Nhu Trang, Nguyen Ngoc; Truong, Linh

# Context-aware, Composable Anomaly Detection in Large-scale Mobile Networks

*Published in:*
Proceedings - 2023 IEEE 47th Annual Computers, Software, and Applications Conference, COMPSAC 2023

*DOI:*
10.1109/COMPSAC57700.2023.00032

Published: 01/01/2023

*Document Version*
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

*Please cite the original version:*
Nhu Trang, N. N., & Truong, L. (2023). Context-aware, Composable Anomaly Detection in Large-scale Mobile Networks. In H. Shahriar, Y. Teranishi, A. Cuzzocrea, M. Sharmin, D. Towey, AKM. J. A. Majumder, H. Kashiwazaki, J.-J. Yang, M. Takemoto, N. Sakib, R. Banno, & S. I. Ahamed (Eds.), *Proceedings - 2023 IEEE 47th Annual Computers, Software, and Applications Conference, COMPSAC 2023* (pp. 183-192). ( Proceedings : International Computer Software & Applications Conference). IEEE.
https://doi.org/10.1109/COMPSAC57700.2023.00032

# Context-aware, Composable Anomaly Detection in Large-scale Mobile Networks

Nguyen Ngoc Nhu Trang*
Daienso Lab, Vietnam
nhutrang.nguyen@daienso.com

Hong-Linh Truong
Aalto University, Finland
linh.truong@aalto.fi

***Abstract*—In a large-scale mobile network, due to the diversity of data characteristics, detection purposes of operation teams, and analytics and machine learning algorithm abilities, building big data anomaly detection pipelines without considering different analytics and team situations may not yield expected quality of analytics, including detection relevancy, performance and quality. This is especially for analytics subjects, such as mobile network zones, of which characteristics are dynamic and contextual. Moreover, due to the lack of labeled data and the high cost of creating labeled data, building anomaly detection analytics models based on (supervised) deep learning or advanced models is even more challenging from various aspects of effort, cost and deployment. In this paper, we present a novel framework that enables anomaly detection through context-aware, composable components to provide efficient detection pipelines suitable for lightweight, resource constrained and geographical operation teams. First, we identify and categorize different types of analytics feature contexts and evaluate existing algorithms suitable for these contexts, mapping anomaly detection algorithms, patterns and configurations for data pre-processing and unsupervised detection tasks in individual analytics functionality. These context-specific pipelines detect anomalies and their relevancy for dynamic analytics subjects such as mobile network zones. Then we develop dynamic configuration and combination techniques for such pipelines to produce highly relevant, multi-context detection of anomalies. Our framework provides flexibility and configurations for team contexts to carry out the anomaly detection in the team's operations. We will demonstrate our work through real data gathered for a large-scale mobile network covering multiple types of sites with different geographical zones and equipment. We especially focus on district zones and user-defined zones as analytics subjects that must be managed by teams in our experiments.**

## I. Introduction

There have been many products as well as research results for anomaly detection in mobile networks in the telco domain that focus on detecting abnormalities in individual network elements such as mobile cells [1], monitoring devices [2], or procedure signalling [3]. However, we face numerous challenges when applying these products and research to our network, including (i) a centralized, integrated anomaly detection system across a large number of mobile sites/cells (approximately 30 thousand cells in our focus) could result in a waste of resources as well as inaccurate and irrelevant types of anomalies for dynamic analytics subjects like mobile network zones; (ii) the suitability/flexibility of existing detection algorithms have not been evaluated at different levels (zones with different sizes based on geographical or network equipment) and types of network measurements for operation and business purposes; (iii) the required different types of network measurements data may not be available at the analytics time and do not have the same timelineness suitable for analytics because of the delay (could be 2–3 hours) in the collection of raw data, preventing the utilization of deep anomaly algorithms with voluminous data; and (iv) the relationship between anomalies detected from different types of data for supporting complex operations is not well-researched.

Solving these issues requires examining the role of multiple types of contexts in anomaly detection due to the variety of businesses, scale, and capabilities of mobile networks in different geographical zones. The current approaches for anomaly detection in mobile networks seek the best algorithms for detecting anomalies for atomic analytics subjects (such as site, cell or customer). For collective, dynamic subjects, such as mobile zones, a detection system using these algorithms cannot cope well with *the context of data*, e.g., different data types, granularity, and timeliness, and with *the context of analytics subject*, e.g., domain knowledge in supporting composition of detection to identify context-relevant anomalies. In this paper, we focus on detecting abnormalities related to performance of dynamic mobile network zones. Given analytics subjects as dynamic zones, we investigate and categorize different data types (alarms and radio network measurements) from various data sources and time periods for different *analytics feature contexts* under *the context of team operations*, due to the characteristics of data and anomalies. We evaluate and apply different algorithms/pipelines suitable for individual analytics feature contexts, based on algorithms abilities for anomaly detection with our data, judged by the domain expert. After that, we combine multi-context anomaly detection based on *the context of team operations* to draw the anomaly situations associated with analytics subjects, producing accuracy detection from multiple views on the same subject. Using this way, we tackle the problem of huge data integration (requiring resources and may cause data quality and may not be suitable for our team constraints), empowering decentralized anomaly detection and lightweight algorithms to be deployed under the governance of teams working without powerful machine learning infrastructures. This paper contributes: (i) a systematic model of contexts for anomaly detections for dynamic

* Work was carried out when the author was with MobiFone Corporation, Vietnam

zones, (ii) an evaluation and mapping of anomaly detection algorithms and suitable pipelines for analytics feature contexts, and (iii) flexible, context-aware composition techniques of different analytics feature contexts for team operation contexts.

We apply our method for anomaly detection without labeled data and provide extensive experiments and examples based on real data from a large-scale mobile network of 12 provinces with approximately 7000 mobile sites and 30000 mobile cells. For the rest of this paper, Section II presents background, motivation and our approach. Section III presents our methods and framework. Experiments and examples are given in Section IV. We discuss related work in Section V. Section VI concludes the paper and outlines future activities.

## II. Background, Motivation and Approach

### A. Background on mobile network operations

For operating Radio Access Networks (RANs) in our mobile networks (also similar to other networks), the management of service quality of dynamic zones is divided into many levels corresponding to the business purposes and operation teams:

- *station/site level*: a mobile station/site at a known location provides the connectivity between mobile devices and the network. One site contains many cells and can serve hundreds of consumers within its coverage area. The number of sites for a zone depends on the network planning. Sites can also include one or more types of technology (e.g., 3G, 4G, and 5G) with equipment from many different vendors (e.g., Nokia, Ericsson, and Huawei).
- *zone level*: a zone is made up of numerous sites based on geographical zoning (such as districts/provinces), specific business purposes (such as business customers in developing markets), or common technical control management components (such as a common transmission node for 4G sites). Pre-defined zones have been identified by the telco operator based on administrative conditions. We call "user-defined zones" for other zones that can be identified based on different parameters during the analytics (such as an area surrounding a point of interest).

We focus on anomaly detection used for the purposes of network operating, planning, optimizing, and finding root causes of performance problems in mobile network zones. Thus, anomaly detection should be aware of the diversity and variety of expected performance in different locations.

### B. Motivating Examples

The quality of services for customers is dependent on the quality of sites in the connected zone. Based on various reports about the services (SMS, voice, or data) and their situations, error times, and geographical locations, the engineer must identify/be informed if there are any anomalies in the alarms and network measurements at the error time (and before), and must determine root causes based on these anomalies. The detection of these anomalies should be performed for dynamic zones. Zone-based analytics are also used by the operator for network planning, network management, and business purposes (such as categorizing market regions). Furthermore, as illustrated in Fig. 1, with many types of anomalies detected by different algorithms, identifying the anomalies of interest to the team operation is an important need.

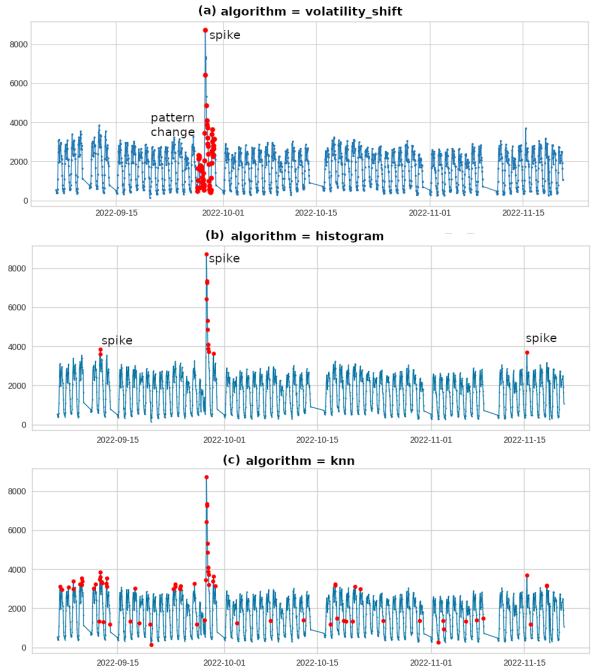



Fig. 1. An example of using different anomaly detection algorithms to detect anomalies in an analytics subject: both the volatility_shift in (a) and the histogram in (b) can detect spikes at `2022-09-28`. The volatility_shift algorithm can detect pattern changes before spikes, providing earlier warnings. But it cannot detect the other two spikes while the histogram one can. Many anomalies or change types in (c) are irrelevant to the team.

Our goal is to identify, at runtime, relevant anomalies for operational and business needs. However, anomalies for dynamic zones have no precise definition and fixed constraints in the view of operations in a large-scale network. They are based on specific analytics subject contexts, such as characterized by the type of business or operation and the constraints and expectations at different times in different locations. Anomalies are also characterized by different properties that may indicate different utilizations of anomaly detection. Therefore, the major research questions are: (i) which algorithms are suitable for which contexts, (ii) is the combination of different algorithms better than the combination of data for single algorithms, and (iii) which types of anomalies are of interest to the team?

### C. Approach

We should note that in practice, especially in our work, the following challenging issues exist:

- we have voluminous data but no labeled data. However, the correctness of any detection from historical data could be verified by human experts. There is also no complete view of all real anomalies occurred, but only known ones.
- many anomaly detection algorithms exist but their suitability for the telco domain must be evaluated according to many contextual constraints.
- *the context of data* includes complex data currencies, timelineness and other data properties in mobile networks

in which some types of data are real-time, while others have delays but are valuable for anomaly detection.

As a result, anomaly detection pipelines must be constructed and tailored based on the best types of data, capable of adaptability and high accuracy for dynamic analytics subjects. To address the above-mentioned goals, our approach is to (i) identify contexts and data for context-specific anomaly detection, (ii) define constraints associated with zone-based analytics subjects and criteria for evaluating anomaly detection algorithms, (iii) evaluate existing anomaly detection algorithms' accuracy for near-real-time telco data at different zone-based analytics subjects, and (iv) build flexible and composable zone-based anomaly detection models. Fig. 2 depicts our approach to the anomaly detection.

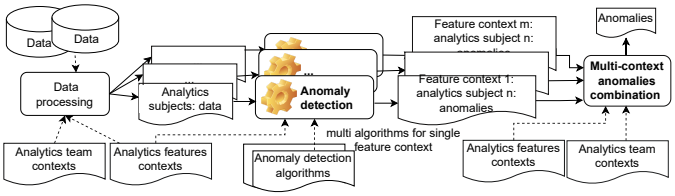



Fig. 2. Combine separated context-based detection for relevant anomalies.

## III. Framework Methods and Design

### A. Classifying contexts in analytics and team operations

Anomaly detection results need to ensure accuracy and timeliness to support the operation and optimization of the network. Each team concentrates on its own analytics subjects and cares about specific anomalies. For example, the site engineer needs to monitor the network's availability for customers at any time. Therefore, the requirement of the anomaly detection system is to warn about the spike (decrease) in the network's availability and related alarms. Given an analytics subject, its behavior and anomalies are contextual. For example, consider a mobile zone, depending on the type of markets (key, potential or developing market regions), the type of technologies, and the location of the zone, we have different business constraints and different types of infrastructure (e.g., equipment from Ericsson vs from Nokia); these factors reflect *the context of the analytics subject* and such a subject must be analyzed according to its context. Besides capturing the contexts of data and of analytics subjects, we introduce a systematic view of two other types of contexts for anomaly detection based on the purpose and structure of the operations, illustrated in Fig. 3:

- *analytics feature contexts*: reflect situations of the detection from the perspective of data analytics, based on different types of data that can be analyzed together in a meaningful way to detect anomalies, given existing algorithms. Features are "analytics functionality" and are divided due to the variety of input data, algorithms, deployment, and potential applications.
- *analytics team contexts*: reflect the team's constraints on the analytics for certain purposes. Analytics teams will have their own context w.r.t. time, performance, accuracy, etc., for the analytics.

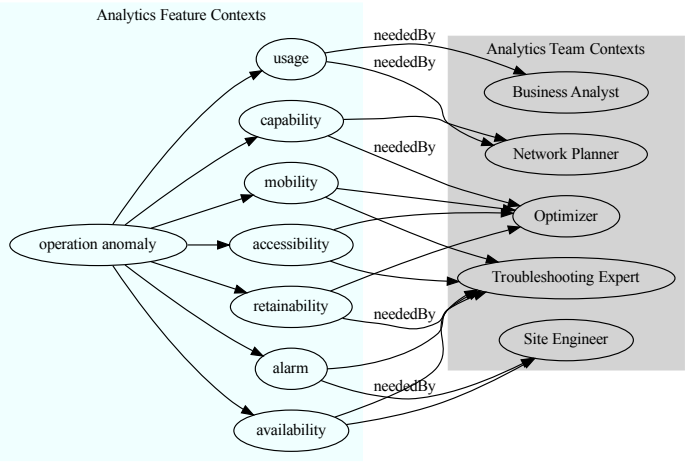



Fig. 3. Current categories of analytics feature contexts and team contexts.

Analytics feature contexts are associated with team contexts to provide suitable analytics of anomalies for different situations. Table I explains current categories of analytics feature contexts and corresponding characteristics and purposes. An *analytics feature context* will be identified by a context type and include the following main information:

- *fields of data*: are required for the analytics
- *constraints*: include *conditions* associated with a *time window*, ceiling thresholds, *a type of analytics subject*, etc. Given a type of feature context, the team will interpret anomalies detected for an analytics subject within the constraints, as anomalies are not the same for all subjects at all time (the context of analytics subjects).

A team context will include information about possible operations and which analytics feature contexts are needed for which operations, as well as expected performance constraints for the operation (e.g., response time of the detection, accepted accuracy, maximum costs and infrastructural resources) that help them decide which algorithms/pipelines they should use.

The context of data is captured via data services and observability, whereas the context of analytics subjects is defined based on domain knowledge. We develop current analytics feature contexts and team contexts based on the telco KPIs [4] and the best practices of operations in our networks. However, our contexts are extensible as one can identify and add new contexts and define constraints for the newly-created contexts. The way to organize the analytics feature and team contexts also allows us to move to a new way of analytics with decentralized data and anomaly detection deployment in the edge. Furthermore, it enables "data as a product" principles [5] in which analytics features are mapped to required data and analytics that would deliver the anomaly outcomes for the team, which needs to handle the anomalies.

### B. Mapping data for context-specific anomaly detection

For each analytics feature context, we need to map and obtain suitable data. Our data includes radio network measurements data for radio network performance and alarm data

| Analytics feature context: detection purposes | Feature context characteristics for anomalies | Team operation |
|---|---|---|
| usage: analytics of the network traffic (SMS, voice, and data traffic) | customer behavior and their demand for using the services (data and voice) | business and network planning |
| alarm: analytics of multi-layered components faults reported from semi-structured realtime alarms | hardware/software component faults from many points and layers that affect various services, correlating to anomalies in other contexts | operations and troubleshooting |
| capability: analytics of traffic bandwidth and throughput at the network layer | (automatic) hardware/software upgrade/downgrade and provisioning configurations affecting quality of customer experiences | network planning and optimization |
| availability: analytics of the network availability for customers using the SMS, voice, and data services | hardware faults, power failures, or transmission issues in a zone causing the unavailability and performance degradation of the services | operations and troubleshooting |
| accessibility: analytics of the radio link connection establishment among user devices and networks | the connection signaling between devices and network components in the setup procedure (attempts, failures, success rates) | optimization and troubleshooting |
| mobility: analytics of the handover and location switching of the network mobility function | the ability to handle continuous services for the devices in a zone (intra-site, inter-site, inter radio access technologies) | optimization and troubleshooting |
| retainability: analytics of the stability of the network while customers use the services | the reliability of service connections during service consumption (failure/drop causes and drop rates) | optimization and troubleshooting |

TABLE I

DETECTION PURPOSES AND DATA DIVERSITY ARE REASONS FOR SEPARATING DIFFERENT TYPES OF ANALYTICS FEATURE CONTEXTS

for network failures. These types of data are delivered via data services, including near-real time data streaming systems based on multiple industrial network management systems. Network measurements include raw data collected from cells, or processed data for sites. Alarm data is collected for cells, sites and other components in the network. Both radio network measurements and alarm data are in timeseries. Based on our intensive domain knowledge and the evaluation of anomaly detection algorithms, we classify and map types of data suitable for contexts. Table II gives data fields according to different feature contexts and records examples of data used in testing our anomaly detection approach. Listing 1 gives a simplified structure of configuration that is used to obtain and pre-process suitable data, including the context of data (quality and time constraints), for suitable analytics feature contexts.

```
{
    "context_type": "ctx_usage",
    "feature_context": {
        "analyticsubject": {
            "subject_fieldname": "DISTRICT",
            "constraints": [
            ]
        },
        "required_data_fields": [
            "TRAFFIC4G"
        ],
        "time_constraints": {
        },
        "dataquality": {
        }
    },
}
```

Listing 1. An example of configuration for obtaining data for a context

### C. Relating feature contexts to anomaly patterns

Existing algorithms detect different types of abnormalities, which may follow different common, well-documented patterns of data changes, such as spikes (increase, decrease, and both sides), level shifts, or pattern changes [6], [7], as illustrated in Fig. 4. Given data selected for feature contexts, an algorithm can detect many patterns of anomalies. However, these anomalies are from the algorithm viewpoint based on data and do not necessarily reflect the anomalies that the team needs in the team context, given the context of the analytics subject. A team wants to obtain only important and relevant anomalies for its activities. Fig. 5 and Fig. 6 give two different examples of anomaly detection, of which not all found anomaly patterns are relevant to a team.

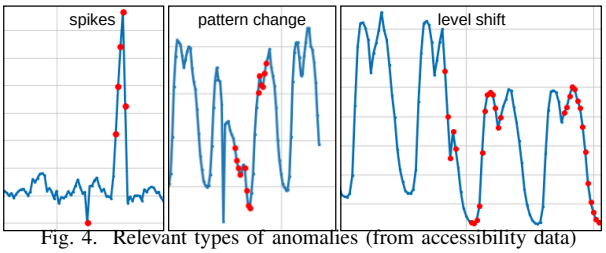



Fig. 4. Relevant types of anomalies (from accessibility data)

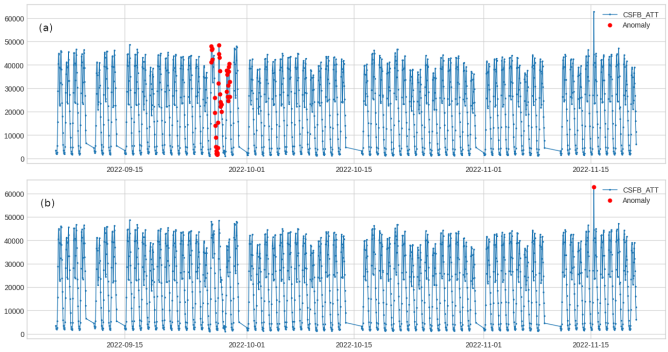



Fig. 5. An example of anomaly detection for accessibility context based on CSFB_ATT data using different algorithms (level_shift in (a) and Quantile in (b)) resulting in different patterns of anomalies: pattern change and spike. While accessibility context is used by the optimization team, it is necessary to detect all types of anomalies to support the team in doing multiple tasks such as finding root causes and optimizing network performance and quality

To address the relevancy of patterns in specific contexts, we have conducted a human-in-the-loop approach that evaluates different algorithms and techniques to detect abnormalities in sample datasets from network measurements and alarm data. The results are then reviewed by domain experts to determine what types of abnormalities need to be monitored and in which situations and contexts they can be used. The upper part of Fig. 7 presents steps w.r.t. the evaluation of algorithms and patterns. Mapping between context and patterns will be used to choose suitable algorithms and pipeline deployments. Table III presents identified patterns associated with contexts.

### D. Evaluating detection algorithms for contexts

Identifying and evaluating the suitability of anomaly detection algorithms for the contexts in Fig. 7 are performed

| Analytics feature context types: Description of example data | Data: record example |
|---|---|
| usage: hourly time series of traffic TRAFFIC4G: the total traffic of all 4G cells in the zone | ('DATE' 'DISTRICT' 'TRAFFIC4G'): ('2022-09-06 09:00:00' '****' '1286.79716') |
| alarm: real-time time series alarm: including duration, type of alarms, etc. by window time, alarm starttime (SDATE) and endtime (EDATE) | ('VENDOR' 'SITE' 'CELL' 'NETWORK' 'SDATE' 'EDATE' 'ALARM_TYPE' 'SEVERITY' 'ALARM_NAME'): ('****' '****' '****' 'RAN_4G' '2022-09-01 00:30:00' '2022-09-01 00:32:49' '1' 'A3' '61631') |
| availability: hourly time series of availability calculated from the rate of total serving time per hour of all 4G cells in the zone | ('DATE' 'DISTRICT' 'AVAILAIBILITY'): ('2022-09-07 10:00:00' '****' '99.67') |
| capability: hourly 4G downlink throughput (THP_DL) as the rate of successful message (THP_VOL_DL) per time (THP_TIME_DL) of all 4G cells in the zone combined with 4G downlink traffic | ('DATE' 'DISTRICT' 'THP_DL' 'TRAFFIC4G_DL'): ('2022-09-07 10:00:00' '****' '16459.69' '1196.95') |
| accessibility: hourly time series of access attempt (RRC_ATT, E_RAB_ATT, CSFB_ATT), calculated as the total attempt of all 4G cells in the zone | ('DATE' 'DISTRICT' 'RRC_ATT' 'E_RAB_ATT' 'CSFB_ATT'): ('2022-09-06 09:00:00' '****' '1202554' '1079454' '45026') |
| mobility: hourly data of handover attempt (INTERRAT_HO_ATT, INTRAFEQ_HO_ATT), calculated as the total inter-rat and intra-frequency attempt of all 4G cells in a zone | ('DATE' 'DISTRICT' 'INTERRAT_HO_ATT' 'INTRAFEQ_HO_ATT'): ('2022-09-06 09:00:00' '****' '3128' '442321') |
| retainability: hourly data of failure (ERAB_ABNORMAL), calculated as the total abnormal release of all 4G cells in zone | ('DATE' 'DISTRICT' 'ERAB_ABNORMAL'): ('2022-09-06 09:00:00' '****' '3053') |

TABLE II

EXAMPLES OF DATA USED AND DESCRIPTION. **** MASKS SENSITIVE DATA

| Context type | Anomaly patterns | Reason for considering relevant anomalies in team operations |
|---|---|---|
| usage | spike (both sides) | the sudden increase in a wide scale due to large traffic loads may cause the service congestion or the sudden decrease may indicate a large-scale problem; they need to be troubleshooted |
| | level shift in a window | capture a long-term change in a wide zone that necessitates monitoring to determine whether more sites are required for deployment or if capacity needs to be upgraded or downgraded |
| alarm | spike of increase in a window or high severity faults | the sudden, severe or broad disruption in a large scale affecting zones indicates severe problems in core elements or common parts of the network, not individual problems of sites or single elements |
| capability | spike of decrease | a sudden wide congestion in a zone that affected network performance that must be resolved |
| | level shift in a window | they are usually related to a change in network capacity (degrade) that leads to a repeated anomaly that we need to analyze the influence of configuration change on services and customers |
| availability | spike of decrease | the sudden decrease of network availability in a wide zone for a period of time that affected customers using the services, which requires engineers to identify the root cause and fix the problem as soon as possible |
| accessibility, mobility, retainability | spike of decrease in success rate/attempt or spike of increase in attempt | they indicate a wider network change or failure (such as a transmission issue, synchronization loss, or parameter change) affecting the customer experience in a wide zone (for example, the customers will be unable to connect to the network, calls will be dropped continuously) |
| | level shift or pattern change in a window | they indicate a repeated (negative) behavior change in a wide zone that requires an optimization team to evaluate root causes influencing network performance, services, and customer experience |

TABLE III

EXAMPLES OF FEATURE CONTEXTS AND ANOMALY PATTERNS FOR ZONES THAT USUALLY INDICATE THE PERFORMANCE IN A LARGE-SCALE SETTING

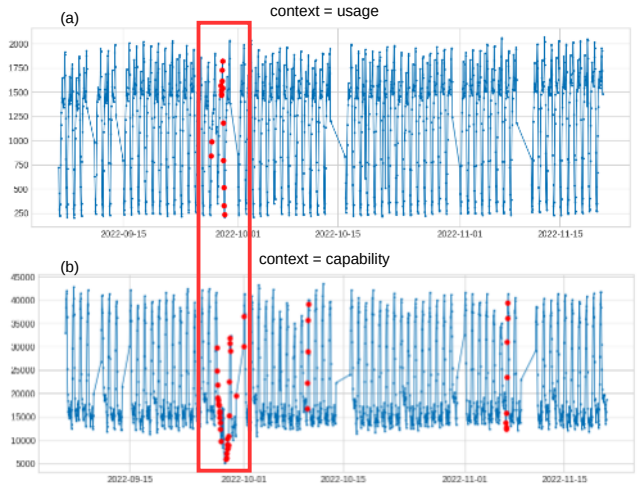



Fig. 6. An example of anomaly detection for (a) TRAFFIC4G data (usage context) and (b) DL_THP data (capability context) using volatility_shift algorithm from ADTK [6]. The algorithm shows anomalies in both contexts at the same time that had the same root cause. However, the pattern change anomalies in TRAFFIC4G data had no value to the business analyst team, which only considered an spike (increase) or level shift anomaly. A spike (decrease) in DL_THP data, which could affect the customer data service, should be raised with the optimization team to quickly find root causes.

together with the identification of contexts and anomaly patterns (discussed before). We select and evaluate many common algorithms from existing frameworks, such as the level_shift, volatility_shift, Interquartile, Autoregression, Histogram, SVM, KNN, Isolation Forest, Local Outlier Factor (LOF), Spectral Residual (univariate), PCA, MinclusterDetector, OutlierDetector (multivariate) for different contexts. They are selected based on well-known evaluation about their accuracy [8], [9], [10], [11], [6], easy-to-use, and deployment possibility in constrained resources in different locations. Furthermore, for a given analytics feature context, the following aspects must be considered: (i) no labeled data, (ii) data quality problems, (iii) varied data sizes (e.g., years of daily data or months of hourly data), and (iv) software requirements.

Since there is no labeled data, to evaluate unsupervised results we perform two steps. For each algorithm, the output anomaly detected is evaluated based on historical data through a validation by humans/experts to check correct and wrong results. However, we do not know how many real anomalies, given a set of data. Second, we compare differences among anomalies detected by algorithms for a given context. By doing so, for each analytics feature context, only a subset of suitable algorithms is selected, shown in Table IV as an example.

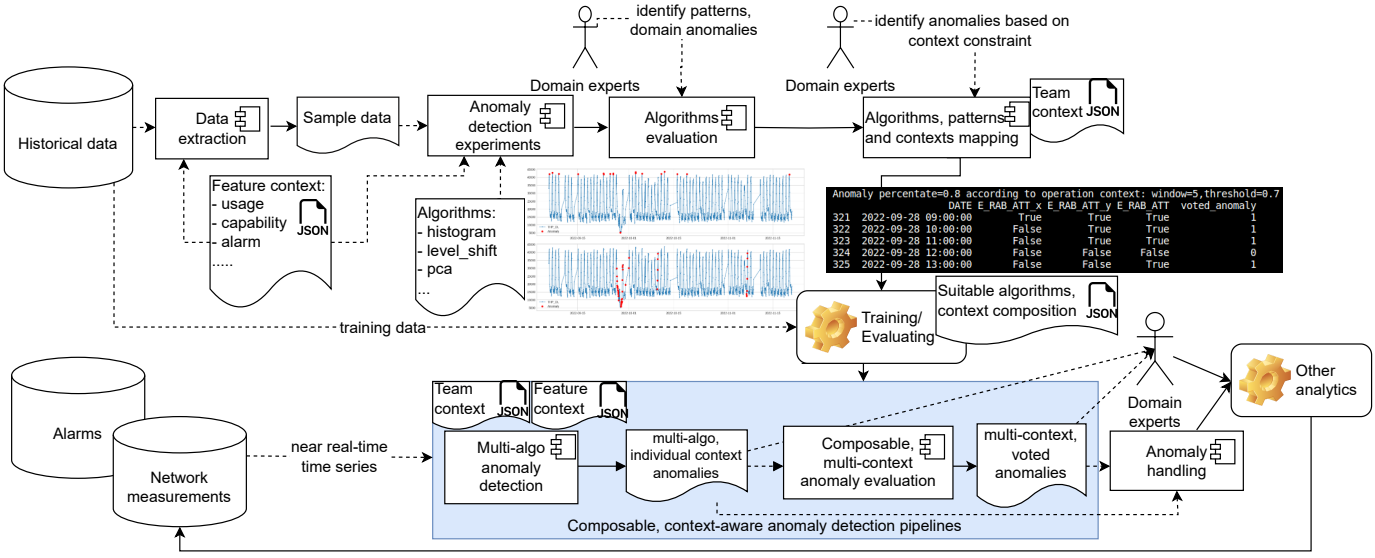


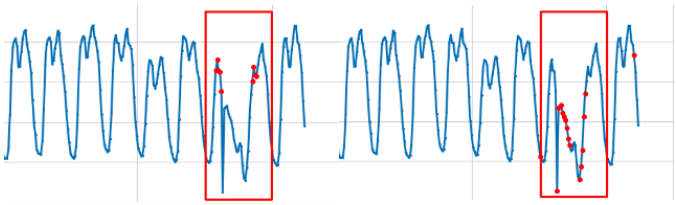

Fig. 7. Composable, context-based anomaly detection using domain experts, multi algorithm pipelines and multi context detection

| Market type | Analytics context type | (Anomaly pattern, [selected algorithms]) |
|---|---|---|
| Key market region | alarm | (spike, [interquartilerange, knn]) |
| | capability | (spike, [pca, outlierdetector]) |
| | availability | (spike, [histogram]) |
| | accessibility | (spike, [histogram, spectral residual]), (pattern change/level shift, [autoregression, volatility_shift, level_shift]) |
| Developing market region | capability | (spike, [pca]) |
| | accessibility | (spike, [histogram, spectral residual]), (pattern change/level shift, [autoregression, volatility_shift, level_shift]) |
| Potential market region | accessibility | (spike, [histogram, spectral residual]), (pattern change/level shift, [volatility_shift, level_shift]) |

TABLE IV
EXAMPLES OF ALGORITHMS FOR ANALYTICS FEATURE CONTEXTS

### *E. Enabling composable, operation-aware detection using multiple algorithms and multiple contexts*

Based on our tests, we have seen that no single algorithm would be suitable for an analytics feature context. Furthermore, depending on the resource constraints and the expected quality of analytics imposed by a team context, the way to choose algorithms for an analytics feature context is different. Therefore, one of the key novelties of our work is to combine different algorithms for an analytics feature context and to combine different pipelines of different analytics feature contexts for a single analytics subject, given a team context (carried out within the block *Composable, context-aware anomaly detection pipelines* in Fig. 7). The key idea in our work is to carry out two levels of combinations. First, we vote for a pattern type: the results from all contexts (based on the algorithms used in the analytics feature contexts) are voted based on type of patterns. Then, another vote is performed among voted anomaly patterns. Thus, our framework can provide various anomaly patterns and scores. The team can configure a suitable one for the team's purposes. Essentially, the technique is "voting" and checking the fitness of voted anomalies against the expectations specified by the team contexts. However, unlike voting in machine learning ensembles, our work uses several types of domain knowledge for anomaly outputs detected by different pipelines.



(a)level_shift (b) autoregression

Fig. 8. Different anomalies detected by two algorithms

*1) Combine multiple algorithms within a single context:* Fig. 8 present two anomaly outputs from two algorithms for a single subject within a single analytics feature context. They provide hints for a real anomaly in the view of the team, but show different patterns. When an expert examines these patterns, Fig. 8(a) provides a better detection w.r.t. the anomaly times whereas Fig. 8(b) shows the change better. However, both cannot tell the window of the anomalies as the expert wishes. In this case, the anomaly window would be ideal based on the combination of two results. In practice, due to configuration, we may have both algorithms or only one of them deployed. Therefore, the best way is to combine them in a flexible way to produce accurate anomalies for the team. In our work, we allow the configuration of different algorithms for a specific analytics feature context. This is done by selecting suitable algorithms for a context (based on Section III-D) and specifying them in the analytics (within the component *Multi-algo anomaly detection* in Fig. 7). Listing 2 presents a structure used to configure the algorithms used (combined with the data selection based on Listing 1).

```
"parameters": {
   "volatility_shift": {
        "c": 1.5,
        "side": "both",
        "window": 21
```

```
    },
    "level_shift": {
        "c": 2.0,
        "side": "both",
        "window": 24
    },
    "spectralresidual": {
        "threshold":3.0,
        "window_amp":24,
        "window_local":24,
        "n_est_points":24
    },
}
```

Listing 2. An example of specifying algorithms for a context

Context-based input data for different algorithms is automatically preprocessed to match the required format of specific algorithms. Outputs from different algorithms will be post-processed to create a unified anomaly output representation, allowing the integration of a wide range of algorithms with different input data and output anomaly formats. The results from multiple algorithms can be used separately but teams can specify configurations to vote/combine the anomaly results from different algorithms. When voting multiple algorithms of the same context, we will vote based on type of anomaly pattern, before providing an overall vote. Listing 3 illustrates configurations for contexts, patterns and algorithms used for voting. An example of a vote for a single context based on types of anomaly patterns and overall vote is:

```
level_shift,histogram,volatility_shift,spectralresidual,
     voted_anomaly_spikes,voted_weight_spikes,voted_ano
maly_pattern_changes,voted_weight_pattern_changes,
     voted_anomaly,voted_weight
1.0,0,1.0,0.0,0.0,0.0,1.0,0.66,1.0,0.5
0.0,0,1.0,0.0,0.0,0.0,1.0,0.33,1.0,0.25
```

```
"potential_market_region": {
    "accessibility": [
    {
    "pattern_name": "spikes",
    "window": "1 hour",
    "algorithms": [ "histogram", "spectralresidual"]
    },
    {
    "pattern_name": "pattern_changes",
    "windows": "5 hours",
    "algorithms": [ "volatility_shift", "level_shift"]
    }
    ]
},
```

Listing 3. Example of feature contexts and patterns linked to subject contexts

*2) Combining results from different analytics feature contexts to detect suitable anomalies for a team context:* Algorithms produce different anomaly results for a subject (as partially illustrated in Fig. 8) in a given context. These anomalies reflect a view (from the given context) in which they are detected. Thus, they may not be necessarily considered "anomalies" for the subject in the team context, as a team context may examine the anomaly from multiple analytics feature contexts. A further step (implemented in the component *Composable, multi-context anomaly evaluation* in Fig. 7) is used to combine anomalies from different algorithms based on domain knowledge. In general, given an analytics subject $as$, a set of anomaly outputs $AO$ from multiple feature contexts will be as an input of a function $f(t_{ctx}, as, AO)$ to determine real, relevant anomalies for the team context $t_{ctx}$.

## IV. Experiments and Applications

We have implemented our prototype based on Python. For the data preprocessing, we use a combination of Spark, Pandas and Dask. We use common detection algorithms provided by Pycaret[1], ADTK [6], and Alibi Detect[2].

### A. Experiment settings

We used historical network measurements and alarms data from September 6 to November 20, 2022 to analyze analytics subjects as pre-defined district zones (based on district-level municipality) and user-defined zones (based on GPS data and H3 distance[3]). For network measurements data, which is cell level in hourly time series raw data, we have 136 districts and six analytics feature contexts. For alarm data, which is at all network levels in real-time time series raw data, the size of data for each alarm context at the district level is based on the number of alarm records. We initialized with 9 districts among 136 districts from 3 types of market regions to get our first views about algorithms and contexts. Then, we added more districts (48 districts) for our experiments on anomaly detection. The market regions (a factor of the analytics subjects context) are divided by the operator based on traffic volume and the number of customers in each district. For user-defined zones, we choose 11 zones with different numbers of sites (from 5 to 20 sites per zone) and different coverage purposes (such as industrial zones and tourism zones). Our experiments are carried out with different types of resources (laptops and workstations) and execution time is not the key for the evaluation (e.g., time and algorithm-level accuracy are extensively covered in [8], [9], [10], [11]). Thus, we present how the detection provides relevant anomalies for team operations.

### B. Sensitivity of algorithms and contexts for different zones

When a network issue happens (such as constantly dropping calls), relevant network measurements associated with the issue could result in spikes, which are an important pattern for troubleshooting and finding root causes.

*1) Sensitivity in pre-defined district zones anomaly:* As an example, Fig. 9(a) and the top sub-figure in Fig. 9(b) show different algorithms for subject `Nha Trang` in which histogram is the best for detecting spikes. Then we test the histogram algorithm for other subjects of different types of market regions in the accessibility context in Fig. 9(b), the results confirm that histogram can detect the most spikes which the experts can use to find out the relations between these spikes and operation issues (such as the number of attempts to access the network increases or decreases suddenly due to the change in the number of customers according to the travel times or because of a system or hardware fault). However, in the usage context in Fig. 9(c), the histogram produces wrong anomalies for the key market regions and irrelevant anomalies for the potential market regions for the team.

[1]https://pycaret.org/
[2]https://github.com/SeldonIO/alibi-detect
[3]https://h3geo.org/

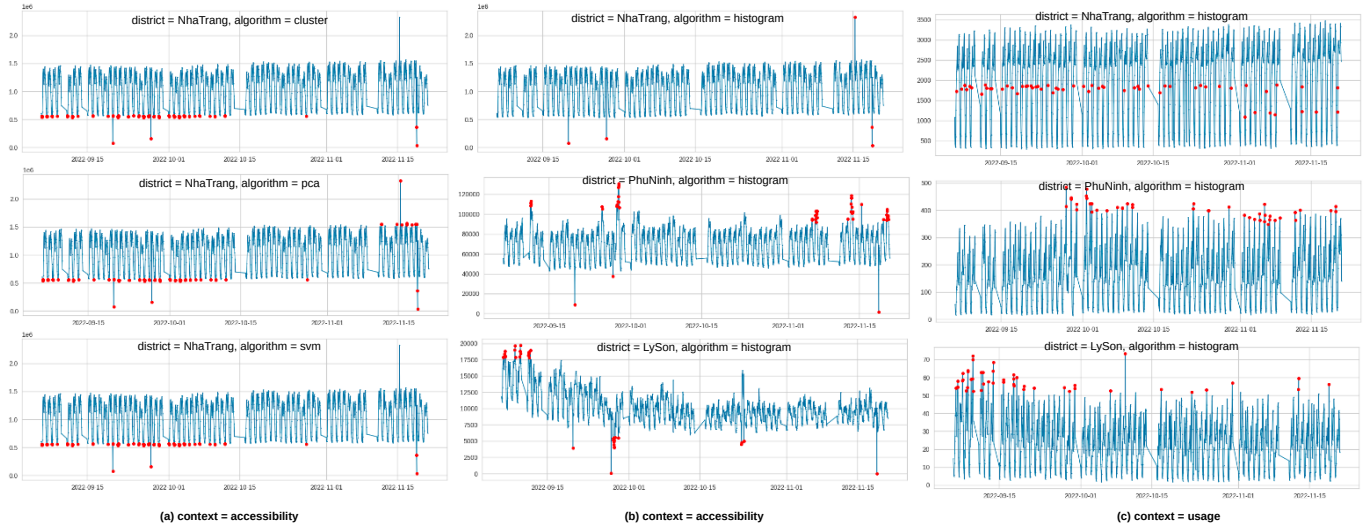



Fig. 9. Results when we applied different algorithms in a zone in the key market region (a) and applied the histogram algorithm to different contexts (accessibility and usage) in different zones in different market regions to detect spikes (b) and (c)

*2) User-defined zone and pre-defined district zone anomaly:* We carry out the same sensitivity tests (using histogram algorithm to detect spikes for accessibility context) for user-defined zones, compared with district zones. As one example, a user-defined zone, within in a district zone, serves a village in a tourist area. The anomalies illustrated in Fig. 10 show that the same anomalies at the same times detected in both district and user-defined zones in (a) and (b) signal the same problems and root causes in a large scale. Thus, engineers should identify the root cause at critical, common components of networks. However, when the user-defined zone (b) does not have the anomaly patterns as the district zone (a) does, the reason for (a) anomalies should come from others, not the sites in the user-defined zone (b). Besides, anomalies in user-defined zones (b) that cannot be detected in the district zones (a) help site engineers identify issues specific in user-defined zones to find problems to solve the related customer feedback.

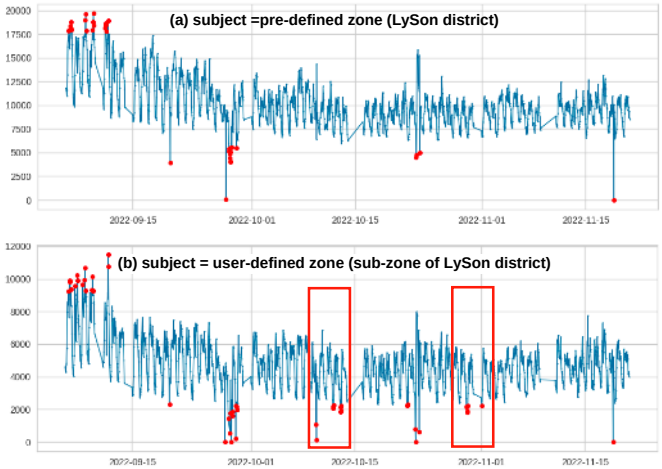



Fig. 10. Differences between patterns detected by the histogram algorithm for accessibility context: more spikes found in user-defined zone (b)

### C. Exploring multi-algo, single-context detection

In many situations, multi-algo pipelines must be used to provide highly relevant anomalies. Fig. 11 shows our experiment in many types of market regions for detecting level shift and pattern change anomalies. Such anomalies have to be considered in a window of time (hours, days, or weeks) according to the *analytics subject and team contexts*. Using a single algorithm we cannot detect all relevant anomalies. However, by combing different algorithms with a suitable window defined by the domain we can detect changes earlier and more completely. Furthermore, changes in pattern or level always have a greater impact than spikes due to longer anomalies. As a result, timely relevant anomalies would assist the operator in troubleshooting, avoiding waster of time and effort due to irrelevant anomalies given by a single algorithm.

### D. Context-aware repeated anomaly patterns

Repeated anomaly patterns may occur in different subjects. Without understanding the context, it is hard to know the reasons and suitable optimization. We look at two cases.

*1) Repeated anomaly across multiple subjects indicates system-wide problems:* usually, due to context differences, different analytics subjects would have different anomalies, as our focus is on the performance of the zone level. However, if they have a common, repeated anomaly at the same time window, e.g. `2022-11-18 19:00:00` for different zones of vendor `A` shown bellow:

```
zone_type,vendor,E_RAB_ATT,histogram_anomalyresults
pre-defined district zone 1,A,809,1
pre-defined district zone 2,A,2369,1
pre-defined district zone 3,A,445,1
user-defined zone 4,A,0,1
user-defined zone 5,A,0,1
pre-defined district zone 6,B,1377083,0
user-defined zone 7,B,6351,0
```

this might signal a system-wide problem. We see that they happened in all zones related to a common radio equipment vendor `A`, and anomalies are only in the accessibility context. Thus, we must check monitoring systems for vendor `A`.

*2) Repeated anomalies for individual subjects:* in this case, the repeated patterns happened in many user-defined zones as

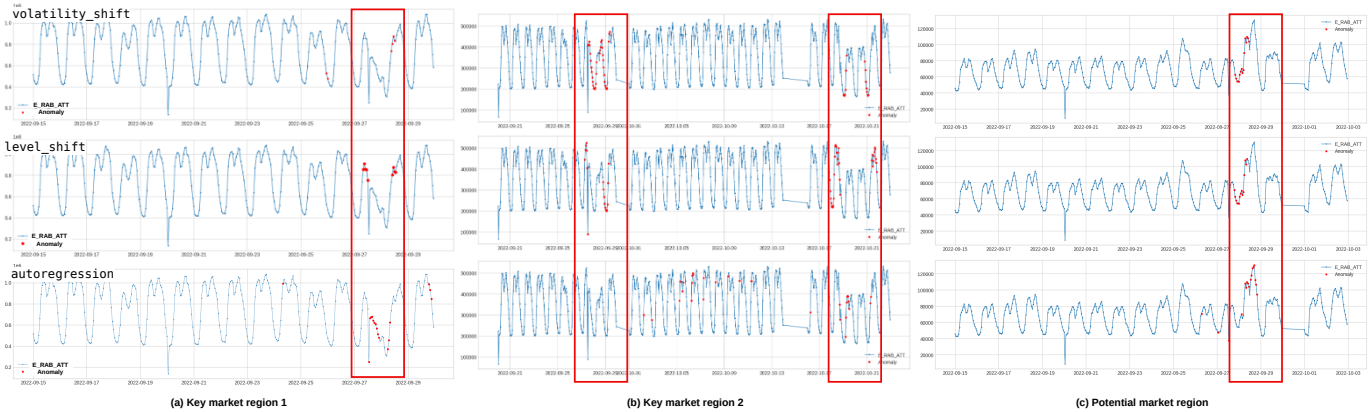



Fig. 11. Pattern change/level shift anomalies detected by 3 different algorithms: volatility_shift, level_shift, and autoregression

shown in Fig. 12, while there were no repeated ones in the district zones. After examining these patterns using domain expert analysis, we see the common points are: (i) time – the repeated patterns occur every weekend for months, (ii) type of business – these two zones are located in office areas or high-tech industrial areas, and (iii) pattern – the anomaly patterns show decreases in attempts and traffic context. Therefore, the anomalies are due to *zone (analytics subject) contexts* and could help the operator determine which areas and times (e.g., on the weekends when the network traffic sudden increase/decrease) they can automate the capacity/power saving.

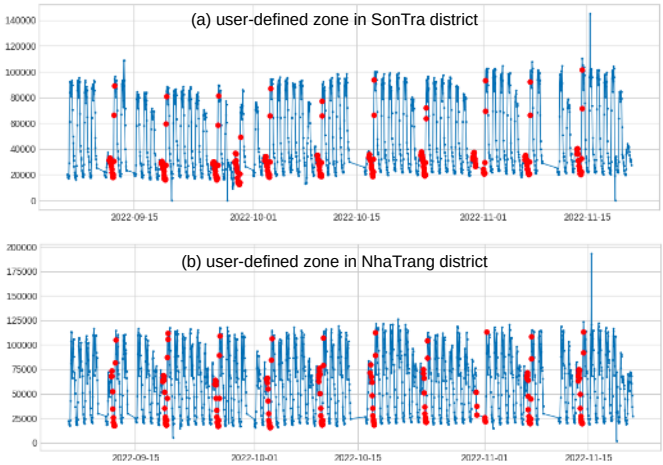



Fig. 12. Repeated usage anomaly patterns happened in two user-defined zones

### E. Using anomaly to monitor configuration change

In this case, we replayed a situation in which the optimization team carried out a change to downgrade the MIMO (multi-input multi-output) configuration in a zone. This configuration change was related to network capacity. The team needs to monitor network performance and quality before and after the action. To emulate the replay, a lightweight anomaly detection pipeline was used to support the team. Figure 13 illustrated anomalies detected from multiple algorithms in multivariate detection in a capability context based on historical data replayed. Because of a periodic abrupt decrease in throughput in a key market region, which could impact the quality of experience of lots of customers, the team could consider to fallback their configurations. The example configuration change was done manually by operators for 4G. Thus observation and fallback could be done by the team. However, we foresee in 5G where algorithms perform auto configuration (and future configuration deployment), this anomaly detection feature will be useful to be combined with control algorithms.

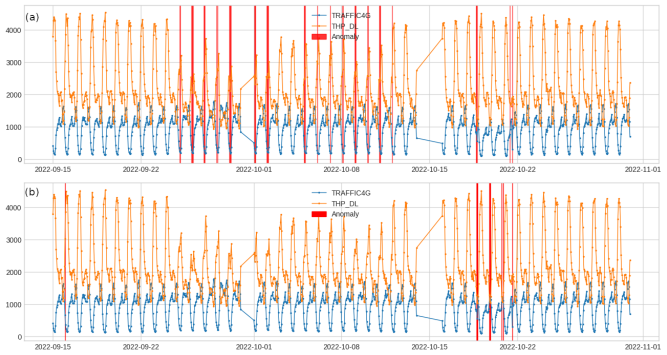



Fig. 13. Multivariate anomalies detected by PCA (a) and OutlierDetector (b)

### F. Multi-context anomaly and alarm for operations

One application scenario is to use multiple algorithms for multiple feature contexts to monitor windows of anomalies for responses to network troubleshooting. For example, when monitoring the anomalies in a district zone using the multi-context voting with a window of 7 hours, the troubleshooting team gets the result that there is a strong relation between accessibility anomalies and availability anomalies from `2022-09-28 01:00:00` to `2022-09-28 20:00:00`:

```
DATE,availability,accessibility,voted_anomaly,voted_weight
2022-09-28 05:00:00,0,1,1,0.5
2022-09-28 06:00:00,1,1,1,1
2022-09-28 07:00:00,0,1,1,0.5
2022-09-28 08:00:00,1,1,1,1
2022-09-28 09:00:00,1,1,1,1
2022-09-28 10:00:00,1,1,1,1
2022-09-28 11:00:00,1,1,1,1
```

A further close look into accessibility and multi-context anomalies are shown in Fig. 14 (a) and (b). Since domain knowledge suggests the availability issues mostly because of hardware faults, the team can carry out the alarm context and will find many anomalies of alarms `Cell_Faulty` which is root cause, shown in Fig. 14 (c).

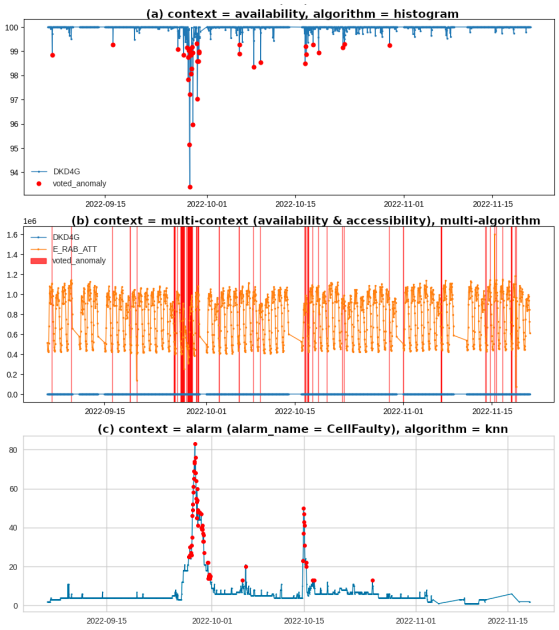


Fig. 14. Anomalies in the multi-context using multiple algorithms

## V. Related Work

Using multiple types of data for a single anomaly detection algorithm for an analytics subject usually requires all data to be concentrated, e.g., in datalakes, but it is not suitable when these types of data have different granularity and timeliness. This common approach can enable deep learning or powerful anomaly detection algorithms [12] but it is not suitable for anomalies defined based on dynamic contexts.

Many papers have performed extensive benchmarks of anomaly detection algorithms [8], [9], [10], [11], although not many detection benchmarks have been carried out for real-world telco data. We rely on these benchmarks to pickup anomaly detection algorithms for our framework but we have to evaluate suitable algorithms for suitable contexts based on our domain expertise and combine different algorithms for different contexts. Benchmarks do not discuss context-aware detection and how to combine results from different contexts.

Several papers focus on anomaly detection for atomic subjects. CellPAD [1] focused on patterns of sudden drops and correlation changes at the level of mobile cells by analyzing active users, radio resource usage, and data transmission load using statistical algorithms, Random Forest Regression, Regression Tree, etc. PCA [3] focused on network log records of Per Call Measurement Data in 4G-LTE networks and analyzed anomalies and root causes using Principal Component Analysis and finite-state machine. Watchmen Anomaly Detection (WAD) [2] focused on anomaly patterns due to issues in devices and equipment using log data obtained from real-time monitoring systems. Clearly these works have not focused on collective, dynamic subjects like zones in our work.

In some aspects, the context-aware and composable pipelines resemble the ensemble models in machine learning. For example, the work in [13] examines input data into different subtypes of data trained by different machine learning algorithms and combines different models to create a "unified model". Our approach utilizes contexts along the analytics pipelines (from data extraction to anomaly detection) and allows context-based configurations and combinations of data and software components for both data/analytics and teams.

## VI. Conclusions and Future Work

Although there exist many anomaly detection algorithms, without being context-aware, often anomaly detection does not bring relevant anomalies where analytics subjects are not static/atomic. In this paper, we present a novel framework that considers context in all steps of big data anomaly detection for a large-scale network. We have devised analytics feature contexts and team contexts and embedded contexts into different steps of the anomaly detection. By providing flexible ways to configure pipelines and their software components, we enable composable, context-aware anomaly detection that can be deployed in different locations for different teams. We are exploring our context model and associated components for composable anomaly detection for other domains. We will improve the voting and also evaluate and incorporate further algorithms into our framework. We will extend our evaluations to other subjects, especially user-defined zones.

## References

[1] J. Wu, P. P. C. Lee, Q. Li, L. Pan, and J. Zhang, "Cellpad: Detecting performance anomalies in cellular networks via regression analysis," in *2018 IFIP Networking Conference (IFIP Networking) and Workshops*, 2018, pp. 1–9.

[2] M. Mdini, A. Blanc, G. Simon, J. Barotin, and J. Lecoeuvre, "Monitoring the network monitoring system: Anomaly detection using pattern recognition," in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, 2017, pp. 983–986.

[3] C. Kim, V. B. Mendiratta, and M. Thottan, "Unsupervised anomaly detection and root cause analysis in mobile networks," in *2020 International Conference on COMmunication Systems NETworkS (COMSNETS)*, 2020, pp. 176–183.

[4] ETSI, "Key performance indicators (kpi) for evolved universal terrestrial radio access network (e-utran): Definitions," "ETSI TS 132 450 V9.3.0 (2011-10)".

[5] Z. Dehghani, *Data Mesh: Delivering Data-Driven Value at Scale*. O'Reilly Media, 2022.

[6] "ADTK," last access: Jan 27, 2023. [Online]. Available: https://adtk.readthedocs.io/en/stable/

[7] J. a. Gama, I. Žliobaitundefined, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Comput. Surv.*, vol. 46, no. 4, mar 2014.

[8] S. Schmidl, P. Wenig, and T. Papenbrock, "Anomaly detection in time series: A comprehensive evaluation," *Proc. VLDB Endow.*, vol. 15, no. 9, p. 1779–1797, jul 2022.

[9] S. Han, X. Hu, H. Huang, M. Jiang, and Y. Zhao, "ADBench: Anomaly detection benchmark," in *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[10] D. Renaudie, M. A. Zuluaga, and R. Acuna-Agost, "Benchmarking anomaly detection algorithms in an industrial context: Dealing with scarce labels and multiple positive types," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018, pp. 1228–1237.

[11] A. Lavin and S. Ahmad, "Evaluating real-time anomaly detection algorithms - the numenta anomaly benchmark," *CoRR*, vol. abs/1510.03336, 2015.

[12] K. Choi, J. Yi, C. Park, and S. Yoon, "Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines," *IEEE Access*, vol. 9, 2021.

[13] M. V. Ngo, T. Luo, and T. Q. S. Quek, "Adaptive anomaly detection for internet of things in hierarchical edge computing: A contextual-bandit approach," *ACM Trans. Internet Things*, vol. 3, no. 1, oct 2021.