ML5G-PS-005: Network Failure Prediction

Team : The Big Fools

Member: Emmanuel Basikolo

1. Abstract

Network outage or failure has devastating effects on both the telecommunication company and its customers [1]. Telecommunication companies need to know when network outage will occur so they can prevent it from happening. Using metrics from the network and machine learning models, telecommunication companies can detect or predict when network outage will occur as early as possible and with high accuracy to prevent the outage from happening.

2. Introduction

Since 2020, telecommunication companies around the world have started commercial 5G deployments [2]. As the technology underpinning 5G has become more developed, a realization of the complexity of 5G has come along with it. 5G specifications have grown in richness and capabilities. Having such a diverse "toolbox" of capabilities is a great benefit to service providers; it enables them to customize and optimize their networks tailored to customers' needs and to add new and innovative services. However, this richness of capabilities could also present some difficulties. The companies developing 5G products and services now have a much broader spectrum of features they need to develop. The breadth of features may lead to a slower rollout of 5G features and sometimes results in network failures.

Network outage refers to the inaccessibility of a network due to the failure of a particular system, application, or the entire network [1]. This outage may occur due to maintenance activities, power cuts, or unexpected technical failures. The impact of such interruption may vary for every organization, depending on its size, customer base, and service capabilities. This may have devastating effects which includes: revenue loss, customer attrition, and share price [1]. Additionally, network outage may affect other businesses and institutions that depends on the network by disabling communication affecting supply chain, service centers and delivery processes [1].

It is of great importance to prevent network outage so that, telecommunication company provides their customers with the best experience possible [2]. Using machine learning, unplanned network outages caused by network failures can be detected earlier before they occur and the company may put in place corrective measures to avoid or prevent network outages from happening.

The problem statement ML5G-PS-005 by KDDI for the 2022 ITU AI/ML in 5G Challenge required participants to predict future network outage in a very short amount of time possible with a higher f1-score using machine learning models and time series data consisting of thousands of metrics [2].

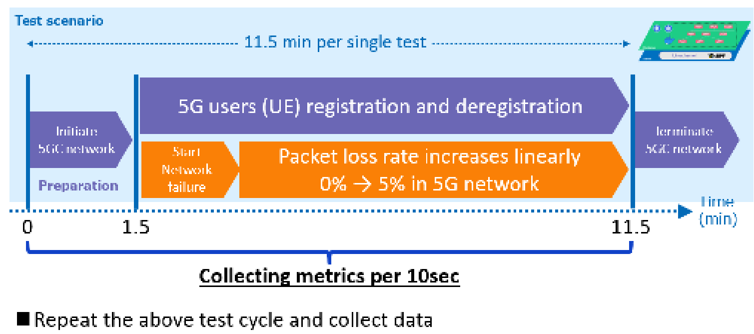

Figure 1: Single test scenario

3. Dataset

The dataset provided contains 42,000 examples as training set and 21,000 samples as test set. From this dataset, 3 types of metrics are provided; basic metric obtained by cAdvisor, fine-grained metric by eBPF, and 5G metric by counting 5G logs.

3.1 Dataset description

In failure scenarios, a packet loss event occurs and the loss rate increases linearly while 5G users repeating registration and deregistration [2] as shown in Fig. 2. The main task is therefore to predict *how early and accurately the future network failures can be predicted* using time-series data consisting of thousands of metrics.

The data was collected from a test-bed in which the preparation phase ranges from 0 sec to 90 sec. (1.5 min.). This is the initiation of the 5G core and no event occurs during this phase. From 90 sec. (1.5 min.). to 690 sec. (11.5 min.) is registration and deregistration phase. The metrics were collected between these two phases at 10 sec intervals from 0 to 11.5 minutes (called test scenario/cycle). Given 42,000 examples, there are only 600 (42,000/70) test cycles. From each test scenario, one example was selected at particular time (detection time). All selected examples were combined and they form new dataset [2] which was used for training and testing the machine learning models.

	Total examples	Test cycle size	Selected examples
Training set	42,000 examples	70 examples	600 examples
Test set	21,000 examples	70 examples	300 examples

Table 1: Provided dataset examples

3.2 Problem statement tasks

The problem's solution has been divided into two tasks; Task 1 and Task 2. In the case of task 1, every metric provided is used to predict network failure in the future [2]. Target value for prediction is User equipment (UE) registration failures that will occur at 10 minutes (600 seconds excluding preparation phase). If the predicted UE registration failure is above certain value, then network failure occurs and if is below same value the network failure does not occur and this value is called threshold value [2]. Figure below shows examples of data samples collected when network failure occurs (abnormal) and when network is normal and no failure occurs.

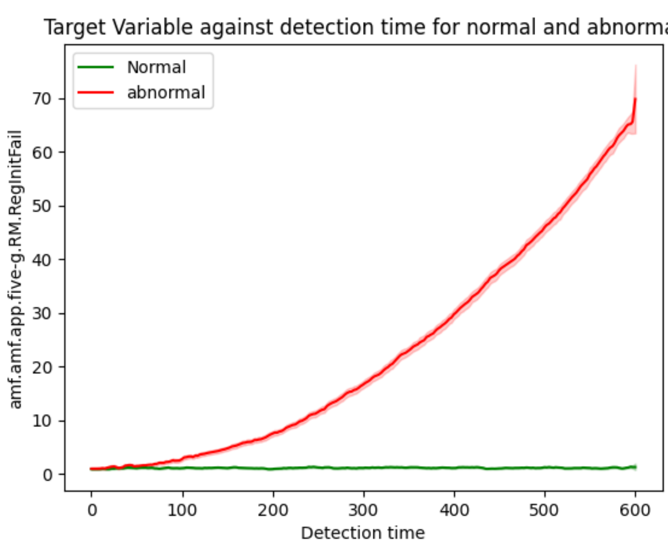

Figure 2: Plot of dataset example for normal and abnormal scenarios.

For task 2, part of metrics provided are used to determine if network failure will occur at 10 minutes or not. Similar to task 1, UE registration failure at 10 minutes will be used together with threshold to determine if network will fail or not. This task requites smaller detection time, smaller number of metrics used and the F1-score obtained must be over 90 percent. The metrics are selected based on their importance or the contribution to final f1-score. That means, most important feature is one which contributes more to the final f1-score [3].

Following horizontal bar graph shows top ten of most important features and their contribution to final f1-score.

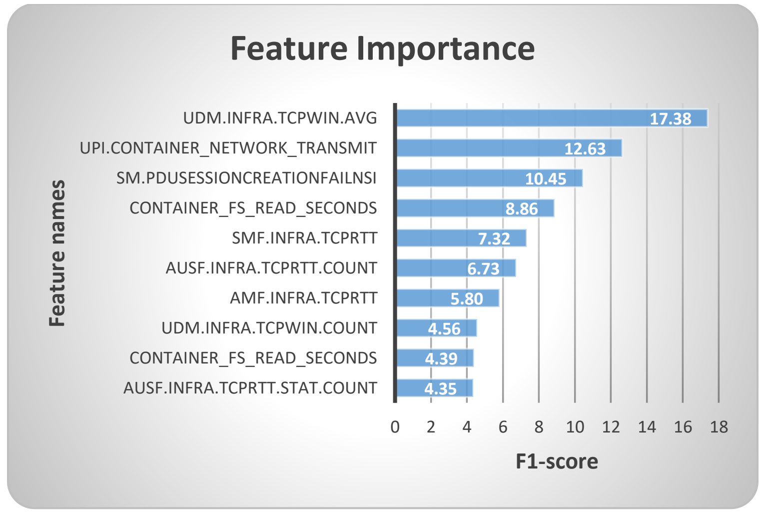

Figure 3: Feature importance.

4. Machine Learning Models

In this section we explore and discuss some well-known and commonly used machine learning methods for the prediction of network failure. ML classifiers can be used to estimate the most likely failure cause, after being trained with a comprehensive set of time series collected in presence of known failures. From the dataset provided as discussed in section 2, regression models and decision trees were selected to perform the task at hand. The models used as solution include: Support Vector Regressor (SVR), Bayesian Ridge Regression, Cat Booster Regressor, Extreme Gradient Boosting Regressor (XGBR) and Random Forest Regressor.

From these models, SVR performed well in the initial analysis and further improvements were made to this algorithm (or model). We proposed to change the model's kernel from radial basis function (RBF) to sigmoid since part of the provided data was non-linear and sigmoid works better with such data [4]. Secondly, regulation parameter was changed from 1 to 0.095 as this solves fitting problem. SVR had good performance in all cases, while other algorithms work better when number of features were reduced. This may be caused by data overfitting.

5. Results

This section presents and discusses the results obtained for the Two tasks that was solved as part of this problem statement.

5.1 Task 1 Results

For task 1, the accuracy, precision, recall and F1-score were all below 90 percent from 0 seconds to 9 seconds then the value increases to over 90 percent at 10 seconds then it decrease again below 90 percent from 11 seconds up to 18 seconds after that the values keep increasing up to 1.0 and remain there up to when detection time was 600 seconds which is last possible value for detection time. This can be observed in the figure below;

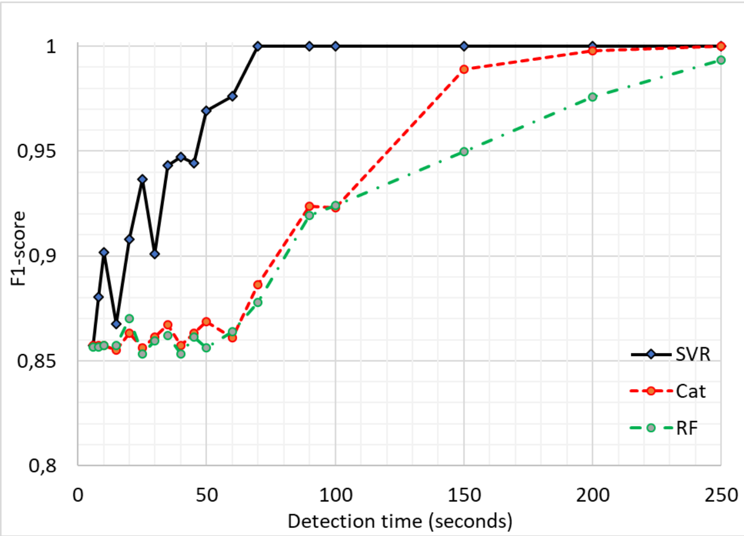

Figure 4: Plot of detection time against f1-score.

5.2 Task 2 Results

For task 2, the values for accuracy, precision, recall and F1-score increases when both detection time and number of features are increased. For instance, 1 feature has minimum detection time of 64 seconds to reach more than 90 percent accuracy and 200 features require 36 seconds to acquire similar results.

Time (seconds)	# of Features	F1-Score	Threshold	Model
64	1	0.980392	1.32	Bayesian Ridge
61	50	0.901010	1.74	SVR
52	100	0.914286	1.14	SVR
40	120	0.902287	1.30	SVR
36	200	0.907975	1.11	SVR
36	300	0.917197	0.62	SVR
35	400	0.905579	0.55	SVR
20	527	0.910603	0.57	SVR

Table 2: Relationship between detection time, number of features and F1-score for different models.

Using all feature is not efficient way, as it requires lots of computation power to train the model or to get prediction. The tradeoff of number of features and detection time is crucial to get smaller detection time with smaller number of features.

6. Conclusion

Using data of how packets are transferred on network, telecommunication companies may predict if network failure will occur or not and this has great advantage on the telecommunication company. Supervised models were used and generally, SVR model do better in all cases. Decreasing number of features used in training the model, increases detection time in order to get same f1-score.

7. References

[1]. A. Wright, "How Much Does a Service Outage Really Cost a Telecom Company?" Mapware.com https://mapware.com/blog/how-much-does-a-service-outage-really-cost-a-telecom-company/ (accessed 20th October, 2022)

[2]. 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)

[3]. J. Brownlee, "feature importance and feature selection with xgboost in python." Machine Learning Mastery. https://machinelearningmastery.com/feature-importance-and-feature-selection-with-xgboost-in-python/ (accessed 5th October, 2022)

[4]. DeepAI, "Sigmoidal Nonlinearity." deepai.org https://deepai.org/machine-learning-glossary-and-terms/sigmoidal-nonlinearity#:~:text=Sigmoidal%20functions%20are%20frequently%20used,linear%20relationships%20between%20data%20features. (accessed 28th October, 2022)