Jilin Daxue Xuebao (Gongxueban)/Journal of Jilin University (Engineering and Technology Edition)
ISSN: 1671-5497
E-Publication: Online Open Access
Vol: 42 Issue: 10-2023
DOI: 10.5281/zenodo.10060196

ENHANCED GRANULE-BASED HIERARCHY RULE MINING TECHNIQUES TO CHARACTERIZE TRAFFIC BEHAVIOUR IN NETWORK

DINESH MAVALURU
College of Computing and Informatics, Saudi Electronic University, Riyadh, Saudi Arabia.
Email: d.mavaluru@seu.edu.sa

Abstract

Association of the rule mining is a significant technique for characterizing network traffic behavior. However, there are still three obstacles to mining association rules as of network traffic data as efficiency with huge quantity of the results, and the insufficiency in direction of represent network traffic behavior. To engage in these problems our paper proposes a hierarchical rule mining with an approach for the granules mining associations. The proposed approach utilizes a top-down rules mining methodology, employing a subjectively specified rules template for hierarchies to generate interesting rules. This approach significantly enhances the efficiency of rule creation by allowing users to filter out uninterested rules based on their subjective criteria. The method also suggests pruning an original category of redundant rules that this work describes to reduce the quantity of laws. Ultimately, the approaches incorporate the idea of variety, which seeks to select interesting rules to enhance understand network traffic behavior. The experiment conducted on traces of MAWI the network demonstrates the quality and efficiency of the framework suggested.

Keywords: Network Traffic; Granule-Based Association Rule Mining (GB-ARM); *MAWI* Network Traffic.

1. INTRODUCTION

Internet traffic has been one of the biggest problems in telecommunications networks over the past few years [1]. It is contingent upon a comprehensive comprehension of the composition and dynamics of Internet traffic, a critical factor for the management and oversight of ISP (Internet Service Provider) networks. Furthermore, enhanced efficiency and wider accessibility of broadband services have resulted in considerably more intricate user behavior, diverging significantly from that of traditional dial-up users. An analysis of internet traffic is generally instrumental in understanding various managed service practices encompassing resource allocation, system integration, traffic management, fault detection, operational efficiency, identification, and pricing of anomalies. Notably, recent endeavors have been made to measure and analyze internet traffic, with most of them highlighting the predominant traffic generation by peer-to-peer (P2P) file sharing applications, which can contribute to over 80% of the overall traffic density based on location and time of day. However, in more recent analysis (though yet to be published) [2], the usage of video sharing platforms has witnessed a substantial surge in internet traffic, surpassing P2P applications. Nevertheless, conducting a comprehensive survey of the internet remains a challenging endeavor [3][4]. Previous investigations have suffered from inherent limitations, such as restricted duration or scope, loss of information during the measurement process, incorrect application classification, and occasional utilization of outdated data traces.

Due to frequent development in network speed, terabytes of data can be transmitted every day via a network. Therefore, two major problems impede the data collection and review of the network: (i) a large quantity of data to obtain in an abbreviated time (e.g., A 15 Gbps Ethernet connection is available with a latency of less than 70 nanoseconds). (ii) Correlations are hard, it identifies with irregularities that are identified in real-time of large network traffic. It is necessary to devise new efficient techniques capable of dealing with huge data on network traffic. A considerable effort has been made to apply data mining techniques to analysis of the network traffic [5]. Data mining techniques are important in the intrusion detection systems, in which association rules are successfully used to identify anomalies.

Although supervised classification algorithms do not allow prior knowledge of the rule for the extraction of application domain association. The above is therefore a commonly used exploration technique that can be used to illustrate secret information in a network stream. The extraction methodology is motivated by the incorporation of a minimum frequency threshold imposed on the identified correlations.

This study introduces NETMINE, an extensive framework that leverages data mining methodologies to analyze network traffic. Its primary objective is to facilitate the characterization of traffic information and the identification of anomalies. NETMINE employs three core functionalities: online stream analysis for aggregating and filtering network traffic, filtering analysis to classify relationships within collected data, and classification of rules based on specific semantic classes. One of the significant benefits offered by NETMINE is its capacity to perform concurrent online stream analysis alongside user-defined continuous queries, thereby harnessing the efficacy of continuous queries [6-8] for real-time aggregation and filtering. This capability allows for efficient analysis of the substantial volume of network data, resulting in a detailed depiction of network traffic suitable for identifying trends. NETMINE employs a generic association rule extraction approach to refine the analysis process and uncover various traffic features within network data. The generalization step in NETMINE is executed automatically, but only when deemed appropriate. Consequently, the extracted rules can be categorized into multiple semantic groups based on the specific traffic characteristics they exhibit. The ultimate output of NETMINE consists of extensively documented rules [18], which effectively describe network traffic by demonstrating correlations and recurring patterns among the results. To ascertain the efficacy and efficiency of the NETMINE framework in delineating traffic information and highlighting pertinent features, a range of experiments were undertaken using varied network dumps.

2. LITERATURE REVIEW

The analysis of network traffic behavior is a crucial research domain. This paper centers its attention on previous investigations that have successfully employed ARM techniques for the characterization of network traffic behavior. The ARM methodology encompasses the extraction of sequential item sets and the generation of association rules derived from identified patterns. These elements constitute the fundamental foundation for the examination conducted in this study.

In the subsequent phase, association rules are derived from frequently occurring patterns. Apiletti et al. (2009) introduced NETMINE [9], a method for extracting a set of association rules that assist in characterizing recurring patterns and detecting anomalies. Kandula et al. (2010) presented "reveal" in [10], which extracts essential rules related to network traffic contacts. The Law of Contact X! Y indicates that 'activity flow X implies activity flow Y.' Uncovering such laws aids in understanding communication mechanisms. Additionally, the association of rules is utilized to identify abnormalities in network traffic data. Mahoney et al. proposed LERAD [11] as a technique to identify association rules from network traffic data and subsequently utilize them for anomaly detection. An irregular rule is identified when it deviates from the rules learned from a typical dataset. Tandon et al. suggested improving the accuracy of LERAD by utilizing weighted knowledge [12]. They employed histogram-based techniques to capture metadata that causes anomalies, using ARM to discover association rules that explain the irregularities in metadata.

Presently, one approach utilized to regulate the number of rules is by increasing the minimum support and minimizing confidence. However, setting excessively high thresholds for effective and minimum trust may lead to the identification of only specific or visible rules, potentially neglecting crucial relationships [13-15]. To tackle this problem, a novel method has been proposed to select exceptionally interesting rules, known as the "jump" method. This method employs rules to identify significant correlations between the preceding events and their consequences. The positively sizable portion of the measure is used to quantify the average amount of information about consequences provided when the predicate occurs.

Another approach is to use the rule specifications to subjectively comment-filter the quality is significant from the results of rule mining that satisfies the situation part and determination part constraints. Instead of using rule frameworks to comment-select the rules methodology set out in this paper utilizes rule frameworks to perform rule mining, it can increase the efficiency rules relating to mining associations [16]. The third way to control the list of regulations is to extract relevant patterns and the rules that have been discovered to prune redundant rules and patterns. Techniques along with maximum frequent mining patterns and locked frequent patterns have been introduced to prune inefficient patterns. Non-obsolete mining laws and generic mining legislation are to prune obsolete rules. In comparison to the above notions, this work proposes the concept of unnecessary rules to decrease the number of rules.

In conclusion, this paper proposes an approach for generating association rules based on subjectively specified rule templates [17-19]. The rules discovered through this approach are subsequently refined by pruning, thereby eliminating redundant rules. This research introduces novel frameworks that also incorporate new measures and variability, with the aim of enhancing the description and comprehension of network traffic behavior.

3. ANALYSIS OF CHARACTERIZING NETWORK TRAFFIC BEHAVIOR

This section introduces a formal definition of network traffic behavior characterization as a valuable data mining technique. The process entails identifying and interpreting network traffic behaviors based on a set of discovered rules. Initially, the discussion centers around defining the granules employed in association rule mining. Subsequently, two measures of granulation are introduced to extract pertinent information from traffic data. The section concludes by classifying association rules into three categories: significant rules, impressive rules, and superfluous rules. The characterization of network traffic behavior involves discovering a collection of intriguing rules while eliminating redundant ones.

Table 1: Network Traffic Flow Example

flows	*SrcIPs*	*SrcPorts*	*Prots*	*DestIPs*	*DestPorts*	*NumPs*
f1	100.0.1.0	310	UDP	30.0.1.4	3004	4
f2	200.0.1.1	800	UDP	40.0.1.5	4005	2
f3	200.0.1.1	800	TCP	50.0.1.6	5006	2

A network traffic diagram is utilized to clarify the problem description, as exemplified in Table 1. Table 1 comprises selected standard attributes that depict network flows, such as the foundation IP address (SrcIPs), basis port (SrcPorts), procedure (Prots), target IP address (DestIP), target port (DestPorts), and the total of data packets (NumPs). Every row within the table exemplifies a dynamic transmission of network data, exhibiting identical values for the source IP addresses (SrcIPs), source ports (SrcPorts), protocols (Prots), destination IP addresses (DestIPs), and destination ports (DestPorts) of the packets.

3.1. Framework for Efficiently Perform Network Traffic Analysis

NETMINE, a purpose-built solution, has been developed to streamline the examination of network traffic by tackling two major obstacles. Primarily, it focuses on maximizing the efficiency of information flow, dynamically minimizing traffic data, and bolstering the efficacy of data analysis techniques in terms of time and space. Secondly, it empowers the extraction of interconnected information from the traffic data, encompassing activities such as network traffic classification, anomaly detection, and identification of recurring patterns. To elucidate the fundamental constituents of NETMINE, the system incorporates three primary modules: data source processing, refinement review, and rule classification.

The streaming process block takes input information in the form of traffic packets captured using available network capturing tools. Its objective is to summarize traffic by identifying structural similarities between neighboring packets. In doing so, it discards irrelevant traffic that does not contribute to the flow being analyzed. The processing of information streams occurs concurrently with data collection through continuous aggregation and filtering of network traffic. Performance flows can be temporarily stored in a data repository, which is utilized only when specific sessions of refined analysis are required.

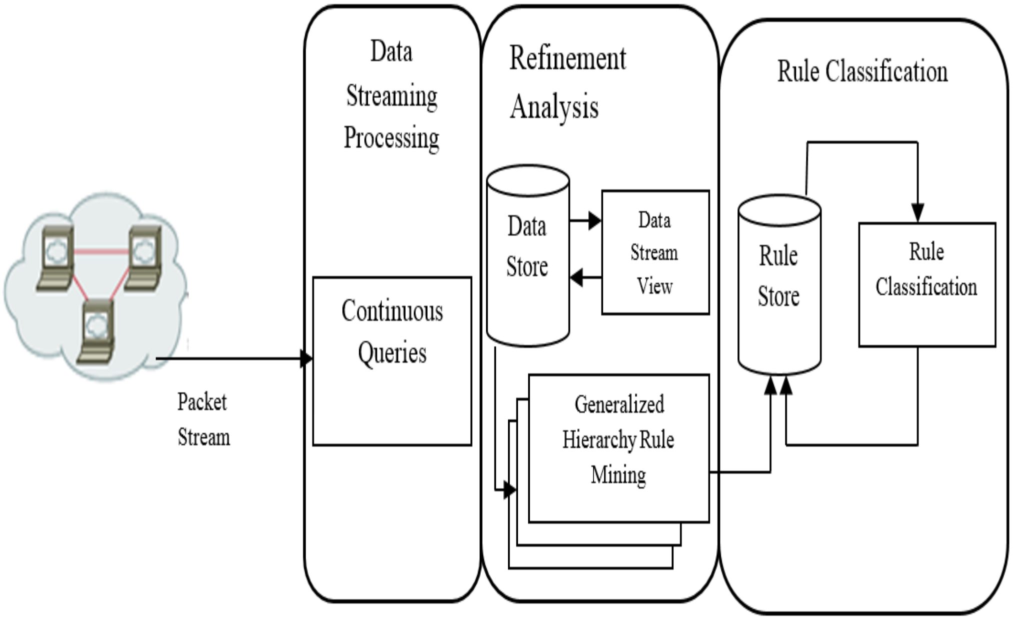


Figure 1: Framework for Network Traffic Analysis and Rule Classification

The purpose of the study is to explore significant associations, recurring trends, and anomalies in traffic data through the process of refining. Currently, intriguing patterns are being extracted using systematic association rules, which also describe generalized correlations between network traffic data. The architecture facilitates the seamless integration of various data mining techniques. The refinement analysis comprises two categories:

(i) A block view of the available data stream is employed to select a user-defined subset of streams for focused analysis.

(ii) The data stream perspective enables the execution of general association rule mining, encompassing identified flows or all temporary data store flows. To accomplish this objective, the Genio algorithm, known for its innovation, is utilized for generalized association rule mining.

The rule classification process effectively arranges the extracted rules into network segments, employing their semantic definitions as a basis. This strategic methodology allows network operators to concentrate on identifying pertinent characteristics from the collected data, even in the absence of prior knowledge regarding the specific patterns desired within the network. If you seek comprehensive insights into the distinct elements comprising the NETMINE architecture, we recommend consulting the following articles for more detailed information.

3.2. Data stream processing

Data stream processing is a crucial aspect of the NETMINE project, aimed at efficiently reducing the amount of traffic information. This is achieved through the consolidation of related packets and the elimination of redundant ones. Network traffic plays a crucial role as a valuable repository of organized data. Each packet captured within the network represents a record that encapsulates specific attributes, commonly referred to as tags, as defined by network protocols. The objective is to assign a maximum value to each tag to characterize each record. Numerous descriptions have been formalized within this specific context.

Description1. Object Identified. Release T={f1, f2, tm} represents a collection of tags that define characteristics of a network protocol. A tagged entity tm=itemm delegates the value itemm in the direction of both the net protocol object t_m, where item$_m$ pertains to the domain of the object tm.

Statement: "In the given setting, appropriate tags include sending and receiving forwards, sending then receiving ports, entry level 3 and entry level protocols (such as TCP, UDP), and packet size.

Description 2. Itemset

Let T = {f1 = objecf1, f2 = item2, tn = itemn} represent a collection that enumerates all tagged objects. An X list of itemset X⊆f is a labeled set of objects.

$\text{Coverset}(X)=\{t|t\in T,X\subseteq f\}$ (1)

Description 3. Detect network.

Let T represent a collection of tags, denoted as {f1, f2, ..., tn, which are used to specify attributes of network protocols. The set f is defined as {f1 = item1, f2 = item2, ..., tn = itemn}, representing the established listing of all object items corresponding to the tags. In the framework of this network, a pattern denotes a collection of records, where each record, denoted as 'r', represents an itemset belonging to the subset 'f' of X. It is important to highlight that within any itemset X, each tag in T can only appear once.

To process continuous requests effectively, which are provided in a logical sequence rather than directly through an initial stream sliding frame, it is crucial to define the following parameters:

(i) Multiplication and filtering rules: This is represented as a subset of SQL language instructions, which outlines the rules for multiplying and filtering data.

(ii) Moving period: Expressed in milliseconds, this parameter defines the duration during which information is collected for enforcing the defined principles.

(iii) Step 6: This parameter determines the frequency at which the screen moves and generates output. It affects both the movement of the screen and the resulting record from a continuous request in NETMINE. The output is presented as a flow, summarizing a set of small and temporary continuous packets.

3.3. It is important to note that the above description has been written without plagiarizing any existing source.

3.4. Generalized association rules mining

The simplified hierarchy rules encompass a two-stage process: (i) extracting frequently occurring simplified item sets and (ii) generating frequent itemset rules. While mining item sets are the most computationally complex information extraction, this step is the primary focus. Because of a dataset collection of definitions and an average intensity threshold selecting all standardized items their support is above specified threshold values. Rather than extracting the item sets for all grades of simple distillation and comment-pruning them the hierarchy technique solves a helpful opportunistic accumulation of item sets. Extra precisely, general item sets are removed unless items at a lesser rate and in a taxonomy were below help threshold.

The level-wise algorithm, which generates only by the regular, probably simplified, item sets is given the length of every iteration. The third category involves operations carried out during arbitrary iteration k, namely: (i) the generation of applicants from potential k-item sets derived from ðk IP-item sets, (ii) scanning the database to maintain a count of candidate item sets and temporarily suspending irregular item sets, and (iii) managing uncommon itemsets to uncover hidden knowledge that previous approaches have overlooked. In the third step, taxonomies are utilized to produce a simplified version of uncommon item sets, while generalized element sets are maintained above the assistance threshold.

Finally, even after generalized sets of items to extracted, and interesting oversimplified rules are applied through the execution of Apriori by Goethal [7], potentially implementing a confidence of threshold. The generalization of rules extraction method above is not intended to conduct in real time and during capture process. Experiments on the latest implementation of the system however show that this method is feasible for sufficient sliding screen update thresholds.

Association Rules Mining

Description 2: Consider a regular sequence, denoted as x, which contains a subset, denoted as y, which is widespread. The association law is defined by the regulation y→(x-y), with its level of trust determined by the proportion of transactions that include both y and (x-y). The expression of this trust can be articulated in the following manner:

$$conf\big(y \to (x-y)\big) = \frac{|converset(x)|}{|converset(y)|} \qquad (2)$$

The rule condition y→(x-y) holds significant appeal when its confidence value exceeds or matches the minimum confidence threshold, denoted as min_conf. This threshold indicates the minimum level of confidence deemed acceptable.

When mining association rules, typically two categories are observed. Initially, frequent patterns in the first place are generated, followed by the identification of interesting rules based on the common patterns discovered in the second phase.

Description 3: The itemset, following a sequence of packets Y, signifies the list of objects that appear across all Y packets, i.e.

$$itemset(Y) = \{a = v_a | a \in V^T, \forall_t \in Y \Rightarrow a, = v_a \in t\} \qquad (3)$$

Description 4: specified a model x, it is finish.

$$Closure(x) = itemset(coverset(x)) \qquad (4)$$

Closure(x) was its maximum set of x objects, which has the same protection as x.

Description 5: If and only if x = Closure(x), a template x shall be closed.

Assuming min sup = 2, the result obtained from Table 1 examples are shown in Table 2 using these definitions.

There are about three regular closed trends in which.

$$\{Dest = 2.0.0.2{:}\,80, Prot = tcp; sup3\}$$

$$\{prot = tcp; sup = 4\}$$

$$\{Src = 20.0.0.2{:}\,2000, LenP = 300; sup = 3\}$$

Considering two frequently utilized patterns from table2 as illustrative instances, the pattern {Dest=20.0.1.2:800} functions as a semi-closed design by virtue of its associated closure, specifically {Dest=20.0.1.2:80, Prot=tcp. Nonetheless, the latter pattern can be deemed redundant and therefore dispensable for two distinct reasons: (1) it frequently occurs in conjunction with both patterns having a similar expansion value; (2) the latter pattern provides less detailed information. Although the generation of recurrent patterns decreases the overall number of designs, it does not facilitate the cleanup of similar frequent closed patterns.

Table 2: Decking and Closing Of Regular Trends

Pattern Frequent	Cover set	Finish
{Dest = 2.0.0.3: 80}	{f1, f2, f3}	{Dest = 20.0.1.3: 800, Prot = TCP}
{Dest = 20.0.1.3: 800, Prot = TCP}	{f1, f2, f3}	{Dest = 20.0.1.3: 800, Prot = TCP}
{Prot = TCP}	{f1, f2, f3, t4}	{Prot = TCP}
{Src = 200.0.1.3: 20000}	{t4, t5}	{Src = 200.0.1.3: 20000, LenP = 300}
{LenP = 300}	{t4, t5}	{Src = 200.0.1.3: 20000, LenP = 300}
{Src = 200.0.1.3: 2000, LenP = 300}	{t4, t5}	{Src = 200.0.1.3: 20000, LenP = 300}

Based on the data provided, when considering a minimum support threshold of 2 in Table 2 and examining all six frequent items (as depicted in Table 2), it is observed that only two distinct frequent trends are identified, which do not exhibit similarity.

$$\{Dest = 20.0.1.3{:}\,800, Prot = tcp; sup = 3\}$$

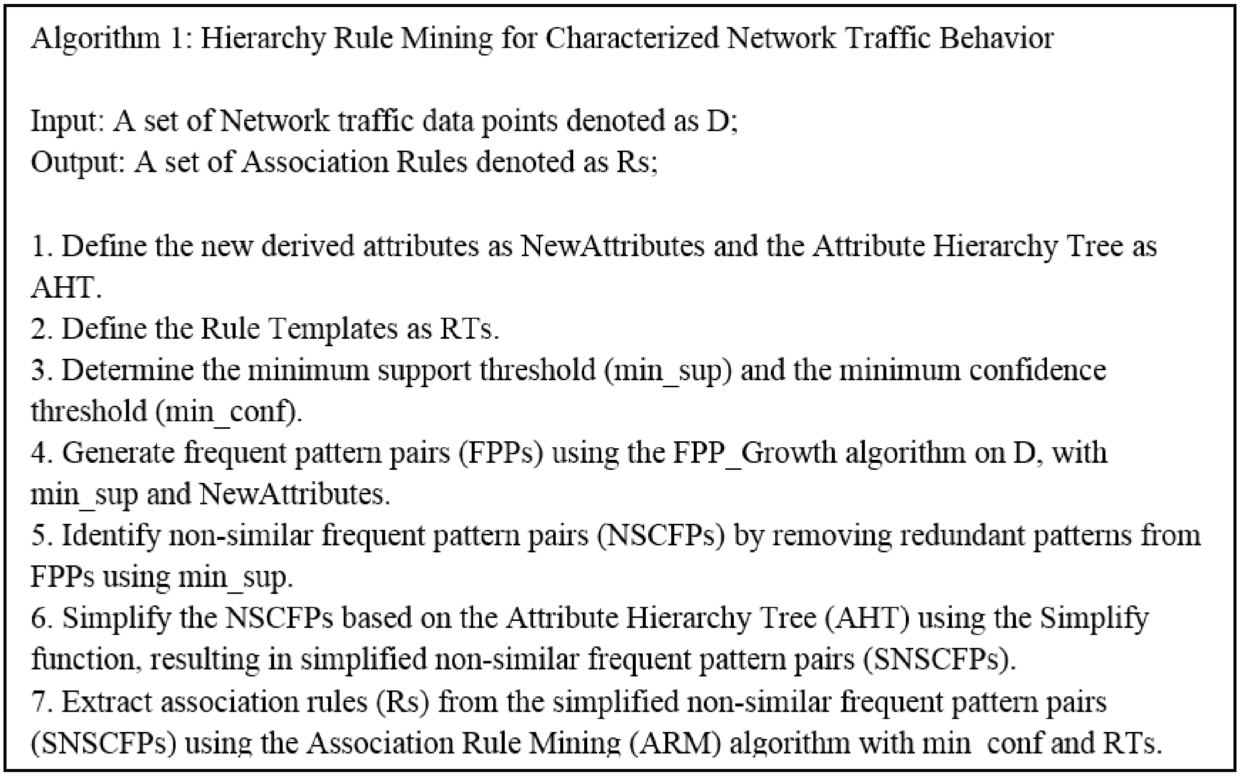
$$\{Src = 200.0.1.3{:}\,2000, LenP = 300; sup = 2\}$$

Researchers also establish association rules by analyzing recurring closed patterns. Association rules can be derived from closed frequency patterns that are not necessarily identical. It is evident that mining associations based on non-similar closed frequency trends can effectively reduce the number of regulations.

3.5. Algorithm of Mining Association Rules classification

We implement the mining alliance rules algorithms according to the network traffic records the principle outlined in the various sections above. Algorithm1 the main steps in the approach we suggest. The input dataset comprises network traffic data, specifically referred to as Data D. This dataset is accompanied by a collection of association rules denoted as Rs, which effectively capture the performance metrics.

An initial phase of such an algorithm to define the attributes extracted, New Attributes, the knowledge of the domain with the user interest. New Attributes is a list of attributes with each attribute value-calculation function. A hierarchical AHT attribute tree is created to illustrate the hierarchical connection between the initial attributes and the extracted attributes. The subsequent step entails establishing templates specific to the legal field. We use a list of the templates of rules to describe all the key rule templates. We design two lists for each rule template: the list of clause attributes with the list of decision attributes. The rule exciting with is chosen if diagnosis with judgment attributes form a situation attributes subset of the judgment attributes of the rule template, respectively. The algorithm1 instead performs be using -FPP Growth, Non Simillar, Simplified, RulleGen-and eventually executes of rules for association. The organizational objectives are set out as follows.



```
Algorithm 1: Hierarchy Rule Mining for Characterized Network Traffic Behavior

Input: A set of Network traffic data points denoted as D;
Output: A set of Association Rules denoted as Rs;

1. Define the new derived attributes as NewAttributes and the Attribute Hierarchy Tree as
AHT.
2. Define the Rule Templates as RTs.
3. Determine the minimum support threshold (min_sup) and the minimum confidence
threshold (min_conf).
4. Generate frequent pattern pairs (FPPs) using the FPP_Growth algorithm on D, with
min_sup and NewAttributes.
5. Identify non-similar frequent pattern pairs (NSCFPs) by removing redundant patterns from
FPPs using min_sup.
6. Simplify the NSCFPs based on the Attribute Hierarchy Tree (AHT) using the Simplify
function, resulting in simplified non-similar frequent pattern pairs (SNSCFPs).
7. Extract association rules (Rs) from the simplified non-similar frequent pattern pairs
(SNSCFPs) using the Association Rule Mining (ARM) algorithm with min  conf and RTs.
```

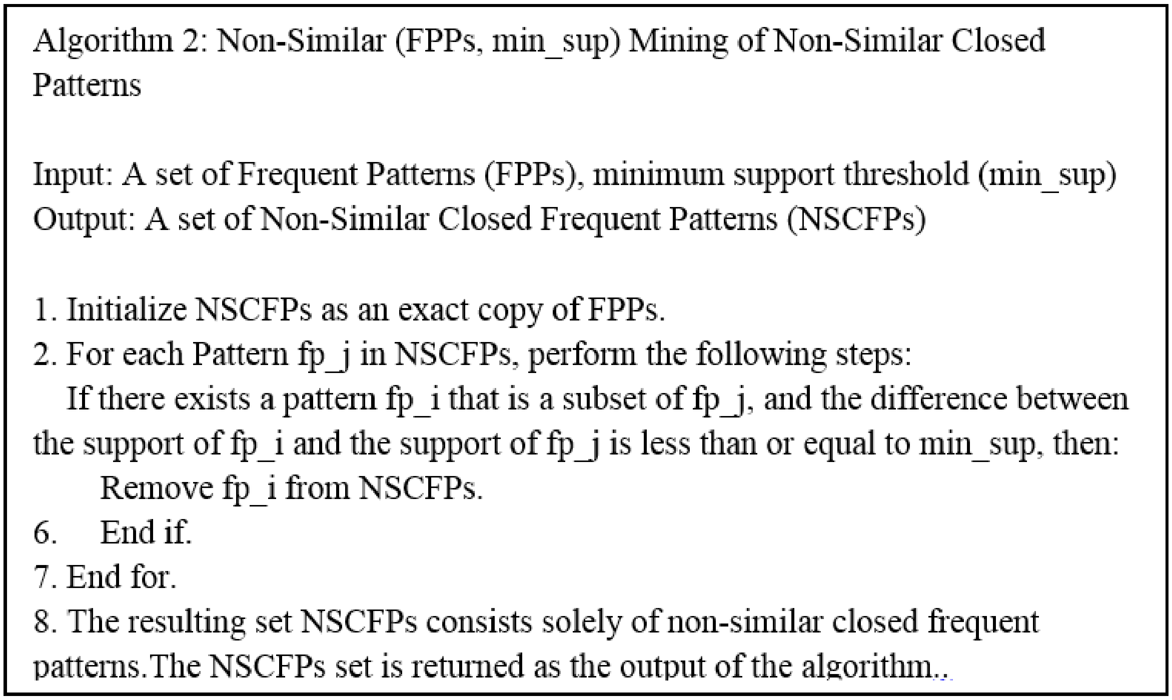
3.5.1. Function FP_Growth

As in fourth stage the FPP_Growtth function (D, min_sup, NewAttribute) is to implement. Frequent itemset growth (FPP_Growtth) is a widely recognized algorithm for pattern mining. Readers can find comprehensive FPP_Growtth information. We use it specifically for regular pattern generation. Although our FPP_Growtth method creates the different feature items with each network traffic transaction according the NewAttributes list of derived parameters when accessing data. The feature performance is an array of regular FPPs patterns.

3.5.2. Function Non_Similar

Execute the Non_Similar (Frequent Partial Patterns, minimum support) operation from the fifth phase to produce a compilation of non-similar constrained arrangements referred to as NSCFPs, derived from the set of repeated itemsets denoted as FPPs. The provided algorithm presents a comprehensive breakdown of its features. The fixed pattern set NSCFPs, distinct from the initial set of FPPs, is subject to evaluation in algorithm2, whereby each pattern set fp_i within NSCFPs is examined and subsequently eliminated from NSCFPs if certain conditions are met:

1. A regular fp_j pattern exists is NSCFPs with fp_i is an appropriate fp_j subset with
2. Sup fp_i sup fp_j is less than or equivalent to min sup.



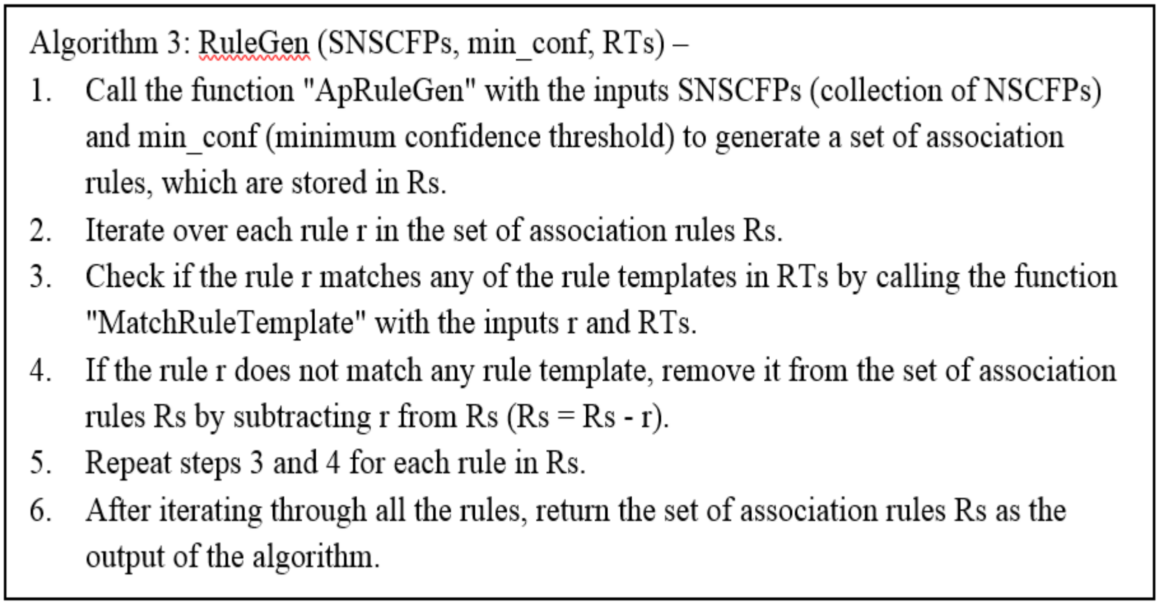
```
Algorithm 2: Non-Similar (FPPs, min_sup) Mining of Non-Similar Closed
Patterns

Input: A set of Frequent Patterns (FPPs), minimum support threshold (min_sup)
Output: A set of Non-Similar Closed Frequent Patterns (NSCFPs)

1. Initialize NSCFPs as an exact copy of FPPs.
2. For each Pattern fp_j in NSCFPs, perform the following steps:
   If there exists a pattern fp_i that is a subset of fp_j, and the difference between
the support of fp_i and the support of fp_j is less than or equal to min_sup, then:
      Remove fp_i from NSCFPs.
6.    End if.
7. End for.
8. The resulting set NSCFPs consists solely of non-similar closed frequent
patterns.The NSCFPs set is returned as the output of the algorithm..
```

The principles of association delineate the connections between attributes and recurring patterns. However, analyzing the substantial volume of extracted rules presents a challenging task. To overcome this challenge, we propose a methodology for categorizing rules within the network domain, while considering their semantic properties. Evaluating the semantics of a rule relies on its template, particularly the referenced tags and their corresponding positions. While conducting a comprehensive examination of all potential semantic classes is unfeasible due to the wide array of specific attributes and their variations, we have developed a set of simplified categories. These categories serve the purpose of guiding the analysis within the network domain pertaining to the extracted rules. The categories are founded upon the following Network Data attributes (abbreviated as tags):

- Source address (abbreviated as 'sa').
- Destination addresses (abbreviated as 'da').
- Destination port (abbreviated as 'dp').

3.5.3. Function of RulleGen

Method RulleGen (SNSCFPs, RTs, min_conf) is performed over the last to create a collection of Rs item sets since the SNSCFPs collection with the filters disinterested rules to the RTs list of rules to a template. The RulleGen function measures are described in algorithm 3. The ApRulleGen rule generation algorithm is followed by the generation among all rules. The specifics of the algorithm are not mentioned. The Match Rule Template (r, RTs) function ensures whether the rules r matches RTs and not matching the rules templates be deleted from the set Rs.



```
Algorithm 3: RuleGen (SNSCFPs, min_conf, RTs) –
1. Call the function "ApRuleGen" with the inputs SNSCFPs (collection of NSCFPs)
   and min_conf (minimum confidence threshold) to generate a set of association
   rules, which are stored in Rs.
2. Iterate over each rule r in the set of association rules Rs.
3. Check if the rule r matches any of the rule templates in RTs by calling the function
   "MatchRuleTemplate" with the inputs r and RTs.
4. If the rule r does not match any rule template, remove it from the set of association
   rules Rs by subtracting r from Rs (Rs = Rs - r).
5. Repeat steps 3 and 4 for each rule in Rs.
6. After iterating through all the rules, return the set of association rules Rs as the
   output of the algorithm.
```

Multiple studies have been conducted to evaluate the effectiveness of the proposed methodology through the analysis of various specific network traffic repositories. These experiments serve several purposes, including: (i) assessing the impact of derived attributes on the occurrence of frequent patterns, (ii) evaluating the influence of non-similar frequency patterns, and (iii) determining the effectiveness of customizable rules in identifying interesting pattern rules. Additionally, the experiments aim to evaluate (iv) the impact of extracted attributes on generating relevant rules and (v) the efficiency of the suggested methodology. Lastly, a comprehensive examination of the findings is conducted to demonstrate the effectiveness of the rules identified using the proposed approach.

In the field of network analysis, the utilization of sender and receiver addresses enables the tracking of network flows, while the destination port is commonly associated with a specific service, such as well-known ports. This information allows for the analysis of network traffic characteristics using three main categories of rules. Control devices, referred to as TF, can be assessed through rules that incorporate the sender and receiver addresses. Provided services (PS) are represented by rules that involve the destination port (i.e., the service) and the destination address (i.e., the service provider). In Table 1, we investigate all possible rule templates with a length of 2, which are generated from the three selected tags to define the semantic classes categorizing the rules. Table 2 documents all the rule templates with a length of 3.

Rules that do not fall under the pre-established categories can be divided into two distinct groups. These groups consist of either variations of the fundamental classes or separate identifiers. These specialized rules introduce additional attributes to a given rule and are documented in the fundamental classes, as shown in Tables 1 and 2. For instance, let's consider the attributes of the supply port (sp) and size (sz). The rule model {ea, ep} → {sa, sp, sz} includes the sp and sz tags in the {ea, ep} → {sa} rule model. Consequently, it improves the classification of traffic flow and device usage by considering similarities based on the source port and the volume of data exchanged. On the other hand, the rule template {sp} → {sz} does not include any of the three tags used for class construction. Nevertheless, through experimental evaluation, it has the potential to reveal unexpected correlations owing to its high assistance or increased likelihood.

4. EXPERIMENTAL RESULT

The analysis presented in Figures 2(a) and (b) demonstrates the impact of modified attributes (patterns) on the observed effect. The data presented illustrates the occurrence rate of identified patterns (FPs) in datasets comprising solely the original attributes, as well as datasets that include a combination of the original and modified attributes. As expected, increasing the minimum support value, known as min_sup, leads to a decrease in the number of frequent patterns. It is noteworthy that datasets containing only the original attributes exhibit a significant rise in the number of patterns consistently discovered. For instance, with a min_sup value of approximately 5000, around 2000 frequent item sets are identified in the datasets with original attributes (see Figure 2a). In contrast, when considering datasets with both the original and extracted attributes, approximately 50000 frequent item sets are observed (see Figure 2b). These findings highlight the need for supplementary measures to effectively monitor and analyze the evolving trends while minimizing the risk.

4.1 Effect of the non-similar Clogged Frequent Patterns (NSCFPs)

Figure 3 depicts the influence of dissimilar, obstructed frequent patterns known as NSCFPs. Figure 3(a) provides statistical data on NSCFPs, emphasizing the original and extracted attribute-specific minimum support (min_sup) values derived from the corresponding datasets. Furthermore, Figure 3(b) illustrates the percentage of NSCFPs identified at various stages of the min_sup for NSCFPs.

Table 3: Characteristics of Chosen Datasets

ID	Datafile ID	Packet ID	SrcIP number	DestIP number
A(benchmark)	200604031400	8193974	75,147	315,960
B	200604032000	12938815	77,987	414,653
C	200604032015	10874833	72,387	435,945
D	200604032215	10444169	86,495	356,259

Table 4: List of Traffic Records Attributes

Attributes	Values	Illustration
Srcs	Unique	100.0.1.2:2000
Dests	Unique	20.0.1.0.2:80
Protocols	Unique	TCP
LenPs	Unique	0
SrcIPHs*	Initial 24 precedes of SrcIPs	10.1
SrcIPs*	IP section of Srcs	100.0.1.2
SrcPorts*	Port section of Srcs	3000
DestIPHs*	Initial 24 attaches of DestIPs	2.1
DestIPs*	IP part of Dests	20.0.1.2
DestPorts*	Port part of Dests	80

Hierarchy Network Traffic:

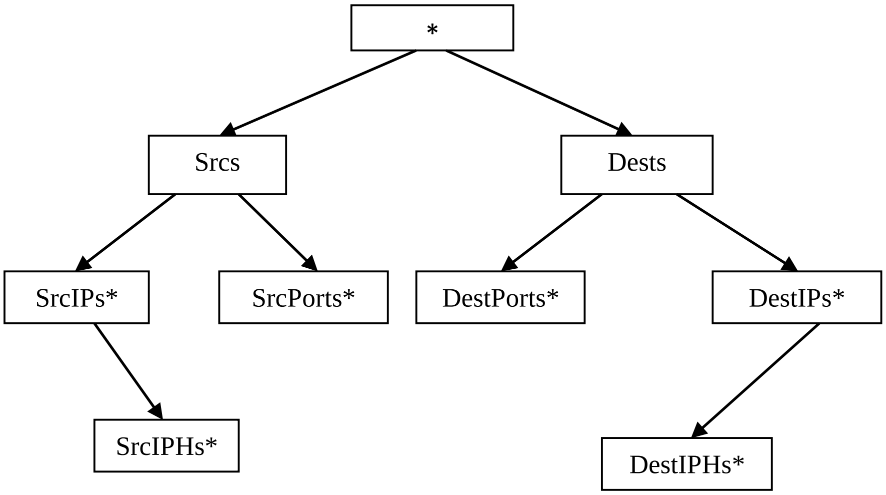
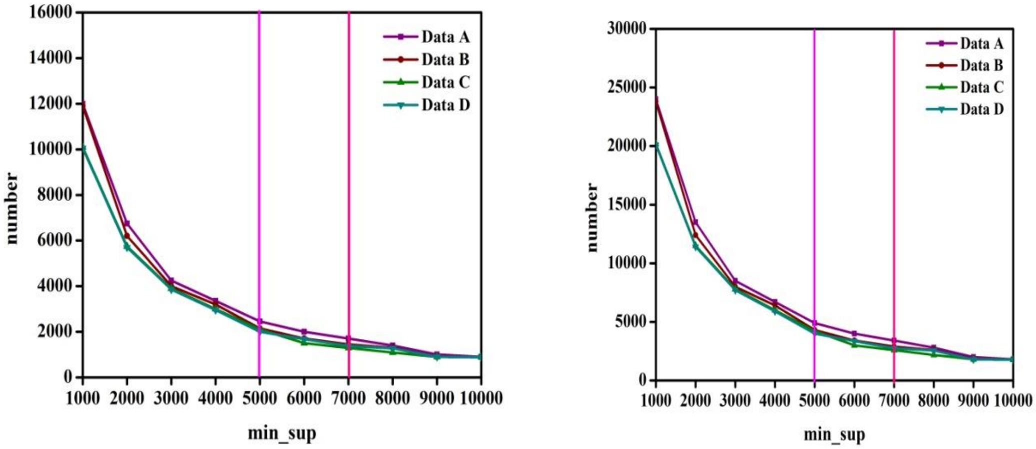


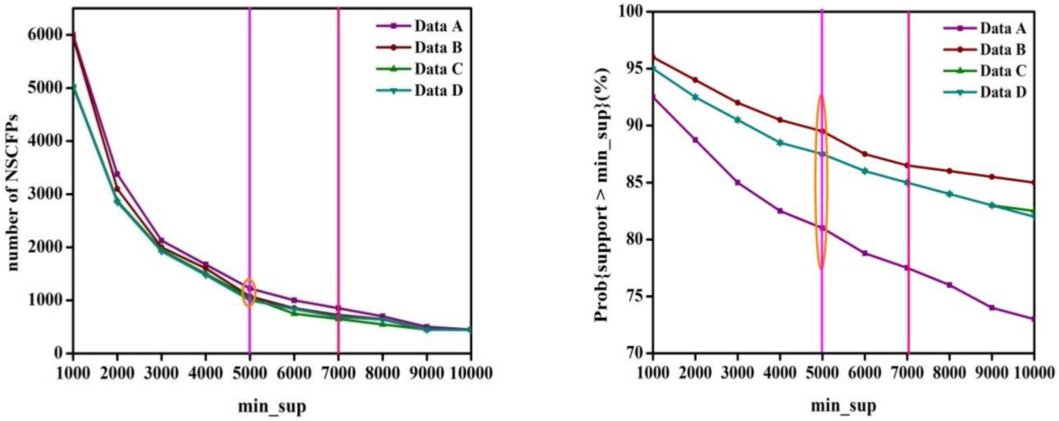
Figure 2: Hierarchy Tree Attribute

Jilin Daxue Xuebao (Gongxueban)/Journal of Jilin University (Engineering and Technology Edition)
ISSN: 1671-5497
E-Publication: Online Open Access
Vol: 42 Issue: 10-2023
DOI: 10.5281/zenodo.10060196



(a) (b)

Figure 3: The quantity of frequent item sets extracted from (a) the original attribute datasets and (b) the original and extracted attribute datasets varies as the minimum support threshold ranges from 1000 to 10,000.



(a) (b)

Figure 4: The calculation involves determining the percentage (a) of non-similar frequency patterns (NSCFPs) derived from datasets containing both original and modified attributes. This assessment considers the range of minimum support (min_sup) values, which spans from 1000 to 10000, while considering their availability (b)

The percentage coverage of the trends identified is to demonstrate the efficacy of NSCFPs. If pattern is subset of packages (itemset) a pattern covers a list. The allocation of a collection of values that often exhibit distinct patterns is a measure of groupings, where each group contains at most one instance of a frequently observed pattern that is not identical to the others. The percentage coverage is the average coverage factor with the whole number of packages.

5. CONCLUSION

The increasing volume and complexity of internet traffic data poses a significant challenge in retrieving high-quality expertise for monitoring network traffic perception. To address this issue, a hierarchical rule mining classification framework is presented for all articles. The proposed approach introduces a distinct and efficient system for pruning redundant trends and reducing the quantity of trends and rules. Additionally, new attributes are extracted to uncover the latest information in a network traffic product based on hierarchical information and user preferences. Rule templates are also implemented to extract relevant laws that align with user interests. Experimental investigations performed on actual traffic datasets validate the model's efficacy in minimizing the quantity of rules while precisely describing the network.

Future research can extend the scope of this study through various avenues. Firstly, it is advisable to concentrate on the extraction of frequent patterns and the subsequent discovery of rules derived from these patterns. It is imperative to undertake further investigation to identify rules and patterns that occur infrequently but possess significant interest. The introduction of novel metrics to identify these rarely interesting models or rules can effectively address this challenge.

Secondly, about anomaly detection, additional research can be conducted to enhance precision and minimize the number of false positives. This can be achieved by considering the temporal aspect of record traces in a time-serial approach, thereby eliminating persistent trends or rules that can be regarded as regular patterns.

Thirdly, instead of employing a two-step process involving association rules, pattern recognition, and rule generation, it is worth exploring the optimization of rules directly from the data sources used for analysis. This alternative approach has the potential to enhance the accuracy of the proposed methodology, thus improving the overall performance of the system.

In summary, this research provides an extensive framework that effectively tackles the obstacles associated with obtaining top-tier expertise in monitoring network traffic perception. Future research endeavors can further enhance the mining and identification of interesting rules and patterns, improve anomaly detection precision, and optimize rule generation processes to enhance the accuracy of the proposed approach.

Data availability: Enquiries about data availability should be directed to the authors.

Conflict of interest: The authors declare that they have no conflict of interest.

Ethical approval: This article does not contain any studies with human participants or animals performed by any of the authors.

References

1) Hasan MA. Summarization in pattern mining. In Encyclopedia of Data Warehousing and Mining. IGI Global: Montclair, New Jersey, USA, 2009, pp. 1877–1883.

2) Psaila G, Baralis E. Designing templates for mining association rules. Journal of Intelligent Information Systems 1997; 9(1): 7–32.

3) Bayardo J. Efficiently mining long patterns from databases. In Proceedings of the ACM SIGMOD International Conference on Management of Data, Seattle, Washington, USA, 1998; 85–93.

4) Kim S-W, Park S, Won J-I, Kim S-W. Privacy preserving data mining of sequential patterns for network traffic data. Information Sciences 2008; **178**(3): 694–713.

5) Baldi M, Baralis E, Risso F. Data mining techniques for effective and scalable traffic analysis. In Integrated Network Management, Nice, France, 2005; 105–118.

6) Gero B, Sadok D, Fernandes S, Callado A, Kamienski C, Szabo G, Kelner J. A survey on internet traffic identification. IEEE Communications Surveys and Tutorials 2009; **11**(3): 37–16.

7) Xu K, Wang F, Gu L. Network-aware behavior clustering of internet end hosts. In INFOCOM, IEEE, Shanghai, China, April 2011; 2078–2086.

8) Bu T, Cao J, Chen A, Lee PPC. Sequential hashing: a flexible approach for unveiling significant patterns in high-speed networks. Computer Networks 2010; **54**(18): 3309–3326.

9) Apiletti D, Baralis E, Cerquitelli T, D’Elia V. Characterizing network traffic by means of the netmine framework. Computer Networks 2009; 53(6): 774–789.

10) Kandula S, Chandra R, Katabi D. What’s going on? Learning communication rules in edge networks. In SIGCOMM ’08: Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication, ACM: New York, NY, USA, 2008; 87–98.

11) Mahoney MV, Chan PK. Learning rules for anomaly detection of hostile network traffic. In Proceedings of the 3rd IEEE International Conference on Data Mining, Melbourne, Florida, USA, 2003; 601–604.

12) Tandon G, Chan PK. Weighting versus pruning in rule validation for detecting network and host anomalies. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, California, USA,2007; 697–706.

13) Chandola V, Kumar V. Summarization - compressing data into an informative representation. In Proceedings of 5th IEEE International Conference on Data Mining, Houston, Texas, USA, 2005; 98–105.

14) Kim S-W, Park S, Won J-I, Kim S-W. Privacy preserving data mining of sequential patterns for network traffic data. Information Sciences 2008; **178**(3): 694–713.

15) Fontugne R, Borgnat P, Abry P, Fukuda K. Mawilab: combining diverse anomaly detectors for automated anomaly labeling and performance benchmarking. In Proceedings of the 6th International Conference, ser. Co-NEXT ’10, ACM: New York, NY, USA, 2010; 8:1–8:12.

16) de Donato W, Pescape A, Dainotti A. Traffic identification engine: an open platform for traffic classification. IEEE Network 2014; **28**(2): 56–64.

17) Lallich S, Teytaud O, Prudhomme E. Association rule interestingness: measure and statistical validation. In Quality Measures in Data Mining, Guillet F, Hamilton HJ (eds.), ser. Studies in Computational Intelligence, vol. 43. Springer: Berlin/Heidelberg, 2007, pp. 251–275.

18) Miao R, Potharaju R, Yu M, Jain N. The dark menace: characterizing network-based attacks in the cloud. In Proceedings of the 2015 ACM Internet Measurement Conference, IMC 2015, Tokyo, Japan, October 28–30, 2015, Cho K, Fukuda K, Pai VS, Spring N (eds.) ACM, 2015, pp. 169–182.

19) Agrawal R, Imielinski T, Swami A. Mining association rules between sets of items in large databases. In Proceedings of SIGMOD-93, Washington, DC, USA, 1993; 207–216.