IMPLEMENTATION OF A MACHINE LEARNING METHOD WITH THE METROLOGY PRINCIPLES IN A SIMULATED COMPUTER NETWORK

[1]Vladimir Pejanović, [2]Boris Stanojević, [3]Platon Sovilj

Keywords: computer network, performance metrics, metrology, Machine learning, Support Vector Machine model, Cisco Packet Tracer

ABSTRACT

This paper presents an example of a defined and constructed computer network represented by a simulation in the Cisco Packet Tracer software, which virtualizes the transmission of data packets from one host/server to another through network infrastructure such as switches, routers and other supporting computer infrastructure. Using this simulation in Cisco Packet Tracer, we defined the necessary input values in the Machine learning process, such as performance metrics in the transportation of data packets in a computer network system under normal circumstances and extraordinary circumstances, such as network congestion and other issues affecting the functionality and optimization of computer network processes. These performance metrics were then implemented as input values in a Machine learning method, specifically the Support Vector Machine (SVM) model, a binary model that supports the use of a small data set, enabling the prediction of extraordinary events in the simulated computer network. By using the principles of metrology and Machine learning, the prevention in the process of optimizing the functioning of computer networks is improved, potentially avoiding congestion, bottlenecks, and other issues that have so far been mostly resolved curatively, i.e., reactively by network administrators. The field of computer networks is of great importance in the functioning of all computer systems and will become even more significant with the increased connectivity of people and devices on the Internet, the development of the Internet of Things (IoT) systems, ubiquitous computing and corporate Intranet networks.

INTRODUCTION

The use of Machine learning, with a dataset obtained by measuring the performance of a simulated computer network and generating artificial data for the purpose of obtaining sufficient amount of data, can predict or forecast errors, delays, congestion of computer networks or routers. This prediction can be significant as a preventive method in case of frequent problems in the reliability of computer network performance and efficiency. Machine learning is used for signal processing, filtering, processing, and classification od data and output results, as well as learning and training. In this way, Machine learning, in addition to numerous application examples in many different areas, can also be applied in metrology, in this case, the performance metrics of computer networks. The method of Machine learning, the Support Vector Machine method, can be used, i.e. binary model that supports the use of a small data set. Feedback can also be included, so that metrics and problems in the performance of computer networks in real time are automatically used as input values for predicting future situations.

METHODOLOGY

In this paper, example of an average computer network is defined and presented through simulation in Cisco Packet Tracer software. The simulation enables the virtualization of sending packets between two hosts/servers through network infrastructure that includes switches, routers, servers, and other accompanying Local Area Network infrastructure. Using this simulation, we obtained the necessary input values such as performance metrics of packets and signals sent between hosts. The network infrastructure was simulated in Cisco Packet Tracer, and system performance metrics were measured and recorded, both under normal and extraordinary circumstances such as network congestion and other problems affecting the functionality and optimization of the network. These metrics were then used as input values in the Machine learning model, specifically in the Support Vector Machine model, which supporting the use of a small data set allows more accurate prediction compared to other Machine learning models. Also, the quantity of these measured data can be increased based on the set of samples obtained by simulating the computer network. Data samples are extracted using regression analysis, thus obtaining a larger data set. This allows the use of other Machine learning methods, contributing to a greater reliability of predictions as well as variety of output data. In this way, we can be informed about the current and future state of the computer network based on Machine learning predictions, which allows organizing preventive measures for the functionality of computer networks with a certain degree of reliability. This can significantly contribute to solving potential problems for each computer network individually, as well as eliminating latent interventions that cause time loss and reduce the efficiency of business operations and the functionality of the used computer network.

Integrating Machine learning into computer network management not only helps in predicting and identifying potential congestion, but also provides tools to manage and mitigate these issues in real-time. This proactive approach ensures high network performance, minimizes downtime, and maintains a smooth experience for all users. By leveraging Machine learning, network administrators can transform reactive network troubleshooting into a proactive, performance-enhancing strategy.

Key steps in this process are:

- Extracting data for SVM input values, i.e. collecting network performance data from simulations in Cisco Packet Tracer. Collecting data on traffic contribution by percentage, counter bytes, counter packets, and time differences from the simulations. Recording these metrics at various intervals and under different network conditions to create a comprehensive dataset.
- Analyzing and formating the data. Once the data has been collected, anppropriate formatting is of utmost importance for use with an SVM model. This typically involves normalizing the data and possibly converting categorical data into a format suitable for Machine learning (e.g., one-hot encoding).

- Preparing for Machine learning, i.e. preparing the data for training and testing in the Machine learning model. With the data prepared, it can be implemented into an SVM model to train it on recognizing patterns associated with network congestion. Trained SVM model can be used for predicting network efficiency based on input parameters. Specifically, the trained model can be used to predict congestion based on live data or simulated scenarios to test the model's effectiveness.

By following these steps, researcher can effectively use Cisco Packet Tracer to generate and gather data for training an SVM to predict network congestion. This approach allows you to simulate various network conditions and gather detailed metrics that are crucial for evaluating the accuracy and performance of the trained model.

Analyzing factors that affect network efficiency, i.e. the performance metrics of the simulated computer network, the obtained input values, and Machine learning (SVM) model, were implemented to predict network efficiency.

Key factors identified as influencing network performance include [1]:

- Bandwidth utilization
- Delay in packet transmission
- Number of transmitted packets

This Machine learning-based approach enables automated monitoring and optimization of computer networks in any area, identifying bottlenecks and measures to improve efficiency.

For the classification of predictions based on simulated computer network performance data, we used the Support Vector Machine, a powerful supervised learning algorithm and classifier. This paper describes the process of data collection, preprocessing, feature extraction techniques, basic principles of the SVM model, and metrics for evaluation and classification.

The Machine learning technique used for this purpose is the Support Vector Machine method. The research aims to assess the accuracy and efficiency of the SVM model in predicting levels of performance metrics based on simulated and augmented input data. To evaluate the performance of the model, several metrics are used, including Accuracy, Precision, Recall (sensitivity) and F1 Score.

PRACTICAL EXAMPLE OF COMPUTER NETWORK APPLICATION

The subject of observation in this research is a hypothetical new, small company on the west side of Chicago, which currently has about 35 to 40 employees, each with their workstation, i.e. computer. Employees use computers for typical administrative office functions such as [2]:

- Word processing
- Shared access to high-quality laser printers
- Access to a centralized database
- Internet access.

The company owner concludes that the best way to offer these services is by installing a local area network. Since all the workstations are only one to two years old, they do not need to be replaced, and all of them contain a Network Interface Card (NIC) for a LAN (Local Area Network) [2]. However, the computer network will also require UTP cables, a server, and software to connect the workstations to the local network. The owner could consider funding the following list of alternative local network systems [2]:

- 100 Mbps Ethernet
- 1000 Mbps Ethernet
- Wireless Ethernet

The 100 Mbps Ethernet seems like the best choice in this case. It is the most popular local network system and, therefore, has some of the lowest and affordable prices [2].

Based on these considerations a simulation of the average office computer network was defined, created and ran in Cisco Packet Tracer software for the purpose of this research.

RESEARCH RESULTS

Regarding the results of the research, with only one feature implemented, performance metrics are not of satisfactory level, primarily accuracy which is 44 percent.

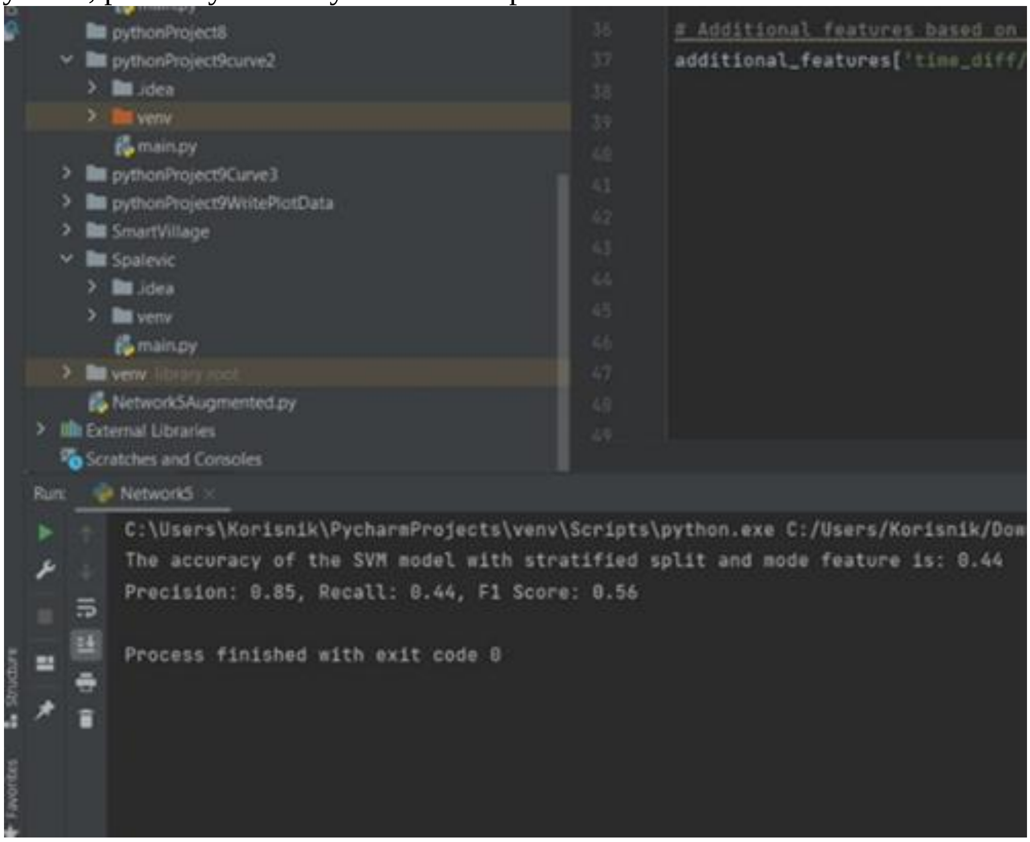

Figure 1. Results of the research with one feature implemented

With several applied features, the accuracy as performance metric of the SVM model was 91 percent, and with the augmentation of a small dataset (around 150-160 data points measured in the simulated network) to 157,000 data points, accuracy, as well as other performance metrics, increased to 99 percent.

Description of Python code for Machine learning prediction process involves using the SVM (Support Vector Machine) to predict network load. The developed code works as follows:

- Data loading and processing – Data is loaded from an Excel file using the pandas library. After loading, additional statistical features such as minimum, 75th percentile, interquartile range, skewness, kurtosis, and other features based on combinations of specific attributes are added. The additional features are then merged with the original dataset.
- Data preparation – Input features and the output target variable are separated. The output target variable is encoded using LabelEncoder, and then the dataset is stratified into training and test sets.
- Class balancing – To address the imbalanced class problem, Random Oversampling is applied to the training set.
- Input feature scaling – Input features are scaled using StandardScaler.
- SVM model creation and training – An SVM model with an RBF kernel is created and trained on the pre-prepared training set.
- Model Prediction and Evaluation – The trained model is used to predict values on the test set.

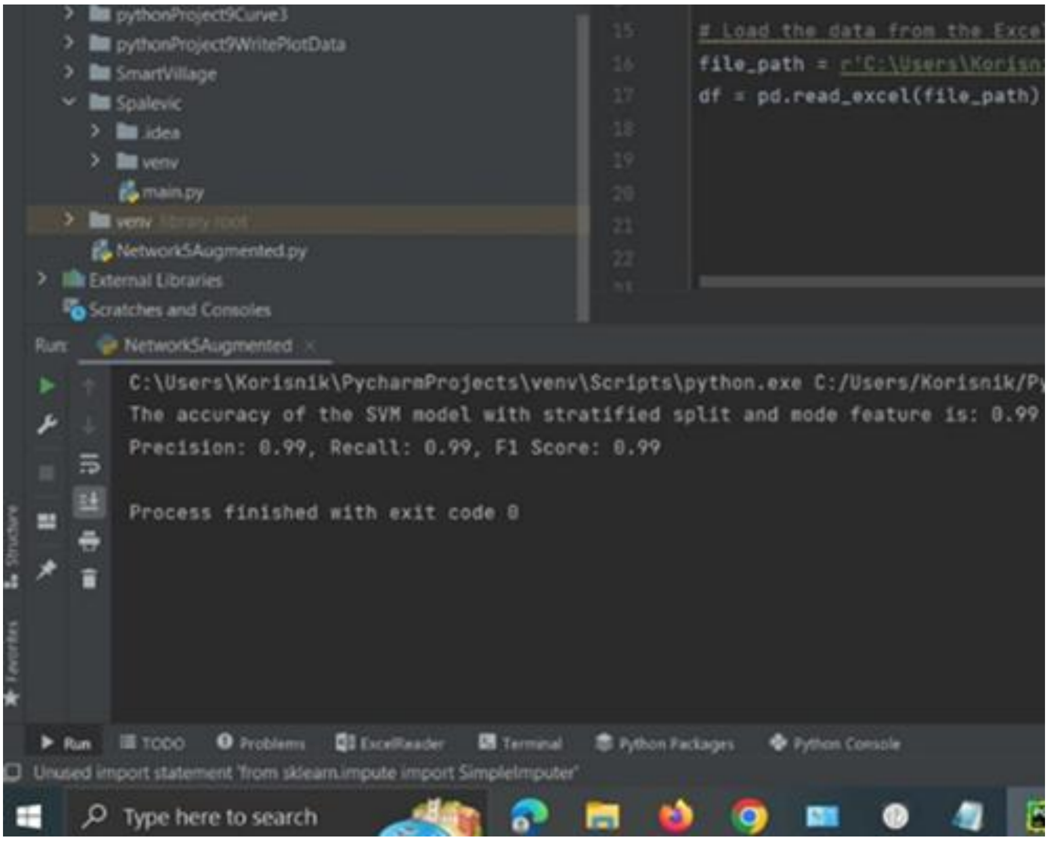

Figure 2. Results of the research with several features and data augmentation implemented

Model evaluation metrics such as accuracy, precision, recall, and F1 score are then calculated and printed.

PERFORMANCE EVALUATION METRICS OF SVM MODEL

The performance of the SVM model was evaluated using specific classification performance metrics:

Accuracy – The ratio of true positive predictions to the total number of positive predictions, measuring the model's ability to avoid false-positive predictions.

Precision – Assesses the quality of positive predictions made by the model. It is defined as the proportion of true positive predictions relative to the total number of positive predictions made (including false positives).

Recall – The ratio of true positive predictions to the total number of actual positive cases, assessing the model's ability to detect positive cases.

F1 Score – Provides a balance between precision and recall by taking their harmonic mean. It is especially useful when you need a single metric to reflect the model's performance with respect to both false positives and false negatives.

CONCLUSION

By utilizing Machine learning with a dataset obtained from measuring the performance of a simulated computer network and generating augmented data based on a small dataset for the improvement of performance metrics of Support Vector Machine model (Machine learning model), predictions or forecasts of errors, delays, and congestion in computer networks or routers can be made. These predictions can be of great significance as a method of prevention in the case of frequent problems with the reliability of computer network performance, as well as in the case of ensuring more efficient functioning of computer network systems. Machine learning is used for data processing, which by filtering, processing, and classification, as well as learning and training of the model, necessary predictions, output data are obtained. In this way, Machine learning, in addition to numerous examples in other areas, can also be applied in metrology, in this case, in assessing performance metrics of simulated computer network, and thus can be an integral part of the servers or cloud computing connected to the computer network. These predictions do not necessarily have to be based on a large amount of data, because the Machine learning method, the so-called Support Vector Machine model (which does not require a large amount of data) can be used. Feedback can also be included to automate current metrics in the performance of key metrology indicators of the computer networks in real-time and use them as input values for predicting such future situations [3].

Potential applications of this models of Machine learning are:

- Real-time congestion prediction and management using Machine learning model

Network administrator can predict congestion in real-time using Machine learning model, and consequently take several proactive steps to manage and mitigate potential network issues.

- Dynamic adjustment of network parameters

Machine learning algorithms can analyze traffic patterns and predict potential congestion before it becomes problematic. With these predictions, network parameters such as Quality of Service policies, bandwidth allocations, and priority settings can be dynamically adjusted. For instance, critical traffic can be prioritized over less urgent data, ensuring that important applications run without interruptions.

- Traffic shaping and load balancing

Based on the predictions made by Machine learning models no single part of the network is going to be overwhelmed. This is called traffic shaping and it can be implemented to control the data flow across the network. Besides that, load balancing can be adjusted in real-time to distribute traffic evenly across multiple servers or network paths, thus avoiding hotspots that lead to congestion.

- Adaptive congestion control protocols

Traditional TCP congestion control mechanisms like slow start, congestion avoidance, and fast recovery can be enhanced with Machine learning insights. By understanding current and predicted traffic conditions and managing data flow these protocols can be adapted more effectively to prevent congestion from occurring. For example, the congestion window adjustments can be optimized based on predicted traffic conditions, allowing for more efficient data transmissions.

- Preventive retransmission strategies

By predicting packet loss or identifying trends that lead to increased error rates, the network can preemptively and in real-time adjust its retransmission tactics, reducing unnecessary traffic that could potentially lead to further congestion.

- Real-time network reconfiguration

In some cases, the network configuration itself might need to be changed based on Machine learning predictions. This could involve activating additional pathways, switching traffic to underutilized network segments, or even temporarily enhancing bandwidth capacity in anticipation of high traffic volumes.

LITERATURE

[1] L. L. Peterson, B. S. Davie, Computer Networks, Morgan Kaufmann, 2021.
[2] C. M. White, Data Communications & Computer Networks, Cengage Learning, 2016.
[3] L. Cocco, Modern Metrology Concerns, IntechOpen, 2012.

[1] Faculty of technical sciences, University of Novi Sad
[2] Evropski Univerzitet, Belgrade, Republic of Serbia
[3] Faculty of technical sciences, University of Novi Sad