On Non-parametric Models for Detecting Outages in the Mobile Network

Eric Falk, Ramino Camino, Radu State
SnT, Interdisciplinary Centre for Security, Reliability and Trust
Luxembourg, Luxembourg
{*eric.falk,ramino.camino,radu.state*}@*uni.lu*

Vijay K. Gurbani
Bell Laboratories, Nokia Networks
Naperville, Illinois, USA
vkg@bell-labs.com

***Abstract*—The wireless/cellular communications network is composed of a complex set of interconnected computation units that form the mobile core network. The mobile core network is engineered to be fault tolerant and redundant; small errors that manifest themselves in the network are usually resolved automatically. However, some errors remain latent, and if discovered early enough can provide warnings to the network operator about a pending service outage. For mobile network operators, it is of high interest to detect these minor anomalies near real-time. In this work we use performance data from a 4G-LTE network carrier to train two parameter-free models. A first model relies on isolation forests, and the second is histogram based. The trained models represent the data characteristics for normal periods; new data is matched against the trained models to classify the new time period as being normal or abnormal. We show that the proposed methods can gauge the mobile network state with more subtlety than standard success/failure thresholds used in real-world networks today.**

I. Introduction

In the second quarter of 2016, the wireless mobile network in the US supported 410 million devices, and these numbers are expected to continue on an upward trajectory. The packet based 4G-LTE network provides the infrastructure support for these devices. The two out of many network components of interest for this paper are: the eNodeB (Evolved Node B) and MME (Mobility Management Entity). User devices access the eNodeB over the radio network; the eNodeB is a base station that controls the mobile devices in one or more geographical areas. The MME is the primary node in the core network that controls important aspects related to mobile devices: it communicates with the eNodeB to establish radio communication channels with the mobile devices, it is responsible for user authentication, and for finding the mobile endpoint. In fact, the MME acts as a central orchestrator that controls the mobile device; as it communicates with the mobile device and the core network, it logs these communications as events in a specific format, called the *Per Call Measurement Data* (PCMD) format. A PCMD record contains about 250 fields, the most important for our work are the fields that contain the messages exchanged between the MME and other core entities, to provide services to the mobile device, and how long it took the message exchanges to complete.

Problem Statement: MME logs are commonly monitored by network operators, but not in real-time. In most cases the log files are consulted offline by domain experts after a network outage has occurred, in order to determine the magnitude and cause of the outage. Furthermore, anomalies of short duration are of minor impact, are often hidden by the automatic network resilience mechanisms, and may not raise alarms. These resilience mechanisms may mask anomalies that have the potential to forecast a large outage. With greater complexity in the networks, it becomes crucial to perform network health measurement real-time, with a model sensitive to the minimal shifts of the mobile network state.

Contributions and summary of results: Our previous work [6] studied a mix of models, parametric (Gaussian Mixture Model) and non-parametric (χ^2); here we focus exclusively on the latter. Our assumption remains that in the mobile network few anomalous procedures happen during normal periods, and that the network's resiliency mechanisms allow it to perform some useful work even during outages. However, such mechanisms may mask an impending outage; we expect our work to detect even the minor perturbations that could be used to signal the degradation of the network. The proposed approaches in this work are based on *Isolation Forests* (*IF*) [8, 9] and on *Histogram-Based Outlier Scores* (*HBOS*) [5]. Using the data from a normal period, a model of normality is trained and subsequently used to predict new incoming data. The analysis of two datasets from recorded network outages, shows that both *IF* and *HBOS* are able to accurately detect the state of the network, and in fact are resilient in the face of an outage that went undetected for 20 minutes using traditional mechanisms that network operators use. The rest of this paper is structured as follows: In Section II related work on anomaly detection and mobile network data is surveyed. Section III describes how *IF* and *HBOS* are used with PCMD data. In Section IV both approaches are evaluated on two real-world PCMD log data-sets from recorded network outages. Section V concludes the paper.

II. Related Work

Anomaly detection is applied in a variety of areas from finance to health care. A survey can be found in [3].

978-3-901882-89-0 @2017 IFIP

Anomaly detection methods for communication networks based on unsupervised learning are presented in [11]. The initial challenge of anomaly detection is to define what constitutes the anomaly [1, 3]. In cases in which anomalies are not clearly defined, a system can be trained to detect data outliers taking values far outside the expected intervals. In such cases histogram based methods as *HBOS* or *IF*, can be employed [1]. An example of anomaly detection using *HBOS* is given in [5], a use case employing *IF* is described in [13]. For our intended application of gauging the mobile network state in real-time both techniques are a good fit. They are robust against data noise, they can be applied to uni-variate data, and the evaluation of novel data is fast, which is mandatory for real-time monitoring.

For mobile networks in particular, a widely studied topic has been the investigation of human mobility patterns using Call Detail Records (CDR) [10, 2, 7]. PCMD records can also be employed for the purpose of locating users, as demonstrated in [4]. In [12] the information drawn from PCMD records is used to efficiently pinpoint the position of a mobile device, improving mobile paging. To our best knowledge this work is the first to consider PCMD data to detect shifts in the mobile network state in real-time.

III. Methodology

A common advantage of *HBOS* and *IF* is that the inclusion of anomalies in the training stage interfere very little in the accuracy of detection. This feature is very convenient for our experiments, because instances in our dataset are not individually labeled as anomalies or non-anomalies. Another property of both methodologies is that they detect anomalies even when they are clustered (masking effect) [8, 9, 5], something that happens when a lot consecutive errors emerge during mobile network outages. Finally, both techniques are capable of handling datasets that exhibit long-tail distributions, as it is the case with PCMD records.

The Dataset: MMEs produce a PCMD log file in 1 minute time intervals and each minute produces in the range of 300K-800K records. The dependability of mobile networks implies a small number of anomalous procedures per unit time, while the fault tolerant algorithms of the network further masks these anomalous procedures. The PCMD attributes of relevance for this work are the procedure identifier (*ProcID*), the procedure duration (*ProcDur*), and the procedure fault code (*FaultCode*). Working with domain experts, we have identified *ProcID* 11 (Pid_{11}) and 16 (Pid_{16}) to be of particular interest. Pid_{11} corresponds to a User Equipment (*UE*) bearer release request and Pid_{16} to the *UE* paging request. Our dataset is composed of a set of observations, S, where each element $s_i \in S$ is produced by the MME every minute. Each s_i contains q records. More concretely, each s_i corresponds to a PCMD file with $1..q$ records. For training the models, we divide S into S_{norm}, 6 minutes of data from a normal period; S_{train}, is composed of a separate set of 6 minutes from a normal period and used for threshold determination in *IF*; and S_{out}, composed of 6 minutes of data corresponding to an outage. The data is univariate and measures the time between network components sending requests and getting back responses (ProcDur).

Isolation Forest: Using different sub-samples IF builds an ensemble of binary trees by randomly splitting the value ranges of each sub-sample; anomalies are those instances with a short average path length on the ensemble of trees in the *IF* [8, 9]. *IF*, in essence, isolates anomalies. *IFs* define an anomaly score, s, of an instance such that $0 \leq s \leq 1$: if instances return s very close to 1.0, they are anomalies. If instances have s much smaller than 0.5, they can be regarded as normal.

There are two parameters to the IF algorithm: t, or the number of trees (default: 100) and ψ, or sample size (default: 2^8). We achieved good performances with default t, but we found out that leaving ψ at a default of 2^8 leads to underfitting during training and failed to generalize well. ψ of 2^5 proved to be the right fit, as anything less than that would run the risk of overfitting. In the evaluation stage, an anomaly score s is derived for each record in the PCMD log file. Here we faced a dilemma because we are not interested in scoring each PCMD record, rather, we are interested in characterizing the entire minute represented by the PCMD log file. To solve this problem we derived scores s_i for each record in the PCMD file and derived an overall anomaly ratio, ar, as follows:

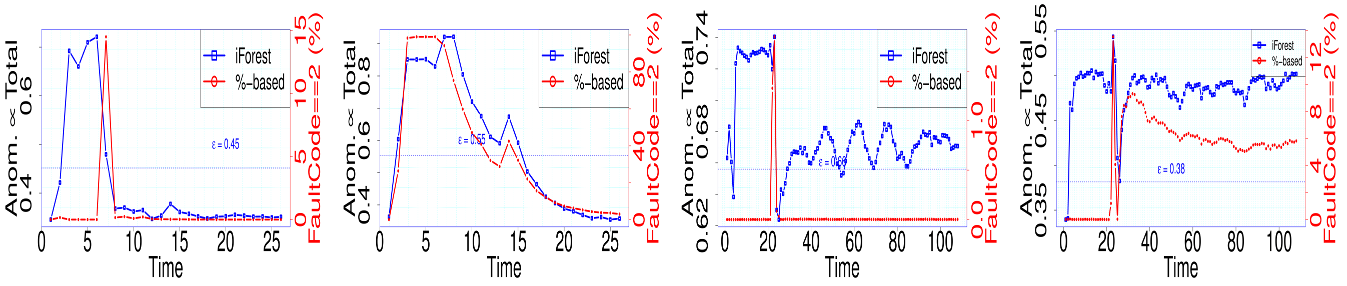
$$ar = (\sum_i score(s_i))/total\ records \tag{1}$$

where $score(s_i)$ returns 1 if the record with score s_i is anomalous and 0 otherwise. For the training of the *IF* model S_{train} was split: 4 minutes dedicated to the actual training phase, and 2 minutes for the threshold determination. The latter is designated as S_{thres}. To set the threshold ϵ, n equally sized samples are generated from S_{thres} and S_{out}. Every measure x of the samples is evaluated to obtain the anomaly score for the measure. The ratio is computed for each of the n samples. ϵ is then determined by:

$$\epsilon = (min(ar_{normal}) + max(ar_{outage}))/2 \tag{2}$$

For validation the trained *IF* models were evaluated using the S_{norm} and S_{out}.

Histogram-based Outlier Score: *HBOS* [5] is an unsupervised anomaly detection algorithm based on histograms. Anomaly detection methods based on histograms rate novel data by assigning it to histogram bins. Data items assigned to a bin estimate the degree of normality: A large bin represents frequent items and is therefore presumed to contain normal data, whereas small bins contain outliers. *HBOS* uses a dynamic bin-width strategy: feature values are sorted first



(A) Pid_{11} Outage-1 (B) Pid_{16} Outage-1 (C) Pid_{11} Outage-2 (D) Pid_{16} Outage-2

Figure 1: Network performance measurement for the *IF* model for Pid_{11} and Pid_{16} on Outage-1 and Outage-2.

and a fixed amount of $\frac{N}{k}$ successive values are assigned to a single bin, where N is the number of total instances and $k = \sqrt{n}$ is the number of bins. *HBOS* makes an exception for long-tailed distributions of the type exhibited by *ProcDur* by allowing more than $\frac{N}{k}$ values in the same bin. The final bin widths are normalized using the number of elements in the bin, the total amount of records, and the value range covered by the underlying bin. The normalized bin width is then used as a score for the evaluation of novel data. The score is expressed via a color coding: the most normal score taking a green color, while the most abnormal bin is red.

The training phase uses S_{train} to build the *HBOS* model, which we validate using minutes from S_{norm} and S_{out}. As with *IF* we seek to determine whether an entire minute is normal or anomalous. To that extent, we extract the average of all observations that occur in a minute after they have fit the model and used it to determine the outcome of that particular minute. The outcome will be color coded with gradients from the familiar $[green, red]$ range.

IV. Evaluation

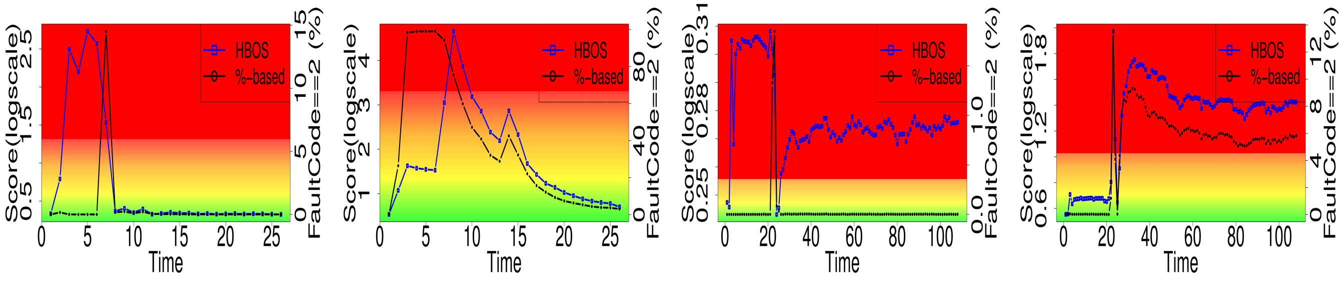
We use two real-world outages, Outage-1 and Outage-2, produced by two distinct MMEs serving different geographical areas in the US. Outage-1 occurred abruptly and spans a time period of 25 minutes; the network transitions abruptly from normal to an outage at minute 2, and then it gradually recovers. Outage-2 spans 108 minutes, and during the whole time period the network remained unstable. An important aspect of Outage-2 uncovered by our models was that while the peak of the outage occurred at minute 20, the network was constantly trending towards an outage even during the first 20 minutes.

Models have to be trained on the MME whose data they will analyze; models do not generalize well across different MMEs. This is because the distributions of the $ProcDur$ across each MME is distinct and includes artifacts related to how each network operator configures their core network and the expected traffic on the network. Models are trained according to the description given in Section III.

Isolation Forest: The results of Outage-1 for *IF* are in Figure 1. The second y-axis contains a %-based model (red curve). The %-based model is used in production networks to trigger a warning when the number of PCMD records with fault code value of 2 (error) are observed. This is a very coarse model that largely fails to detect outages, only detecting them when the network behaves extremely. Furthermore, it does not contain the nuances of the *IF* and *HBOS* as we discover below. Nonetheless, in the absence of better models, %-based models are used widely in service-provider networks.

Each Panel in Figure 1 also contains a horizontal line, which corresponds to our threshold, ϵ (c.f., Equation 2). Points below the threshold are normal and those above are anomalous. For Outage-1, *IF* correctly detects the first two minutes as normal, and then the abrupt outage starts at minute 3. The network starts to return to normality at around minute 8-10, as is reflected in Panels 1A and 1B. The outage remains undetected by the %-based system until the peak occurs at around minute 6, after that, the %-based system simply assumes that the network has recovered. Clearly, this is not the case as the network has *mostly* recovered but some variations are still occurring as shown by the *IF* curve. There is a difference in how quickly the network is tending towards normalcy between Panels 1A and 1B; this has to do with the procedure represented by the PIDs. PID_{11} implies that the UEs are releasing bearer sessions in response to the outage, and most of those have been released by minute 10. On the other hand, PID_{16} is used to establish new bearer sessions, thus the move towards normalcy is gradual after the peak as UEs establish new sessions. As more UEs establish bearer sessions, it takes less for remaining UEs to do so, and hence the gradual downward slope. The %-based model performs better for PID_{16} than it does for PID_{11}, which implies that it depends strongly on the PID being tracked. Our models, by contrast, do not.

Panels 1C and 1D track Outage-2. Our models show that the outage builds up in the first 20 minutes and reaches a critical point at around minute 20. The %-based model is not able to detect the outage until minute 20. This is important because the network operator can take proactive actions with a 20 minute advance warning.



(A) Pid_{11} Outage-1 (B) Pid_{16} Outage-1 (C) Pid_{11} Outage-2 (D) Pid_{16} Outage-2

Figure 2: Network performance measurement for the *HBOS* model for Pid_{11} and Pid_{16} on Outage-1 and Outage-2.

Histogram Based Outlier Score: Unlike *IF*, *HBOS* does not have a threshold; instead, evaluating test data for outage is relative since the results are presented on a color scale that goes from green to red. Minutes in between can be calibrated according to the preferences of the operator to be conservative or aggressive. In the context of mobile networking where the core is running on an infrastructure of highly dependable nodes, it may be justified to consider every slight deviation as anomalies, although such an interpretation may be too aggressive.

The graphs for Outage-1 (2A and 2B) are as expected. The first minutes are characterized as normal and then the outage occurs and exacerbates before stable state is reached. The downward slope of PID_{11} and PID_{16} tracks the equivalent slopes of the *IF* model for Outage-1 (Panels 1A and 1B). Because there is no specific threshold in *HBOS* the first couple of minutes fall in the yellow zone on the graphs of Panels 2C and 2D, thereby making them suspect.

Summary: Overall the *IF* and *HBOS* models give a more nuanced representation of the network state than the existing percentage model. For Pid_{11} the %-based models registered a peak of anomalous procedure terminations late into the outage periods. The amount of Pid_{11} requests drops during outage periods. For this reason less errors are registered by the percentage model, which is why it is less accurate as the two proposed methods also considering $ProcDur$. For Pid_{16} there exists a symmetry between the threshold technique and the *IF* and *HBOS* predictions. The conclusions are twofold: (a) During outages the amount of Pid_{16} requests increases because of retries. They imply previously failed requests which are captured by the percentage model, improving its accuracy. (b) Unlike the %-based model, which depends on the particular PID, *IF*/*HBOS* are independent of the specific PID. Thus they are more general than the %-based model and accurately reflect the state of the network by tracking the Procedure Duration distribution instead of raw counts on how many times a procedure was invoked.

V. Conclusion and Future Work

We have presented novel approaches to detecting outages in a mobile network using non-parametric anomaly detection methods. To run the models only few attributes are required from the dataset. The *IF* and *HBOS* approaches perform better than the percentage-based model, which only responds to the peak of the outage, thereby making it ineffectual for detection until the outage occurs. The models we used, by contrast, are able to characterize discrete states of the network: stable, trending towards outage, and then trending back towards stable.

Future work includes building large-scale analytic pipelines that use these real-time models to not only detect, but also predict the state of the network. We are exploring how to evolve the models so they can be complemented by the evaluation of additional data to contextualize the network state.

References

[1] Charu C. Aggarwal. *Outlier Analysis*. Springer New York, 2013.

[2] Richard A. Becker et al. "Route Classification Using Cellular Handoff Patterns". In: *Proceedings of the 13th International Conference on Ubiquitous Computing*. New York, NY, USA: ACM, 2011.

[3] Varun Chandola, Arindam Banerjee, and Vipin Kumar. "Anomaly detection: A survey". In: *ACM computing surveys (CSUR)* (2009).

[4] M. J. Flanagan et al. "Wireless network analysis using per call measurement data". In: *Bell Labs Technical Journal* (2007).

[5] Markus Goldstein and Andreas Dengel. *Histogram-based Outlier Score (HBOS): A fast Unsupervised Anomaly Detection Algorithm*.

[6] Vijay K. Gurbani et al. "Detecting and predicting outages in mobile networks with log data". In: *Under submission at IEEE International Conference on Communications, ICC*. 2017.

[7] Sibren Isaacman et al. "Identifying Important Places in People's Lives from Cellular Network Data". In: *Pervasive Computing*. Springer Berlin Heidelberg, 2011.

[8] F. T. Liu, K. M. Ting, and Z. H. Zhou. "Isolation Forest". In: *2008 Eighth IEEE International Conference on Data Mining*. 2008.

[9] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation-Based Anomaly Detection". In: *ACM Trans. Knowl. Discov. Data* (2012).

[10] Gyan Ranjan et al. "Are Call Detail Records Biased for Sampling Human Mobility?" In: *SIGMOBILE Mob. Comput. Commun. Rev.* (2012).

[11] Marina Thottan, Guanglei Liu, and Chuanyi Ji. "Anomaly detection approaches for communication networks". In: *Algorithms for Next Generation Networks*. Springer, 2010.

[12] Hui Zang and Jean C. Bolot. "Mining Call and Mobility Data to Improve Paging Efficiency in Cellular Networks". In: *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*. New York, NY, USA: ACM, 2007.

[13] Guang-Tong Zhou et al. "Relevance Feature Mapping for Content-based Image Retrieval". In: *Proceedings of the Tenth International Workshop on Multimedia Data Mining*. Washington, D.C.: ACM, 2010.