EasyChair Preprint

№ 13165

Security System Performance Analysis Analytics Based

Kristine Wau, Wana Yumini and Dedy Hartama

May 1, 2024

Security System Performance Analysis Analytics based

Kristine Wau[1], Wanayumini[2], Dedy Hartama[3]

[1,2,3] *Magister of Computer Science, Potensi Utama University*

JL. KL. Yos Sudarso Km. 6.5 No. 3-A, Medan, Indonesia

[1]kristinewau1@gmail.com

[2]wanayumni@gmail.com

[3]dedyhartama@amiktunasbangsa.ac.id

Abstract **— Information is a very important asset in making decisions. With the development of network technology which leads to network virtualization and the increasing use of IoT, data results in increasingly diverse data. The variety of data becomes a problem in itself in management, monitoring and network security. The emergence of the concept of Big Data Analytics has become one solution in data management, Big data is a collection of data that has a very large volume or size consisting of structured, semi-structured and unstructured data that can develop over time. Security analytics is a combination of tools used to identify, protect and solve security events that threaten IT systems using real-time and historical data. This research uses a big data analytical approach to process network traffic data. This research uses two algorithms, namely Naive Bayes and KNN. This is to compare the two algorithms which have the best level of accuracy. Apart from that, the two algorithms are used to produce information from the network traffic dataset used. The Naïve Bayes algorithm is one of the algorithms used for statistical classification which can be used to predict probability of membership of a class and the KNN algorithm is a supervised learning algorithm that is used to classify new objects based on nearby objects. The aim is to find out the extent of attacks on the network. This research uses a dataset (Network traffic data). Spark was chosen as the big data analytical framework used which is utilized. big data analytics in the performance of normal data network security systems or those indicated in the training process and network traffic classification.**

Keywords: Classification, Security System, Network Traffic, Naive Bayes Algorithm, KNN Algorithm

I. INTRODUCTION

Information security is a very valuable asset for organizations because it is one of the strategic assets for creating business value. Therefore, protecting information security is an absolute problem that requires serious thought at all levels of the organization. Information security includes policies, procedures, processes and activities. aimed at protecting information, the increasing importance of information and data requires a security procedure to safeguard information [1]. An analytical-based security system is an approach to computer security that uses data analysis techniques to detect security attacks and threats. Security analytics involves collecting security data from various sources such as system logs, network data, and other information. This data is then analyzed using algorithms and analytical techniques to identify suspicious patterns and behavior [2]. Monitoring and detecting attacks based on traffic data on computer networks is a complex task. Several problems include: large traffic volumes, transmission system speed, developments in services, developments in types of attacks, and various data sources and methods for obtaining data security systems [3]. In research carried out in the analytical academic field by predicting student performance with datasets obtained from public datasets, where big data analysis is operated using Apache Spark, then to the data grouping process uses the k-mean clustering algorithm, part of the machine learning algorithm [4]. In the field of network security, the use of a big data framework focuses on *the Volume*, *Veracity* and *Variety* characteristics of big data in network traffic and attacks [5]. thus security measures can be taken more quickly to protect the system and improve attack classification through training and classification trials with the aim of improving attack classification using various algorithms and machine learning techniques [6]. In this research, we try to utilize the potential of the Big Data Analytics framework for analyzing network security systems. In research comparing the performance of accuracy, recall, precision and f1-score levels of two Naïve Bayes and K-Nearest Neighbor methods in analytical-based network traffic data classification. The assets used in this research use network traffic from network traffic data. The dataset is a

comprehensive collection of network activity data for studying network infrastructure and traffic.

II. RESEARCH METHODS

In this stage, research was carried out to compare the performance levels of accuracy, recall, precision and f1-score of the two Naïve Bayes and K-Nearest Neighbor methods in analytical-based network traffic data classification . Fig. 1 shows a general research methodology flow diagram.

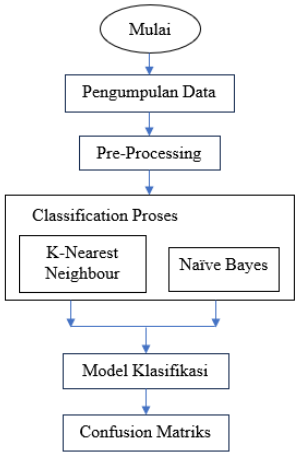


Fig. 1 Stages of Research Methodology

A. *Big data analytics*

Big data analysis identifies trends, patterns and correlations in large amounts of unstructured data for data-based analysis and decision-making processes. This technique uses modern technology to apply techniques such as clustering and regression, data aggregation. big data analytics processes include:

1. Collection data structured And No structured:from various sources, such as network traffic logs, network analysis package applications and application monitoring network.Next data will saved in format storage data.
2. Pre-processing Data:After data collected And stored,data the must sorted in accordance with need process furthermore changing data from unstructured data to structured data accordingly with framework big data.
3. Cleaning Data:cleaning data refers on process For determine data that is inaccurate, incomplete, or unreasonable And Then change or delete data the For improve data quality. k general framework for cleanup data consists from five steps :
   1) Defining And determine type error,
   2) Search And identification example error,
   3) Repair error,
   4) Documenting example error And type error,
   5) Modify procedure entry data For reduce error in Century coming.

B. *Big Data Analytic Frameworks (Spark)*

Apache Spark, is a powerful, scalable, and rapidly distributed hybrid data processing engine, the most active open-source project for Big Data. Spark was developed at UC Berkeley in 2009. Spark provides APIs in Scala, Java, Python, and R. To process very large data, Spark must be fast enough to process large data at once. Therefore, the Spark architecture is available in cluster mode , not on one machine. The results of processes executed by Spark are not written to disk but stored in memory. This all-in-memory capability is a high-performance computing technique for advanced analytics, making Spark 100 times faster than Hadoop. Spark also has an ecosystem of libraries that can be used to machine learning, interactive queries that can have important implications for productivity.Spark has been progressively enriched to provide the current complete ecosystem of library support in Spark shown in Fig. below[8]

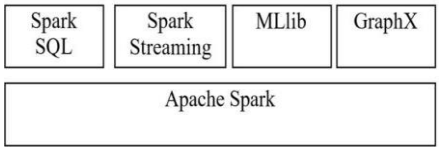


Fig. 2 Ecosystem Apache Spark

C. Data collection

The data used as primary data in this research is internet traffic usage data taken from internet network data. Meanwhile, secondary data in this research comes from literature studies related to the research title. The data that will be obtained is student performance data as part of academic analytical in the application of big data analytics. In this data preprocessing process, internet traffic usage data is taken using Wireshark software. The data taken is then saved in .csv file format. After obtaining internet traffic usage data, the next step is the preprocessing process with Weka software.

D. *Datasets*

At this research stage, the dataset is prepared, then the data is divided into two, namely training data and testing data with a percentage of 60% for training data and 40% for testing data. Training data is used to form patterns or models, while testing data is used to test models that have been built. The model used is classification with the Naïve Bayes classifier and K-Nearest Neighbor algorithms which are then evaluated on the results of the classification performance in writing the accuracy level.

E. *Pre-Processing*

The preparation stage for data processing is to check the data, ensuring that there are no records, empty attributes or the format that Python will process. The Preprocessing stage aims to prepare text documents into data ready to be processed in the next process. The preprocessing stages carried out are:

1. Tokenizing

   Stages for dividing words in a document. Spaces are used as separators between words. At this stage filtering will also be carried out by removing certain characters, such as punctuation marks.

2. Case folding

   Stages for changing all capital letters in a document to lower case. characters other than the letter az will be considered as delimiters.

F. *Process Classification*

The classification phase consists of two processes, namely training and testing the predicted labels. These tasks are implemented using Sklearn, a Python library for data mining, data analysis in machine learning with the K-Nearst Neiqboard and Naïve Bayes methods.

1. Training Process

   The training and classification process of network data is carried out by first classifying the data into attack classes or not (attack data or normal data). Classification classes are then used as additional data features when classifying data into attack categories. Fig. 2 shows the following training steps:

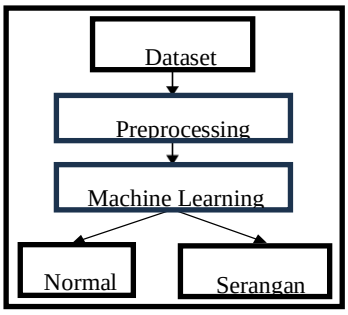


Fig. 3 Training Steps

G. *Classification Models*

After the data has gone through the classification process from the two models, a data testing process is carried out using the Naive Bayes and K-Nearest Neighbor algorithms. In the final stage of the modeling stage, the model is tested with unseen data. The magical data used at this stage is the result of a test set of split data (20%). Testing is carried out to assess how the model represents the data and how well it will perform in the future, then from the classification of each method model the accuracy level results are analyzed.

H. *Confusion Matrix*

At this stage we will test whether the performance of the prediction results is effective so that it can be used as a model recommendation for use. One way to describe the performance of a classification model is the number of instances that are classified correctly and incorrectly. These values are usually represented in a matrix. A matrix is a tabulated visualization of the performance of a supervised learning algorithm. Rows represent the number of instances in the actual class, while columns represent the number of instances in the predictive class. The confusion matrix testing method can produce calculations with 4 outputs, including:

$$Precision = \frac{TP}{TP+FP}\ x\ 100\%$$

$$Recall = \frac{TP}{TP+FN}\ x\ 100\%$$

$$Akurasi = \frac{TP+TN}{TP+TN+FP\_FN}\ x\ 100\%$$

$$Error\ = \frac{FP+FN}{TP+TN+FP+FN}\ x\ 100\%$$

III. Results and Discussion

This chapter explains a series of trials and performance evaluations of research carried out starting from data collection in the form of network activity logs, security attack data or datasets related to analytical-based security systems.

A. *Data analysis*

The data analysis stage consists of data collection and labeling of data obtained from the UWF-ZeekData22 network traffic dataset. This data is processed into student performance data as part of academics analytical in the application of big data analytical.

TABLE I
CLASSIFICATION OF SECURITY SYSTEMS

No	Name	Description
1	Reconnaissance	Attack patterns can include a series of actions or behaviors that are typical of specific attacks such as DDoS attacks, brute force attacks, SQL injection attacks and others.
2	None	This pattern includes a traffic activity that is common and expected in a normally operating network.

B. *Pre-processing*

The preparation stage for data processing is to check the data, ensuring there are no records, empty attributes or the format is appropriate . In this data preprocessing process, network traffic usage data is taken using Wireshark software , the data taken is then saved in .csv file format. After obtaining network traffic usage data, the next step is the preprocessing process with Weka software.

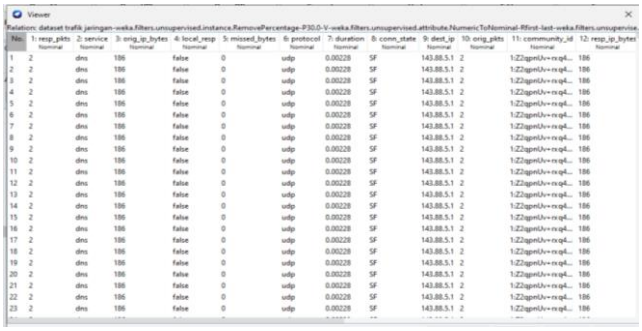
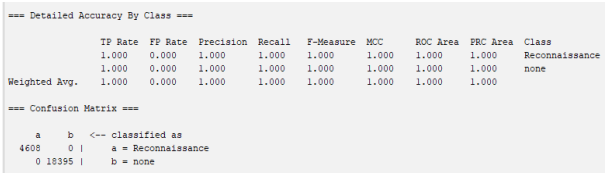


Fig. 4 Network Traffic Result Data processed in WEKA

1) Naive Bayes Classification

After the data has passed data preprocessing, the data is classified using the Naïve Bayes classification method. The results of the dataset using Naïve Bayes can be seen in Fig. the.



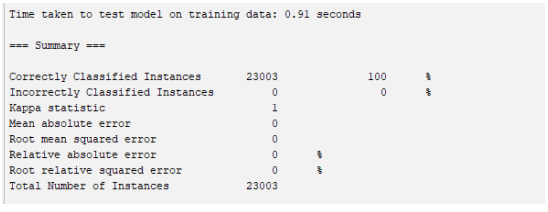
```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Reconnaissance
               1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     none
Weighted Avg.  1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000

=== Confusion Matrix ===

     a     b   <-- classified as
  4608     0 |     a = Reconnaissance
     0 18395 |     b = none
```

Fig. 5 Naïve Bayes Classification

Based on the results from Fig. II can be concluded as follows: -The first line “4608 0” indicates that there is an anomalous level of interference in network traffic and 0 is incorrectly classified as an attack pattern. -The first line “0 18395” indicates the normal interference level.

a. Evaluation

The evaluation results carried out a quantitative comparison by considering the accuracy and error values of the results of the naïve Bayes algorithm classification using network traffic data, shown in Fig. that.



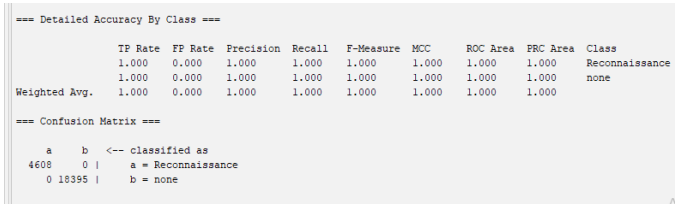
```
Time taken to test model on training data: 0.91 seconds

=== Summary ===

Correctly Classified Instances       23003               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances            23003
```

Fig. 6 Naïve Bayes evaluation results

Based on the results of the table above, it can be concluded that the accuracy level of the Naive Bayes classification results has an accuracy value of 100% and an error value of 0%.

2) K-Nearest Neighbor (KNN) Classification

In data classification using the K-Nearest Neighbor algorithm, the dataset results can be seen in Fig. the



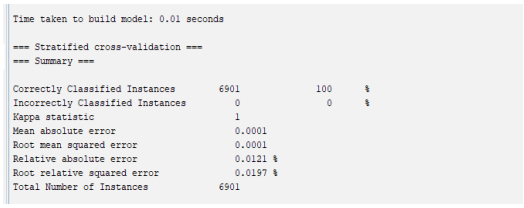
```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Reconnaissance
               1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     none
Weighted Avg.  1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000

=== Confusion Matrix ===

     a     b   <-- classified as
  4608     0 |     a = Reconnaissance
     0 18395 |     b = none
```

Fig. 7 K-Nearest Neighbor Classification

Based on the results from Fig. III can be concluded as follows: -The first line "4608 0" indicates that there is an anomalous level of interference in network traffic and 0 is incorrectly classified as an attack pattern. -The first line “0 18395” indicates that the interference level is normal.

a. Evaluation

The evaluation results carry out quantitative comparisons by considering the accuracy and error values of the KNN algorithm classification results using network traffic data, shown in Fig. that.



```
Time taken to build model: 0.01 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        6901               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0.0001
Root mean squared error                  0.0001
Relative absolute error                  0.0121 %
Root relative squared error              0.0197 %
Total Number of Instances             6901
```

Fig. 8 KNN Evaluation Results

Based on the results of the table above, it can be concluded that the accuracy level of the K-Nearest Nighbour classification results has an accuracy value of 100% and an error value of 0%. However, the relative absolute error is 0.0121%, which shows how big the average absolute error in prediction is and in the root relative squared error, it is 0.0108%, which is the average of the squared differences between the predicted value and the actual value.

C. Analytical Approach

Big data analytics identifies trends, patterns, and correlations in large amounts of unstructured data for data-driven analysis and decision-making processes. Apache Spark, is a powerful, scalable, and rapidly distributed hybrid data processing engine, the most active *open-source project* for Big Data.

IV. CONCLUSION

Based on the test results carried out, the implementation of the security system performance analysis algorithm based on analytics results obtained from system testing produces the highest level of accuracy from the Naive Bayes algorithm test data which is better than the K-Nearest Neighbor algorithm with the accuracy level of the Naive Bayes classification results having an accuracy value of 100% and an error value of 0%. Meanwhile, the accuracy level of the K-Nearest Nighbour classification results has an accuracy value of 100% and an error value of 0%. However, the relative absolute error is 0.0121%, which shows how big the average absolute error in prediction is and in the root relative squared error, it is 0.0108%, which is the average of the squared differences between the predicted value and the actual value. Based on the parameter results precision, recall and F1-Score, naive Bayes tends to be better.

REFERENCE

[1] Casas, P., D'Alconzo, A., Zseby, T., & Mellia, M. (2016). Big-DAMA: Big data analytics for network traffic monitoring and analysis. *LANCOMM 2016 - Proceedings of the 2016 ACM SIGCOMM Workshop on Fostering Latin-American Research in Data Communication Networks, Part of SIGCOMM 2016* , 1–3. https://doi.org/10.1145/2940116.2940117

[2] Gökdemir, A., & Çalhan, A. (2022). Deep learning and machine learning based anomaly detection in internet of things environments. *Journal of the Faculty of Engineering and Architecture of Gazi University , 37* (4), 1945–1956. https://doi.org/10.17341/gazimmfd.962375

[3] Kanakis, M.E., Khalili, R., & Wang, L. (2022). Machine Learning for Computer Systems and Networking: A Survey. *ACM Computing Surveys , 55* (4). https://doi.org/10.1145/3523057

[4] Novianto, E., Herman, E., Ujito, H., Rianto,), Information, MT, Yogyakarta, UT, Ring, J., Utara, R., Lor, J., & Yogyakarta -Indonesia, DI (2023). *Some rights reserved BY-NC-SA 4.0 International License INFORMATION SECURITY IN HUMAN RESOURCE MANAGEMENT INFORMATION SYSTEM APPLICATIONS 1) . 8* (1), 10–15. https://doi.org/10.36341/rabit.vx8i1.2966

[5] Prasetyo Nugroho, F., Wariyanto Abdullah, R., & Wulandari, S. (2019). *BIG DATA SECURITY IN THE DIGITAL ERA IN INDONESIA* (Vol. 5).

[6] Purnomo, R., Priatna, W., & Putra, T.D. (2021). Implementation of Big Data Analytics for Higher Education Using Machine Learning. *Journal of Information and Information Security (JIFORTY) , 2* (1), 77. https://archive.ics.uci.edu

[7] Wang, L., & Jones, R. (2021). Big Data Analytics in Cyber Security: Network Traffic and Attacks. *Journal of Computer Information Systems , 61* (5), 410–417. https://doi.org/10.1080/08874417.2019.1688731

[8] Wang, S., Balarezo, J.F., Kandeepan, S., Al-Hourani, A., Chavez, K.G., & Rubinstein, B. (2021). Machine learning in network anomaly detection: A survey. *IEEE Access , 9* , 152379–152396. https://doi.org/10.1109/ACCESS.2021.3126834

[9] “https : / / s p a r k .a p ache.o r g /.”