**ATCN - Anomalies in Telecom Cellular Network Performance**

**Modification History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comments** |
| 0.1 | 13-10-2020 | Sai Keerthi Krishna | Initial Version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Problem Statement:**

Telecom Company wants to minimize technical disruptions experienced by their customers is critical to retain competitiveness, especially in the wake of stronger competition. Client wants to detect an array of anomalies in network performance, including hidden ones that do not manifest themselves in explicit downtime.

**Assumption:**

* In this work, we do not consider the correlation across multiple cells. Also, we do not identify the root causes of the anomalies due to insufficient information in our dataset.

The main task of network administrators is to identify any KPI anomalies, which refer to unexpected patterns that occur at a single time instant or over a prolonged time period.

Today’s network diagnosis still mostly relies on domain experts to manually configure anomaly detection rules such a practice is error-prone, labour-intensive, and inflexible.

**Issues:**

1. How should we define useful KPI anomalies that correspond to practical cellular network performance degradation problems?
2. Can we design a unified anomaly detection framework that can incorporate various anomaly detection algorithms and detect various types of anomalies for one or multiple KPIs?
3. Can our anomaly detection framework be automated with limited manual intervention, while still achieving accurate detection?

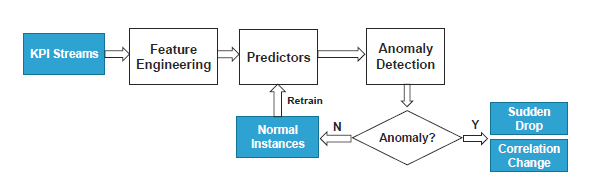
**Value to Client:**

* Increased network availability and utilization
* Minimized and mitigated network outages
* Better Operating & Maintenance planning
* Higher service levels and customer satisfaction.

**Design:**

In the following discussion, we propose a unified framework that can effectively detect both sudden drops and correlation changes.

Our anomaly detection focuses on a per-cell basis by inspecting the time-series instances of multiple KPIs in each cell. It takes the time-series data of multiple KPIs as inputs, and detects both sudden drops and correlation changes with high accuracy by taking into account both seasonality and trend components in KPI time-series data.



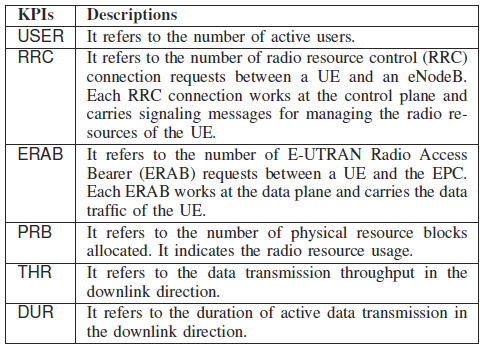
It takes the time-series data of multiple KPIs as inputs, and detects both sudden drops and correlation changes with high accuracy by taking into account both seasonality and trend components in KPI time-series data.

It also provides a feedback loop to incrementally update the prediction models based on the past detection outputs, thereby eliminating the manual efforts of specifying labelled data (i.e., ground truths) for model training.

In particular, it takes into account both seasonality and trend components in KPI time-series data, and provides a feedback loop for retraining the prediction models using prior detection results to improve detection accuracy.

The six types of KPIs address the cellular network performance in three aspects:

1. user population (i.e., USER),
2. radio resource usage (i.e., RRC, ERAB, and PRB), and
3. data transmission load (i.e., THR and DUR).



UE User Equipment

eNodeB evolvedNodeB

EPC Evolved Packet Core

<https://yatebts.com/documentation/concepts/lte-concepts/>

**Data Set:**

Network administrators deploy probes in the EPC and every eNodeB to periodically collect KPI values, which will be sent to a centralized network management system (NMS).

We call each collected input an instance, which specifies the time and value for a KPI. In this work, we collected per-cell KPI instances from the NMS of an operational LTE network

deployed in a metropolitan city in China.

Each instance is recorded on an hourly basis and describes the performance of a cell in the latest hour. We consider six types of KPIs.

Our KPI dataset covers three collection periods for a total of 17 weeks: (i) November 7, 2016 to January 8, 2017, (ii) February 13, 2017 to March 12, 2017, and (iii) April 10, 2017

to May 7, 2017. We only select the cells that have the complete KPI data over the entire 17 weeks;

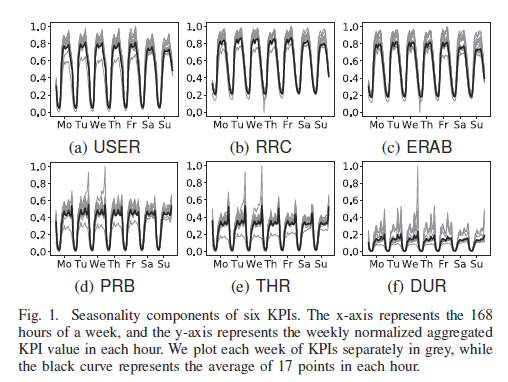


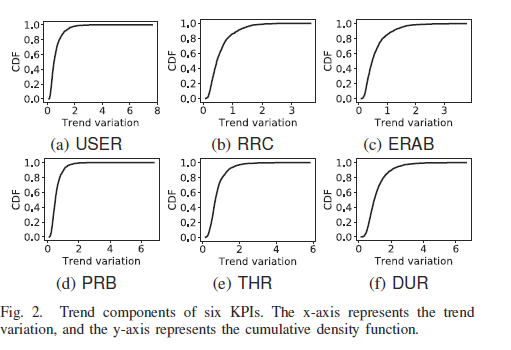
We first examine the statistical properties of our collected dataset, so as to understand the behaviours of the cellular network. Our observations are summarized as follows:

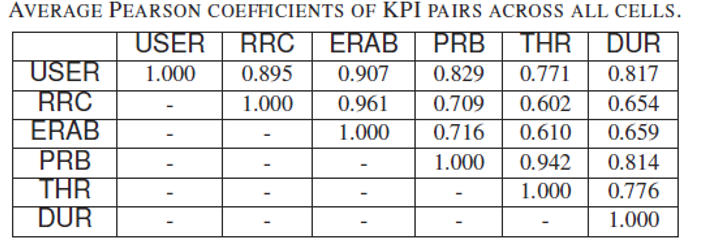
(i) there exist strong seasonality and trend components in the dataset;

(ii) some KPIs are strongly correlated; and

(iii) there exist non-negligible variances in KPI values across the same hour of different days.







We observe all six KPIs have positive linear correlation. In particular, the pairs (RRC, ERAB), (PRB, THR), and (USER, RRC) are the top-3 pairs with the strongest correlation.

**Predictors**

CELLPAD supports three families of predictors:

* simple statistical modelling
* linear regression, and
* tree-based regression

The latter two belong to machine-learning-based regression approaches. Each predictor returns a predicted value for each hour based on the underlying prediction algorithm.

****

We consider two types of anomalies that are of practical interest to cellular network management based on our internal communication with network administrators:

1. **sudden drops**, which indicate the unexpected degradations of a KPI, and
2. **correlation changes**, which indicate the inconsistency between the current and historical correlations of two correlated KPIs.

Both sudden drops and correlation changes are complementary to each other in characterizing performance anomalies of cellular networks. In practice, if either one of the KPI anomalies persists for a prolonged period (e.g., a few hours), it may indicate the presence of network failures and requires network administrators to investigate further.

**Sudden Drops:**

A sudden drop refers to the sudden performance degradation of a KPI instance within a cell. For example, if there exists a sudden drop in USER, it may imply that a cell fails to provide connectivity to a significant portion of users. In general, a sudden drop happens when a KPI value is significantly less than the expected one.

• In sudden drop detection, random forest regression with trend removal achieves high PRAUC using different features, although some simple statistical modelling algorithms such as EWMA and WMA can also achieve high PRAUC.

**Correlation Change:**

It refers to the large deviation of two correlated KPI instances within a cell. For example, a cell failure may increase the number of RRC request attempts (i.e., RRC), while the number of active users (i.e., USER) remains relatively un-changed.

• In correlation change detection, Huber regression without trend removal achieves the highest PRAUC across all predictors.

CELLPAD builds on regression analysis to predict the normal values of KPI instances in order to detect anomalies.

***At a high level, CELLPAD takes multiple time-series streams of KPI instances at different time intervals (hours in our case) as inputs. It first performs feature engineering to extract a set of features, whose values are derived from the KPI instances that are observed up to the current hour.***

The feature values serve as inputs to different predictors, each of which performs a specific prediction algorithm and outputs a predicted KPI value, which is the expected value for a KPI at each hour in normal situations (i.e., without anomalies).

For sudden drop detection, CELLPAD returns one predicted KPI value for each KPI instance being considered.

For correlation change detection, CELLPAD returns two predicted KPI values for each pair of KPI instances being considered.

CELLPAD performs anomaly detection based on the prediction at each hour by checking the deviations between the actual and predicted KPI values. It concludes that the current KPI instances are either anomalies (i.e., sudden drops or correlation changes) or normal instances.

For the latter case, CELLPAD also feeds back the normal instances to retrain the prediction models for improved detection accuracy.

**How to call different algorithms?**

• For **DropController**, "predictor" can be "RF", "RT", "SLR", "HR", "WMA", "EWMA", "HW".

• For **ChangeController**, "predictor" can be "RF", "RT", "SLR", "HR","LCS".

**How to perform feature selection?**

• "feature\_types" can be a subset of ["**Numerical**", "**Indexical**"]

• "feature\_time\_grain" can be a subset of ["**Hourly**", "**Daily**", "**Hourly**"]

• "feature\_operations" can be a subset of ["**Raw**", "**Mean**", "**Median**", "**Wma**", "**Ewma**"]

**How to remove any trend components?**

* DropController(to\_remove\_trend=True, trend\_remove\_method="center\_mean")
* ChangeController(to\_remove\_trend=False, trend\_remove\_method="center\_mean")
* "to\_remove\_trend" is **True** or **False** to indicate whether to remove the trend or not.
* "trend\_remove\_method" is "**center\_mean**" or "**past\_mean**".
* "center\_mean": the trend at time i is the mean of the points in [i-84, i+83].
* "past\_mean": the trend at time i is the mean of the points in [i-167, i].

Note that the accuracy of "past\_mean" is slightly less than that of "center\_mean" in general, as the latter considers the time interval closer to time i.

**sudden drop**

cd ./example

python example\_drop.py

**correlation change**

cd ./example

python example\_change.py

**Simple statistical modelling:** CELLPAD implements four algorithms:

• **EWMA (Exponentially Weighted Moving Average)**: It computes the predicted value based on the weighted values of a set of instances, such that the weights are exponentially decayed for older instances.

**• WMA (Weighted Moving Average):** Its prediction is also based on the weighted instances as in EWMA, except that the weights are linearly decayed.

**• HW (Holt-Winters):** It is a triple exponential smoothing method that extends EWMA to deal with seasonality and trend. It computes the predicted value as a function of the weighted inputs of both instances as well as the seasonality and trend components. It also estimates the seasonality and trend components from the instances using EWMA.

• **LCS (Local correlation score):** It measures the correlation of two time-series. It holds two synchronous sliding windows to compute the auto-covariance matrices continuously and then aggregates the matrices using their exponentially weighted averages. We mainly use LCS for detecting correlation changes.

**Linear regression:** CELLPAD implements two linear regression algorithms to model the linear relationships between features and predicted values:

**• SLR (Simple linear regression):** It computes the predicted values based on the optimal linear combination of the values of a feature that can minimize the mean square deviation.

• **HR (Huber regression):** It enhances simple linear regression to be robust against noise, by controlling whether instances are classified as outliers via an epsilon parameter (a smaller epsilon is more robust to outliers). For example, how Huber regression excludes outliers from modelling as opposed to simple linear regression.

**Tree-based regression:** To model the non-linear relationships between features and predicted values, CELLPAD also implements two tree-based regression algorithms:

**• RT (Regression tree):** It organizes the feature space into a tree structure, in which each non-leaf node is a

decision-making process that splits the feature space based on a selected feature, while each leaf node holds a local predictor that averages all instances that fall into the feature partition.

**• RF (Random forest):** It is an ensemble learning algorithm. It samples different subsets of instances and features to form multiple regression trees and take their average prediction result. It is robust against irrelevant features and noises than a single regression tree in general.

**Anomaly Detection**

To perform anomaly detection, we first calculate the degree of deviation.

For **Sudden Drop Detection**,

CELLPAD computes the drop ratio **D = (KPI**a**−KPI**p**)/KPI**p

where KPIa and KPIp denote the actual and predicted KPI values, respectively.

**If D is much less than 0, it likely implies a sudden drop.**

We consider the metric PRAUC (Area Under Precision-Recall Curve), which is shown to be robust when the distributions of normal instances and anomalies are highly imbalanced. Here, we use the drop as the prediction input to PRAUC, which computes various precision and recall pairs against different thresholds to obtain an accuracy measure between 0 and 1 (higher means more accurate).

To detect **Correlation Changes** of two KPIs (say, KPI1 and KPI2), CELLPAD computes the change ratio

for KPI1 by **C1 = (KPI1**a**−KPI1**p**)/ KPI1**p,and that

for KPI2 by **C2 = (KPI2**a**−KPI2**p**)/KPI2**p

where KPI1a and KPI1p (resp. KPI2a and KPI2p) denote the actual and predicted KPI values of KPI1 (resp. KPI2), respectively.

We use PRAUC as the accuracy metric. We use the average of two absolute change ratios

**1/2 (|C1|+|C2|)** as the input to PRAUC. Here, we focus on the KPI pairs (USER, RRC).

