Annals of Telecommunications
https://doi.org/10.1007/s12243-021-00849-9

Prediction of low accessibility in 4G networks

Diogo Ferreira[2] · Carlos Senna[1] · Paulo Salvador[2] · Luís Cortesão[3] · Cristina Pires[3] · Rui Pedro[3] · Susana Sargento[2]

Received: 8 July 2020 / Accepted: 15 April 2021

Abstract

The increased programmability of communication networks makes them more autonomous, and with the ability to actuate fast in response to users and networks' events. However, it is usually a difficult task to understand the root cause of the network problems, so that autonomous actuation can be provided in advance. This paper analyzes the probable root causes of reduced accessibility in 4G networks, taking into account the information of important key performance indicators (KPIs), and considering their evolution in previous time-frames. This approach resorts to interpretable machine learning models to measure the importance of each KPI in the decrease of the network accessibility in a posterior time-frame. The results show that the main root causes of reduced accessibility in the network are related with the number of failure handovers, the number of phone calls and text messages in the network, the overall download volume, and the availability of the cells. However, the main causes of reduced accessibility in each cell are more related to the number of users in each cell and its download volume produced. The results also show the number of principal component analysis (PCA) components required for a good prediction, as well as the best machine learning approach for this specific use case. In addition, we finished our considerations with a discussion about 5G network requirements where proactivity is mandatory.

Keywords Cellular networks · Root cause analysis · Machine learning

1 Introduction

In communication networks, root cause analysis of network problems or failures is essential, so that a fast reaction to these failures or even an anticipation and prevention of these failures can take place. However, usually, it is difficult to assess the cause of reduced network accessibility, since it may happen due to a large number of issues, and impacting in a large number of metrics simultaneously. Knowing the causes that lead to these events can help to detect them prematurely and it indicates how to act autonomously on the network to mitigate or avoid them.

With the increased requirements proposed for the 5G networks, e.g. 1,000,000 devices per km^2, 20 Gbit/s of download peak data rate [5], the new generation of cellular networks promises to handle more traffic than ever before. The incorporation of network slicing, as well as software-defined networking (SDN) and network function virtualizations (NFVs) in the 5G architecture, overly increases the management complexity of those networks. With so many metrics to monitor, it is becoming harder to detect the cause of an event due to the complex

✉ Carlos Senna
cr.senna@av.it.pt

Diogo Ferreira
diogodanielsoaresferreira@ua.pt

Paulo Salvador
salvador@ua.pt

Luís Cortesão
luis-a-cortesao@alticelabs.com

Cristina Pires
cristina-j-pires@alticelabs.com

Rui Pedro
rui-d-pedro@alticelabs.com

Susana Sargento
susana@ua.pt

1 Instituto de Telecomunicações, Aveiro, Portugal

2 DETI, University of Aveiro and Instituto de Telecomunicações, Aveiro, Portugal

3 Altice Labs, Aveiro, Portugal

Published online: 04 May 2021

combinations of various key performance indicators (KPIs). Traditional approaches to detect the root cause of failures, with a knowledge base and a set of rules, is becoming obsolete due to the flexibility of the network. With the advances in machine learning, it is easier to indirectly analyze dependent variables with reduced complexity, but with increased uncertainty.

This work identifies the KPIs that may cause reduced accessibility in 4G networks, using machine learning techniques. Knowing those KPIs helps to create a proactive management of the network, detecting an eventual future drop in network accessibility and having the possibility to avoid it by acting on the network, adjusting the resources that have the most impact on those KPIs. However, the aim of this work is not to propose a resource management approach, but to identify the problems in the network, the causes of those problems, and give this information to the network operator, so that it can adjust its resources along time, when it is predicted that the accessibility will be reduced. Therefore, we will use statistical analysis and machine learning for this identification and prediction in advance.

In this work, two different approaches for root cause analysis are explained and discussed. The first approach measures the feature importance using internal calculations in the model to determine the importance of each KPI in a reduced accessibility event. However, due to the high number of combinations of the KPIs, it is not feasible to test all possibilities. It is then important to perform feature selection. The second approach proposes a dimensionality reduction algorithm to reduce the number of features and apply the machine learning algorithms.

A stepping approach of our solution is depicted in Fig. 1, which shows the different phases of the work and the relationship between them. Initially, it is chosen the low accessibility metric, the performance metrics, the feature importance methodology and the features of the models. For each scenario defined, models are trained with machine learning algorithms. The feature importance of the best model is measured to determine which are the most important features to predict the low accessibility in a 4G network.

In the evaluation results, we present the most important KPIs that are able to predict if the number of E-UTRAN radio access bearer (E-RAB) setup failures is above a specific threshold. Then, we present the most important KPIs that are able to predict if the number of E-RAB establishment failures has high variations, and therefore, are highly correlated to the reduction of the network accessibility.

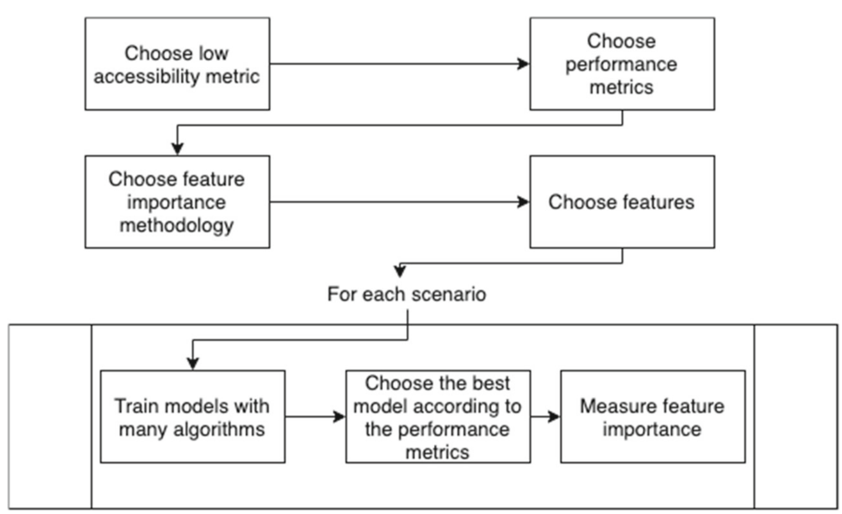
The results showed that the causes of reduced accessibility of the overall network are the number of failure handovers, the number of phone calls and SMSs in the network, and the overall download volume; the causes of reduced accessibility of each cell are related with the number of users in a cell and its download volume. This is extremely important to proactively change the network configurations and avoid the reduction on the network accessibility.

Previous work on this subject was published in [6], where a preliminary study of the root cause analysis has been performed. This article extends this previous one and has the following contributions:

- Analyses the problem of low accessibility in 4G networks;
- Through different approaches, it analyses the important features on the reduced accessibility of the network;
- Applies and extends machine learning algorithms to determine the KPIs that most impact the low network accessibility;
- Determines the difference between overall network and per cell performance in terms of impacting KPIs;

In the remaining of the paper, Section 2 shows related work, Section 3 discusses how KPIs impact the network

Fig. 1 Workflow to determine the root causes of reduced accessibility in the network



accessibility, Section 4 presents our approaches for root cause analysis, Section 5 discusses the results, Section 6 present an architecture to act proactively in 5G networks, and Section 7 presents conclusion and future works.

2 Related work

Understanding the root cause of an observed symptom in a complex system has been a major problem for decades. The main question is how to find appropriate real-time mechanisms to determine root causes [18]. Most root cause analysis approaches for network operators are currently based on Bayesian Networks [14]. They have the capability of representing network metrics and events in nodes, and their relations represent the dependencies, along with a conditional probability. To obtain the most probable cause of an error, a probabilistic inference can be done. In [1], the author proposed a repair system for analyzing four main network problems: IP duplication, link up/down, loopback, and MAC flapping problems. He made the problem classification using rule-based with thresholds. However, the proposed solution does not use machine learning or other intelligent method, making the solution static and with no capability to evolve or understand changes in the network behavior. In [2][3], the authors argue that a Bayesian network is not suitable for large-scale systems with a large number of components, because the complexity of inference increases exponentially with the number of nodes and dependencies between them. To solve that, they combine the Bayesian network with case-based reasoning techniques to prune the nodes needed to analyze in the network. The results show that the technique used reduces drastically the inference time, as well as the need for human intervention.

In [19], a generic framework for root cause analysis in large IP networks was proposed. To determine the root cause of events, two reasoning engines are included: Bayesian inference and rule-based reasoning. The authors discuss that rule-based logic is often preferred over Bayesian inference, because it is easier to configure, it has an easier interpretation of results and it is effective in most applications. However, Bayesian networks are preferred when the root cause is unobservable (no direct evidence can be collected). In our work, the root cause of reduced accessibility is mostly unobservable. In [13], Maccari and Passerini explain how they can perform monitoring, anomaly detection and root cause analysis in mesh networks using Big Data techniques. They describe the architecture of mesh network, they justify the use of Big Data techniques and provide a design for the storage and analysis of Big Data produced by a large-scale mesh network. But, despite introducing the detection of anomalies and root cause analysis, and presenting a framework to deal with them, the discussion is superficial and no results are presented on the use of machine learning methods or algorithms.

The proposed solution in [4] determines which are the parameters that are most relevant across all different types of failure modes, and use them to build a Bayesian network to model the cause-effect relationship between the degradation parameters (cause) and failure modes (effect) that occur on the field. Two real-life field issues are used as examples to demonstrate the accuracy of the network once it is modelled and built. This paper shows that accurately modelling the hardware system as a Bayesian network substantially accelerates the process of root cause analysis.

A self-healing method based on network data analysis is proposed to diagnose problems in future 5G radio access networks [9, 15]. The proposed system analyzes the temporal evolution of a plurality of metrics, and searches for potential interdependence under the presence of faults. The work in [10] is the only one that we are aware of that uses the concept of "variable importance" ("feature importance" in our work) to measure how much a feature contributes to predicting an objective variable on a machine learning model. The influence of each variable is then represented in an influence matrix that represents the influence that each variable has for each event. However, only the Random Forest algorithm is used to create a model. The variable importance is then calculated using the permutation feature importance approach.

Our work studies RAC through different approaches. It analyses the important features on the reduced accessibility of the network. The diversity of algorithms used in the works shown above is restricted, with few of them using Logistic Regression[1] or Random Forest.[2] We extended our preliminary study applying a wide range of algorithms (Logistic Regression, Extra Trees, Random Forest, Gradient Boosting,[3] and AdaBoost[4]) to determine the KPIs that most impact the low network accessibility. Moreover, we discuss the difference between overall network and per cell performance, in terms of impacting KPIs.

3 Problem statement

The objective of this work is to understand what are the most important KPIs to anticipate reduced accessibility in a 4G network. A 4G network is chosen, since it is a running network with real data. The data available on 5G networks

[1] https://www.statisticssolutions.com/what-is-logistic-regression/

[2] https://towardsdatascience.com/an-intuitive-explanation-of-random-forest-and-extra-trees-classifiers-8507ac21d54b

[3] https://towardsdatascience.com/understanding-gradient-boosting-machines-9be756fe76ab

[4] https://towardsdatascience.com/understanding-adaboost-2f94f22d5bfe

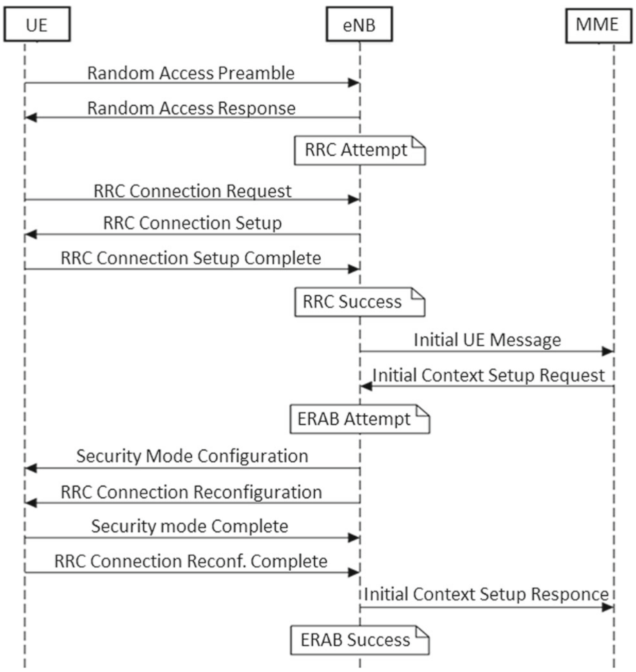


Fig. 2 Sequence diagram indicating the E-RAB setup phase—success

is still scarce and can lead to distorted conclusions. There are still few 5G devices available and, in general, they are still in the testing phase. Even so, at Section 6, we discuss RCA within the 5G context.

In this work, a 31-day data set with 4G network KPIs from an overall country was used. The interval period between each measurement of a KPI is 1 h, which means that there are 24 values for each KPI (total of 744 measurements) for each cell every day. A metric must be chosen to represent the reduced accessibility. After that, with time-series analysis, it is possible to calculate the importance of each KPI to the reduced accessibility metric.

The metric used to indicate low accessibility in the network is the number of E-UTRAN radio access bearer (E-RAB) setup failures per hour in the network. The E-RAB setup in a 4G network is a major KPI for accessibility. The E-RAB is a bearer that the user agents (UEs) need to establish communications in the network [11, 17]. Figure 2, inspired by the OTP blog[5], shows the E-RAB setup phase. After the UE has established a connection with the E-UTRAN Node B (eNB), it is needed to setup a context with the mobility management entity (MME), to enable the UE to communicate and send data to the network.

When the MME sends a context setup request, it is called an E-RAB setup attempt. After some configuration messages between the UE and the eNB, the context setup response from the eNB to the MME is called an E-RAB setup success. There are more E-RAB setup attempts than E-RAB setup successes. When the network is congested, the difference between the E-RAB setup attempts and the E-RAB setup successes is higher, because some messages after the E-RAB setup attempt are lost due to network problems, such as congestion.

A new accessibility metric is used to measure the accessibility of the network: the number of E-RAB setup failures. If the number of E-RAB setup failures has a high value, the network is congested. The sequence diagram indicating E-RAB setup failure is shown in Fig. 3, and the E-RAB setup failure formula is presented in Eq. 1, where esf is the number of E-RAB setup failures, esa is the number of E-RAB setup attempts, and ess is the number of E-RAB setup successes.

$$esf(t) = esa(t) - ess(t) \tag{1}$$

The E-RAB setup failure is depicted in Fig. 4a. The biggest congestion happened on the 17th day. The objective of this paper is now to understand which KPIs most contribute to the forecast of this metric. To do this, it is important to analyze the KPIs individually and also their aggregation. Figure 4b shows that there were failures related to Handover (intrafrequence and interfrequence) at exactly the same period as the E-RAB failures. However, Fig. 4c shows that in the same period, no abnormal behavior related to Circuit Switched Fallback (CSFB) was identified. But both of them are considered important KPIs related to E-RAB as we will demonstrate in Section 5.1.

Other metrics could also be used to measure accessibility problems, such as the Radio Resource Control (RRC) setup

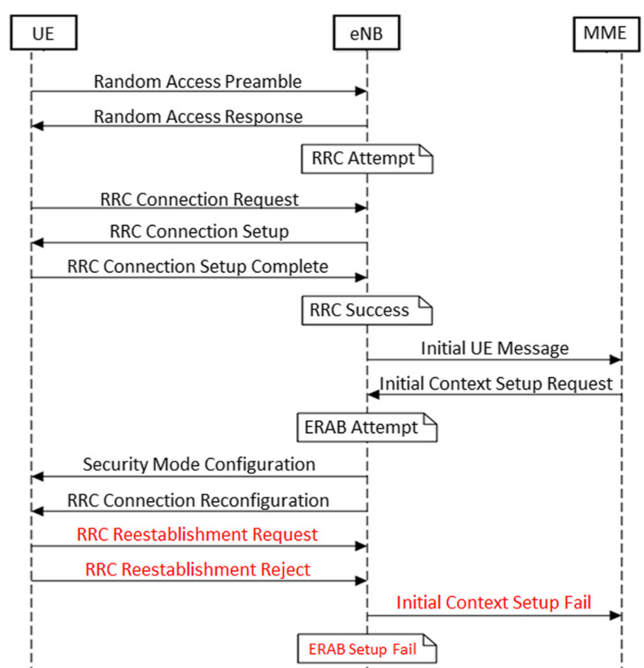


Fig. 3 Sequence diagram indicating the E-RAB setup phase—fail

[5] https://ourtechplanet.com/lte-erab-success-rate/

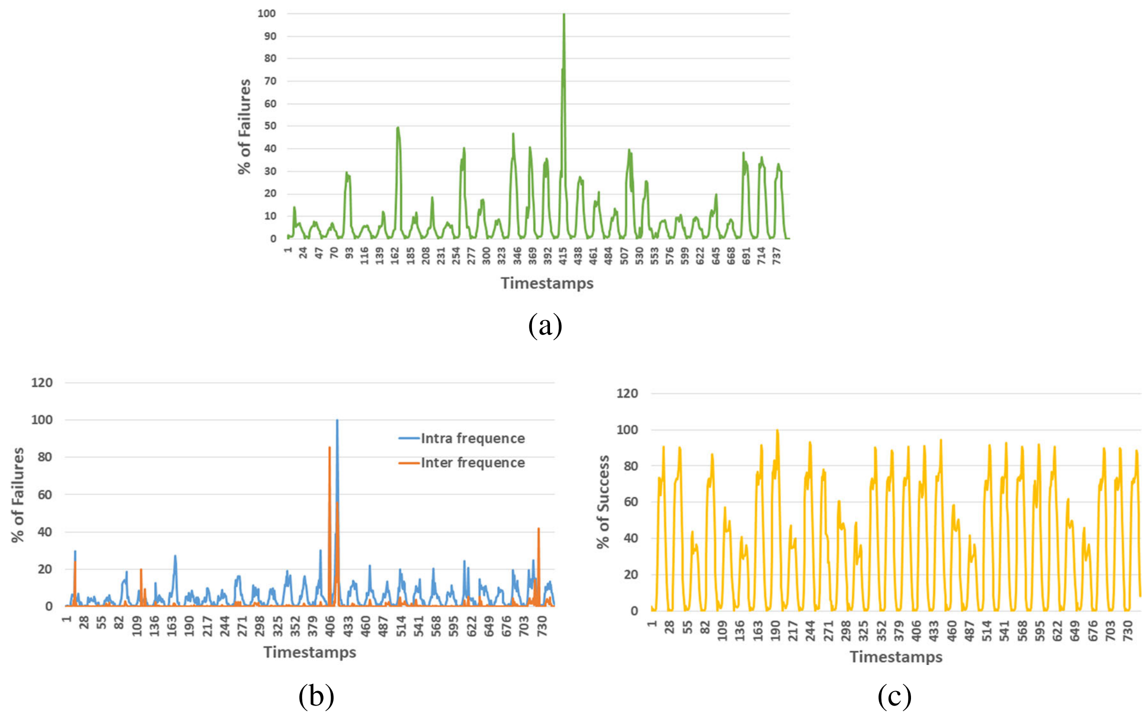


Fig. 4 Important KPIs related with E-RAB

failure. As it will be shown in Section 5, this KPI is highly correlated with the ERAB setup failures of an individual cell, as it is involved in the first phase of an E-RAB setup (Fig. 3). Other metrics also commonly used to measure the network accessibility are the contention-based (CB) random-access channel (RACH) procedure failure and the S1 establishment failure [20]. However, both metrics were not present in the available dataset.

4 Methodology for root cause detection

In this section, we describe two different approaches for root cause analysis using machine learning techniques. One approach that measures feature importance using internal calculations in the model is used to determine the importance of each KPI in a reduced accessibility event. Our second approach proposes a dimensionality reduction algorithm to decrease the number of features (feature selection).

4.1 Approaches to measure the feature importance

A simple test to check if any KPI can accurately forecast the number of E-RAB setup failures is to calculate the Pearson correlation coefficient[6] between the number of E-RAB setup failures and all other KPIs. Since the goal is to understand which KPIs can forecast low accessibility before the number of E-RAB setup failures increases, the KPI values will be shifted (lagged) behind the number of E-RAB setup failures by one hour (the values are sampled hourly; one hour is the minimum time interval).

There are two disadvantages of analyzing the KPIs' importance with the Pearson correlation coefficient. For each KPI, it is only measured its linear contribution to the number of E-RAB setup failures. Besides, combinations of KPIs that can be important are not being taken into account, because it is assumed that the KPIs are independent of each other.

There are other approaches to measure the importance of input features to an output value that takes into account these considerations. These approaches take advantage of machine learning techniques, and they can be divided into two categories: some approaches take into account the error of the model in a test set to calculate the importance of the input features; other approaches measure the feature importance with an internal calculation (algorithm-specific) of the coefficients associated with each input feature.

Approaches considering the model error One of the approaches that fall into the first category is the drop column feature importance. In this approach, the importance of a feature is measured by comparing the test error of a model when all features are available as input, with the test error of a model when one feature is dropped for training. The higher the error for the model with one feature dropped, the more importance is given to that feature.

A big disadvantage of this approach is that, for each feature, it is needed to train a new model with that feature

[6] https://www.statisticssolutions.com/pearsons-correlation-coefficient/

dropped, which causes the approach to be inefficient for many features, or for models that take a significant training time. In the permutation feature importance approach [8], a similar approach is used without the need to re-train the model for each feature. Instead of dropping the feature, it is applied random shuffling to the test values of that feature among the various examples, to preserve the distribution of that variable. If the model error is almost unchanged, the feature is not much important for the forecast. However, if the model error increases much, it is a sign that the feature is important for the forecast.

Both previous approaches have some advantages. It is possible to apply them to black-box models, given that the feature importance is measured by the model error. They also take into account all interactions between features, which is an advantage when compared with the correlation tests. In the permutation feature importance approach, a disadvantage is that the results are dependent on the shuffling of the features. If the tests are repeated, the results may vary.

Approaches considering feature importance using internal calculations The approaches that measure the feature importance by inspecting the internals of the models are algorithm-dependent. For some algorithms, like neural networks or support vector machines (SVMs), it is impossible to calculate the importance of each feature, due to the non-linear transformations applied. However, for other algorithms such as Logistic Regression, Extra Trees, Random Forest, Gradient Boosting, or AdaBoost, it is possible to estimate the importance of each feature. For linear regression, the importance of each feature can be measured by the absolute value of the coefficient associated with each input (if all features are within the same scale). For the other four tree-based ensemble algorithms (Extra Trees, Random Forest, Gradient Boosting, and AdaBoost), the feature importance can be calculated with resort to the mean decrease impurity (or Gini impurity[7]). The importance of a node j in a decision tree is computed as described in Eq. 2, where w_j is the weighted number of samples in node j, C_j is the impurity of this node, and $left(j)$ and $right(j)$ are the respective children nodes.

$$ni_j = w_j C_j - w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)} \quad (2)$$

The feature importance of feature i across all nodes is computed as described in Eq. 3,[8] where a node splitting in feature i means that the node uses the value of the feature i to divide its children nodes.

$$Fi_i = \frac{\sum_{j:\text{node j splits on feature i}} ni_j}{\sum_{j \in \text{all nodes}} ni_j} \quad (3)$$

[7] https://victorzhou.com/blog/gini-impurity/

[8] https://stats.stackexchange.com/questions/311488/summing-feature-importance-in-scikit-learn-for-a-set-of-features/

For this approach, it is important that the features are all normalized within the same scale, and it is recommendable that they are from the same type (continuous/categorical) for better importance estimation.

The advantage of the approaches that measure the feature importance by inspecting the internals of the models is that they do not depend on the test set, only on the model. If the model is accurate and it is not underfitting or overfitting, the feature importance can be calculated more reliably than the previous methods. Otherwise, the feature importance will be highly biased. The biggest challenge is to create accurate models that do not overfit the train set. This second category of approaches (measure feature importance with an internal calculation algorithm-specific) will be used to measure the importance of the KPIs.

4.2 Feature selection

If all KPIs are included as features, the performance of the model will be degraded, because some features are uncorrelated with the output and do not contribute to the output classification. It is then important to perform feature selection [12]. The ideal scenario is to test all combinations of features in the input, and determine the best features by the test error. However, it is not feasible to test all combinations, due to their high number.

The approach chosen is to use a dimensionality reduction algorithm to reduce the number of features. Principal component analysis (PCA) [16] will be used as the algorithm to perform dimensionality reduction. The five algorithms chosen to train the model and to calculate the feature importance are the following: Logistic Regression, Extra Trees, Random Forest, Gradient Boosting, and AdaBoost. Those algorithms were chosen because it is possible to measure the importance of each input feature for each prediction. For some algorithms, like neural networks or SVMs, it is impossible to calculate the importance of each feature, due to the non-linear transformations applied.

The feature importance of each model will calculate the importance of each PCA component. Each component importance is then multiplied by the PCA coefficients, to get the KPI importance for each component. Finally, all the importances of the same KPI are added, as described in Eq. 4, where fi_j is the feature importance of the j PCA component and $pca\_coefficient(j, i)$ is the PCA coefficient i for the component j, where i is the number of a KPI and j is a PCA component.

$$kpi\_i_i = \sum_{j=0}^{n\_pca\_components} fi_j * pca\_coefficient(j, i) \quad (4)$$

To get the best possible model, the number of PCA components cannot be too small, because relevant features can be lost; however, the number cannot be too big either, with the risk of creating overfitted models. The number of PCA components tested will vary from 1 to the number of KPIs, for the five algorithms. The model with a lower test error will be used to calculate the KPI importance.

4.3 Defining the input and the output of the algorithms

The input values will be based on the KPIs. There will be two types of input values: normalized values and normalized variations. For each KPI, the normalized value of the previous hour will be used as input. The number of lags could be increased besides one hour, but in this test it is considered that the low accessibility indicators appear at most one hour before the congestion. For each KPI, the normalized variation will be calculated according to the Eq. 5. The normalized variations are added as features because the network accessibility can depend, not only on the previous values but also on sudden variations of other KPIs.

$$normalized\_variation(t) = \frac{value(t-1) - value(t-2)}{value(t-2)} \tag{5}$$

Two types of classification problems will be used to calculate the importance of the KPIs. It is important to understand which KPIs are more important when forecasting the possibility of a low accessibility event in a network. It is also important to understand which KPIs are more important to forecast sudden increases and decreases in the network accessibility. For each case, a different output value will be calculated. For the first case, it will be set a threshold in the 90th percentile of the data, with all the values above the threshold being classified as one, and all other values as zero, turning the problem into a binary classification problem. For the second case, the output will be classified as one if the absolute difference between two consecutive values is higher than the 90th percentile of the data, and zero otherwise. Doing that, it is possible to analyze which KPIs are most important for classifying low accessibility events, and also for forecasting bigger increases and decreases in data, which can be important for resource allocation.

The two classification problems will be applied in two different ways, according to the data split. First, the tests will be done with aggregated data. The network KPIs will be added and will be taken into account in the whole network. In this way, it is possible to forecast low accessibility of the network. Another way of performing the tests is to split the data per cell. In this approach, the data will not be aggregated, and 75% of all cells in the network will be used to train, with the other 25% cells used as test set. The threshold for the tasks will be defined using the specific values of each cell (each cell will have a different threshold, based on its values). With this approach, it is built a model that is capable of detecting low accessibility per cell. This case is expected to perform worse than the aggregated network, because forecasting low accessibility per cell is harder than forecasting low accessibility in the network. However, for a network operator, it is very important to forecast low accessibility per cell for various reasons. For a cell in a crowded region, it can be made management adjustments to avoid the low accessibility of the cell, such as the installation of another temporary cell, or the allocation of resources for that cell. Besides, it can be made a time-series analysis about the future accessibility of the cells in a region, for expansion purposes.

4.4 Performance metric

For both classification problems, a performance metric must be used. Since in the problems previously described the number of positive samples is lower than the number of negative samples, the performance metric on accuracy would give similar cost to the false negative and false positive errors. The performance metric used will be the F1-score (6). The F1-score is the harmonic mean of precision (7) and recall (8), and it encompasses the False Negative and False Positive errors, weighted according to the number of samples of each class. The precision and recall were also calculated. They are not included in the article because they provide similar information and conclusions to $F_1 - score$.

$$F_1 = 2 * \frac{precision * recall}{precision + recall} \tag{6}$$

$$precision = \frac{true\_positives}{true\_positives / false\_positives} \tag{7}$$

$$recall = \frac{true\_positives}{true\_positives / false\_negatives} \tag{8}$$

5 Root cause detection results

Traffic monitoring is essential to provide a good quality to the services in communication networks. Network conditions such as bandwidth, packet loss, delay and jitter are important for traffic engineering to track the quality needs of applications. Therefore, it is important not only to measure the network conditions, but also to analyze them to understand the causes of reduced accessibility in cellular networks. In the following, we show how machine learning techniques can help to understand which network KPIs can indicate a low accessibility event that will happen in the future.

5.1 Aggregated network tests

In this Section, the results for the aggregated network tests will be presented. As explained before, two scenarios are proposed. First, it will be presented (scenario i) the most important KPIs for predicting if the number of E-RAB setup failures is above a threshold. Second, it will be presented (scenario ii) the most important KPIs for predicting if the number of E-RAB establishment failures has high variations.

In Fig. 5 it is shown that the best model for scenario (i) is the Extra Trees algorithm, achieving an F1-score of 86.6%, with 23 PCA components. With a higher number of PCA components, the performance of most of the algorithms starts to deteriorate.

Table 1a shows the ten KPIs that are considered more important for predicting if the number of E-RAB setup failures is above a threshold. The handover failure is considered to be the most important KPI, both inter- and intrafrequency. The third KPI is the Circuit Switched FallBack (CSFB) preparation success. The CSFB is a technology to create circuit-switched calls over a 4G network that does not support LTE voice call standard (VoLTE), which has to fall back on the 3G network. This indicates that the number of phone calls and SMS messages has a high impact on the network accessibility. The Packet Data Convergence Protocol (PDCP) download transmission time, the variation of the inter-frequency handovers and the cell available time are also important KPIs to predict variations in the number of E-RAB setup failures. Finally, the ERAB normal release, handovers intrafrequency success and attempt, and upload traffic volume have also a significant importance.

The best model for scenario (ii) is achieved with the AdaBoost algorithm, achieving an F1-score of 40.0%, with 24 PCA components. The F1-score is lower than in scenario (i) because the task of predicting variations is harder than the one of predicting if the value is above a threshold.

Table 1 Ten important KPIs in the aggregated network

KPI	Value
(a) Scenario (i)	
HO interfreq failure	0.194
HO intrafreq failure	0.188
CSFB Prep Success	0.181
PDCP Download TX Time	0.169
Variation HO interfreq failure	0.150
Cell availability	0.131
ERAB Normal Release	0.124
HO intrafrequency success	0.121
HO intrafrequency attempt	0.121
PDCP Upload Volume (Mb)	0.115
(b) Scenario (ii)	
CSFB Prep Success	0.188
HO intrafreq failure	0.170
PDCP Download TX Time	0.157
HO interfreq failure	0.148
Variation HO interfreq failure	0.141
Download AS (Max)	0.138
Active Subscribers (AS) (Max)	0.125
RRC Setup Failure	0.121
Cell availability	0.120
Upload AS (Max)	0.119

Fig. 5 Different algorithms in the aggregated network (scenario i)

The number of PCA components is almost the same as in scenario (i); however, in Fig. 6, it can be seen that the algorithms' F1-score is very unstable, and that it is hard to build a model to predict accurately the variations of the number of E-RAB establishment failures in the aggregated network.

Table 1b shows the ten KPIs that are considered more important for predicting if the number of E-RAB establishment failures has high variations (scenario ii). The handover failures are still important, but they are not the most important KPI. In this case, the CSFB preparation success is the most important KPI to predict the variations in the number of E-RAB setup failures. Similar to the scenario (i), PDCP download transmission time, the variation of the inter-frequency handovers, the number of active subscribers, download and upload ones, and the cell available time are also important KPIs.

5.2 Individual cells tests

In this subsection, the results for the individual cells will be presented. In this case, 75% of the cells in the network will be used for training and the other 25% for testing. Just like in the previous subsection, first it will be presented (scenario i) the most important KPIs for predicting if the number of E-RAB establishment failures is above a threshold, and then it will be presented (scenario ii) the most important KPIs for predicting if the number of E-RAB setup failures has high variations.

The best model for scenario (i) achieved an F1-score of 30.79%, using the Gradient Boosting algorithm, with 66 PCA components. For the cell prediction, more information is needed to obtain the best model when compared with the previous subsection. Figure 7 shows the F1-score varying with the number of components and the different algorithms. In the PCA component 62, there is an increase in the F1-score of all algorithms.

Table 2a shows the ten KPIs that are considered more important for the scenario (i). Since they are directly correlated with the network accessibility, two KPIs have much more importance than all others:

– RRC setup success ratio ($\frac{\text{RRC setup success}}{\text{RRC setup attempt}}$) and
– E-RAB setup success ratio ($\frac{\text{ERAB setup success}}{\text{ERAB setup attempt}}$)

The best model for scenario (ii) has an F1-score of 20.39%, using the AdaBoost algorithm with 68 PCA components. The F1-score for different PCA components is similar to the scenario (i), where the F1-score improved its performance significantly after 60 PCA components (Fig. 8).

Table 2b shows the ten KPIs that are considered more important for the scenario (ii). Like in the previous scenario, RRC setup success ratio and E-RAB setup success rate are the most important KPIs for the output. In this scenario, other KPIs have also similar importance, such as the variation of the cell availability, the average connected subscribers, and the average number of radio bearers. A radio bearer is a connection between the eNB and the UE at

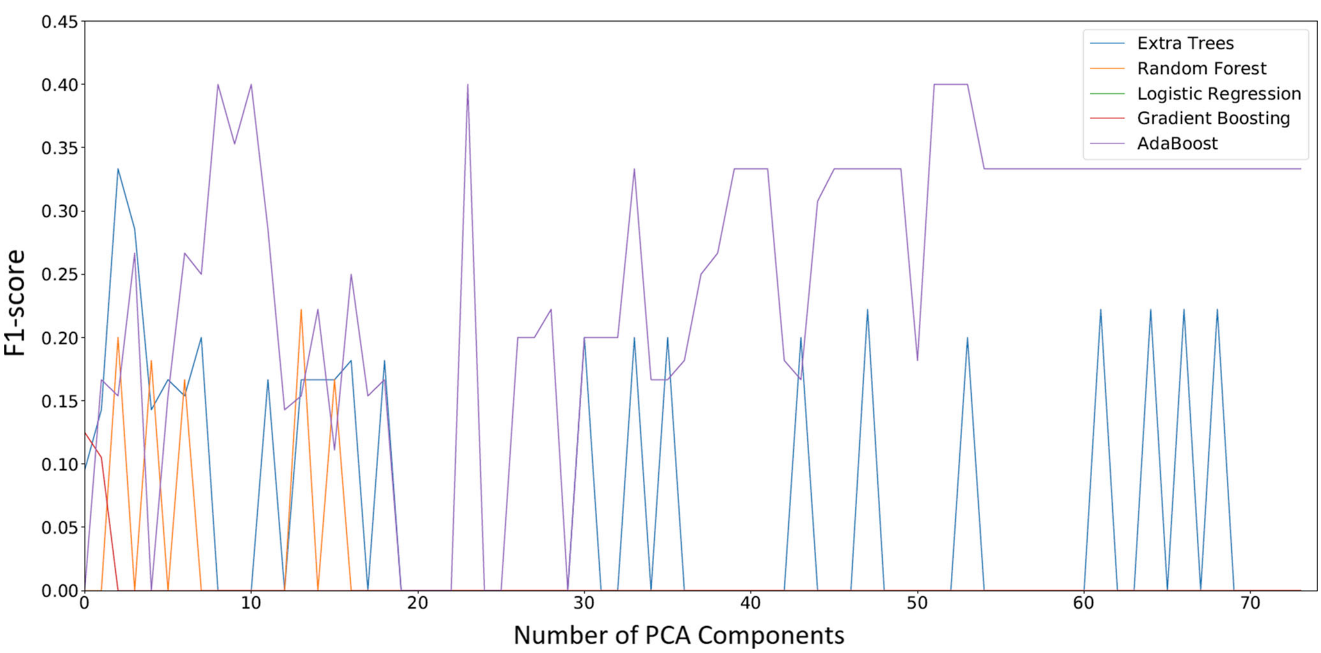


Fig. 6 Different algorithms in the aggregated network (scenario ii)

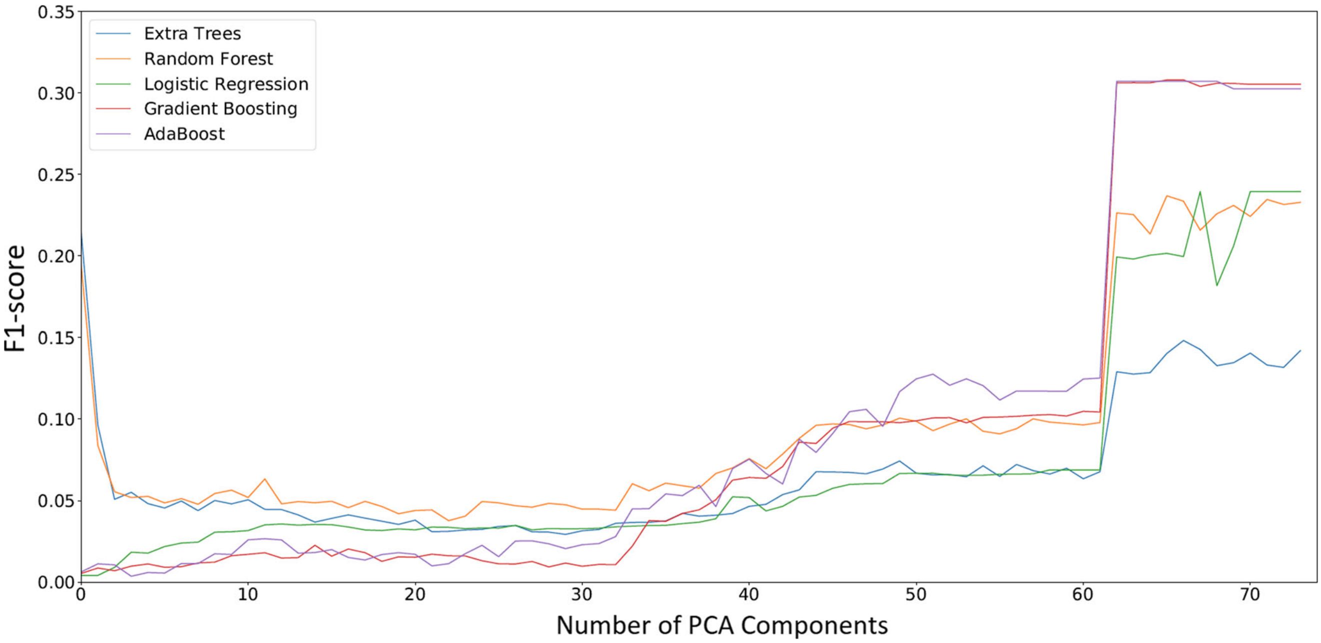


Fig. 7 Different algorithms in individual cells tests (scenario i)

layer 2, and defines the communication configurations for upper layers.

5.3 Discussion

Aggregated network tests From the results of both aggregated network tests, it can be concluded that most KPIs that cause the number of E-RAB setup failures in a network to be above a threshold are the same that cause it to have high variations. Those KPIs are the number of failed handovers (intrafrequency and interfrequency), the CSFB preparation success (number of phone calls and SMSs in the network), the PDCP download volume, and the cell availability. The maximum number of active subscribers (downloading subscribers and overall subscribers) also causes the number of E-RAB setup failures to vary.

Interpreting the KPIs, the results achieved are according to the intuition about lower network accessibility. When the number of failure handovers is high, the cells are crowded with user sessions and cannot accept any more sessions, which leads to lower network accessibility in the next hour. The high number of CSFB preparation success shows that there is a clear relationship between the high number of phone calls and SMS messages in the network with its lower accessibility. The KPIs of PDCP download volume and the maximum number of active subscribers also show that the number of active subscribers and their download volume influence the network accessibility (more than the number of connected subscribers). Finally, the cell availability indicates that, if many cells are unavailable in the current hour, it is likely that the network accessibility will be lower in the next hour.

Individual cells tests The results of the forecasts of low network accessibility in individual cells had higher error than the results with the aggregated network. In these tests, the best models needed more data than the aggregated network models to achieve the best result, because more features were needed to be able to generalize the predictions for different cells. The results show that, just like in the aggregated network tests, the most important KPIs for forecasting the number of E-RAB setup failures in a cell to be above a threshold are the same that cause it to have high variations. However, the most important KPIs that cause lower accessibility in a network are different from the KPIs that cause lower accessibility in a cell. The two most important KPIs are the RRC and the E-RAB setup success rate. These KPIs cannot be understood as the cause for lower network accessibility, but as a consequence. Because they are intrinsically related to the E-RAB setup failure, being themselves accessibility metrics, they can be understood as an indicator of the high autocorrelation between consecutive hours. These results show that the network accessibility per cell is highly dependent on the network accessibility of that cell in the previous hour. If a cell has lower network accessibility in an hour, it is likely that the network accessibility in the next hour for that cell is still low, and vice-versa. It is essential for a network

Table 2 Ten important KPIs in individual cells tests

KPI	Value
(a) Scenario (i)	
RRC Setup Success Rate (SSR)	0.560
ERAB SSR	0.547
Connected Subscribers (CS) (Max)	0.069
Connected Subscribers (Avg)	0.066
Active CS (Avg)	0.043
Variation CS (Max)	0.034
Variation of Active CS (Max)	0.032
ERAB normal release	0.031
Variation Radio Bearers (Avg)	0.031
Variation CS (Avg)	0.031
(b) Scenario (ii)	
ERAB SSR	0.147
RRC SSR	0.147
Variation Cell availability	0.138
CS (Avg)	0.102
Radio Bearers (RB) (Avg)	0.095
Variation RRC SSR	0.093
Variation ERAB SSR	0.092
ERAB Normal Release	0.088
CS (Max)	0.087
Variation RB (Avg)	0.081

operator to monitor the right KPIs for the different tasks: forecast lower accessibility in a network or forecast lower accessibility per cell.

As opposed to the results for the aggregated network, the important KPIs for low network accessibility are not related with the CSFB preparation success, or with any of the handover metrics. The results by cell show that, besides the RRC and the E-RAB setup success ratio, the most important KPIs are counters related to the number of users in a cell and its utilization: maximum number of connected subscribers, average number of connected subscribers, average number of active subscribers download data, variation of the maximum number connected subscribers, variation of the maximum number of active download subscribers, average number of radio bearers or variation of the average number of radio bearers. It is expected that, as these KPIs have higher values, the accessibility of a cell decreases.

Because the available dataset provides KPIs only in 1-hour intervals, it is not possible to predict low accessibility with a smaller time interval. Time intervals of more than one hour were tested (2, 4, 8, 16 and 24 hours), but the $F_1 - score$ was lower than with the 1 hour prediction.

The time required to train the models lies between 0.1 and 1 second, using a personal laptop with 8 GB of RAM, an Intel Core i7-6500U processor, and GeForce 940M as a Graphics card, and implemented with the scikit-learn library.

6 The proactive approach in 5G networks

The effort made in recent years by the research community and the Telecom industry in defining a new network architecture (5G) that supports the new set of requirements

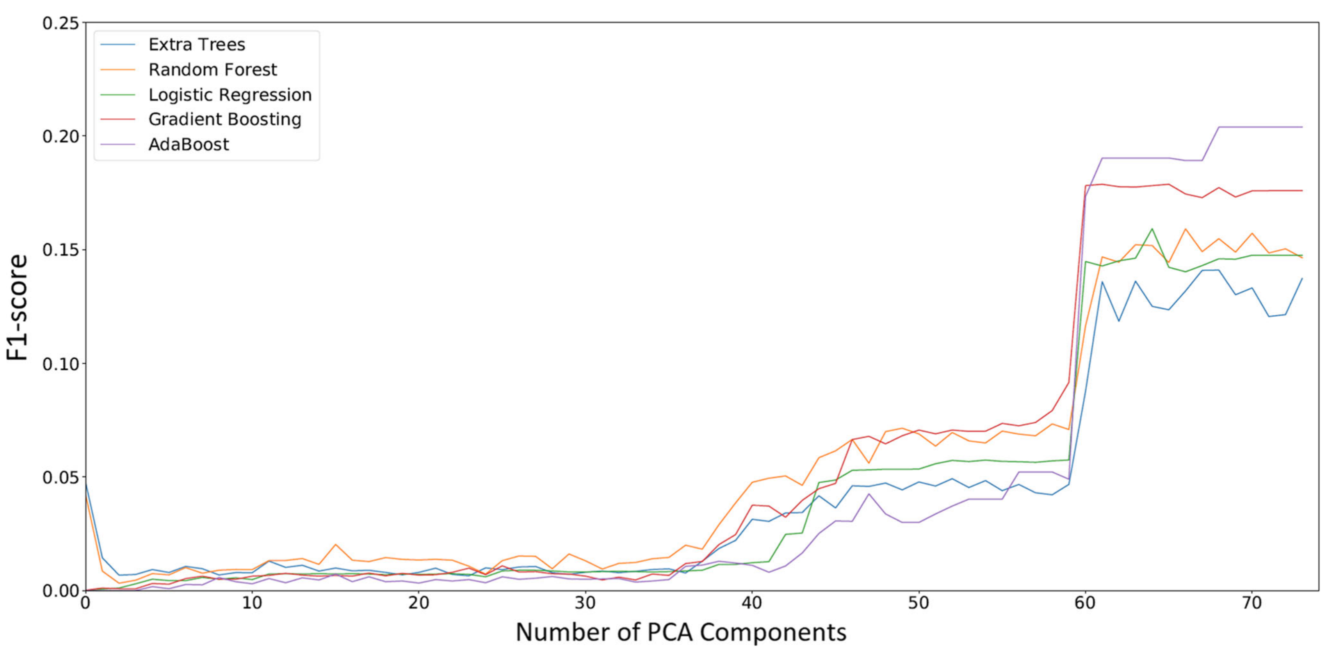


Fig. 8 Different algorithms in individual cells tests (scenario ii)

is finally reaching the market. The urge for efficient monitoring of those networks is decisive for their successful management. Due to its dynamic load and flexible topology, forecasting of the network state is a must to ensure that the user requirements are met. Automation of the network management is also mandatory, here made only possible thanks to advances in virtualization, mainly in Software Defined Networking (SDN) and Network Functions Virtualization (NFV). In 5G environment, traditional routing protocols cannot react in real-time to avoid congestion. That is, the 5G scale requires a proactive prediction-based approach, to faster detect a possible congestion and to act accordingly in the network to avoid it.

6.1 Prediction framework

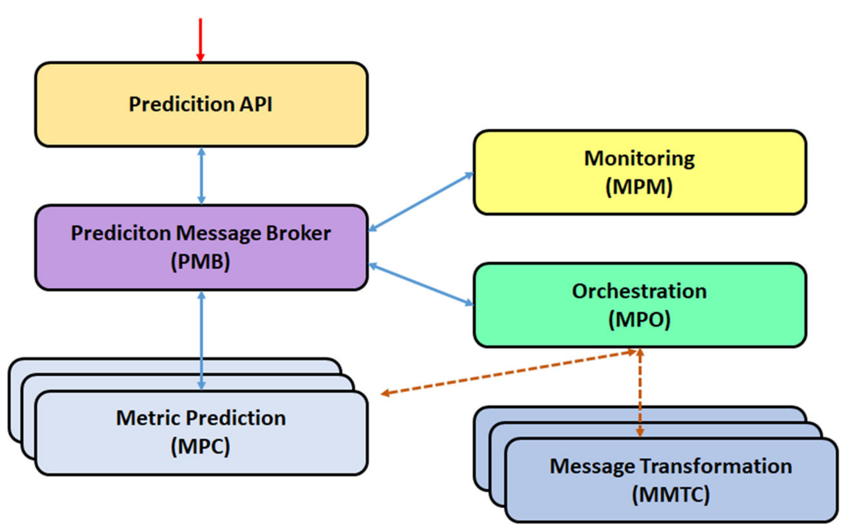
To meet these automation requirements, we design and implement a real-time distributed forecasting framework. The proposed framework is able to test, in real-time and simultaneously, different algorithms with different metrics, including ensembles of the best algorithms [7]. With the proposed framework, it is straightforward to create ensemble predictions (predictions with the results of other predictors), and include a message transformer, useful for combining the result of different predictors, or for changing the predictions. The modularity of the architecture is adequate for enforcing data privacy policies. The predictor components, where the data is stored, can be relocated to a computational resource with reinforced security.

The high-level view of the prediction architecture, depicted in Fig. 9, is composed of six main components: the Prediction API, the Predictor Message Broker (PMB), the Metric Prediction Component (MPC), the Metric Message Transformation Component (MMTC), the Metric Prediction Monitoring (MPM), and the Metric Prediction Orchestrator (MPO). The Prediction API is the interface that allows a client to: (i) predict a value for a time series system/metric, (ii) train the predictor for a given metric, (iii) save new time series data for later training, and (iv) change the training parameters. The Prediction API is a REST microservice that encapsulates in a simple and convenient way the functionalities of the predictors without any operability loss, to be used for non-experts. The Predictor API receives the requests from external clients and, according to an internal message protocol, sends a message to the Predictor Message Broker (PMB). PMB forwards the message to the corresponding MPC or MMTC, enabling the multicast distribution of messages, allowing it to build several prediction components for the same metric, each one with a different prediction model.

The core component of the architecture is the MPC. Each instance of the MPC is responsible for predicting the time series data for a given dynamic system/metric. The MPC does online or offline training of the model to update it with the newly received data. For offline training, the training task parameters (date of the next training, training data window, etc.) can be adjusted via the Prediction API. For each new model uploaded, the MPC stores the data to optimize the read and write operations for time-series data, to be used when training the model. To enable an ensemble prediction, it is also needed to perform a reduction step, to join the predictor results and perform the adequate transformation. The Metric Message Transformation Component (MMTC) performs that transformation by receiving the messages in the prediction message broker and applying custom functions for each predictor component messages, before inserting the transformed messages into the broker time-series data, to be used when training the model.

The instances of the MPC and the MMTC are orchestrated by the MPO. The MPO is able to manage the created instances of metric predictors and transformations, ensuring their availability and reliability, restarting them when they

Fig. 9 Macro architecture of the prediction framework



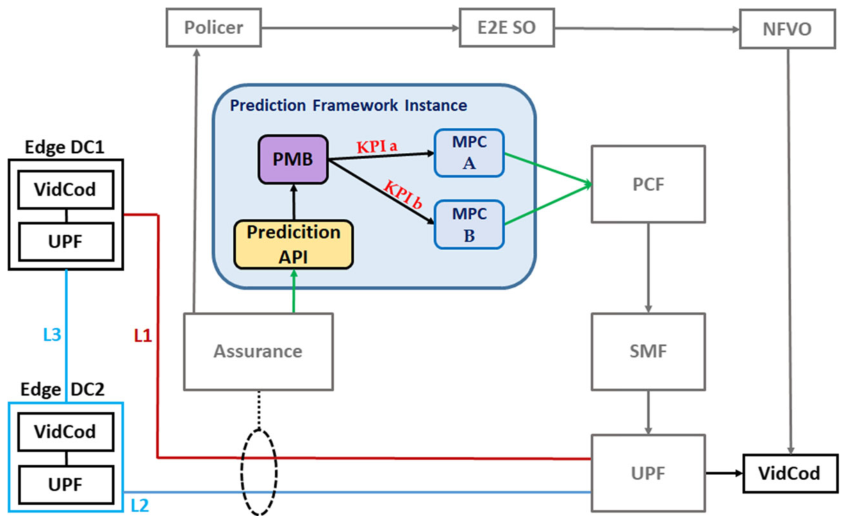


Fig. 10 Anomaly prediction from two KPIs simultaneously

crash. When the instances of metric predictors and transformations start their execution, they establish a connection with the message broker (PMB), to be able to report their activity to the monitoring component. The MPO works in conjunction with Metric Prediction Monitoring (MPM). The MPO can access the logs of the predictor and transformer (the logs are sent to the message broker) and the exchanged messages, to better understand the current state of operation of the various components. The configuration and management module is able to instantiate and manage the life cycle of the predictor and the transformation components. These components can be dynamically added or removed when ordered by MPO.

6.2 The prediction framework in action

The 3rd Generation Partnership Project[9] (3GPP) proposes a 5G system architecture to leverage service-based interactions between the network functions of the control plane [21]. Some of the main components of this reference 5G architecture are the following:

- Policy Control Function (PCF): uses its information about the slice usage (bandwidth, active services, clients, etc.) to manage the slice/network behaviour. PCF provides and maintains the policy rules to control plane functions;
- Session Management Function (SMF): supports session management, selection, and control of user plane functions, downlink data notification, and roaming;
- User Plane Functions (UPFs): handle the user plane path of Packet Data Unit (PDU) sessions. UPF selection is performed by the SMF;
- Network Data Analytics Function (NWDAF): represents the operator managed network analytics logical control, and provides slice specific network data analytics.

These components are directly related to our Predictor Framework (PF). Our PF can be seen as an implementation of NWDAF. This architecture has been implemented in Mobilizer 5G Project,[10] a portuguese 5G project that aims to foster 5G implementation and deployment in Portugal. Our prediction framework has been deployed in this architecture and has been integrated with both Assurance and PCF modules. These modules are implemented in Docker containers, and orchestration, management, and automation of network and edge computing services is performed through ONAP.[11]

Figure 10 depicts a simplified version of the 5G architecture deployed in our premises. To illustrate the functionalities of PF, we exemplify an anomaly forecast potentially caused by two KPIs that can affect QoS/QoE in a use case of a virtual Video Coding scenario (VidCod). Policer, end-2-end Service Orchestrator (E2E SO), and Network Function Virtualization Orchestrator (NFVO) are the entities needed to orchestrate the VidCod services in Edge DC1 and DC2.

The Prediction API of the PF receives both KPIs (KPI a and KPI b) from the Assurance module, and activates two instances of the Metric Predictor Component (MPC) through the Predictor Message Broker (PMB). Each of the instances, MPC A and MPC B, evaluates the received KPIs against the expected behaviour model (learned over time). The *KPI a* evaluated by MPC A could be E-UTRAN radio access bearer (E-RAB), and Circuit Switched Fallback (CSFB) could be used as *KPI b* in MPC B. If any instances of MPC predict that link 1 (L1—red line in Fig. 10) has some type of anomaly, they send a notification to the PCF.

[9]https://www.3gpp.org/

[10]https://5go.pt

[11]https://www.onap.org/

The PCF and SMF do all the preparation for changing the traffic from L1 to L3-L2 (blue lines). The intelligence of this operation lies in the fact that the PF does not recognise anomalies when they are occurring, but rather tries to predict the occurrence of the anomaly in the near future. Upon receiving the prediction, the PCF, considering the current slice status (bandwidth, services, options), can act preventively by changing the slice policy, or even indicating the need for more resources to the orchestrator. The result of this joint operation can prevent the anomaly from occurring, and consequently, maintain the slice's QoS/QoE and consequently the entire network.

Our framework was designed to be easy to use (no requirements on programming knowledge), easy to deploy and can be used in automated deployment workflows, and lightweight to work in slices. The framework is modular in its components, since it is implemented with microservices, and it is able to train and use multiple prediction models simultaneously, an essential characteristic for 5G environments when there are many time series systems.

7 Conclusion

Understanding the causes of events in a network, such as low accessibility, helps the network operators to forecast and avoid them to happen, by adjusting network resources that influence their causes. In this work, the goal was to determine the causes of reduced network accessibility in 4G networks, using only historic data.

Two different analysis were made. Besides analyzing the causes of reduced accessibility in the whole network, it was also analyzed the causes of reduced accessibility for each cell. The results showed that the causes of reduced accessibility for each analysis are very different. While for the overall network, the KPIs that most influence the accessibility are the number of failure handovers, the number of phone calls and SMSs in the network, the overall download volume, and the availability of the cells; the KPIs that most influence the accessibility of each cell are related with the number of users in a cell and its download volume. For a network operator, it is important to know if it is important to monitor low accessibility in a cell, in a network, or in both, to make the right measurements in the network. In addition, we ended our considerations with a discussion of the requirements of 5G networks, where proactivity is mandatory.

As future work, the next step will be to detect the patterns of those KPIs that indicate future low accessibility, to be able to predict it and adapt the network to prevent it to happen. For example, if the network operator knows that the network accessibility will be lower in the next hour when the number of handover failures intrafrequency and the maximum connected users both exceed a threshold, he can take proactive measures to adapt the network and avoid the low accessibility.

Funding This work is supported by the European Regional Development Fund (FEDER), through the Regional Operational Programme of Lisbon (POR LISBOA 2020) and the Competitiveness and Internationalization Operational Programme (COMPETE 2020) of the Portugal 2020 framework (Project 5G with Nr. 024539 (POCI-01-0247-FEDER-024539)), and by FCT/MEC through national funds and when applicable co-funded by FEDER—PT2020 partnership agreement under the project UID/EEA/50008/2019.

References

1. Anusas-Amornkul T (2018) A network root cause analysis and repair system. In: 2018 6th International symposium on computational and business intelligence (ISCBI), pp 69–73. https://doi.org/10.1109/ISCBI.2018.00023
2. Bennacer L, Ciavaglia L, Ghamri-Doudane S, Chibani A, Amirat Y, Mellouk A (2013) Scalable and fast root cause analysis using inter cluster inference. In: 2013 IEEE International Conference on Communications (ICC), pp 3563–3568. https://doi.org/10.1109/ICC.2013.6655104
3. Bennacer L, Amirat Y, Chibani A, Mellouk A, Ciavaglia L (2015) Self-diagnosis technique for virtual private networks combining bayesian networks and case-based reasoning. IEEE Trans Automation Science Eng 12(1):354–366. https://doi.org/10.1109/tase.2014.2321011
4. Chigurupati A, Lassar N (2017) Root cause analysis using artificial intelligence. In: 2017 Annual reliability and maintainability symposium (RAMS), pp 1–5. https://doi.org/10.1109/RAM.2017.7889651
5. Kennedy D (2017) D2.6 Final report on programme progress and KPIs, https://5g-ppp.eu/wp-content/uploads/2017/10/Euro-5G–D2.6_Final-report-on-programme-progress-and-KPIs.pdf
6. Ferreira D, Senna C, Salvador P, Cortesão L, Pires C, Pedro R, Sargento S (2020) Root cause analysis of reduced accessibility in 4g networks. In: Boumerdassi S, Renault É, Mühlethaler P (eds) Machine learning for networking. Springer International Publishing, Cham, pp 117–133. https://doi.org/10.1007%2F978-3-030-45778-5_9, https://link.springer.com/chapter/10.1007%2F978-3-030-45778-5_9
7. Ferreira D, Senna C, Sargento S (2020) Distributed real-time forecasting framework for iot network and service management. In: Fifth IEEE/IFIP International Workshop on Analytics for Network and Service Management (AnNet 2020), IEEE/IFIP Network Operations and Management Symposium (NOMS)
8. Fisher A, Rudin C, Dominici F (2018) All models are wrong but many are useful: variable importance for black-box, proprietary, or misspecified prediction models using model class reliance
9. Gómez-Andrades A, Muñoz P, Serrano I, Barco R (2016) Automatic root cause analysis for lte networks based on unsupervised techniques. IEEE Trans Veh Technol 65(4):2369–2386
10. Gonzalez JMN, Jimenez JA, Lopez JCD, Parada GHA (2017) Root cause analysis of network failures using machine learning and summarization techniques. IEEE Commun Mag 55(9):126–131. https://doi.org/10.1109/mcom.2017.1700066
11. Imoize AL, Orolu K, Atayero AA (2020) Analysis of key performance indicators of a 4G LTE network based on experimental data obtained from a densely populated smart city. Data in Brief

29:105304. https://doi.org/10.1016/j.dib.2020.105304. http://www.sciencedirect.com/science/article/pii/S2352340920301980
12. Khatib E, Barco R, Serrano I (2017) Degradation detection algorithm for lte root cause analysis. Wireless Personal Communications 97. https://doi.org/10.1007/s11277-017-4738-6
13. Maccari L, Passerini A (2019) A big data and machine learning approach for network monitoring and security. Security and Privacy 1. https://doi.org/10.1002/spy2.53
14. Mfula H, Nurminen JK (2017) Adaptive root cause analysis for self-healing in 5g networks. In: 2017 International conference on high performance computing simulation (HPCS), pp 136–143. https://doi.org/10.1109/HPCS.2017.31
15. Muñoz P, de la Bandera I, Khatib EJ, Gómez-Andrades A, Serrano I, Barco R (2017) Root cause analysis based on temporal analysis of metrics toward self-organizing 5G networks. IEEE Trans Veh Technol 66(3):2811–2824. https://doi.org/10.1109/TVT.2016.2586143
16. Pearson K (1901) Liii. on lines and planes of closest fit to systems of points in space. The London Edinburgh, and Dublin Philosophical Magazine and Journal of Science 2(11):559–572. https://doi.org/10.1080/14786440109462720
17. Rai D, Dwivedi A (2018) Lte theory to practice – kpi optimization (a 4g wireless technology). International Journal of Innovative Technology and Exploring Engineering (IJITEE) 8. https://www.ijitee.org/wp-content/uploads/papers/v8i2/B2534128218.pdf
18. Solé M, Muntés-Mulero V, Rana AI, Estrada G (2017) Survey on models and techniques for root-cause analysis. arXiv:1701.08546
19. Yan H, Breslau L, Ge Z, Massey D, Pei D, Yates J (2012) G-RCA: A generic root cause analysis platform for service quality management in large IP networks, vol 20
20. Santos R (2018) Machine learning techniques using key performance indicators for the configuration optimization of 4G networks
21. ETSI: 3GPP TS 123 501 V15.9.0 (2020-03) – 5G; System architecture for the 5G System (5GS) 3GPP TS 23.501 version 15.9.0 Release 15), ETSI, March 2020, https://www.etsi.org/deliver/etsi_ts/123500_123599/123501/15.09.00_60/ts_123501v150900p.pdf