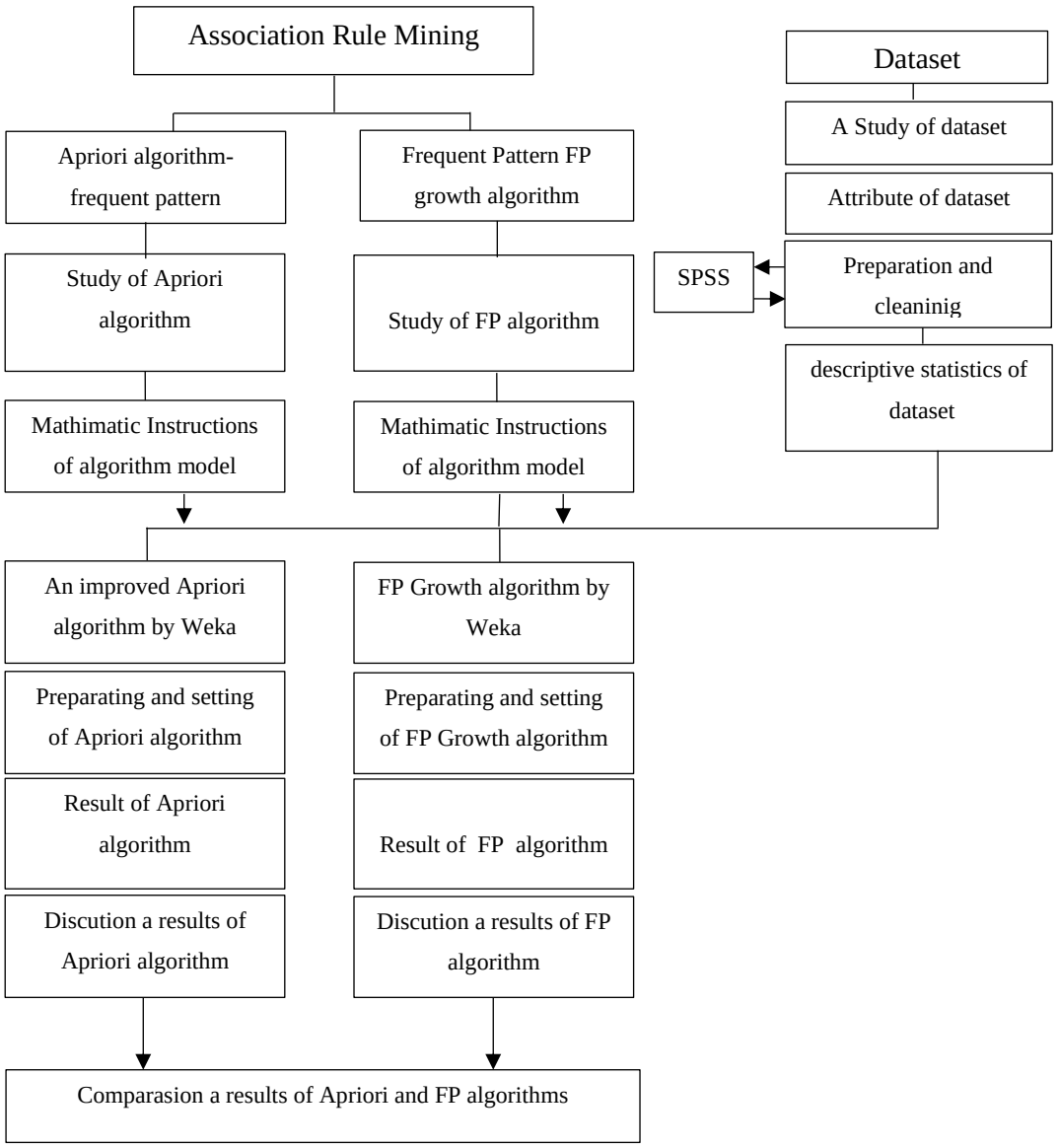


Figure 5.1. Methodology of BGP anomaly detection by association rule mining algorithms.

The figure 5.1. shows a flowchart of the methodology used in this study to detect the anomaly of the BGB using the technique of association rule mining algorithms, where apriori and FP algorithms are used to detect anomalies rules based on anomalies value of the attributes on the dataset of BGP, and compare that results and take the best of them.

5.2. STUDY OF DATASET

Code Red I is the one well-known attacks. These attacks served as the training data for the model. Code Red I make up the training dataset. As a test dataset, Code Red I was used. Based on their pearsons correlation coefficient value, a few selected features and the top 5 features were chosen. To be able to distinguish between anomaly and non-anomaly BGP update signals. The dataset for this study was downloaded from a website in CSV format and was ready to use (Kaggle,2020) [32]. Several announcements, average as-path length, and maximum edit-distance are all included in this dataset. Code Red I was one well-known Border Gateway Anomalies (BGP) that happened in january 2003, september 2001, and july 2000. BGP update messages from the Reseaux IP Europeens (RIPE) are accessible to the public via the Network Coordination Centre (NCC) and include Code Red I. At the model preparation stage, data is collected, data is integrated, outliers and extreme values are removed, data is converted into data mining language, and data is reduced by removing superfluous variables from the model. The dataset was reviewed to improve data quality and discover missing or noise, as well as whether there is a deviation or anomalous value in the study data. It is required to eliminate variables that will not be used in the analysis to prepare the dataset [32]. The dataset was evaluated using weka preprocessing and SPSS statistics.

5.2.1. Attributes of Dataset

When we used the NCC Dataset, we saw that it is necessary to stop at its own attributes and perform a clustering process to sort the attributes according to their category, and they were sorted into three clusters and clarified. As the table below shows, attributes from 1 to 4 were sorted into the cluster time, as well as the attributes from 5 to 41 were

sorted into the clustering features, and attribute 42 was assigned as a label to the dataset.

Table 5.2. Attributes of dataset

Column number	Name	Type
Columns 1-4	hour and minute, hour, minute, second	Time
Columns 5-41	Features	Features
Column 42	Label	labels for the regular (-1) and anomalous (1) data

Table 5.2. is the most important table that clarified and sorted all the fields of the dataset and classified them according to their type and description. We note that columns from 1 to 4 were classified under the time category, as well as columns from 5 to 41 were classified under the features category. furthermore, column No. 42 was classified under the so-called label that specified two values regular or anomaly, in addition to that classification was arranged in the dataset during its preparatory process.

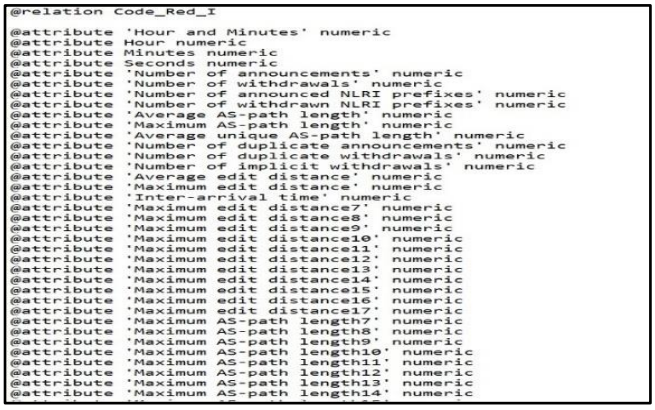
5.2.2. Descriptive Stattistics of Dataset

Before starting the experiment, according to the model used to detect anomalies in the BGP using the machine learning technique or mining using association rules algorithms, and depending on the dataset, it was necessary to conduct analysis, checking, and purification processes for that data, and a method of descriptive analysis or so-called descriptive statistics must be carried out to ensure the validity and integrity of the data in it. For this, we used the statistical analysis program SPSS Statistic, which enabled us to carry out the process of descriptive statistics with its two types, descriptive and explorers, where it classified the attributes according to their type. Attributes that are of the type of features for the BGP, and the third cluster is the regular or so-called label, and it includes the values of the regular or anomalies value, and finding the number of values valid for all the attributes in the dataset , In addition to finding the maximum and minimum value and the mean for each attribute, addition to finding the average standard deviation for each value of the attribute. Descriptive

statistics explorers analyze each attribute separately and finds the resulting values from the descriptive statistics process that was mentioned above.

There are important procedures that must be followed to understand and analyze the dataset that is used in the model of the experiment that we are doing, and to know the details and type of data it contains to suit the processes that are carried out in that model. For example, in our experiment, the information and details of the data type, the number of attributes, the number of instances and classes for that dataset were clarified as shown in Table 5.3., and because the model used in this study is a weka application, the dataset was set up with an extension of ARFF.

Table 5.3. Information of dataset



```
@relation Code_Red_I

@attribute 'Hour and Minutes' numeric
@attribute Hour numeric
@attribute Minutes numeric
@attribute Seconds numeric
@attribute 'Number of announcements' numeric
@attribute 'Number of withdrawals' numeric
@attribute 'Number of announced NLRI prefixes' numeric
@attribute 'Number of withdrawn NLRI prefixes' numeric
@attribute 'Average AS-path length' numeric
@attribute 'Maximum AS-path length' numeric
@attribute 'Average unique AS-path length' numeric
@attribute 'Number of duplicate announcements' numeric
@attribute 'Number of duplicate withdrawals' numeric
@attribute 'Number of implicit withdrawals' numeric
@attribute 'Average edit distance' numeric
@attribute 'Maximum edit distance' numeric
@attribute 'Inter-arrival time' numeric
@attribute 'Maximum edit distance7' numeric
@attribute 'Maximum edit distance8' numeric
@attribute 'Maximum edit distance9' numeric
@attribute 'Maximum edit distance10' numeric
@attribute 'Maximum edit distance11' numeric
@attribute 'Maximum edit distance12' numeric
@attribute 'Maximum edit distance13' numeric
@attribute 'Maximum edit distance14' numeric
@attribute 'Maximum edit distance15' numeric
@attribute 'Maximum edit distance16' numeric
@attribute 'Maximum edit distance17' numeric
@attribute 'AS-path length7' numeric
@attribute 'Maximum AS-path length8' numeric
@attribute 'Maximum AS-path length9' numeric
@attribute 'Maximum AS-path length10' numeric
@attribute 'Maximum AS-path length11' numeric
@attribute 'Maximum AS-path length12' numeric
@attribute 'Maximum AS-path length13' numeric
@attribute 'Maximum AS-path length14' numeric
```

Dataset	code_red_i
Type	csv.arff
Type of data	nominal
N.Attributes	42
Instance	7199
Class	label

Figure 5.2. The dataset as ARFF file

Figure 5.2. shows a snapshot of the shape of the dataset when configured along the ARFF, whereas, the dataset must be converted to the archive extension accompanied by a netbook file to describe the features and specifications for each attribute.

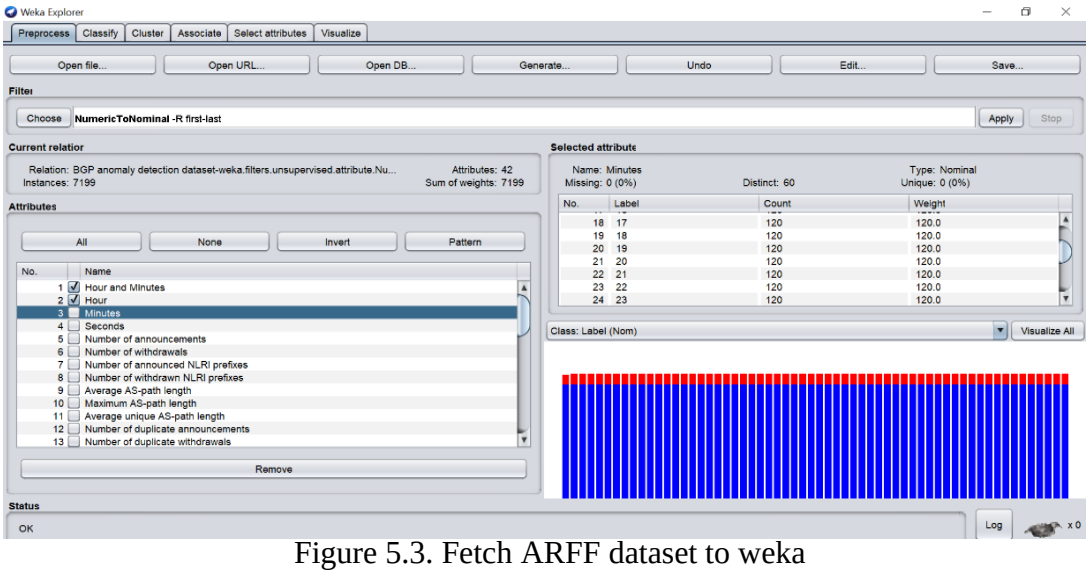


Figure 5.3. Fetch ARFF dataset to weka

Figure 5.3. shows the process of fetching the dataset to the weka model, where it is possible to perform the preprocess and visualization process for all attributes, change the data type of the attributes in the dataset, and also assign label to them. The data type of the attributes in this experiment was modified from numeric to numinal to suit the apriori and FP growth anomaly detection algorithms using the association rule technique.

5.3. EXPERIMENT SETUP

The setting for the experiment is a series of basic data mining processes. The six stages that comprise this experiment are the following: data preparation, feature selection and preprocessing, normalization, mining, and experiment results.

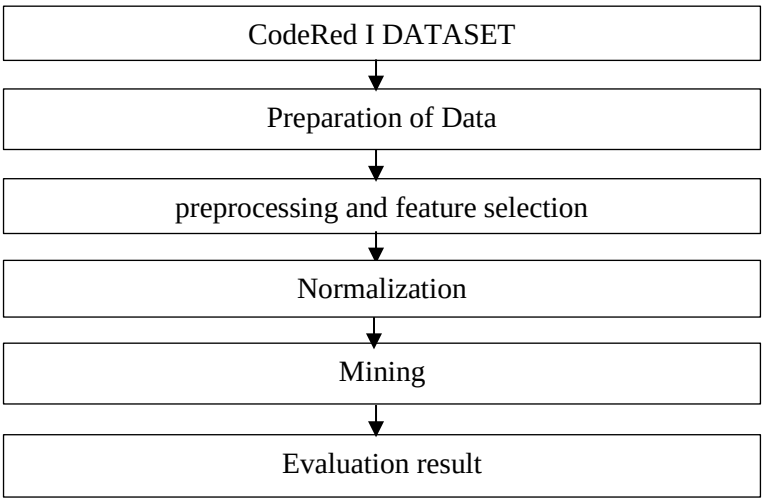


Figure 5.4. Steps of experiment

Figure 5.4. shown that stages which comprise an experiment, where in the stage of data preparation, all data is analyzed and checked, and prepared in the dataset. If there are empty or missing values in one of the attributes, in this case, the missing values for all attributes are filled in, and the dataset is configured with an extension so that it is compatible with the model used, the technique or the algorithm used in the experiment. In the feature Selection stage, all the attributes that are of the feature type are identified to clarify their nature and type, as well as the process of preprocessing for the dataset. In the normalization stage, the data type of all the attributes in the dataset is set to a

special type. In the mining stage, exploration and search for the anomalies associated values and rules that resulted from the experiment as a result of anomalies in the values of the attributes or the emergence of anomalies associated rules compared to the rules associated with the regular. In the final stage, the results of the experiment are written down and all the required observations and results are taken and evaluated.

5.4. APRIORI ALGORITHM IN WEKA

The apriori algorithm, which is one of the top ten data mining methods, is a similar technique to mining association rules. The basic idea is to mine repeated element groups in two stages: first, create a candidate group and then build a descending closed list. There is a wide variety of applications for apriori algorithms, such as in the business world and the protection of computer networks. The goal of association rule mining is to identify hidden associations between entities. It is an important subject of study in data mining and a long-standing issue. The mining of recurrent element sets is divided into two stages: the formation of the item set and the drawing of a decreasing closed list. We use the apriori algorithm in the weka application to determine the rang association rule based on minimum support and minimum metric(confidence). We can distinguish between correlation rules and support by looking at the best rules that result from apriori algorithm implementation and focusing on the correlation ratio through the value of confidence, support, and left.

5.4.1. Properties Setting of Apriori Algorithm

There are many apriori algorithm properties that we set it in the beginning of work as shown follow:

5.4.1.1. Car

If true is selected, class association rules rather than global association rules will be mined.

5.4.1.2. Classindex

what an index attributes class is. If it is set to -1, the last attribute is regarded as the class attribute.

5.4.1.3. Delta

Use these values as an iterative lowering unit, and then keep lowering the support until it hits the minimal support, or you can come up with a rule that satisfies the quantity requirement. The algorithm for determining and comprehending the number of turns is: **(upperBoundMinSupport)-((Number of cycles completed)-1)delta >= (LowerBoundMinSupport).*

5.4.1.4. Lowerboundminsupport

LowerBoundMinSupport.

5.4.1.5. Metrictype

Define the metric type and the basis for the sorting rules. Trust (note: class correlation) is an option.

Rules can only use trust mining, leverage (leverage), and trust (persuasion).

5.4.1.6. Minmtric

The minimal scale value denotes the minimum scale value you chose in the previous stage. For instance, if confidence is provided by default, the number specified by minMtric is the minimal confidence value. If confidence is lower than the minimal value for this statistic, it will be instantly erased and wont show up in best rules created later. For instance, minMtric = 0.9 and Conf = 0.8 wont show up.

5.4.1.7. Numrules

If there are more rules than can be detected, they will be sorted, and the highest numbers will only ever be displayed. For instance, if numRules = 10, only ten rules will ever be displayed.

5.4.1.8. Output Itemsets

Set to true to output the set size of L(X) large data sets rather than merely the number of item sets for the set of items supplied in the result.

5.4.1.9. Remove Allmissing Cols

Eliminate any columns that have default values.

5.4.1.10. Significance Level

Meaningful level, meaningful test.

5.4.1.11. Upperboundminsupport

Starting from this value, upperboundminSupport repeatedly decreases to more than or equal to minimal support.

5.4.1.12. Verbose

The algorithm will execute in repeat mode if true is set.

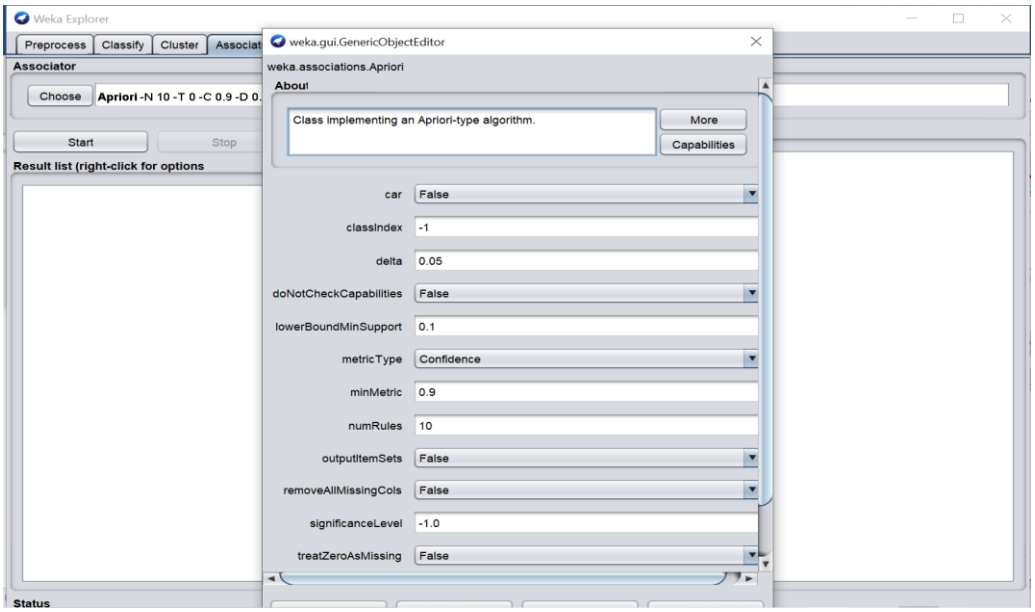


Figure 5.5. Setting of apriori algorithm properties

Figure 5.6. shown many apriori algorithm properties that we set it in the beginning of work lik car property that used to class association rules rather than global association rules will be mined and delta where Use these values as an iterative lowering unit, and then keep lowering the support until it hits the minimal support……..etc.

PART6

RESULTS & DISCUSSION

It is known, both apriori and FP association rule mining algorithms have their scheme and their relation methodology, which they follow while doing the work. In our study, regarding the dataset model, the number of its instances is 7199, and the number of attributes that it formed is 42. Also the association rule type model, is a full training set. The following table shows an illustrative comparison of the methodology of each of the two algorithms.

Table 6.1. Type of relation and association model of apriori and FP algorithm

Name	APRIORI algorithm	FP growth algorithm
Scheme:	weka.associations.APRIORI -N 5 -T 0 -C 0.9 -D 0.05 -U 1.0 –M 0.1 –S -1.0 –C -1	weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1
Relation:	Code_Red_I-weka.filters.unsupervised.attribute. -R breadth-first -weka.filters.unsupervised.attribute. NumericToNominal-R breadth-first	Code_Red_I-weka.filters.unsupervised.attribute. NumericToBinary-Rfirst-last-weka.filters.unsupervised.attribute. NumericToNominal-Rfirst-last
Instances:	7199	7199
Attributes:	42	42
APRIORI found top rules	5	5
Associator model	(full training set)	(full training set)

Table 6.1. shows the scheme of each of the apriori and FP growth algorithms, as well as the type of relation used in terms of those two algorithms, where the relation in the apriori algorithm is unsupervised-attribute-R breadth-first, while the relation at the FP growth is unsupervised-attribute-NumericToBinary-Rfirst-last. The table also shows the number of instances in the dataset model is 7199, on which the work of the two algorithms will be applied. As we noticed, the number of attributes is 42, the number of required and set rules in all the two algorithms is 5, and the methodology of the association model is full-training.

Table 6.2. Measures and range to important measures point

Symbole	Measure	Range	Formula
Ø	Ø-coefficient	-1……. 1	$\frac{P(A,B)-P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$
S	Support	0 ……. 1	$P(A,B)$
C	Confidence	0 ……. 1	$\max(P(B\backslash A), P(A\backslash B)$
V	Conviction	0.5 …… ∞	$\max(\frac{P(A)P(\bar{B})}{P(A\bar{B})}, \frac{P(B)P(\bar{A})}{P(B\bar{A})}$
⅄	Lift	0 …… ∞	$\frac{P(A,B)}{P(A)P(B)}$

Table 6.2. discusses the top symbol we used in our experiment and the apriori results. Moreover, two algorithms have the support that defines a number to appear from the set of items in the dataset model and the confidence that shows the rate of verifying the two itemsets that combine transaction or rule, coefficient appears as Ø if the itemset in candidate cycle is null or there is not a candidate according to support value.

6.1. DISCUSSION THE RESULTS OF APRIORI ALGORITHM

When experimenting, the apriori algorithm was used by applying weka and tuning its properties to implement the process of detecting anomalies value from the BGP anomaly detection dataset (Code Rede I), where the minimum support S value was set to 0.95 that take its result from *P(A,B)* equition where A&B is transaction of itemset and set the confidence to be 0.9 that calculate from max $(P(B\backslash A), P(A\backslash B)$ and number of cycles performed=1 and the result of the total frequent itemset was 39 as summarized in Table 6.3.

Table 6.3. Support and confidence of apriori algorithm

Support S	0.6	0.7	0.8	0.9
Confidence C	0.6	0.7	0.9	0.9
A number of cycles performed:	9	6	4	3
Size of the set of large itemsets L(1):	28	22	22	20
Size of the set of large itemsets L(2):	297	227	164	96
Size of the set of large itemsets L(3):	2404	1166	530	240
Size of the set of large itemsets L(4):	6397	3484	922	346
Sec.	0.6	0.6	0.3	0.2

In Table 6.3. as we noticed, when we gave different values for the support, starting from 0.6 to 0.9, how did the confidence values associated with it change. The first cycle in size of the set of large itemsets L(1) = 28, and so are the candidates in size of the set of large itemsets L(2)= 297, and the candidates in size of the set of large itemsets L(3)= 2404, and also the candidate items in size of the set of large itemsets L(4)= 6397. This means that the number of anomaly itemsets detected by the algorithm in L(1) = 28, as well the number of anomaly itemsets in L(2) = 297, and the number of anomaly itemsets that appeared in L(3) = 2404 and the number of anomaly itemsets in L(4) = 6397 , and it is important to note that the time taken by the algorithm to complete this cycle is 0.6 Thus, with the rest of the values for the support, each cycle took a number of candidates, as indicated by the point corresponding to each cycle in the table. It is also important to clarify that the time taken by the algorithm when the support was at value 0.6 took 0.06, and at each value of the support the algorithm took a time of seconds.

Table 6.4. Calculated frequent item sets with 0.6 support

NO	Name	Itemset	support	Anomalies count
1	Itemset1	Maximum AS-path length15	0.6	15
2	Itemset2	Maximum edit distance12	0.6	18
3	Itemset3	Maximum edit distance16	0.6	54
4	Itemset4	Maximum edit distance15	0.6	51

Table 6.4. represent the calculated frequent item sets after adjusting the NumRules property in the apriori algorithm to the value of 5 and setting the values of the minimum support in 0.6 and the confidence values as we discussed it in Table 6.4, as we noticed, the number of times the anomaly Itimset1, which is called Maximum AS-path length15, appears as an abnormal value in L(1), L(2), L(3), and L(4) is 15 times. On the other hand, the number of times the anomaly Itimset2, which is called Maximum edit distance12, appears as an abnormal value in L(1), L(2), L(3), and L(4) is 18 times, and the same is the case with the anomaly Itimset3, which is called Maximum edit distance16, as the number of times it appeared as an anomaly in the four levels L1, L2, L3, and L4 is 54 times, this case with the anomaly Itimset4, which is called Maximum edit distance15, as a number appeared 51 times as an abnormal value in the Dataset, and converting the data type in the dataset to NumericToNominal,

We obtained the itemsets shown and detect an anomaly value count to all of them was produced.

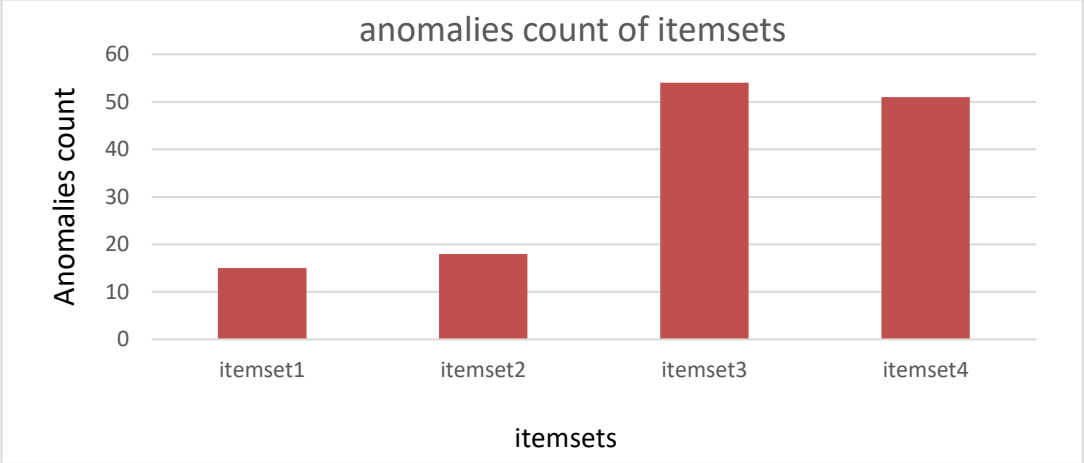


Figure 6.1. Anomalies values count of itemsets in apriori algorithm

Figure 6.1. accurately shows the number of occurrences of the abnormal value for item sets detected by the apriori algorithm, where the number of occurrences of the anomaly value for the itemsets was set as it appeared in all sets of large itemsets L(1) and set of large itemsets L(2) and set of large itemsets L(3) and set of large itemsets L(4). For example, the sum of the number of occurrences of the value of itemset1(Maximum AS-path length15) in the L(1) to L(4) was discovered to be 5, and so on with the rest of the itemset2, itemset 3, and itemset 4, as shown in the table6.5 in anomalies count.

Table 6.5. Calculated frequent itemsets with many support values

Min. Support	Itemset1	Itemset2	Itemset3	Itemset4
0.6	15	18	54	51
0.7	9	18	36	33
0.8	21	24	42	39
0.9	15	18	54	51

Table 6.5. performed calculations of the recurrence of anomaly values for each itemset in more than one cycle. For example, when the value of 0.6 was given for the minimum support, the repeating values of the itemsets were discovered as an anomaly value that appeared after executing the operation in the apriori algorithm, and the itemste1 took the value 15 and the itemset2 took the value 18, It is the value of the number of occurrences of the occurrence of the anomalous value of that itemset after executing

the algorithms work, and also itemset3 took the value 54, and the value of the recurrence of itemset4 was also the value 51. This explains that the anomaly itemset1 was repeated 15 times as an abnormal value in the four levels L(1), L(2), L(3), and L(4) when the minimum support value was equal to 0.6 and was repeated 9 times when the value of the minimum support was equal to 0.7 and was repeated as an abnormal value 21 times when it was the value of the minimum support is equal to 0.8 and its value is repeated as an abnormal 15 times with the value 0.9 for the minimum support. The same is the case with Anomaly itemset2, where it was repeated 18 times as an abnormal value in the four levels L(1), L(2), L(3), and L(4) when the minimum support value was equal to 0.6 and was repeated 18 times when the value of the minimum support was equal to 0.7 and was repeated as an abnormal value 24 times when it was the value of the minimum support is equal to 0.8 and its value is repeated as an abnormal 18 times with the value 0.9 for the minimum support. This also applies to the anomaly itemset3 and the anomaly itemset4 as shown in the values associated with them in Table 6.5.

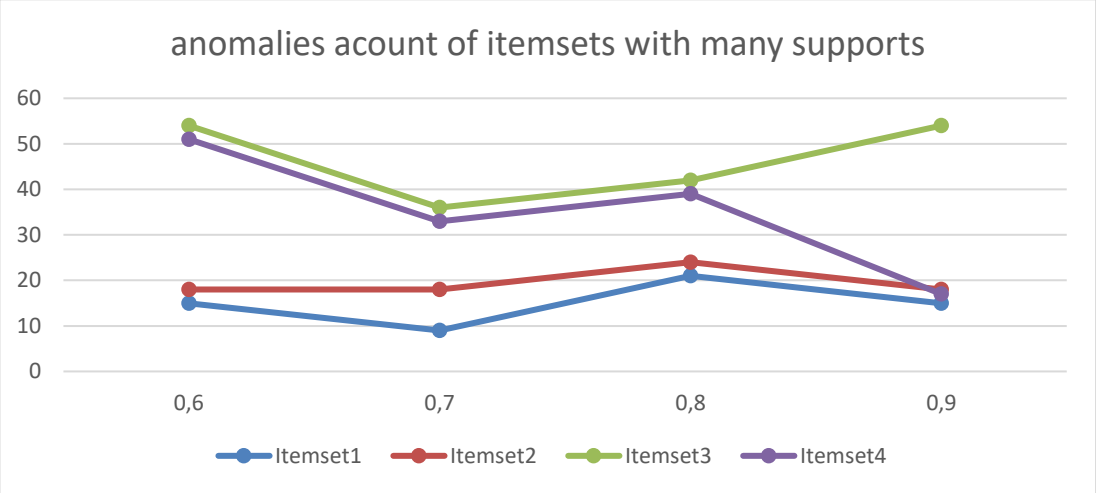


Figure 6.2. Calculate anomalies count of itemsets with supports

Figure 6.2. Calculate anomalies count for all itemsets. furthermore,The graph threshold represents the anomalies for the itemsets of the dataset model. Where it is noticeable that the values for itemset1 are repeated 15 times when the support ratio = 0.6 as well as the value of itemset2 is repeated 18 times with the same support rate, and the value of itemset3 is 54, and the anomaly value of itemset4 is 51 at the same support rate, which is 0.6 . Thus, the histogram showed all the values of the repeated anomalies of the itemsets in the dataset model, and those values for those datasets were

changing with the change of the support ratio, so when the support ratio was raised from the value 0.6 to the value 0.7, we saw a change in all the values of the itemsets repeated, and by returning to the table we will notice that calculated for those itemsets and its values. After calculating the minimal support threshold for each frequent itemset, we constructed the rules with a larger than 85% support and confidence level associated to the label attribute. The best five abnormal association rules resulting from the experiment are explained as follows.

Table 6.6. Calculated association rules

Rule NO.	Association rules
1	Maximum AS-path length15 6898 ==> Maximum edit distance12 6898 <conf:(1)> lift:(1.04) lev:(0.04) [288] conv:(288.41).
2	Maximum edit distance12 6898 ==> Maximum AS-path length15 6898 <conf:(1)> lift:(1.04) lev:(0.04) [288] conv:(288.41).
3	Maximum edit distance16 Maximum AS-path length15 6859 ==> Maximum edit distance12 6859 <conf:(1)> lift:(1.04) lev:(0.04) [286] conv:(286.78).
4	Maximum edit distance12 Maximum edit distance16 6859 ==> Maximum AS-path length15 6859 <conf:(1)> lift:(1.04) lev:(0.04) [286] conv:(286.78).
5	Maximum edit distance15 Maximum AS-path length15 6839 ==> Maximum edit distance12 6839 <conf:(1)> lift:(1.04) lev:(0.04) [285] conv:(285.95).

Table 6.6. shows the emergence of the best five anomalous association rules based on the frequency of the occurrence of anomalous values of the associated attributes in them, as the anomalous association rule is 1 it is the result of the recurrence of the anomalous value of the Maximum AS-path length15 correlates with the appearance of the anomalous value of the Maximum edit distance12 at Instance 6898, and the anomalous correlation rule 2 is the result of the recurrence of the anomalous value of the attribute Maximum edit distance12 correlates with the appearance of the anomalous value of the Maximum AS-path length15 at instance 6898. The anomalous association rule 3 is the result of the repetition of the anomalous values of each of the attributaries, Maximum edit distance16 Maximum AS-path length15 and correlates together with the appearance of the anomalous value of the attributes Maximum edit

distance1 at instance 6859, and the anomalous correlation rule 4 is the result of the repeated occurrence of the outliers for each of the attributes Maximum edit distance12 and Maximum edit distance16 Maximum edit distance15 and Maximum AS-path length15 outliers are correlated together with the appearance of the anomalies Maximum edit distance12 and Maximum AS-path length15 at instance 6859, and the anomalous correlation rule 5 is the result of the repeated anomalies.

Table 6.7. Rate of leverage and conviction for anomalous rule

Rule No.	No. Of instance	Confidence	Lift	Leverage	Conviction	label
1	6898	(1)	(1.04)	(0.04)	(288.41)	1
2	6898	(1)	(1.04)	(0.04)	(288.41)	1
3	6859	(1)	(1.04)	(0.04)	(286.78)	1
4	6859	(1)	(1.04)	(0.04)	(286.78)	1
5	6839	(1)	(1.04)	(0.04)	(285.95)	1

Table 6.7. shows the experiment results in good detail after its implementation by the apriori algorithm. The instance values of each attribute were recorded according to the associated rules resulting from the process, as well as the clarification of the leverage ratio and the conviction ratio for each attribute confirming the existence of anomalous values for these attributes and at a value of 1 for the label and the number of the instance specified for them. Where it is noted that the rate of confidence was appropriate and almost reached the value of 1 for each of the associated Rule No. 1, Rule No. 2, Rule No. 3, and Rule No. 4, and it measures the extent to which the correlation is achieved for the anomaly elements that make up those bases and appear in them, and table 6.7 also shows the lift value, which shows the correlation ratio between the itemsets that make up each correlation rule.

6.2. VISUALIZED RESULTS OF APRIORI ALGORITHM

Depending on the algorithms showing the anomalous association rules and the values of the instance and according to the order specified in the dataset and the assignment of the instance label, the value of 1 was determined for the label instance if there is an anomaly in any attributes in the dataset, and the value of -1 for label instance if there is no anomalous value for any attributes in the dataset. Furthermore, the apriori algorithm in which the dataset is configured does not recognize the value -1 and it is denoted by the value 0.

Based on the foregoing, the result of the visualization process for the attributes dataset values by using the weka framework is shown in the figures.

As we will see in the next figures, and based on the anomalous values and rules resulting from the visualization of the apriori algorithm experiment, assigning the value -1 to the regular value, represented by the blue color, and the value 1 to the anomalous values, represented by the red color for the label attribute in the calass colour visualization process, the following appears:

1. In Figure 6.3., anomalies appear for each of the attributes Maximum AS-path length15 and Maximum edit distance12 that represent the x-axis and the y-axis in red at the point (0,0), as well as they, appear more intensely at the point (1,1).
2. In Figure 6.4., anomalies appear for each of the attributes Maximum edit distance12 and Maximum AS-path length15 that represent the x-axis and the y-axis in red at the point (0,0), as well as they, appear more intensely at the point (1,1).
3. In Figure 6.5., anomalies appear for each of the attributes Maximum AS-path length15 and Maximum edit distance12 that represent the x-axis and the y-axis in red at the point (0,0), as well as they, appear more intensely at the point (1,1).
4. In Figure 6.6., anomalies appear for each of the attributes Maximum edit distance12 and Maximum AS-path length15 that represent the x-axis and the y-axis in red at the point (0,0), as well as they, appear more intensely at the point (1,1).
5. In Figure 6.7., anomalies appear for each of the attributes Maximum AS-path length15 and Maximum edit distance12 that represent the x-axis and the y-axis in red at the point (0,0), as well as they, appear more intensely at the point (1,1).

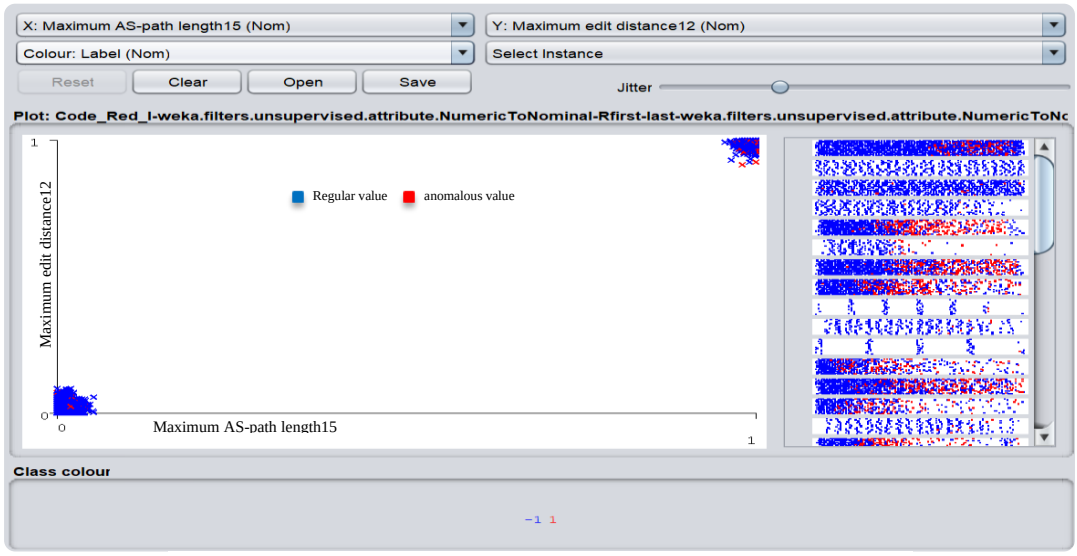


Figure 6.3. Visualization of the firest rule and its values

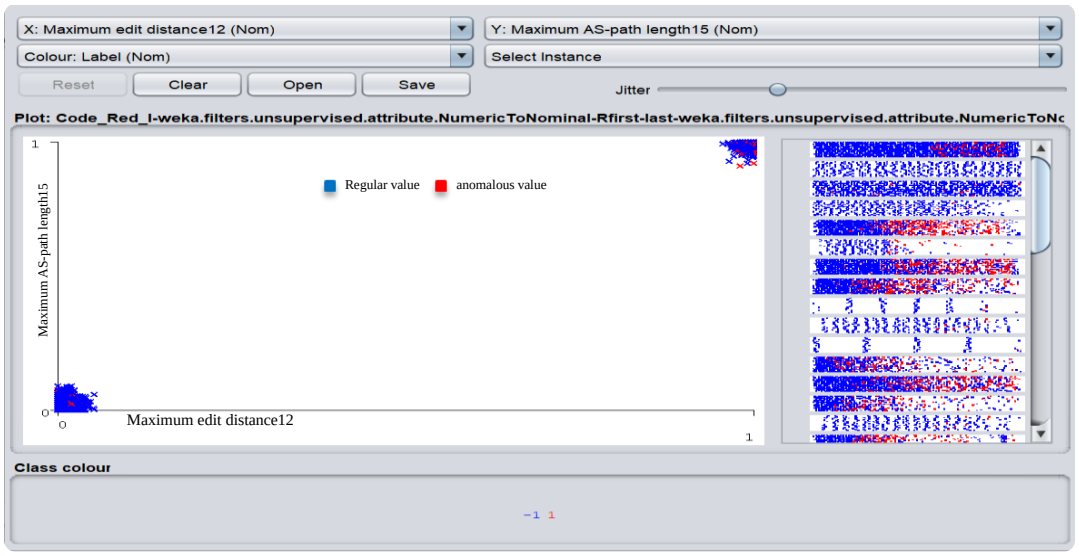


Figure 6.4. Visualization of the second rule and its values

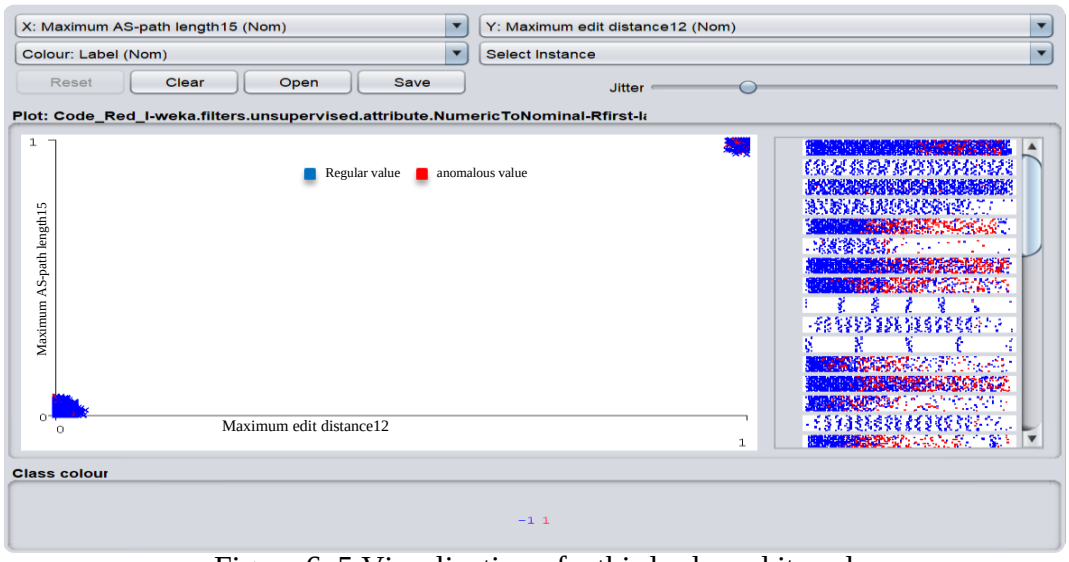


Figure 6. 5 Visualization of a third rule and its values

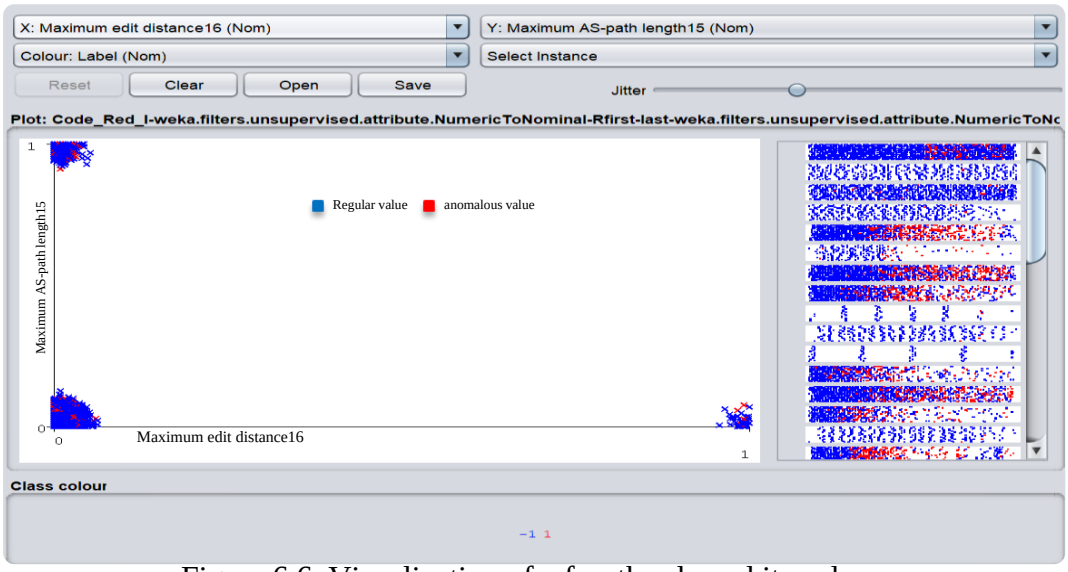


Figure 6.6. Visualization of a fourth rule and its values

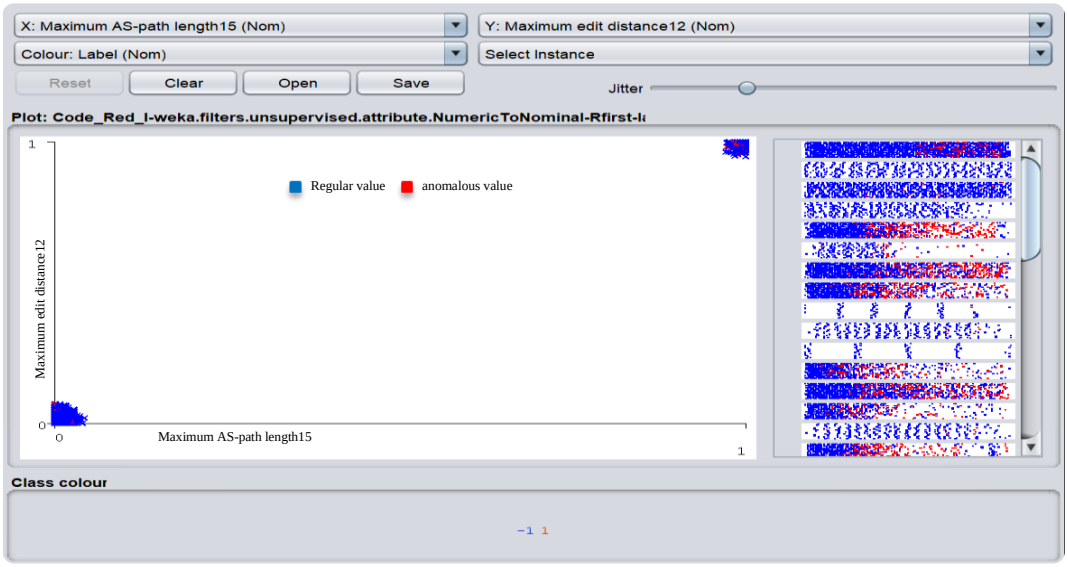


Figure 6.7. Visualization of a fifth rule and its values

6.3. DISCUSSION THE RESULTS OF FP GROWTH ALGORITHM

The FP growth algorithms goal is to find sets of frequently occurring items in a dataset faster than the apriori algorithm. The apriori algorithm goes back and forth through the data set to find out which goods are often found together. The FP algorithm compress the dataset representing frequent items in to frequent-pattern tree or FP-tree that retains the itemsets association information. Then, divide the compressed dataset into aset of conditional datasets, each associated with once frequent item and mines each such dataset separatly [30]. The algorithm saves time due to the tree data structures quicker scanning[30]. In our experiment after adjusting the NumRules property in the FP growth algorithm to the value of 5 and setting the value or rest properties, we get follows results.

Table 6.8. Calculated frequent item sets with support

NO	Name	Itemset	Anomalies count
1	Itemset1	Packet size (B)_binarized	25
2	Itemset2	Number of duplicate withdrawals_binarized	26
3	Itemset3	Number of announcements_binarized	58
4	Itemset4	Number of announced NLRI prefixes_binarized	53

Table 6.8. represent the calculated frequent item sets after adjusting the NumRules property in the FP algorithm to the value of 5 and setting the initial values of the Minimum Support, and converting the data type in the Dataset to NumericToBinary next NumericToNominal, We obtained the itemsets shown and detect an anomaly value count to all of them was produced. we noticed, the number of times the anomaly Itimset1, which is called Packet size (B)_binarized, appears as an abnormal value in L(1), L(2), L(3), and L(4) is 25 times. On the other hand, the number of times the anomaly Itimset2, which is called Number of duplicate withdrawals_binarized, appears as an abnormal value in L(1), L(2), L(3), and L(4) is 26 times, and the same is the case with the anomaly Itimset3, Number of announcements_binarized, as the number of times it appeared as an anomaly in the four levels L1, L2, L3, and L4 is 58

times, this case with the anomaly Itimset4, which is called Number of announced NLRI prefixes_binarized, as a number appeared 53 times as an abnormal value in the Dataset, and converting the data type in the dataset to NumericToBinary and NumericTo Nominal, We obtained the itemsets shown and detect an anomaly value count to all of them was produced.

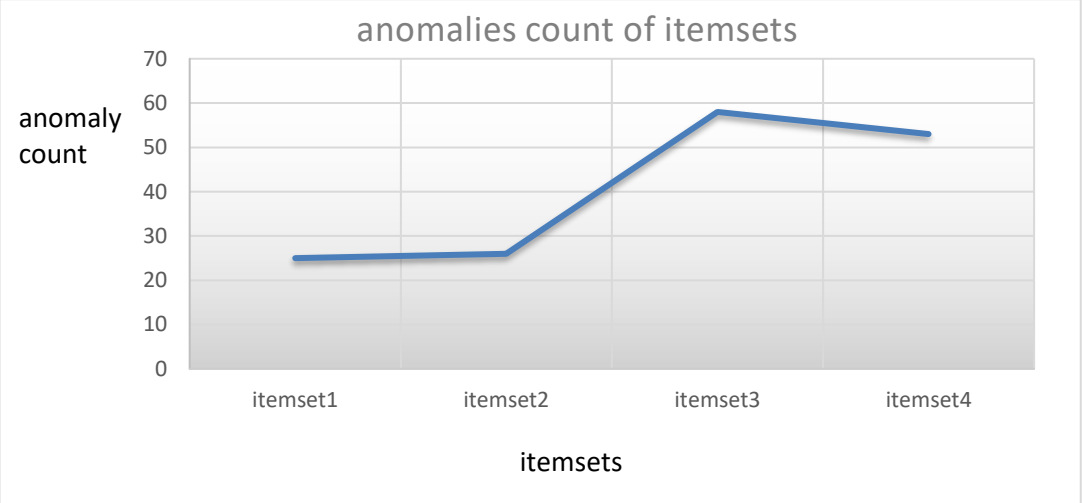


Figure 6.8. Anomalies values count of itemsets in FP growth algorithm

Figure 6.8. accurately shows the number of occurrences of the abnormal value for itemsets detected by the FP growth algorithm, where the number of occurrences of the anomaly value for the itemsets was set as it appeared in all sets of large itemsets as shown in the Table 6.9. in anomalies count.

Table 6.9. Calculated frequent itemsets with many support values

Min. Support	Itemset1	Itemset2	Itemset3	Itemset4
0.6	25	26	58	53
0.7	10	22	36	42
0.8	9	24	42	49
0.9	9	18	39	44

Table 6.9. performed calculations of the recurrence of anomaly values for each itemset in more than one cycle by fp growth algorithm. For example, when the value of 0.6 was given for the minimum support, the repeating values of the itemsets were discovered as an anomaly value that appeared after executing the operation in the apriori algorithm, and the itemste1 took the value 25 and the itemset2 took the value 26, It is the value of the number of occurrences of the occurrence of the anomalous

value of that itemset after executing the algorithms work, and also itemset3 took the value 58, and the value of the recurrence of itemset4 was also the value 53, and this explains that the anomaly itemset1 was repeated 25 times as an abnormal value in the four levels L(1), L(2), L(3), and L(4) when the minimum support value was equal to 0.6 and was repeated 10 times when the value of the minimum support was equal to 0.7 and was repeated as an abnormal value 9 times when it was the value of the minimum support is equal to 0.8 and its value is repeated as an abnormal 9 times with the value 0.9 for the minimum support. The same is the case with Anomaly itemset2, where it was repeated 26 times as an abnormal value in the four levels L(1), L(2), L(3), and L(4) when the minimum support value was equal to 0.6 and was repeated 22 times when the value of the minimum support was equal to 0.7 and was repeated as an abnormal value 24 times when it was the value of the minimum support is equal to 0.8 and its value is repeated as an abnormal 18 times with the value 0.9 for the minimum support. This also applies to the anomaly itemset3 and the anomaly itemset4 as shown in the values associated with them in Table 6.9.

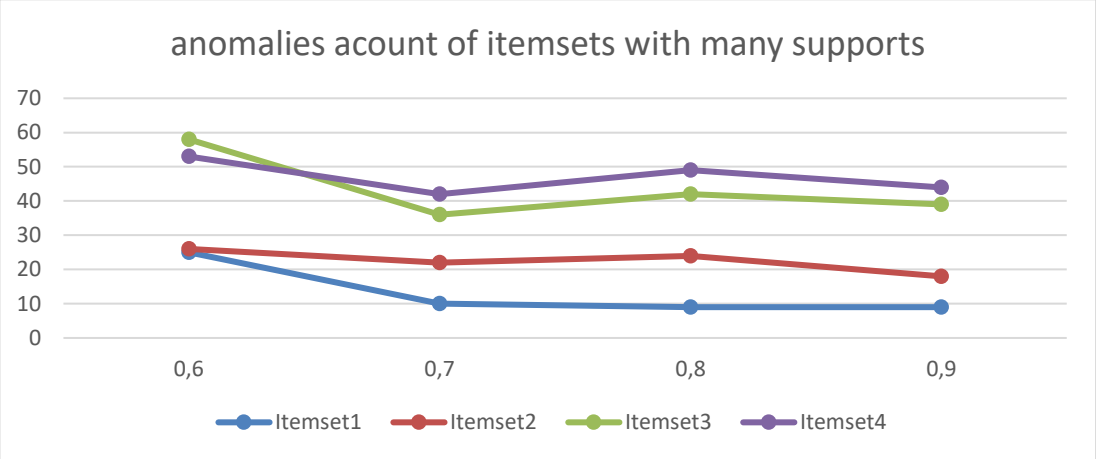


Figure 6.9. Calculate anomalies count of itemsets with supports

Figure 6.9. Calculate anomalies count for all itemsets. furthermore,The graph threshold represents the anomalies for the itemsets of the dataset model. Where it is noticeable that the values for itemset1 are repeated 25 times when the support ratio = 0.6 as well as the value of itemset2 is repeated 26 times with the same support rate, and the value of itemset3 is 58, and the anomaly value of itemset4 is 53 at the same support rate, which is 0.6 . Thus, the histogram showed all the values of the repeated

anomalies of the itemsets in the dataset model, and those values for those datasets were changing with the change of the support ratio, so when the support ratio was raised from the value 0.6 to the value 0.7, we saw a change in all the values of the itemsets repeated, and by returning to the table we will notice that calculated for those itemsets and its values. After calculating the minimal support threshold for each frequent itemset, we constructed the rules with a larger than 85% support and confidence level associated to the label attribute.

The best five anomalous association rules resulting from the experiment are explained as follows:

Table 6.10. Calculated association rules

Rule no.	Association rules
1	[Packet size (B)_binarized=1]: “6982” ==> [Number of duplicate withdrawals_binarized=1]: “6982” “<conf:(1)>" “lift:(1)” “lev:(0)” conv:(0)”
2	[Number of duplicate withdrawals_binarized=1]: 6912==> [Packet size (B)_binarized=1]: 6912“<conf:(1)>" “lift:(1)” “lev:(0)” “conv:(0)”.
3	[Packet size (B)_binarized=1]: “6982” ==> [Number of announcements_binarized=1]: “6982” “<conf:(1)>" “lift:(1)” “lev:(0)” “conv:(0)”.
4	[Number of announcements_binarized=1]: “6961” ==> [Packet size (B)_binarized=1]: “6961” “<conf:(1)>" “lift:(1)” “lev:(0)” “conv:(0)” .
5	[Packet size (B)_binarized=1]: “6942” ==> [Number of announced NLRI prefixes_binarized=1]: “6942” “<conf:(1)>" “lift:(1)” “lev:(0)” “conv:(0)”.

In Table 6.10. after adjusting the NumRules property in the FP growth algorithm to the value of 5 and setting the value of the minimum support to the value of 0.95 and the confidence at the value of 0.9 and converting the data type to NumiricToNominal, NumiricToBinary, we noticed the emergence of the best five anomalous association rules based on the frequency of the occurrence of anomalous values of the associated attributes in them. The anomalous association rule 1 is the result of the repeated appearance of the Packet size (B)_binarized associated with the appearance of the number of duplicate withdrawals_binarized at the instance of 6982, and the anomalous association rule 2 is the result of the repeated appearance of the number of duplicate

withdrawals_binarized associated with the appearance of the Packet size B)_binarized at the instance 6912, the anomalous association rule3 is the result of the repetition of the Packet size (B)_binarized anomaly values associated with the appearance of the number of announcements_binarized anomaly value at the instance 6982, and the anomalous association rule 4 is the result of the number of announcements_binarized anomalous values reappearing associated with the appearance of the Packet size (B)_binarized attribute anomaly on the instance 6961, and the anomalous association rule 5 is the result of the recurrence of the Packet size (B)_binarized anomaly associated with the appearance of the Number of announced NLRI prefixes_binarized on the instant 6942.

Table 6.11. Rate of leverage and conviction for anomalous Rule

Rule No.	No. Of instance	Confidence	Lift	Leverage	Conviction	label
1	6982	(1)	(1)	(0)	(0)	1
2	6912	(1)	(1)	(0)	(0)	1
3	6982	(1)	(1)	(0)	(0)	1
4	6961	(1)	(1)	(0)	(0)	1
5	6942	(1)	(1)	(0)	(0)	1

Table 6.11. shows the experiment results in great detail after it was implemented by the FP growth algorithm. The instance values of each attribute were recorded according to the associated rules resulting from the process, as well as clarification of the leverage ratio and the conviction ratio for each attributes, confirming the existence of anomalous values for these attributes and at a value of 1 for the label and the instant number specified for them. Where it is noted that the rate of confidence was appropriate and equal to the value of 1 for each of the associated Rule No. 1, Rule No. 2, Rule No. 3, and Rule No. 4, and it measures the extent to which the correlation is achieved for the anomaly elements that make up those bases and appear in them, and table 6.11 also shows the lift value, which shows the correlation ratio between the itemsets that make up each correlation rule.

6.4. VISUALIZED RESULTS OF FP GROWTH ALGORITHM

As shown in the best association rules resulting from the visualization process using the FP growth algorithm by the application weka, we notice the appearance visualized of anomalous values and rules as shown in following figures.

As we will see in the next figures, and based on the anomalous values and rules resulting from the visualization of the FP algorithm experiment, assigning the value -1 to the regular value, represented by the blue color, and the value 1 to the anomalous values, represented by the red color for the label attribute in the calass colour visualization process, the following appears:

1. In Figure 6.10., anomalies appear for each of the attributes Packet size (B)_binarized that represent the x-axis and attribute Number of announced NLRI prefixes_binarized that represent the y-axis in red at the point (1,0), as well as they, appear more intensely at the point (1,1).
2. In Figure 6.11., anomalies appear for each of the attributes Number of duplicate withdrawals_binarized that represent the x-axis in red and the attributes Packet size (B)_binarized that represent the y-axis at the point (1,1).
3. In Figure 6.12., anomalies appear for each of the attributes Packet size (B)_binarized that represent the x-axis and Number of announcements_binarized that represent the y-axis at the point (1,1).
4. In Figure 6.13., anomalies appear for each of the attributes Number of announcements_binarized that represent the x-axis and Packet size (B)_binarized that represent the y-axis at the point (1,1).
5. In Figure 6.14., anomalies appear for each of the attributes Packet size (B)_binarized that represent the x-axis and Number of announced NLRI prefixes_binarized that represent the y-axis at the point (1,1).

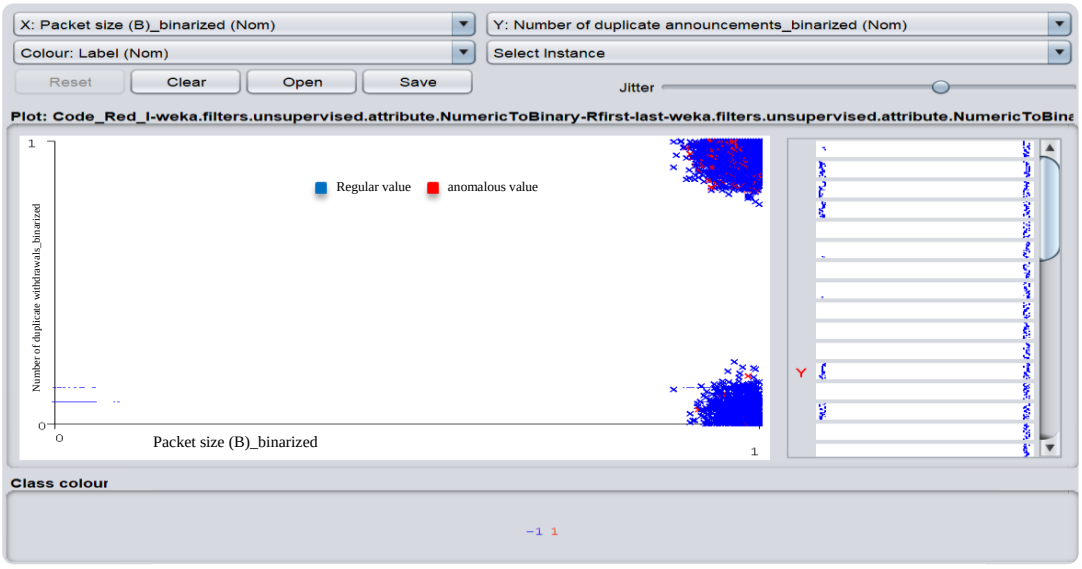


Figure 6.10. Visualization of the Firest rule and its values

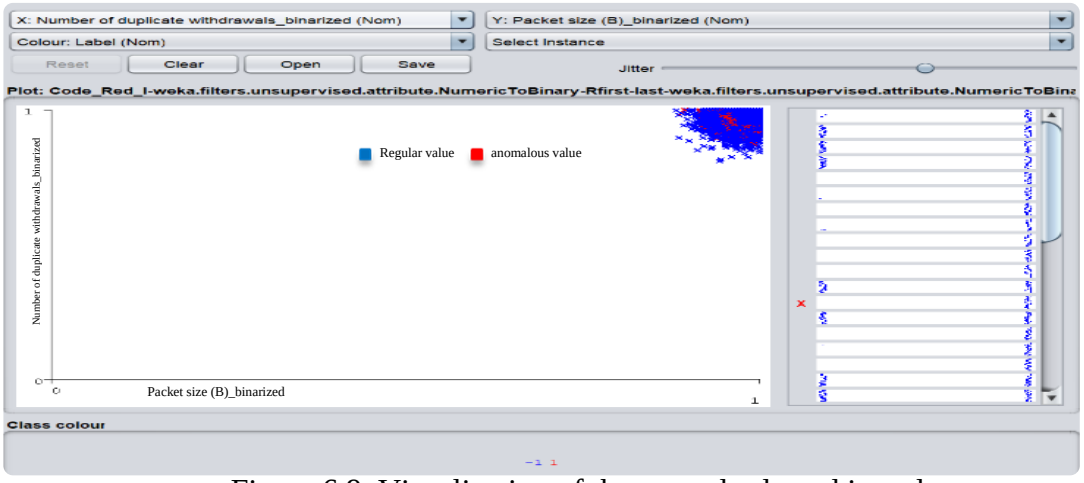


Figure 6.9. Visualization of the second rule and its values

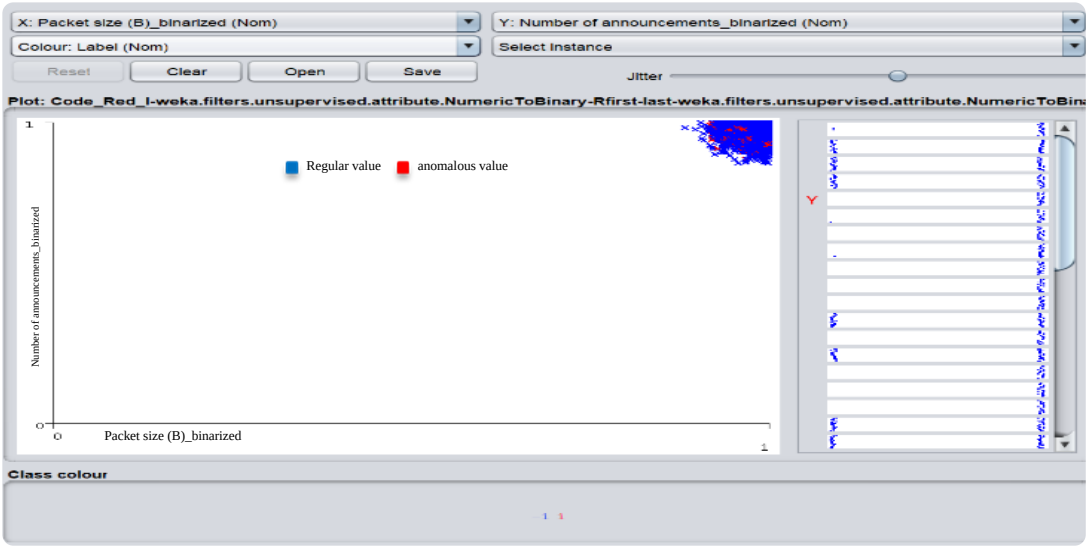


Figure 6.10. Visualization of a third rule and its values

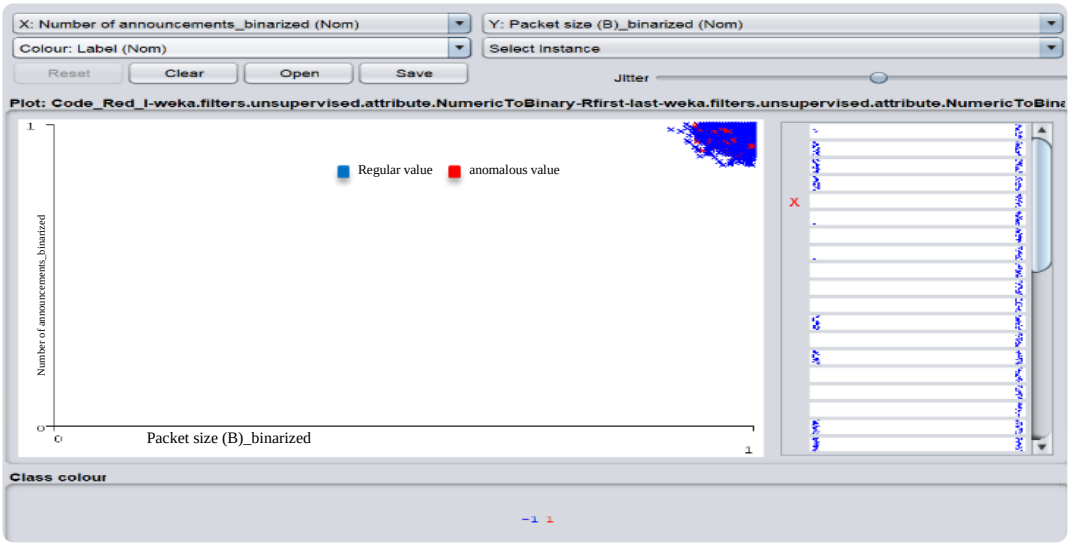


Figure 6.13. Visualization of a fourth rule and its values

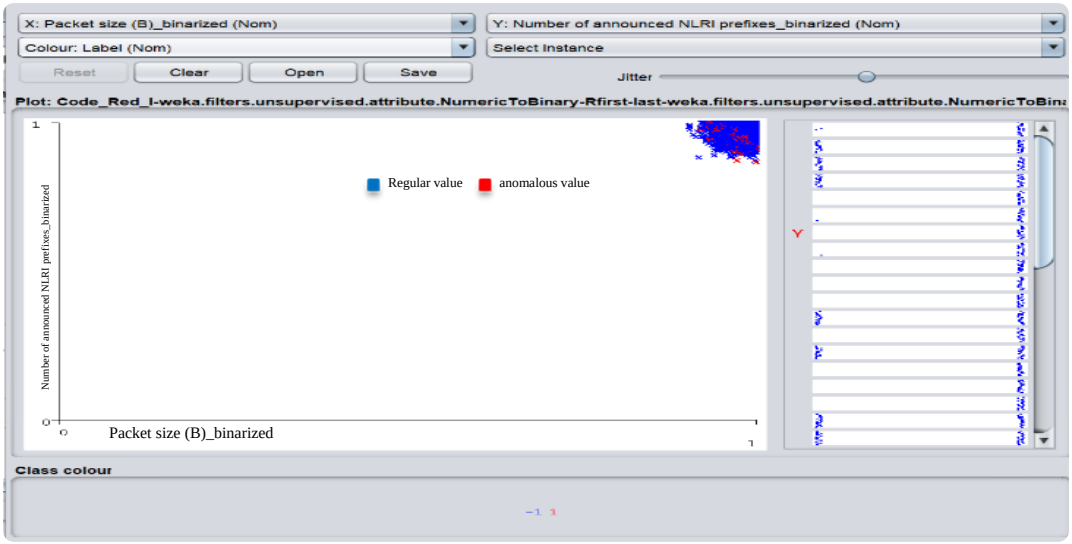


Figure 6.14. Visualization of a fifth rule and its values

6.5. RESULT ANALYSIS

In our research, we employ a Code Red I network traffic dataset model from the Reseaux IP Europeens (RIPE), which the general public can access through the Network Coordination Centre (NCC). Here, a network attributes has been listed in numerical form, with 7199 records and a file size of 720KB. Each record contains 42 columns. In our experiment, two of the most important reasons that showed a variance in the results of the rules for each of the two algorithms lie in that the different methodology for each of them, where the apriori algorithm follows the NumericToNominal-R-breadth-first-search methodology in dealing with the attributes of the dataset model, while the FP algorithm follows the tree methodology and the NumericToBinary-R-first-last-search-NumericToNominal, where it divides the dataset into a sub-datasets and takes one candidate from each one of sub-dataset and compares between them for taking one of them as a target value, and with high accuracy, after that, it starts by pulling out the last candidate that contains the value from the end of the dataset. That makes the FP algorithm the most accurate in its results, depending on support, confidence, and accuracy.The many analyses of this dataset are displayed below.

Table 6.12. Parameter analysis using support & confidence

Support	0.6	0.7	0.8	0.9
Confidence	0.6	0.7	0.9	0.9
Frequent sets in apriori	138	96	126	138
Frequent sets in fp grwoth	162	110	124	110
Time generation of apriori(sec)	0.6	0.6	0.3	0.2
Time generation of FP (sec)	0.3	0.2	0.1	0.1
Rules in aprtiori	5	5	5	5
Rules in FP growth	5	5	5	5

Table 6.12. displayed the analysis of the apriori and FP growth algorithms based on various parameters on the basis of how frequently item sets are generated and how many rules are used to determine the number of sets that may be used to identify any

undesirable anomalous data in the network, various confidence and support values are shown in the table.

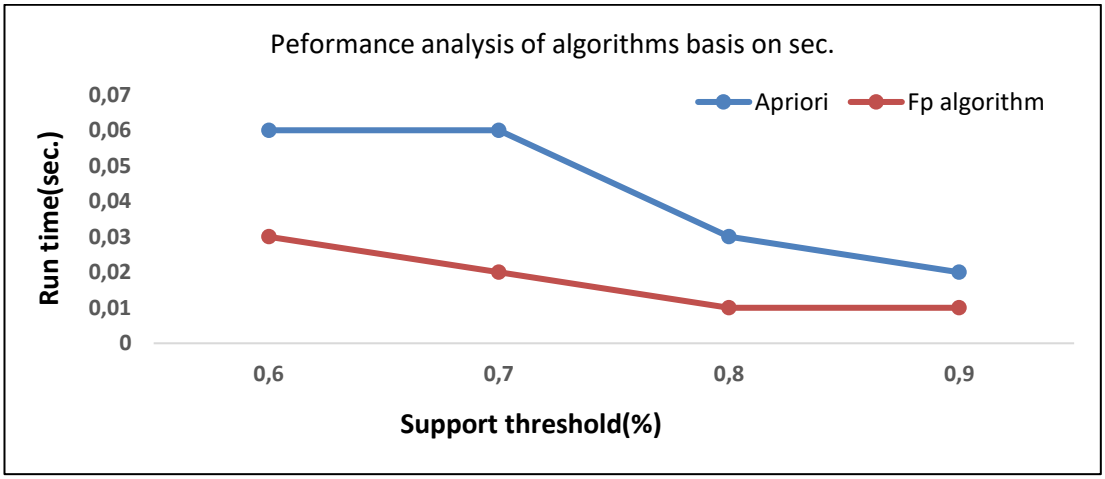


Figure 6.11. Evaluate the performance of the two algorithms

In Figure 6.15., a histogram was made the comparison to analyze the performance of each of the two algorithms, apriori and FP, on the basis of support and the time it took to analyze the dataset model and detect anomalies. We noticed that FP algorithm was more accurate and performs with the fastest time spent in each process and at the support values that were set.

For further study, analysis, and evaluation of the performance of the two algorithms, Table 6. 10 shows their performance in the classification process of the dataset model instances at the support value of 0.9.

Table 6.13. Analysis of algorithms and classifying instances

Algorithm	Correctly classification inistances	InCorrectly classification inistances	Time of building model
Apriori	6898	301(3. 01%)	0.2
FP growth	6982	217(2.17%)	0.1

Table 6.13., it is noticeable after analyzing the performance of the two algorithms, we found that the apriori algorithm adopted 6898 correct instances in classifications

beings out of the total number of 7199 instances that compound the dataset model, where the remaining 301 were classified as incorrect, and this is one of the most important reasons that made the aforementioned algorithm low efficiency, while The FP growth algorithm adopted 6982 correct instances in classifications beings out of the total number of 7199 instances that compound the dataset model, where the remaining 217 were classified as incorrect, and this is what ensures the FP growth algorithm as the best and more efficient of its process. furthermore, the advantages of values shown by the FP algorithm in confidence, leverage, convection, and lift, in addition to the acceptable percentage in the time of building the model.

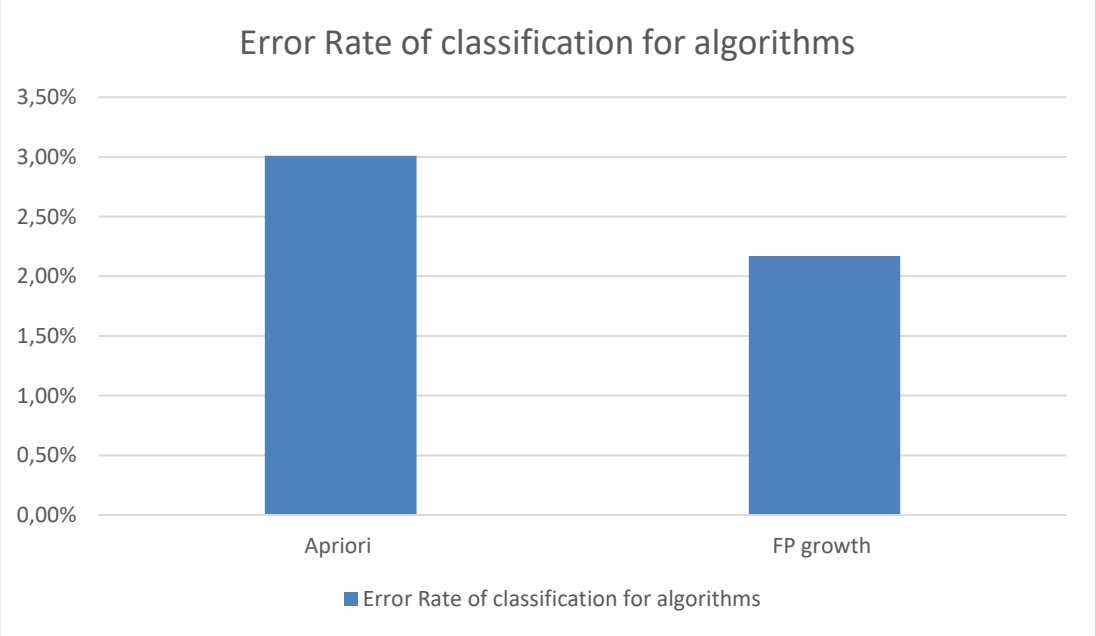


Figure 6.12. Evaluate error rate of the two algorithms

In addition to all of the above, It is noticeable after the implementation and visualization of the experiment using the apriori and FP growth algorithms in weka that the difference between the results of the two algorithms is that the Apriori algorithm shows anomalous values of attributes in (0,0) or (1,1) vectors, while the FP growth algorithm shows anomalous values for attributes in(0,0),(0,1) and(1,1) vectors and all of them shows anomalous association rules, and this is confirmed by the figures

shown in the process of visualization and the values appearing in it at the x and y axes and at the coordinates (0,0), (1,1) and (0,1). Also More accurate in the results, values , and anomalies that were detected after executing the experiment on the dataset by using the FP growth algorithm and in the weka framework.

PART7

CONCLUSION

Attacks, configuration errors, and power outages are examples of abnormal BGP events that should be caught early on because they might lead to anomalous or pathological routing behavior at the global or prefix level. Worm event rules were deduced from BGP data collected during the Code Red I worm outbreaks. Code Red I is one well-known assaults that target BGP networking and produce abnormalities in its operation. These attacks were utilized as the dataset for training the model using network traffic data. Association rule unsupervised algorithms is a key research area in data mining, as well as a long-standing topic whose main aim is to discover the hidden connections between objects. Apriori and FP growth association rule mining algorithms an efficient algorithm that scans the training features dataset model to detect anomalous itemsets, and showed an anomalies rules between these anomalous itemsets. this study aimed at employ a feature selection approach by association rule unsupervised algorithms to detect BGP anomalies and evaluate the performance of these algorithms in terms of values of support, confidence, and accuracy. Where We noticed that FP algorithm was more accurate and performs with the fastest time spent in each process and at the support values that were set. Furthermore, the advantages of values shown by the FP algorithm in confidence, leverage, convection, and lift, in addition to the acceptable percentage in the time of building the model.

.

REFERENCES

1. Hoarau, K., Tournoux, P. U., & Razafindralambo, T. (2021, October). "***Suitability of graph representation for bgp anomaly detection"***. In 2021 IEEE 46th Conference on Local Computer Networks (LCN) (pp. 305-310). IEEE.

2. Garcia-Luna-Aceves, J. J. (2022, August). "***Attaining stable and loop-free inter-domain routing without path vectors"***. In Proceedings of the ACM SIGCOMM Workshop on Future of Internet Routing & Addressing (pp. 58-65).

3. Griffin, T. G., & Wilfong, G. (2019). "***An analysis of BGP convergence properties. ACM SIGCOMM Computer Communication Review"***, 29(4), 277-288.

4. Alotaibi, H. S., Gregory, M. A., & Li, S. (2022). "***Multidomain SDN-Based Gateways and Border Gateway Protocol"***. Journal of Computer Networks and Communications, 2022.

5. Edwards, P., Cheng, L., & Kadam, G. (2019). "***Border gateway protocol anomaly detection using machine learning techniques"***. SMU Data Science Review, 2(1), 5.

6. San Jose, Americas Headquarters, ***"I.P. Routing: BGP Configuration Guide, First Published"***, Cisco Systems, Inc. 170 West Tasman Drive, CA 95134-1706. U.S.A., 12-08-2013.

7. Nael A. Zidan, ***"Implementation of Border Gateway Protocol (BGP) Attributes"***, Article, Modern University College, February 2016.

8. Christopher Kruegel, Darren Mutz, "***Topology-Based Detection of Anomalous BGP Messages"***, Software Group, Santa Barbara, University of California, 2017.

9. Zhou, S., He, J., Yang, H., Chen, D., & Zhang, R. (2020). ***Big data-driven abnormal behavior detection in healthcare based on association rules.*** IEEE Access, 8, 129002-129011.

10. Yan Yang, Xingang Shi, ***"Path Stability in Partially Deployed Secure BGP Routing"***, Department of Computer Science and Technology, Tsinghua University, 3 January 2022.

11. Lu Cheng1, Phil Edwards1, Girish Kadam ***"Border Gateway Protocol Anomaly Detection Using Machine Learning Techniques"***, Southern Methodist University, Dallas, TX 75275 USA,2019.
12. SARA,ANAND,"*https://packetpushers.net/demystifying-bgp-sessionestablishments* " , January 22, 2020.
13. Deshpande, S., Thottan, M., Ho, T. K., & Sikdar, B. (2019). "***An online mechanism for BGP instability detection and analysis"***. IEEE transactions on Computers, 58(11), 1470-1484.
14. Yulanda, R. D., Wahyuningsih, S., & Amijaya, F. D. T. (2019, July). "***Association rules with apriori algorithm and hash-based algorithm"***. In Journal of Physics: Conference Series (Vol. 1277, No. 1, p. 012048). IOP Publishing.
15. El-khoudary, O. A. E. N.(2017), & Baraka, R. ***Apriori Algorithm for Arabic Data Using MapReduce***.
16. Eljadi, E. E., & Othman, Z. A. (2011, June). "***Anomaly detection for PTM's network traffic using association rule"***. In 2011 3rd Conference on Data Mining and Optimization (DMO) (pp. 63-69). IEEE.
17. Moore, D., Shannon, C., & Claffy, K. (2020, November). "***Code-Red: a case study on the spread and victims of an Internet worm"***. In Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment (pp. 273-284).
18. Chandola, V., Banerjee, A., & Kumar, V. (2021). "***Anomaly detection Algorithms every Data Scientist should know"***. ACM computing surveys (CSUR), 41(3), 1-58.
19. Awadlesh, I. (2019). "***Weka: IT For Business Intelligence: Classification and Clustering Analysis"***. Term Paper, April, 19.
20. Satyam Kumar, 5, ***"Anomaly Detection Algorithms every Data Scientist should know"***, Dec 13. 2021.
21. Hyeok Kong., ***"Rare Association Rule Mining for Network Intrusion Detection"***, Faculty of Mathematics, Kim Il, Sung University, D.P.R.K, D.P.R.K, 2017.

22. Yi, F., Zhang, L., Yang, S., & Zhao, D. (2021, October). "***A Security-Enhanced Modbus TCP Protocol and Authorized Access Mechanism"***. In 2021 IEEE Sixth International Conference on Data Science in Cyberspace (DSC) (pp. 61-67). IEEE.

23. Adebayo, O. S., & Abdul Aziz, N. (2019). ***Improved malware detection model with apriori association rule and particle swarm optimization***. Security and Communication Networks, 2019.

24. Douzi, S., Benchaji, I., & El Ouahidi, B. (2018). ***Hybrid approach for intrusion detection using fuzzy association rules plus anomaly and misuse detection***. Int J Mach Learn Comput, 8(5).

25. Qiankun Zhao and Sourav S. Bhowmick, ***"Association rule mining: A Survey"***, Technical Report, C.A.I.S., Nanyang Technological, University, Singapore, No. 2003116, 2013.

26. Jie, X., Wang, H., Fei, M., Du, D., Sun, Q., & Yang, T. C. (2018). ***Anomaly behavior detection and reliability assessment of control systems based on association rules***. International Journal of Critical Infrastructure Protection, 22, 90-99.

27. Gao, M., Ma, L., Liu, H., Zhang, Z., Ning, Z., & Xu, J. (2020). ***Malicious network traffic detection based on deep neural networks and association analysis.*** Sensors, 20(5), 1452.

28. Yanbin YE, ***"A Parallel Apriori Algorithm for Frequent Itemsets Mining"***, Acxiom Corporation,1001 Technology Drive, Little Rock, Arkansas 72223, U.S.A., 2006.

29. Ola Abed El-Nasser El-khoudary, ***"Apriori Algorithm for Arabic Data Using MapReduce"***, The Islamic University – Gaza, Shawal. 1436H, July. 2015.

30. Edastama, P., Bist, A. S., & Prambudi, A. (2021). "***Implementation of data mining on glasses sales using the apriori algorithm***". International Journal of Cyber and IT Service Management, 1(2), 159-172.

31. Xiuli Yuan, ***"An Improved Apriori Algorithm for Mining Association Rules"***, School of Shanghai University, Shanghai 200444, China, 2017.

32. Panjaitan, S., Amin, M., Lindawati, S., Watrianthos, R., Sihotang, H. T., & Sinaga, B. (2019, August). "***Implementation of apriori algorithm for analysis of consumer purchase patterns"***. In Journal of Physics: Conference Series (Vol. 1255, No. 1, p. 012057). IOP Publishing.

33. Entisar E. Eljadi, Zulaiha Ali Othman, ***"Anomaly Detection for P.T.M.'s Network Traffic Using Association Rule"***, School of Computer Science Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia 28-29 June 2011.

34. Zhida Li, Ana Laura Gonzalez Rios, and Ljiljana Trajkovic ", **https:// www.kaggle.com /search?q= bgb +anomaly+dataset***"*, Procedia Comput. Sc, vol 173, 2020.

35. Fan, C., Xiao, F., Zhao, Y., & Wang, J. (2018). ***"Analytical investigation of autoencoder-based methods for unsupervised anomaly detection in building energy data"***. Applied energy, 211, 1123-1135.

36. R.T. Morris, *"A Weakness in the 4.2BSD Unix TCP/IP Software",* **ftp://netlib.att.com/netlib/att/cs/cstr/ll7.ps.Z** Technical Report 117, AT and T Bell Laboratories, Murray Hill New Jersey 07974, Feb. 1985.

37. R. Atkinson, ***"Security Architecture for the Internet Protocol"***, RFC1825, Aug. 1995.

38. S.Q. Zaho, ***"Association Rule Mining: A Survey"***, anyang Technological University, 2003.

39. Feng, C., & Hu, P. (2022). "***Towards Interpretable Anomaly Detection via Invariant Rule Mining"***. arXiv preprint arXiv:2211.13577.

40. Breier, J., & Branišová, J. (2017). "***A dynamic rule creation based anomaly detection method for identifying security breaches in log records"***. Wireless Personal Communications, 94(3), 497-511.

41. Xu, J., Wu, H., Wang, J., & Long, M. (2021). "***Anomaly transformer: Time series anomaly detection with association discrepancy"***. arXiv preprint arXiv:2110.02642.

RESUME

Mubaarak ABDULLAH ALTAMIMI graduated from first and elementary education in Yemen. He completed his high school education at altaawon high school, and then, he obtained a bachelor’s degree from the university of taiz in sciences, department of computer science in 2007. He studied turkish language in karabük, turkey, from 2020 to 2021. To complete his M.Sc. education, he moved to Karabuk/Turkey in 2021. He started his master's education at the department of computer engineering at Karabuk University.