TOWARDS ZERO DOWNTIME: USING MACHINE LEARNING TO PREDICT NETWORK FAILURE IN 5G AND BEYOND

Emmanuel Basikolo[1], Thomas Basikolo[2]
[1]Department Physics and Electronics, School of Natural and Applied Sciences, University of Malawi, Malawi,
[2]Telecommunication Standardization Bureau (TSB), International Telecommunication Union, Switzerland

NOTE: Corresponding author: Emmanuel Basikolo, bsc-phy-22-19@unima.ac.mw

Abstract – *A stable network is very important to both network providers and their customers, as it increases reliability, improves security and helps customers and companies save costs. When network outages occur, they result in significant downtime and financial losses for organizations and network users. Traditional methods of detecting and troubleshooting network failures are often reactive and time-consuming, whereby network administrators rely on traditional methods such as reactive monitoring and manual troubleshooting. These methods are often not effective in detecting and preventing network failures. In this paper, we propose a machine learning-based approach to predict network failures and minimize downtime. Network performance observability data from a 5G core network testbed based on Cloud-native Network Functions (CNFs) is used to train several supervised learning models, including random forest, gradient boosting regressor, conventional support vector regressor and proposed support vector regressor, to predict network failures. Our experiments and analysis show that the proposed model Support Vector Regressor (SVR) produced better results as compared to other models. In a very short amount of time (ten seconds), the proposed SVR model is capable of predicting whether a network failure event will occur or not within the next ten minutes, with an f1-score of more than 0.9. Our results indicate that machine learning-based approaches can significantly enhance the detection and prediction of network failures, leading to zero downtime and improved network performance.*

Keywords – Machine learning, network outage/failure, packet logs, regression models, support vector regressor

1. INTRODUCTION

The fifth generation of telecommunication systems, or 5G, is a global wireless standard that offers higher data capacity and transmission speeds [1, 2, 3, 4]. 5G services are essential for a wide range of innovative applications which have the potential to transform many sectors of our economies and improve citizens’ daily lives [4]. The deployment of 5G networks began in 2019 and proliferated in 2021 and 2022 with network rollouts accelerated [5]. As the technology underpinning 5G has become more developed, there has also been a realization of the complexity of 5G. 5G specifications have grown in richness and capabilities [6]. Having such a diverse "toolbox" of capabilities is a great benefit to service providers; it enables the service providers to customize and optimize their networks tailored to customers' needs and to add new and innovative services [6]. However, this richness of capabilities could also present some difficulties. Companies developing 5G products and services now have a much broader spectrum of features they need to develop [6]. The breadth of features may lead to a slower rollout of 5G features and sometimes results in network failures [7].

Modern businesses and social activities rely significantly on communication networks, making their uptime vital. Network failures can cause significant downtime and financial losses [8]. The negative effects of network outages on enterprises and organizations are numerous. A single disruption in network infrastructure can devastate a revenue stream, and even numerous minor disruptions can have a large negative impact on corporate operations [9]. Some of the negative impacts of network outages include: (a) decreased productivity, since employees are unable to access the necessary applications and tools leading to decreased productivity and missed deadlines [10]; (b) revenue loss especially for businesses that rely heavily on e-commerce or online transactions [11, 12]; (c) damage to reputation when a company faces frequent network downtime which damages a business's reputation and erodes customer trust [11, 13]. This can lead to loss of customers and difficulty in acquiring new ones; and (d) legal and regulatory compliance issues, since in most countries network downtime can result in non-compliance with legal and regulatory requirements, leading to fines and other penalties [5].

Overall, network downtime can have a significant impact on a business's bottom line, reputation and ability to operate effectively. As a result, preventing network downtime through proactive measures such as network failure prediction using Machine Learning (ML) techniques can be critical to maintaining network availability and avoiding these negative impacts. Currently, traditional methods of detecting and troubleshooting network failures are often reactive, meaning network administrators respond to network failures after they occur, resulting in extended downtime and reduced productivity [13, 14]. With the increasing complexity of modern networks and the growing demand for network availability, there is a need for more proactive approaches to detect and predict network failures.

Most of the time, telecommunication companies take an action only when they are certain that network failure is going to occur. In recent years, Machine Learning (ML) techniques have gained much attention as an effective means/tool to predict network failures. Machine learning algorithms can automatically learn patterns from network data, identify potential failure points, and provide predictions before failures occur, allowing network administrators to take preventive measures. In this paper, we propose a machine learning-based approach to predict network failures and minimize downtime.

Using 5G core network packet logs from a 5G testbed, performance of several supervised machine learning techniques was analysed. This paper proposes a modified Support Vector Regression (SVR) approach that improves the traditional SVR. This improved SVR method is called "*proposed-SVR*" in this work. The *proposed-SVR* model therefore achieves the lowest detection time *(t)* on a network failure event with a higher accuracy and *f1-score*.

Using machine learning models to predict when a network failure event will occur, network engineers can prevent network downtime from happening. By tracking how packets are being transferred in the network, predictions can be made as to whether a network failure event will occur or not.

The rest of the paper is structured as follows. In Section 2 we provide an in-depth examination of preliminary and related research work on network outages and investigate the application of machine learning techniques for predicting network failures. Section 3 provides an overview of the dataset employed in this study. Section 4 discusses the methodology applied to solve the problem addressed in this work including the *proposed-SVR* model and the two tasks. Section 5 delves into the experiments conducted in this study, while Section 6 focuses on performance evaluation, discussing the results of various machine learning models and compares them with the proposed method. The last section (Section 7) concludes the paper with some remarks and future direction of the work.

2. PRELIMINARIES AND RELATED WORK

2.1 Network outage

Network outage/downtime refers to an unscheduled period in which a service is interrupted and not usable [15]. The disruption or loss of connectivity within a telecommunication network can cause service disruption for users who rely on the network for communication or data transmission. This inaccessibility may be to a part of, or all of, a network, that can be caused due to failure of hardware (equipment failures including failures of routers, switches, servers or other critical components of the network), software (bugs or glitches) or a combination of the two [16]. Scheduled maintenance, including upgrades or repairs, has the potential to cause network outages if not executed correctly or if unexpected issues arise. Power outages, natural disasters or accidental damage to infrastructure can also cause network outages [16]. When a network outage occurs, it can affect various services, such as Internet access, phone calls, video conferencing and other forms of data communication.

In recent years, several network failure incidences have been reported, which have affected many businesses and individuals. In July 2022, nearly 40 million people were left without mobile phone services in Japan after equipment failure caused a network outage which lasted over 80 hours at the telecommunication company KDDI Corporation [17]. In the same month, a massive network outage at Rogers, a Canadian telecommunication operator, forced more than 10 million customers off their Internet or wireless services for 19 hours [18]. The outage was reported to have been caused by a maintenance upgrade that caused routers to malfunction [19].

In December 2020, a major outage affected O2, the UK's largest mobile network operator. The outage was caused by a software issue and affected voice and data services for several hours [20]. In February 2020, a major outage affected Vodafone Idea, one of India's largest telecommunication operators. The outage was caused by a server issue and affected voice and data services for millions of users [21]. The examples of network outage above not only disrupted users' communications but also affected emergency services, business activities and several other economic activities.

With 5G technology and the increasing complexity of network architectures, the potential for network outages has become even more significant [22]. Telecommunication companies have to work quickly to resolve network outages, as they can cause significant disruptions to businesses and individuals who depend on their services. It is essential for telecommunication operators to have measures in place to monitor the network and detect outages early, as well as protocols for responding to and resolving outages as quickly as possible. It is also important for network operators to have contingency plans in place to mitigate the effects of network outages and minimize downtime for their customers. As a result, there is a growing interest in predicting and mitigating network outages using advanced techniques such as machine learning [22].

2.2 Machine learning models

Within the last decade, machine learning has gained significant interest and seen significant growth with applications in various fields including communication networks [23]. With the evolution of 5G, there has been an increase in the number of devices and amount of data generated, which creates the need for the efficient and intelligent management of network resources. ML algorithms can help network operators to automate decision-making, optimize network performance and enhance user experience [24].

Several studies have applied machine learning to 5G networks, addressing various challenges in network management, optimization and security. In [25] a machine learning-based approach for network slicing in 5G networks is proposed. The approach utilizes ML algorithms to identify and group similar network slices, enabling the efficient management of network resources and enhanced user experience. A study by [26] proposed a traffic prediction approach for 5G networks by utilizing neural networks to predict network traffic patterns, thereby enabling proactive optimization of network resources.

[27] proposed the use of a multilayer feed forward neural network, also known as a Multilayer Perceptron (MLP), for modelling the network outage time of network elements or systems. An MLP network was trained on 150 samples of daily network outage time data obtained from the network operating centre. The model input and output layers were kept constant, while the number of neurons was varied from 1 to 20, resulting in a prediction accuracy of 97.5%. A study in [28] demonstrated that metrics collected by monitoring tools could indicate system failure, providing maintainers with an opportunity to predict system failures from these metrics. However, the sheer volume of collected metrics can obscure relevant hints that may be hidden within combinations of multiple metrics. By employing an SVM algorithm, 15% of failures were predicted 60 minutes in advance.

Overall, the usage of ML in communication networks has been very successful in the last decade. ML has helped to improve network performance, enhance user experience, reduce costs and improve network security.

2.3 Predicting network outage using ML

The studies discussed in Section 2.2 highlight the significant potential of Machine Learning (ML) in addressing multiple challenges in 5G networks. Among these challenges, predicting network outages stands out as an area that can benefit greatly from ML advancements. By leveraging historical data and analysing patterns and anomalies, ML techniques enable the proactive prediction of network outages or failures. This approach empowers network operators to adopt proactive network management strategies, mitigating costly downtime and ensuring uninterrupted service for their customers.

In the past few years, the use of machine learning in predicting network failures has become increasingly popular in the telecommunications industry [29]. There has been some research on the application of ML to network failure prediction. In [30], a Recurrent Neural Network (RNN) model was trained on time-series data of network traffic and performance metrics to predict network outages. In a study by [31] a Deep Neural Network (DNN) was used to predict network outages in a cloud data centre. The DNN model was trained on a large dataset of network logs and performance metrics to predict network outages. Another study [32] used a gradient boosting algorithm to predict network outages in a mobile network. Performance data from base stations were used to identify patterns and anomalies that could indicate an impending outage. In [33], big data analytics were employed to reduce network downtime by predicting and rectifying equipment failures. The prediction utilized system activity and operational parameter data, resulting in a large amount of data that required a dynamic and adaptive algorithm. It was discovered that equipment failure symptoms manifested approximately nine days prior to failure, and predictions made within four days before failure achieved an accuracy of up to 99.9%.

The studies above demonstrate the potential of machine learning in predicting network outages, mobile network management, and improving network reliability. However, the datasets used in most of these studies included either datacentre data only or base station data. In this study, a multivariant time-series dataset from a real 5G core network testbed (CNF-based 5G core test environment) is used. This brings us closer to using ML and predicting network outages for a real commercial 5G network.

By using machine learning to predict network outages, telecommunication network operators can proactively address issues before they become critical, reduce downtime and improve the overall reliability of their networks. As network data becomes more complex and dynamic, machine learning is likely to become an increasingly important tool for network management and optimization.

2.4 Regression models

Regression is a type of supervised machine learning that aims to predict a continuous target variable based on one or more input variables [34]. There are several types of regression models, including linear regression, polynomial regression, logistic regression, and others. The dataset provided for this work contained multivariant time-series metrics which are well suited to regression ML models. In this work therefore, regression models were used to predict the likelihood or probability of a network failure event based on historical data and features related to network performance. The ML model considered various factors (features) that may contribute to network failure, such as network traffic, packet loss, latency and device failures.

In our study, a performance comparison of several regression ML models was conducted including: (a) random forest regressor, (b) Support Vector Regression (SVR), (c) Bayesian ridge, (d) cat boost regressor, and (e) extreme Gradient Boosting Regressor (XGBR).

An XGBR is a type of gradient boosting algorithm that is known for its high performance and accuracy in predictive modelling. It is based on the decision tree algorithm and uses an ensemble of decision trees to make predictions [35]. The XGBR is particularly useful for handling complex and non-linear relationships between features and target variables.

A Support Vector Regressor (SVR) is a type of Support Vector Machine (SVM) algorithm that is commonly used for regression tasks [36]. It works by finding a hyperplane in a high-dimensional space that maximizes the margin between the training data points and the predicted values. The SVR algorithm is particularly useful for handling data with non-linear relationships between features and target variables [36]. This paper proposes a modified Support Vector Regression (SVR) approach that improves on the traditional SVR by modifying the regularization parameter C and substituting a sigmoid kernel for the radial basis function kernel. This improved SVR method is called "*proposed-SVR*" in this work. The *Proposed-SVR* model therefore achieves a lower detection time (t) on a network failure event with a higher accuracy and *f1-score*.

3. DATASET

3.1 Dataset description

The dataset provided is from a commercial network testbed [37]. The target network is a CNF 5G core running on top of Kubernetes with an extended Berkeley Packet Filter (eBPF) [37]. The eBPF enables derivation of fine-grained metrics from Linux kernel space, therefore providing observability. Although this is the case, network failure prediction is still a challenge by using only eBPF without further processing or analysis. Thus, machine learning plays a significant role in predicting future network failures on the 5G core. The dataset provided contains 3 types of metrics: (i) basic metric obtained by cAdvisor [37], (ii) fine-grained metric by eBPF, and (iii) 5G metric by counting 5G logs. Fig. 1 shows the network environment for the dataset generation and associated metrics.

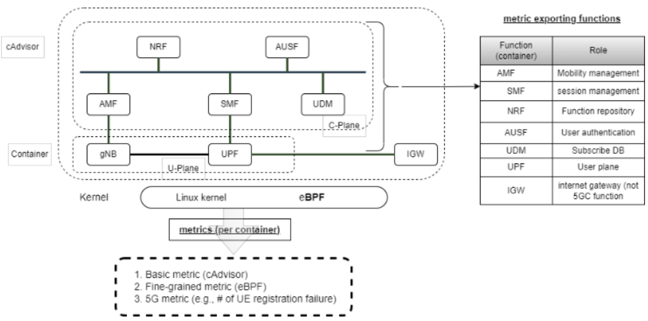

Fig. 1 – Network environment for data collection and associated metrics

Multivariant time-series metrics for each 5G core CNF under normal and failure conditions provided were generated in the CNF-based 5G core test environment. Failure injector caused packet loss on the 5G core, extending the loss rate. The dataset consists of training data (600 test cycles) and test data (300 test cycles), both of which include labels and time-series metrics. The labels include normal (normal state, no network failures) and abnormal (*br-cp_bridge-loss-congestion-with-time-start*, network failure occurs).

During data collection in the testbed, the first 90 seconds was used as a preparation phase (5G network initialization) and the remaining 600 seconds were used for the UE registration and deregistration phase as shown in Fig. 2. During a network failure event, the packet loss increases linearly while users are registered and deregistered continuously. If no network failure event occurs, there is no significant packet loss [37]. This work studies how early and accurately future network failures can be predicted using machine learning and these metrics.

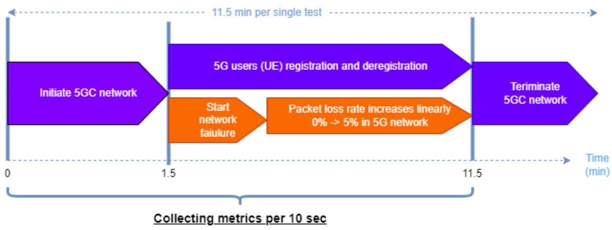

Fig. 2 – Single test scenario

4. METHODOLOGY

4.1 ML tasks

In order to accurately predict network failures as well as reduce the model size and number of features, the work is divided into two tasks.

4.1.1 Task 1

In this task, every metric provided in the dataset is used to train and test ML models for predicting a network failure which will occur in the future. The target value for prediction is the number of UE registration failures (*amf.amf.app.five-g.RM.RegInit Fail*) at 10 min (600 sec). This is the number of failed connection requests from 5G users and is directly related to the QoE of 5G services.

The goal is to find the lowest detection time (t) when the models can achieve over 0.9 of *F1-score* in detection by threshold. This implies that if a predicted value by an ML model is greater than a threshold, then a network failure event will occur at 600 seconds.

4.1.2 Task 2

The second part (task 2), a part of the metrics is selected to develop models that can predict failures at the detection time *t.* This ensures a few metrics are used to train the ML models which significantly reduces the training time, reduces the ML model size, increases the model inference time, leading to lower energy consumptions for ML model training and inference, as well as a higher speed on inference for real-time inference. In this task, a better ML model should achieve a lower detection time *t,* have a lower number of metrics used for training, as well as higher performance (*F1-score*).

4.2 Proposed solution

In this work, packet logs generated at a particular time from the dataset are selected. The selected data is used to train a regression model. In this work, data balancing was performed. A dataset is said to be imbalanced when there is an unequal distribution of classes [38]. This implies that one class has significantly more examples compared to other classes. An imbalanced dataset is challenging for ML training because the class with less examples affects the overall model performance. The target variable of the training set contains two classes (normal and abnormal) of which 450 data points belongs to the normal class and 150 data points to the abnormal class. In this work, we utilized the Synthetic Minority Oversampling Technique (SMOTE) to sample the abnormal class by generating synthetic examples [38]. In order to improve the model's performance and reduce overfitting, a subset of relevant features was selected from the dataset based on its importance. In feature importance, contribution of each feature to final *f1-score* is evaluated. An extreme gradient boosting classifier library is used in this work to determine each feature's contribution to the final *f1-score* thus determining the feature importance of each metric.

In Fig. 3, some of the most important features and their associated *f1-score* contribution for network failure prediction are shown.

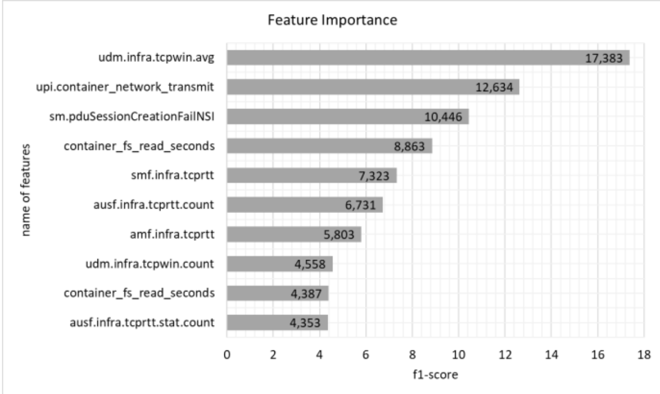

Fig. 3 – Feature importance ranking for network failure prediction

When features have different scales, machine learning models tend to be biased towards features with higher magnitudes and it is difficult to reach a minimum cost function [38]. This reduces the performance of the machine learning model. It is very important to transform numeric values of the features to a similar range.

There are two common methods for feature scaling which are:

(a) Normalization: This method scales the features between 0 and 1.

(b) Standardization: This is s scales feature to have a mean of 0 and standard deviation of 1.

This work utilized the standardization method to improve the final *f1-score* of ML models.

In order to improve on the performance of *proposed-SVR*, the kernel was changed from a Radial Basis Function (RBF) to sigmoid. The sigmoid kernel works better with binary classification and this solution classifies obtained results further into two groups (network failure event occur or not) and hence a binary classification. The modification improves the resulting *f1-score* of the ML model. Furthermore, the reciprocal of regularization parameter C was changed from 1 to 0.09 which increases the regularization parameter, thereby preventing overfitting.

5. EXPERIMENTS

5.1 Machine learning modelling

In this work, regression machine learning models were trained from scratch then predictions obtained from the trained models. The *f1-score* metric was used to evaluate model performance. For the two tasks mentioned in Section 4.1, the same ML models are used to make predictions and performance comparisons. The model feature importance discussed in Section 4.2 is utilized for task 2.

5.2 Thresholds

Regression models return continuous values as the prediction. For a given set of continuous values, there exists a boundary which separates the values into two groups. Any number which is above this boundary represents network failure and the remaining numbers represent a normal network. This boundary is also known as a threshold, and it is used to convert continuous values into binary values.

The occurrence of network failure significantly influences the values of the target variable. This dependency is demonstrated in Fig. 4, where it is observed that the target variable takes similar values for all conditions below a detection time of 60 seconds. This could be attributed to a lower number of lost packets for durations less than 60 seconds. However, for detection times greater than 60 seconds the target variable is observed to take different values, indicating a higher degree of network failure. Based on the observations from Fig. 4, it is evident that the *f1-score* for points with similar values for the target variable will generally be poor. Therefore, a thorough analysis of the target variable values and their dependence on network failure is essential in improving the performance of the *f1-score*. Fig. 4 also illustrates an example of an abnormal (network failure) dataset used to train the regression ML models.

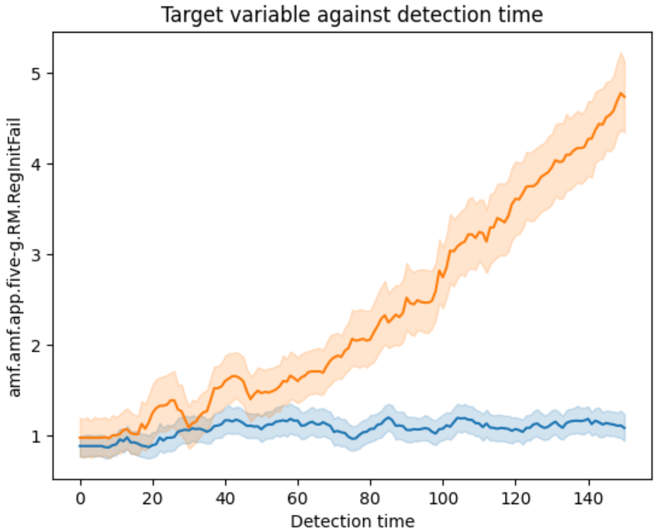

Fig. 4 – Separation between normal and br-cp_bridge-loss-congestion-with-time-start (abnormal) classes

6. PERFOMANCE EVALUATION AND DISCUSSION

6.1 Task 1 results

In the first task, all features (3,325) from the provided dataset were used to train the ML models and perform predictions. In this task (Task 1) no feature selection was performed. In general, using all features yield the best ML model performance results but requires more computational power.

Performance comparison results of different ML models for different detection times (5 – 30 seconds) are shown in Fig. 5 and Fig. 6. In these figures, it is observed that the *proposed-SVR* model obtains the best *f1-score* of more than 0.9 for a detection of a time of 10 seconds. The same observation is made in Fig. 6 in which the *proposed-SVR* accuracy is better when the detection time is 10 seconds or more.

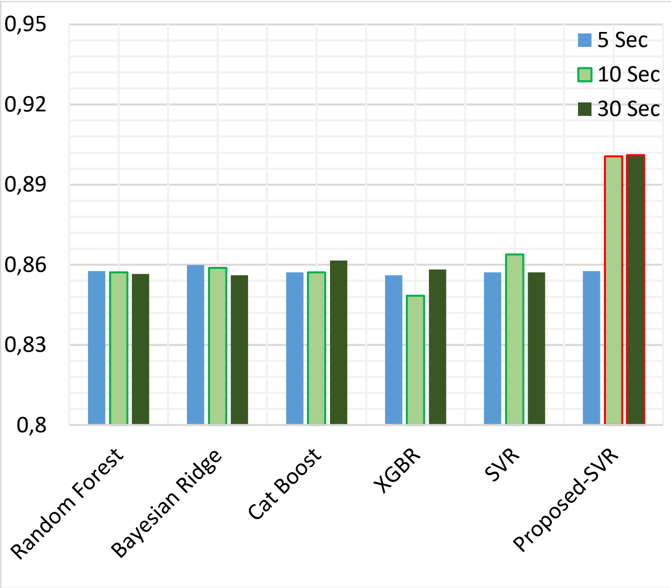

Fig. 5 – *F1-score* vs detection time for different ML models

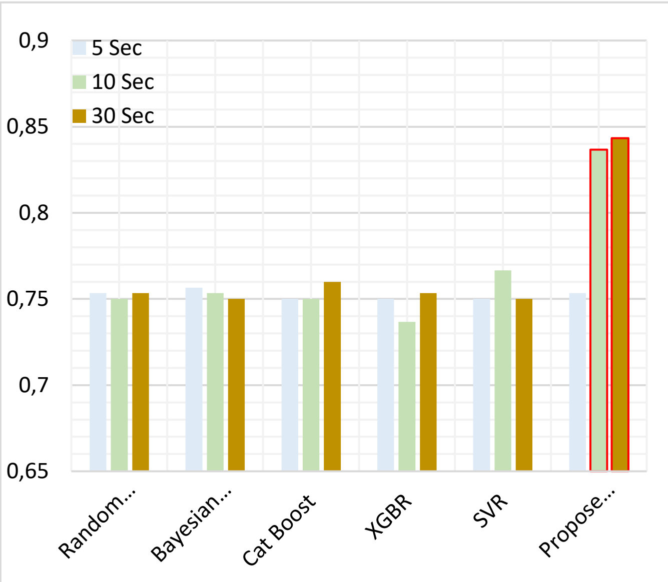

Fig. 6 – ML model accuracy vs detection time for different ML models

To comprehensively evaluate the performance of *proposed-SVR* model in comparison to other ML models, three metrics were employed. These metrics, namely (i) accuracy, (ii) precision, and (iii) *f1-score*, were utilized to assess the performance of these different ML models for a detection time of 10 seconds. The comparison in Fig. 7 reveals that, in general, the *proposed-SVR model* outperforms the other ML models in network failure prediction including the traditional SVR model. Specifically, the *proposed-SVR* model achieves a higher *f1-score* of over 0.9, indicating superior performance in detecting network failure events within a 10-second timeframe. Moreover, the *proposed-SVR* achieves an accuracy of 0.84, surpassing the closest ML models, the traditional SVR model with an accuracy of 0.77 and Bayesian ridge, which achieves an accuracy of 0.75.

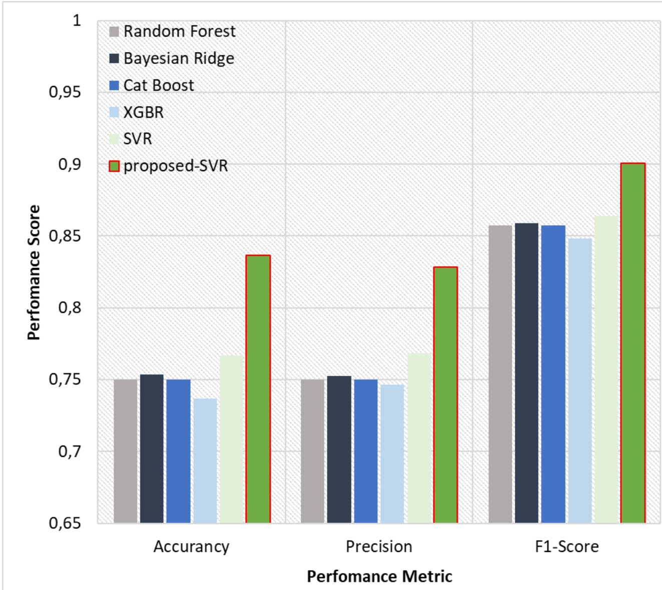

Fig. 7 – Performance comparison of different ML models when detection time is 10 seconds

Fig. 8 shows the performance comparison of the *proposed-SVR* for *f1-score* as a function of detection time. If the detection is increased, the performance of the ML model is improved. This observation also applies to the accuracy of the SVR model. From the results shown in Fig. 8, it can be generalized that the *f1-score* of the model increases with an increase in detection time. This implies that we can be more certain about the prediction of the *proposed-SVR* model as detection time increases.

6.2 Task 2 results

In the second task (Task 2) a number of features were chosen based on their importance to train the ML models, which have been analysed in Task 1 above (Section 6.1). The XGB classifier and feature engine libraries were used to rank features according to their importance. Table 1 shows the SVR model performance as the number of features is reduced; the SVR model produce better results for Task 2 as shown in Table 1.

An analysis of results presented in Table 1 reveals a clear relationship between the number of features and detection time needed to attain a comparable *f1-score*. A significant inverse relationship is observed, indicating that as the number of features increases the required detection time decreases while maintaining a similar *f1-score*.

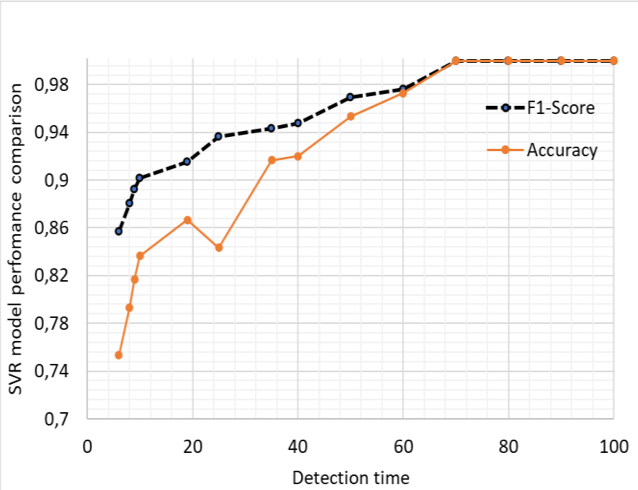

Fig. 8 – F1-score and accuracy performance comparison of *proposed-SVR* against detection time

Table 1 – ML model performance results of *proposed-SVR* for task 2

# of feature	Time	F1-Score	Threshold
All	10	0,901	0.40
527	20	0.911	0.57
400	35	0.906	0.55
300	36	0.917	0.62
200	36	0.908	1.11
120	40	0.902	1.30
100	52	0.914	1.14
50	61	0.901	1.74

From the above data and plots it is clear that there is a relationship between the number of features and *f1-score,* which is not linear. This implies that, some features contribute more to the final *f1-score* compared to other features. Additionally, there is a relationship between the *f1-score* and detection time. Building on the previously discussed correlations it can be confirmed that the number of features and detection time have a nonlinear inverse relationship. This link is depicted in Fig. 9, which shows the indirect relationship between the number of characteristics and detection time.

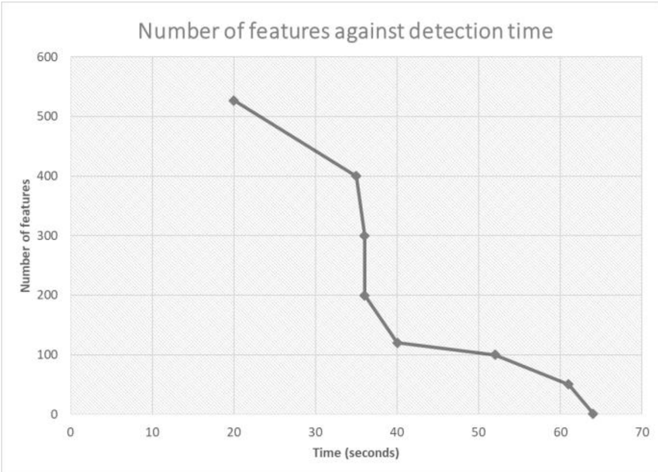

Fig. 9 – Relationship between number of features and detection time for *proposed-SVR* ML model

Another significant aspect of the results obtained in Task 2 is that the resulting ML model is smaller. This has a lot of implications for the training pipeline, as well as the resulting ML model. Training an ML model with a fewer number of features warrants that the training time of the ML model is significantly reduced. This also reduces the ML model size, increases the model inference time, lowers energy consumption and has a higher speed on inference for real-time inference.

In the case of the *proposed-SVR* model, we compare the effect of reducing the number of features to the training time, ML model size, and prediction time when the network failure event is detected at 10 seconds.

Table 2 – Feature size of *proposed-SVR* model

# of Features	Size (MB)	Training Time (sec)	Prediction Time (sec)
all	20.89	2.00	0.65
527	3.31	0.32	0.10
100	0.63	0.10	0.03
64	0.39	0.08	0.03
50	0.30	0.08	0.02
1	0.01	0.03	0.01

There is a trade-off between the complexity of the *proposed-SVR* model and the number of features used. Increasing the number of features increases the complexity of the ML model which improves the model's ability to generalize to new data. It is clear that the number of features must be selected carefully as it affects generalization of the model when it is presented with new data. In Table 2, the training time is for 600 data samples and the prediction time is for 300 data samples.

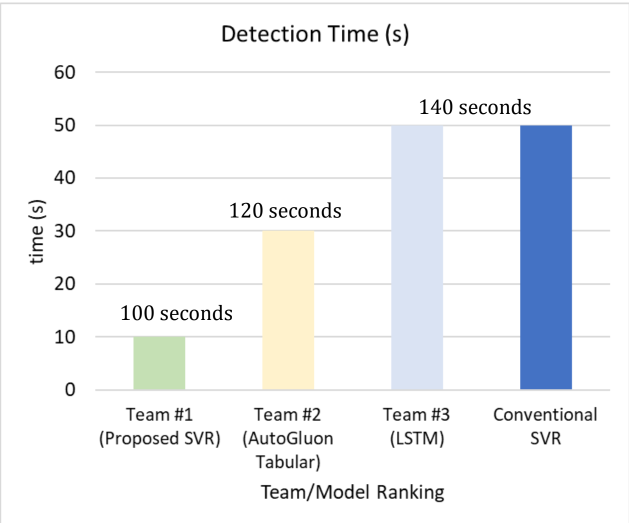

Fig. 10 – Detection time performance comparison between teams from [37]

During the 2022 ITU AI/ML in 5G Challenge [37], different teams attempted to provide the solution to this problem statement of detecting network failure events. Fig. 10 presents the detection time comparison for an *f1-score* of more than 0.9. It is observed that our proposed solution (*proposed-SVR*) obtains the best detection time of 10 seconds (100 seconds if preparation phase is included) as compared to 30 seconds and 50 seconds for a similar *f1-score* reported by other teams who used AutoML and LSTM. Additionally, a comparison with conventional SVR is made to verify the performance gains obtained by our proposed ML model.

7. CONCLUSION

In this paper, the effects of network outages on telecommunication systems and customers have been explored, along with the need for proactive measures to be taken as early as possible to prevent these impacts. Specifically, the paper proposes the use of machine learning models, such as SVR and Bayesian ridge, to predict network failures using packet logs. The study established that the *proposed-SVR* model performed the best, achieving an *f1-score* of more than 0.9 when predicting network failures based on packet logs with a detection time of 10 seconds. Additionally, the proposed ML model was able to accurately predict network failures even after removing several features, achieving an *f1-score* of above 0.9 using 527 features and packet logs with a detection time of 20 seconds. Overall, the paper highlights the importance of proactive network failure prediction and demonstrates the effectiveness of using machine learning models for this purpose.

REFERENCES

[1] R. Stair and G. Reynolds, Fundamentals of Information Systems, 8th ed. Boston, MA: Cengage Learning, 2015.

[2] S. Mattisson, "An Overview of 5G Requirements and Future Wireless Networks: Accommodating Scaling Technology," *in IEEE Solid-State Circuits Magazine*, vol. 10, no. 3, pp. 54-60, Summer 2018, doi: 10.1109/MSSC.2018.2844606.

[3] M. Shafi et al., "5G: A Tutorial Overview of Standards, Trials, Challenges, Deployment, and Practice," *in IEEE Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1201-1221, June 2017, doi: 10.1109/JSAC.2017.2692307.

[4] European Court of Auditors, "5G roll-out in the EU: delays in deployment of networks with security issues remaining unresolved." Special Report No. 03/2022, 2022, available online at https://www.eca.europa.eu/Lists/ECADocuments/SR22_03/SR_Security-5G-networks_EN.pdf

[5] Thales, "A 5G Progress Report: Launches, Subscribers, Devices & more," https://www.thalesgroup.com/en/worldwide/digital-identity-and-security/magazine/5g-progress-report-launches-subscribers-devices

[6] K. Loughran "Facing today's 5G issues and challenges." https://www.jabil.com/blog/5g-challenges.html

[7] X. Foukas, G. Patounas, A. Elmokashfi and M. K. Marina, "Network Slicing in 5G: Survey and Challenges," in IEEE Communications Magazine, vol. 55, no. 5, pp. 94-100, May 2017, doi: 10.1109/MCOM.2017.1600951.

[8] Hall, J.W., S. Thacker, M.C. Ives, Y. Cao, M. Chaudry, S.P. Blainey, and E.J. Oughton. 2017. Strategic analysis of the future of national infrastructure. Proceedings of the Institution of Civil Engineers-Civil Engineering 170(1): 39–47.Return to ref 2017 in article.

[9] Mirielli Edward, J., & Webster Linda (2016). Business continuity and information availability. Journal of Computing Sciences in Colleges.

[10] Tareen, B.K., Yang, J., & Lee, S. (2008). Developing a Cross Platform Server Monitoring Application.

[11] S. Chittor, "Effect of faulty network components on node availability," Proceedings International Phoenix Conference on Computers and Communications, Scottsdale, AZ, USA, 1995, pp. 128-135, doi: 10.1109/PCCC.1995.472500.

[12] Narang, Unnati and Shankar, Venkatesh and Narayanan, Sridhar, "How Does a Failure in a Retailer's Mobile App Impact Purchases in its Online and Offline Channels?" (October 31, 2022). Mays Business School Research Paper No. 3144434, Stanford University Graduate School of Business Research Paper No. 18-18, http://dx.doi.org/10.2139/ssrn.3144434

[13] A. Wright, "How Much Does a Service Outage Really Cost a Telecom Company?" https://mapware.com/blog/how-much-does-a-service-outage-really-cost-a-telecom-company/

[14] T. Horie, G. Hasegawa, S. Kamei and M. Murata, "A New Method of Proactive Recovery Mechanism for Large-Scale Network Failures," 2009 International Conference on Advanced Information Networking and Applications, Bradford, UK, 2009, pp. 951-958, doi: 10.1109/AINA.2009.46.

[15] "Network outage definition," Law Insider, available online at: https://www.lawinsider.com/dictionary/network-outage

[16] How to avoid network failure & outages for business https://kirbtech.com/how-to-prevent-network-interruptions

[17] Y. Obayashi "KDDI aims to restore service Sunday after 40 mln users affected" https://www.reuters.com/business/media-telecom/japans-kddi-says-70-services-restored-after-wide-network-troubles-2022-07-03/

[18] The guardians “Massive network outage in Canada hits homes, ATMs and 911 emergency lines” https://www.theguardian.com/world/2022/jul/08/internet-down-canada-rogers-mobile-network-outage

[19] Koks, E., Pant, R., Thacker, S. et al. Understanding Business Disruption and Economic Losses Due to Electricity Failures and Flooding. Int J Disaster Risk Sci 10, 421–438 (2019). https://doi.org/10.1007/s13753-019-00236-y

[20] The Guardian “O2 outage: more than 30m mobile customers unable to get online” https://www.theguardian.com/business/2018/dec/06/o2-customers-unable-to-get-online

[21] Business Today “Vodafone Idea services restored after hours of being hit” https://www.businesstoday.in/tech-today/trending/story/vodafone-idea-services-restored-after-hours-of-being-hit-368838-2023-02-03

[22] Agiwal, Mamta & Roy, Abhishek & Saxena, Navrati. (2016). Next Generation 5G Wireless Networks: A Comprehensive Survey. IEEE Communications Surveys & Tutorials. 18. 1-1. 10.1109/COMST.2016.2532458.

[23] M. A. Ridwan, N. A. M. Radzi, F. Abdullah and Y. E. Jalil, "Applications of Machine Learning in Networking: A Survey of Current Issues and Future Challenges," in IEEE Access, vol. 9, pp. 52523-52556, 2021, doi: 10.1109/ACCESS.2021.3069210.

[24] D. Rafique and L. Velasco, “Machine Learning for Network Automation: Overview, Architecture, and Applications [Invited Tutorial],” Vol. 10, Issue 10, pp. D126-D143 (2018) https://doi.org/10.1364/JOCN.10.00D126

[25] P. Yu, F. Zhou, X. Zhang, X. Qiu, M. Kadoch and M. Cheriet, "Deep Learning-Based Resource Allocation for 5G Broadband TV Service," in IEEE Transactions on Broadcasting, vol. 66, no. 4, pp. 800-813, Dec. 2020, doi: 10.1109/TBC.2020.2968730.

[26] R. Dangi, A. Jadhav, G. Choudhary, N. Dragoni, M. K. Mishra, and P. Lalwani, “ML-Based 5G Network Slicing Security: A Comprehensive Survey,” Future Internet, vol. 14, no. 4, p. 116, Apr. 2022, doi: 10.3390/fi14040116.

[27] Oduro-Gyimah, F.K., Boateng, K.O., Adu, P.B., & Quist-Aphetsi, K. (2021). Prediction of Telecommunication Network Outage Time Using Multilayer Perceptron Modelling Approach. 2021 International Conference on Computing, Computational Modelling and Applications (ICCMA), 104-108.

[28] Chen, J., Huang, C., & Shih, C. (2017). The exploration of machine learning for abnormal prediction model of telecom business support system. 2017 19th Asia-Pacific Network Operations and Management Symposium (APNOMS), 211-214.

[29] Darlington-NjokuChidinma, A., Mishra, B.K., & Sayers, W.K. (2022). Fault Log Text Classification Using Natural Language Processing And Machine Learning For Decision Support. 2022 14th International Conference on Software, Knowledge, Information Management and Applications (SKIMA), 98-103.

[30] R. Madan and P. S. Mangipudi, "Predicting Computer Network Traffic: A Time Series Forecasting Approach Using DWT, ARIMA and RNN," 2018 Eleventh International Conference on Contemporary Computing (IC3), Noida, India, 2018, pp. 1-5, doi: 10.1109/IC3.2018.8530608.

[31] A. Bhardwaj, V. Mangat and R. Vig, "Hyperband Tuned Deep Neural Network With Well Posed Stacked Sparse AutoEncoder for Detection of DDoS Attacks in Cloud," in IEEE Access, vol. 8, pp. 181916-181929, 2020, doi: 10.1109/ACCESS.2020.3028690.

[32] D. Kushnir, G. Gohil, Z. Sayeed and H. Uzunalioglu, "Predicting Outages in Radio Networks with Alarm Data," 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), Banff, AB, Canada, 2018, pp. 1-2, doi: 10.1109/IWQoS.2018.8624134.

[33] Shuan, L.H. & Tan, Yi-Fei & Soo, Wooi King & Guo, Xiaoning & Mein, L.Z. (2016). Network equipment failure prediction with big data analytics. 8. 59-69.

[34] D. Castillo "Machine Learning Regression" https://www.seldon.io/machine-learning-regression-explained

[35] XGBoost https://www.nvidia.com/en-us/glossary/data-science/xgboost

[36] A. Raj "Unlocking the true power of support vector regression" https://www.towardsdatascience.com/unlocking-the-true-power-of-support-vector-regression-847fd123a4a0

[37] KDDI Research, "ML5G-PS-005: Network failure prediction on CNFs 5GC with Linux eBPF." ITU AI Challenge. https://challenge.aiforgood.itu.int/match/matchitem/64 (accessed 8th August, 2022)

[38] J. Brownlee (2020) "SMOTE for Imbalanced Classification with Python" Machine Learning Mastery https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/

AUHORS

Emmanuel Basikolo is an aspiring physicist and data scientist, currently pursuing a Bachelor of Science in physics. His research interests include data analysis, machine learning and artificial intelligence, and statistical modelling.

Thomas Basikolo is a programme officer with the ITU Telecommunication Standardization Bureau (TSB). He coordinates and manages the ITU AI/ML in 5G Challenge and is an advisor to the ITU-T Focus Group on Autonomous Networks.

Prior to joining ITU, he worked as a research engineer in the Engineering Department of Microwave Factory Co., Ltd, Tokyo, Japan.

He received a PhD in electrical and computer engineering from Yokohama National University, Japan. He is a recipient of multiple best paper awards, the IEEE AP-S Japan Student Award and the Young Engineer of the year award by IEEE AP-S Japan in 2018.

He has co-authored peer-reviewed journal and conference papers, predominantly in the areas of wireless communications and antenna engineering. He serves as a reviewer of IEEE and IEICE journals. His interests include machine learning, deep learning and network science, and their applications in wireless networks.

