International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Evolution of Nb-IWD Based Method for Network Traffic Classification Using KDD Dataset

First Author*1, Second Author2, Third Author2, Fourth Author3
Shahin Quainat[1], Ritesh Kumar Yadav[2]
1 M.TECH Scholar, SRK University, Bhopal
2 Associate Professor, SRK University, Bhopal
E Mail squainat@gmail.com , er.ritesh1987@gmail.com

ABSTRACT

The main focus of this paper is to sorting out the problems which comes while handling network traffic whereas some of the traffic classification methods are unable to find out the special requirements of individual datasets because there are massive measure of network traffic datasets and restricted quantities of resources are accessible to deliver classification examination. The paper uncovers that traffic arrangement should be refreshed normally to keep up the precision and ought to have the capacity to adjust the dynamic conduct of network stream.

Keywords: Research Paper, Technical Writing, Science, Engineering and Technology

I. INTRODUCTION

Traffic classification is the general block that is required to empower any traffic executive tasks, from separating traffic pricing and treatment (e.g., policing, molding and so on.), to security activities (e.g., firewalling, sifting, detection of anomalies etc..).

Properties of Traffic Classification

The most important properties of a traffic classifier, which determine its applicability to different network tasks, are:

- Granularity: The recognize between the coarse-grained algorithms, which always consider huge group of protocols (for example P2P versus non P2P, HTTP versus Streaming) and finegrained classifiers which rather, attempt to recognize the particular protocols (for example BitTorrent versus eDonkey filesharing), or even the particular application (for example PPlive versus SopCast live streaming).
- Timeliness:Some early classification methods can rapidly recognize the traffic, after a couple of packets, hence, being appropriate for such tasks requiring a brief response (for example security). Late classification algorithms take more time to gather traffic properties, and for some situation they even need to wait lot for termination of flow (i.e., after death classification): such systems are shown for observing tasks, for example, charging.
- Computational cost: The preparing power expected to assess traffic and take the classification factor is a vital factor while picking an arrangement algorithm. While executing the packet processing, the most costly activity is typically packet memory access, followed by ordinary expression coordinating.

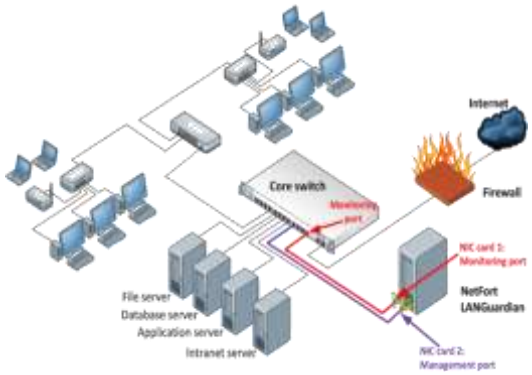

Figure 1.1: Network traffic monitoring

The figure 1.1 above shows a typical setup if you want to detect any unusual traffic on your network. This uses LANGuardiantraffic analysis tool to monitor traffic

coming from a SPAN\Mirror port on our core switch. LANGuardian is deep-packet inspection software that monitors network and user activity. The core switch is configured to send a copy of all traffic going to and from the firewall to the monitoring port which is also known as a SPAN or mirror port.

Supervised Methods: Supervised techniques, otherwise called arrangement or classification strategies, separate information structures to arrange new cases in pre-characterized classes. It is essential to note that is called managed in light of the fact that the yield classes are pre-characterized. The procedure of a regulated ML techniques begin with a preparation dataset TS characterized as,

T S = < x1, y1 >, < x2, y1 >, ..., < xN , yM >, where xi is the vector of estimations of the highlights relating to the I th example, and yi is its yield class esteem. It finds the distinctive relations between the occurrences and yields a structure, normally a choice tree or order governs, that will characterize the cases in a discrete set y1, y2, ..., yM. There is a great deal of related work that utilization administered procedures [1] with a promising outcomes. The supervised traffic grouping strategies dissect the managed preparing information and produce a deduced capacity which can anticipate the yield class for any testing flow. In regulated rush hour gridlock characterization, adequate managed preparing information is a general supposition. To address the issues endured by payload-based traffic characterization, for example, scrambled applications and client information.

Unsupervised algorithms try not to should be prepared with wanted result information. Rather, they utilize an iterative methodology called profound figuring out how to audit information and land at ends. Unsupervised learning calculations - likewise called neural systems - are utilized for more mind boggling handling errands than directed learning frameworks, including picture acknowledgment, discourse to-content and regular language age. These neural systems work by sifting through a great many instances of preparing information and naturally distinguishing regularly unobtrusive connections between's numerous factors. When prepared, the calculation can utilize its bank of relationship to translate new information. These calculations have just turned out to be achievable in the period of huge information, as they require huge measures of preparing information.

Classification using Semi Supervised Technique:Uses a set of supervised training data in an unsupervised approach to address the problem of mapping from flow clusters to real applications

The organization of this paper is as follows. In Section 2 (Methods and Material), I'll give detail of method to be used. In Section 3 (Result and Discussion), present our research findings and analysis of those findings. Section 4 gives (Conclusion).

II. METHODS AND MATERIAL

A Naïve-Bayes (NB) ML algorithm is a simple structure consisting of a class node as the parent node of all other nodes. The basic structures of Naïve Bayes Classifier is shown in Figure 2 in which C represents main class and a, b, c and d represents other feature or attribute nodes of a particular sample. No other connections are allowed in a Naïve-Bayes structure. Naïve-Bayes has been used as an effective classifier. It is easy to construct Naïve Bayes classifier as compared to other classifiers because the structure is given a priori and hence no structure learning procedure is required. Naive Bayes assumes that all the features are independent of each other. Naïve-Bayes works very well over a large number of datasets, especially where the features used to characterize each sample are not properly correlated.

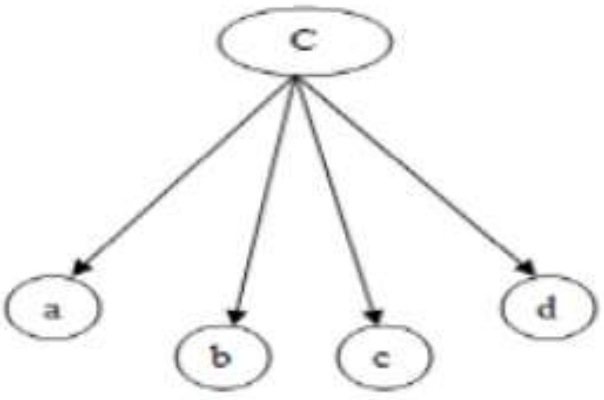

Figure 2.1: Naïve Bayes Classifier

In the above figure 2.1 the Naïve Bayes Classifier is shown.

METHODOLOGY:

In order to achieve high quality truth data which contains the identification of the traffic in every application, the proposed technique works on the cluster analysis that omes under the unsupervised ahine learning algorithm and the supervised classifier training in tandem. A high level overview of the traffic classification scheme is shown in Figure 1 with a description of principal steps as follows.

(i) Preprocessing. Internet traffic is collected from enduser machines and marked with application labels accordingly (e.g., Skype and YouTube) using a localized operational packet-level classifier. Application labelled traffic is afterwards exported as flows using a flow exporting utility for unsupervised cluster analysis.

(ii) Cluster Analysis. Utilizing unsupervised *k*-implies, streams having a place with individual applications are independently bunch broke down to remove one of a kind subclasses for each application, offering a better granularity of the arrangement (e.g., YouTube and Netflix streams would be classed as gushing and perusing).

(iii) Classifier Training. Streams or packets set apart with their *k*-implies groups, showing the subclass they have a place with, are subsequently nourished to a Naive classifier for directed preparing, prompting a choice tree.

Evaluation. Different datasets will be used for the purpose of testing the accurate rate of the proposed algorithm. The Naïve Bayes IWD approach is helpful in creating the application and the sub-class of the parent class of the flow based on their respective attributes, ingrained during decision tree creation.

III. **RESULTS AND DISCUSSION**

3.1 SIMULATION ENVIRONMENT

An application has been developed which provides two-factor authentications. In developing this application, using JDK 1.8 that is a JAVA developing kit. JAVA is a computer language. It helps the developer to write a code as per the requirement and run it anywhere. This type of language is known as high-level language because it is under stable and easily written by a human. JAVA has a set of rules that determine how the instruction is written. These rules are known as its syntax. Once a program has been written the high-level instruction are translated into numeric codes that computers can understand and execute. Web-based content and enterprise software. JAVA development kit a software development kit (SDK) for producing JAVA programs. The JDK is developed by Oracle INC Java soft division [3-7].

Netbeans: The NetBeans IDE is an honor winning incorporated improvement condition accessible for Windows, Mac, Linux, and Solaris. The NetBeans venture comprises of an open source IDE and an Application stage that assistance engineers to quickly make web, endeavor, work area, and versatile applications. It offers an undeniable IDE that keeps running on different stages and has support for pretty much every prominent language you need to code in . IT Consultant, Danial Oz says 'The stages upheld are Java , JavaFX, PHP, JavaScript and Ajax, Ruby and Ruby on Rails, Groovy and Grails, and C/C++" [2]. The NetBeans venture is upheld by an incredibly dynamic and energetic engineer network and incorporates point by point and wide documentation and preparing assets. A broad accumulation of outsider modules are likewise accessible for Netbeans . The NetBeans coordinated advancement condition (IDE) can help your efficiency to an incredible broaden . Visual devices that produce skeleton code are additionally accessible, giving you a chance to make an essential application without composing a solitary line of code [8-14].

3.2 RESULT ANALYSIS

In proposed system, a novel parametric approach is used to deal with the correlated flows in an effective way, which can significantly improve the classification performance.

A. Pre-processing

Here the IP packets crossing across a network is collected and used for constructing the flows by examining the header of packets.

B. Correlation Based Feature Selection

Here measurable highlights are extricated and are utilized to speak to traffic streams that are finished by pre-preparing to apply include choice to expel superfluous and repetitive highlights from the list of capabilities.

C. Feature Discretization

Discretization is a process of converting numeric values into intervals and associating them to a nominal symbol. These symbols are then used as new values instead of the original numeric values.

D. Naïve Bayes Classification

A Naïve-Bayes (NB) ML algorithm is a simple structure consisting of a class node as the parent node of all other nodes. The basic structures of Naïve Bayes Classifier is shown in Figure 2.1 in which C represents main class and a, b, c and d represents other feature or attribute nodes of a particular sample. No other connections are allowed in a Naïve-Bayes structure.

Naïve-Bayes has been used as an effective classifier. . It is easy to construct Naïve Bayes classifier as compared to other classifiers because the structure is given a priori and hence no structure learning procedure is required. Naive Bayes assumes that all the features are independent of each other. Naïve-Bayes works very well over a large number of datasets, especially where the features used to characterize each sample are not properly correlated.

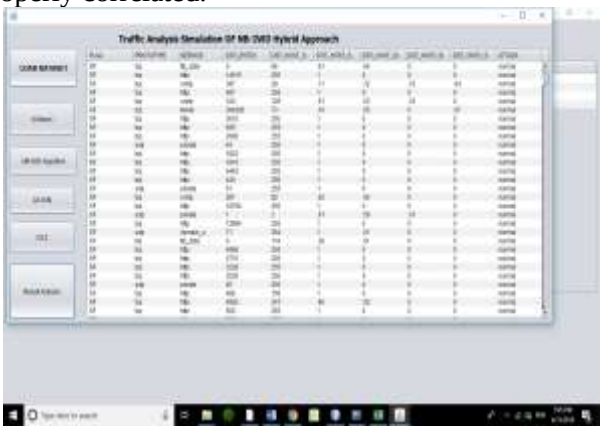

Figure 3.1: Traffic analysis simulation

In the above figure 3.1 the representation of the hybrid approach which includes the loading of the datasets along with applying the proper algorithm.

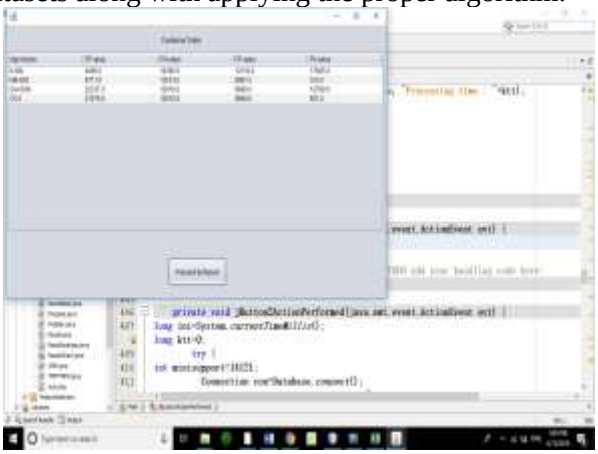

Figure 3.2: Conclusion table.

In the above figure 3.2 the representation of the conclusion table has been shown that consists of the FP value, FN value for various algorithms KNN, SVM, C5.0 along with the resultant values.

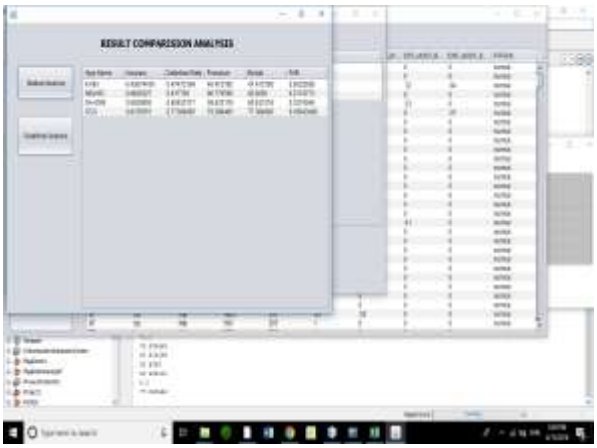

Figure 3.3: Comparison table.

In the above figure 3.3 the comparison table is shown with the respected outputs getting from the algorithms- KNN, SVM, NB-IWD, C5.0.

Performance Evaluation

Performance evaluation of the algorithm is done by using the following metrics: overall accuracy, precision, recall, F-measure, and classification speed.

F-Measure: It can be defined as an information retrieval (IR) system has recall R and precision P on a test document collection and an information need.

Accuracy: It can be defined as the sum of all the values that is divided by the given set of numbers.

Recall: The ratio of True Positives over the sum of True Positives and False Negatives.

Algorithms	Naïve Bayes	BayesNet
Correctly classified	270	331
Incorrectly Classified	86	92
Overall Accuracy (%)	73.232	90.07
Error (%)	3.76	2.92
Time (sec)	0.04	0.5

In the above table 4.2 the algorithms on the basis of the correctly classified ratio, incorrectly classified ratio, overall accuracy, error and time is shown.

CONCLUSION

Traffic classification plays an important role in the network security as the applications and their behavior are changing day to day. As a result there increased the need for accurate classification of the network flows. Here we have proposed a Naïve Bayes model with feature selection for the accurate classification of internet traffic. We have compared the method with three other Bayesian models. Our experiment shows that it provides an accuracy of 96.5% which is better than that of the other state-of-the art methods. NBD is easy to build and is applicable to various real world applications. In this work, a new traffic classification scheme is proposed which can effectively improve the classification performance in the situation that only few training data are available. The proposed scheme is able to incorporate flow correlation information into the classification process. A new Naïve Bayes method was also proposed to effectively aggregate the correlation naive Bayes (NB) predictions. The experiments performed on real-world network traffic datasets demonstrated the effectiveness of the proposed scheme. The experimental results showed thatthe sum rule outperforms existing state-of-the-art methods by large margins. This study provides a solution to achieve high performance traffic classification without time-consuming training samples labelling.

IV. REFERENCES

[1] L. Bernaille, R. Teixeira, and K. Salamatian, Early application identification, in Proceedings of the 2nd Conference on Future Networking Technologies (CoNEXT 06), Lisboa, Portugal, December 2006.

[2] L. Stewart, G. Armitage, P. Branch, and S. Zander, An architecture for automated network control of QoS over consumer broadband links, in Proceedings of the IEEE Region 10 International Conference (TENCON 10), November 2005.

[3] Comparison of integrated development environment (ide) debugging tools: eclipse vs net beans. *1 mrs. kavita s., *2ms. sindhu s.,(july2015).

[4] Dr. j vs. the bird: java ides one-on-one* - Michael olan.

[5] Lee, Whats ahead for embedded software?, Computer, vol. 33,pp. 18–26, Sep 2000.

[6] Kölling, Michael and Bruce Quig, Andrew Patterson, John Rosenberg. The BlueJ System and Its Pedagogy, Journal of Computer Science Education, Vol 13, No 4, December 2003.

[7] Stoler, Brian. A Framework for Building Pedagogic Java Programming Environments, Masters Thesis, Rice University, April 2002.

[8] D. Mazinanian, A. Ketkar, N. Tsantalis, and D. Dig. Understanding the use of lambda expressions in java. Proc. ACM Program. Lang., 1(OOPSLA):85:1–85:31, Oct. 2017.

[9] Microsoft. Sal annotations. https://msdn.microsoft.com/en-us/library/ms235402.aspx, 2015.

[10] A. Mockus and L. G. Votta. Identifying reasons for software changes using historic databases. In Proceedings of the International Conference on Software Maintenance (ICSM'00), ICSM '00, pages 120–, Washington, DC, USA, 2000. IEEE Computer Society.

[11] N. Nagappan and T. Ball. Use of relative code churn measures to predict system defect density. In Proceedings of the 27th International Conference on Software Engineering, ICSE '05, pages 284–292, New York, NY, USA, 2005. ACM.

[12] Oracle. Annotation type suppresswarnings. https://docs.oracle.com/javase/7/docs/api/java/lang/SuppressWarnings.html, 2017.

[13] Oracle. Lesson: Annotations. https://docs.oracle.com/javase/tutorial/java/annotations/, 2017.

[14] Oracle. Annotations. https://docs.oracle.com/javase/7/docs/technotes/guides/language/annotations.html, 2018.