JADE: Data-Driven Automated Jammer Detection Framework for Operational Mobile Networks

JADE: Data-Driven Automated Jammer Detection Framework for Operational Mobile Networks

Caner Kilinc[†], Mahesh K. Marina[†‡], Muhammad Usama[†], Salih Ergut[◇], Jon Crowcroft[‡⋆], Tugrul Gundogdu[○], Ilhan Akinci[○]
The University of Edinburgh[†] The Alan Turing Institute[‡] Turkcell[○] University of Cambridge[⋆] Oredata[◇]

***Abstract*—Wireless jammer activity from malicious or malfunctioning devices cause significant disruption to mobile network services and user QoE degradation. In practice, detection of such activity is manually intensive and costly, taking days and weeks after the jammer activation to detect it. We present a novel data-driven jammer detection framework termed JADE that leverages continually collected operator-side cell-level KPIs to automate this process. As part of this framework, we develop two deep learning based semi-supervised anomaly detection methods tailored for the jammer detection use case. JADE features further innovations, including an adaptive thresholding mechanism and transfer learning based training to efficiently scale JADE for operation in real-world mobile networks. Using a real-world 4G RAN dataset from a multinational mobile network operator, we demonstrate the efficacy of proposed jammer detection methods relative to commonly used anomaly detection methods. We also demonstrate the robustness of our proposed methods in accurately detecting jammer activity across multiple frequency bands and diverse types of jammers. We present real-world validation results from applying our methods in the operator's network for online jammer detection. We also present promising results on pinpointing jammer locations when our methods spot jammer activity in the network along with cell site location data.**

I. Introduction

Jamming is the intentional interference aimed at disrupting wireless communications services and as such can be seen as a denial of service (DoS) attack [1], [2]. The use and sale of jamming devices (commonly referred to as *jammers*) is therefore illegal in many countries (e.g., [3], [4]) and any violations may result in imprisonment or fines (e.g., [5], [6]). Nevertheless, jammers targeting all wireless communications technologies continue to be widely available and in fact quite affordable (see [7]–[9], for example). The threat posed by jammers to robust wireless communications including on mobile networks is expected to become worse in the future with low-cost and open-source software-defined radio (SDR) platforms becoming easily available to arm malicious actors [10]–[12].

In this paper, we consider jamming in operational mobile networks and in particular focus on the problem of detecting jammer activity in this context. The activation of jammers can severely deteriorate the service quality in a mobile network. Fig. 1 shows the impact on selected cell-level key performance indicators (KPIs) due to the presence of a jammer, using data from an operational 4G network. We observe that the jammer activation effectively shuts down the network for users, forcing uplink (UL) and downlink (DL) traffic volumes and throughputs down to near-zero. The levels of received signal strength indicator (RSSI) can also be elevated by over 20dB (100x increase) with a jammer. These results highlight the potential risk posed by jammers to the robustness of mobile networks. This applies not just to currently deployed 3G and 4G networks that the society heavily relies on for personal mobile communications, emergency response systems and public safety networks but also for future 5G networks that aspire to support diverse use cases including ultra-reliable and low-latency communications services (e.g., connected vehicles, remote surgery).

The jammer detection in current practice, however, is a highly manual process that is costly and slow, resulting in degradation in user quality of experience (QoE) while the jammer induced interference is detected and resolved. When significant service quality deterioration or network outage is noticed, which may be prompted by user reporting, radio network engineers manually examine large volumes of multi-dimensional network KPI data to diagnose the problem, which may also require field testing. Moreover, when jammer activity is intermittent it can be perceived as a radio network problem and result in misguided network reconfigurations/optimizations. Thus, in current operational mobile networks, according to the operators themselves, it may take days or weeks after the jammer activation for it to be detected. This suggests that jammer detection is a perfect use case for automation in mobile networks that can not only lead to savings in operating expenditure (OPEX) costs for operators but also enhance user QoE. Jammer detection is also the first step that can enable further troubleshooting to identify and pinpoint the interference source.

In view of the above, we aim at automated jammer detection for operational mobile networks that can automatically and quickly detect jammer (de-)activations, then trigger alarms to kick-start downstream resolution processes. While this clearly involves reliable detection of all kinds of malicious jammers, we also aim to detect the operation of other unintentional interference sources (e.g., malfunctioning devices, DECT phones) that can cause jammer-like impact on the network. Achieving this goal, however, is hard as it requires addressing the following challenges:

1) Any available ground-truth label information on jammer activations to build an automated detection system may be limited to small parts of the network and/or short periods of time due to the scale of the task and manual nature of the process. So practical jammer detection

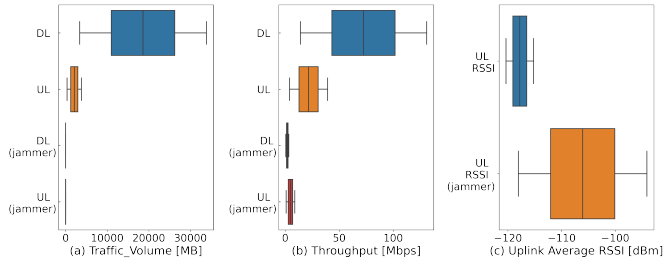

Fig. 1: Impact of jammer presence on three cell-level KPIs: (a) UL/DL traffic volumes; (b) UL/DL throughputs; and (c) uplink RSSI.

methods need to generalize well to be applied to bulk of the data that is "unlabeled".

2) Jammers can have very diverse behaviors. Some jammers once activated may stay continuously active while others exhibit complex discontinuous activation patterns. They also differ from each other in their type (e.g., military, drone specific jammer, DECT phone), the networks (e.g., 3G, 4G) and frequency bands they target. In our real-world dataset we have encountered jammers that can affect as many as ten different frequency bands. Moreover, newer types of jammers with a priori unknown characteristics can emerge over time.
3) Jammer related activity needs to be disambiguated from network KPI dynamics induced from normal and expected behavior (e.g., due to user mobility, network overload). But this can be challenging, for example, with active but distant jammers.

To this end, we present JADE, which to the best of our knowledge is the first automated jammer detection framework for operational mobile networks. By treating jammer activity as 'abnormal' or 'anomalous' from the mobile radio access network (RAN) infrastructure side, *JADE approaches jammer detection as an anomaly detection problem* [13]–[15]. JADE is envisioned for operator side deployment and considers cells at each tower site as measurement vantage points. It continually monitors the time series of various KPIs at each of these cells to detect anomalous behavior over time.

At the core of JADE are two deep learning based time-series anomaly detection methods that we developed; they work with multivariate KPI data at each cell. In the first method, a multi-variate long short term memory (LSTM) neural network model tracks the variation of uplink RSSI observed at each cell to predict its future value and detects an anomaly (jammer activation) when the difference between the predicted and the actual value is more than a *threshold*. The second method models multivariate KPI time series using a LSTM autoencoder model and the associated reconstruction error over time is compared against a threshold to detect jammer activation.

To address the several challenges outlined above (i.e., limited labeled data, ability to detect new and a priori unknown jammer activity, discrimination of jammer activity from normal KPI dynamics), JADE operates in a *semi-supervised anomaly detection mode* [13] in that it relies only on 'normal' data for training the above outlined models. This jammer agnostic nature of JADE also makes it robust for detecting jammers that may exhibit adversarial behaviors. JADE also embeds a mechanism to adaptively set thresholds to detect anomalies, thereby adjust the boundary between normal and abnormal events. To efficiently support the detection of jammers at scale in real-world mobile networks across multiple cells and operating frequency bands, we employ transfer learning [16] to develop one frequency and cell agnostic model instead of a different model for each cell and frequency band.

In summary, we make the following key contributions:

- We propose JADE, a novel data-driven jammer activity detection framework for operational mobile networks (§III) which addresses this problem for the first time. It incorporates two alternative custom-tailored semi-supervised deep learning based anomaly detection methods for the jammer detection task along with an adaptive thresholding mechanism. JADE also leverages transfer learning towards efficient modeling and ease of deployability.
- We extensively evaluate the JADE framework using a 4G radio access network (RAN) dataset from a multinational mobile network operator (§IV).
  - Our results show that the anomaly detection methods developed for JADE outperform a wide range of commonly used anomaly detection methods when applied to the jammer detection task, and also confirm the effectiveness of JADE's adaptive thresholding mechanism.
  - We demonstrate the robustness of the JADE framework powered by transfer learning in accurately detecting jammer activity across multiple frequency bands and diverse types of jammers.
  - We also present real-world validation results by applying our methods in the operator's network for online jammer detection.
  - As a downstream use case of JADE, we consider jammer localization. Specifically, we demonstrate the potential for pinpointing jammer locations based on jammer activity detections in the mobile network using JADE and combining them with cell site location data.

Next section describes our datasets and evaluation metrics.

II. Preliminaries

A. Datasets

A unique and noteworthy aspect of our work is the use of real 4G RAN datasets from a multinational mobile network operator for our evaluations. Overall, these datasets outlined below consist of about a million samples of radio network KPIs measured at an hourly time resolution over several months.

Training dataset. This dataset contains 'normal' data collected during periods with no jammer activation which is verified manually by the radio engineers from the operator's network. It is used for training the two semi-supervised anomaly

detection methods we developed as part of the proposed JADE framework. This dataset consists of around 700,000 measurement samples in total that were collected over a period of three months from 266 different cell tower sites, hosting 339 distinct cells operating over five LTE) uplink frequency bands: 852-862 MHz, 1745-1765 MHz, 1765-1775 MHz, 1970-1980 MHz, and 2500-2520 MHz. Each sample is a time-stamped (at hourly granularity) tuple of 9 different radio network KPIs measured at a cell: Uplink and Downlink Traffic Volumes (MB), Average Uplink and Downlink User Throughputs at PDCP Layer (Mbps), Average Uplink RSSI (dBm), E-UTRAN Radio Access Bearer (ERAB) Setup Success (%), Evolved Radio Access Bearer (ERAB) Drop (%), LTE Random Access Channel (RACH) Success (%), and Voice over LTE (VoLTE) drop (%).

Testing dataset. In contrast to the training dataset, this dataset corresponds to periods with jammer activation events. Specifically, it is made up of three sub-datasets, each corresponding to a different jammer's activity over a 3 week period. These three jammers are arbitrarily named by the operator as J16, J17 and J22. Among these, J16 affects three frequency bands (1745-1765 MHz, 1765-1775 MHz, 2500-2520 MHz) and has a discontinuous (on/off) activation pattern. On the other hand, J17 and J22 have continuous activation patterns, which means once activated they stay active until they are switched off. J17 and J22 operate on 2500-2520 MHz and 1765-1775 MHz bands, respectively. All these three sub-datasets are manually labeled by radio network engineers from the operator with ground-truth on jammer active/absent for each sample. These ground-truth labels are key to our study in that they enable evaluation of different jammer detection methods considered in this paper. Overall the testing dataset consists of 255600 samples with ground-truth annotated KPI data. KPIs captured in each sample are same as in the training dataset. Each of the three sub-datasets contain data for at least 50 cells.

For the above datasets, we have conducted preprocessing steps to impute missing values with the average of neighboring values and normalized each KPI value via min-max normalization. We did further feature extraction based on the time-series of the above listed KPIs, following a similar methodology to the one in [17], which led to 82 features in all. Besides the datasets described above, the proposed JADE framework and constituent jammer detection methods are validated through trials on the operator side. In §IV-C, we present results for two additional different jammers (labeled J23 and J19) encountered during the field trial period. In the case study on jammer localization based on jammer detection events (§IV-D), we also use the ground-truth location data for J16, J17 and J22 jammers along with the location data of surrounding cell tower sites from the operator to assess jammer localization accuracy.

B. Evaluation Metrics

Here we describe our metrics to evaluate the various jammer detection methods developed/considered in this paper. These

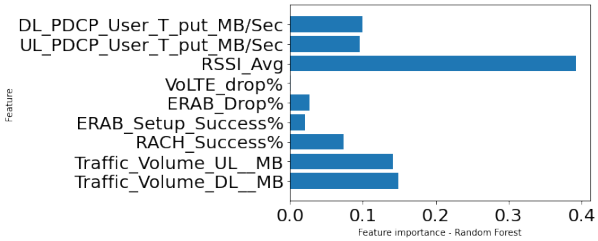

Fig. 2: Relative importance of different features (KPIs) for the random forests based classifier.

metrics are defined in terms of four possible outcomes that can result from applying a jammer detection method, which are:

- True-Positive (TP): Jammer activity detected when such activity is actually present, as per the ground-truth.
- False-Positive (FP): Jammer activity detected when in fact there is no jammer activity.
- True-Negative (TN): Jammer activity not detected when jammer activity is absent as per the ground-truth.
- False-Negative (FN): Jammer activity not detected even though jammer is actually active.

An effective jammer detection method minimizes both FPs and FNs. Two commonly used measures to assess the extent to which a given method achieves these goals are *Precision* and *Recall*[1], as defined below:

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

Higher values for both these metrics are desired. A lower value of precision (equivalently, a higher number of FPs) leads to an increase in OPEX to diagnose, confirm and localize jammer activity (e.g., by field visits and testing), when in reality, there is none. On the other hand, a lower value of recall (equivalently, a high number of FNs) shows that the method in question fails to detect all jammer activation events and thus risks prolonged degradation of user QoE until the jammer activity is eventually detected and stopped. Given the above, having high precision and high recall are equally important. As such, we consider a composite metric called *F1-Score* that weighs precision and recall equally by taking a harmonic mean of the two, as defined below:

$$F1\text{-}Score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \tag{3}$$

III. The JADE Framework

In this section, we describe our proposed JADE framework for data-driven automated jammer detection in operational mobile networks in detail. By way of motivation, we start by examining the limitations of commonly used jammer detection approach based on supervised classification, thereby highlight

[1]Recall is sometimes also referred to as Sensitivity or True Positive Rate.

a challenge that an effective jammer detection method design needs to address.

A. Limitations of Supervised Binary Classification Approach

A straightforward approach to jammer detection is to treat it as a supervised binary classification problem. In fact, most recent works on jammer detection [18]–[22], though aimed at 802.11 networks, take this approach, and show Random Forests (RF) [23] to be the most effective classifier. Here we assess the effectiveness of supervised binary classification approach towards jammer detection in operational mobile networks, considering RF as the classifier.

Since we need a labeled dataset for training and testing with supervised classification, we use one of our 'testing' sub-datasets for the J16 jammer (outlined in section II-A) for this study. Each sample in this dataset is a 9-tuple KPI data labeled as jammer active/absent. The dataset overall spans over 60 cells across three different frequency bands: 35 cells in 1745-1765MHz; 17 cells in 1765-1775MHz; and 9 cells in 2500-2520MHz. It is a fairly balanced dataset with 60% (40%) jammer active (inactive) samples. We do a 70/30 split of this dataset to create training and test data for the RF based binary classifier, which leads to a test set with 11 cells in 1745-1765MHz; 5 in 1765-1775MHz; 3 in 2500-2520MHz.

We first use the RF feature importance [24] test to examine the relative importance of the different features from the classifier's perspective with the results shown in Fig. 2. We find that the average uplink RSSI is the most important feature, followed by UL/DL traffic volumes and throughputs. Interestingly, the percentage of VoLTE drops has the least predictive effect on the classifier, perhaps because call drops could happen due to a myriad of factors beyond the jammer presence e.g., due to network overload or coverage issues.

Fig. 3 shows the box plot results for precision, recall, and F1-score metrics, separately for each frequency band where the J16 jammer operates, with each box plot capturing the distribution for the metric across different cells. We observe that precision and recall values are less than ideal, and range between 0.7-0.8 and 0.55-0.75, respectively. The relatively lower recall results indicate that this supervised classifier method errs more towards missing some jammer activation events than causing false alarms. The combined effect, measured by F1-score, is worse with a median value between 0.5 and 0.7.

Crucially, this poor classification performance is noticed for cases with fewer cells and less data in the training dataset. This highlights a key issue with supervised classification based approach to jammer detection – more and diverse "labeled" data is needed. Compared to getting "normal" class data, it is difficult and very costly to produce a large and diverse labeled dataset with "jammer activation" events. More data can be available for "normal" periods as reflected by our datasets (where 700,000 samples or at least 70% of the total data is for the "normal" category), but supervised classifiers fail to take advantage of such data.

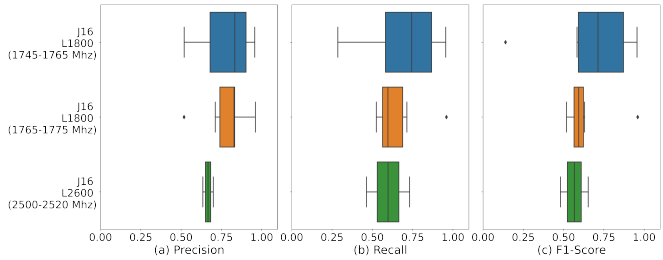

Fig. 3: Precision, recall and F1-score performance of random forests based supervised binary classifier.

B. JADE Overview

The foregoing discussion not only highlights limitations with the supervised classification based approach but a key challenge to be addressed in jammer detection method design, i.e., limited or no data labeled with jammer effect. Another challenge is that there exist numerous types of jammers and each has its own different characteristics and impact on mobile network performance (e.g., due to affecting different sets of frequency bands). Gathering training data that represents all jammer types is simply impractical. Moreover, identifying the decision boundary between normal network behavior and that affected by jammer activity through the radio network KPIs is challenging due to the inherent KPI dynamics.

Our proposed JADE framework (illustrated in Fig. 4) addresses the above challenges. JADE is envisioned for operator-side deployment and considers cells at each tower site as measurement vantage points to aid in online jammer detection. It relies on continual monitoring of time series of various KPIs at each of these cells and collecting this data at the operator RAN data lake facility. This data is then preprocessed to address issues such as missing values by imputing with neighboring ones before putting it through the jammer detection pipeline in JADE.

JADE approaches jammer detection in an operational mobile network as a time-series anomaly detection problem [13]–[15] by considering that jammer activity manifests as 'abnormal' or 'anomalous' in the time series of radio network KPIs. To address the aforementioned challenges, JADE adopts the *semi-supervised* form of anomaly detection [13] by solely relying on and leveraging potentially abundant 'normal' data for model training. This also makes JADE independent of the type and behavior of a jammer, thereby enabling robust detection across diverse types of jammers. At the core of JADE are two alternative deep learning based anomaly detection models. Besides, JADE incorporates an adaptive mechanism for these models to set thresholds that represent the boundary between normal and anomalous samples. Also, rather than have a separate anomaly (jammer) detection model per cell or frequency band, JADE employs transfer learning [16] towards *one* cell and frequency band agnostic model. We elaborate on the above components of JADE in the following subsections.

C. Deep Learning Models for Cell-Level Anomaly Detection

Here we describe the two anomaly detection based models we develop as part of JADE for jammer activity detection:

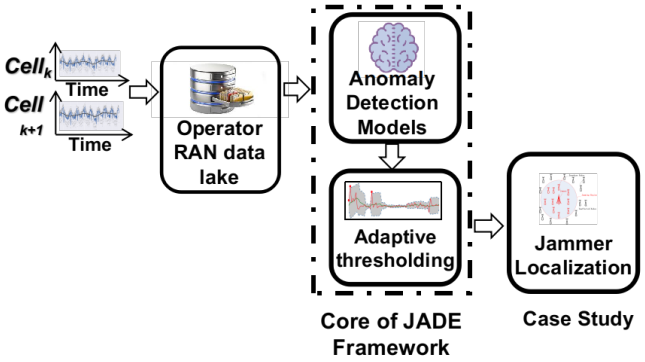

Fig. 4: Illustration of the JADE Framework.

(1) Autoencoder (AE) [25] based model; (2) RSSI prediction based model. Both these models operate at the cell level and take multivariate time series data for radio network KPIs as input but aim to detect anomalous samples in the time series through different approaches, as elaborated below. Since we are dealing with time-series data, we use LSTM [26] cells in the architecture of both models.

*1) **Autoencoder (AE) based Anomaly Detection Model**:* Fig. 5(a) illustrates the proposed LSTM autoencoder based model that takes multivariate time series as input. The core idea is to encode each input sample using an LSTM network and decode it using another LSTM network. Anomaly detection relies on the (in)ability to faithfully decode the input sample. With a sufficient amount of training data (i.e., normal samples), the AE model learns to reconstruct the normal samples. The reconstruction error for normal samples will therefore be lower compared to the reconstruction error of samples in the presence of a jammer.

As mentioned above, the AE model has two components: an encoder and a decoder. The encoder represents the input sample in the latent space, whereas the decoder aims to reconstructs the latent domain back to the input sample. The encoder in our model consists of four layers. The number of LSTM cells in the first layer is equal to the number of features in the input sample while the following three layers have 512, 64, and 4 LSTM cells, respectively. Then the decoder reconstructs the input sample from the 4-dimensional latent space. The decoder also consists of four layers where the number of LSTM cells in its first three layers are 4, 64, and 512, respectively. The number of LSTM cells in the last layer of the decoder equals the input sample size (in terms of features) to the AE model.

We use Mean Squared Error (MSE) as a loss function and trained the model for 20 epochs for every cell in the training dataset. We empirically optimize the hyper-parameters with grid search [24]. We use RMSprop as an optimizer and RELU as an activation function with a batch size of 64. Since the network KPIs have inherent noise and variations, we did not introduce any additional noise in the input of the AE towards better generalization.

*2) **RSSI Prediction based Anomaly Detection Model**:* We now present an alternative anomaly detection model for jammer activity detection that focuses on predicting the uplink RSSI over time, considering that this KPI is seen to be the most important feature in our earlier study on supervised binary classification based jammer detection (Fig. 2). The essential idea here is to train a time-series prediction model that takes as input multivariate time series of KPIs and predicts the RSSI for each following time step. The prediction error is expected to be low for 'normal' samples as the model is trained with data consisting of such samples. But the presence of jammer activity can yield higher prediction errors, which can then be identified as anomalies (jammer activation events).

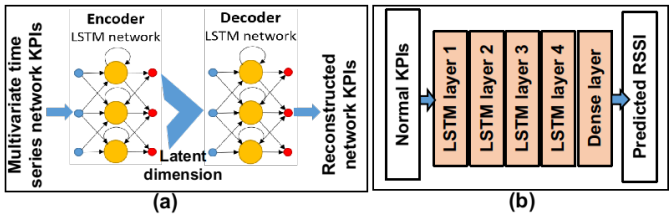

Fig. 5: (a) Autoencoder based anomaly detection model; (b) RSSI prediction based anomaly detection model.

Fig. 5(b) illustrates our RSSI prediction based anomaly detection model, which leverages the state-of-the-art LSTM based time series prediction model architecture. Our model consists of four LSTM layers (the input layer, three hidden LSTM layers) and one fully connected output layer. The LSTM cells in the input layer are equal to the number of features in each input sample. The following three hidden layers have 32, 16, and 8 LSTM cells, respectively. The output layer is a fully connected dense layer with a single neuron that outputs RSSI predictions. Specifically, our model predicts the RSSI for the following 10 time steps based on multivariate KPI time series for the past 10 time steps, along a moving window. Like in the AE model, the hyper-parameters are experimentally optimized with grid search, RELU is used as an activation function, and the batch size is 64.

3) Single and Multi-KPI models: We consider two variants of the above described models:

- *Single KPI model* that considers only one KPI, specifically RSSI. In essence, AE and RSSI prediction versions of this model deal with RSSI KPI time series. Note that each sample in the input time series to these models is multivariate with 74 different features, due to feature extraction during data preprocessing.
- *Multi-KPI model* that considers all 9 KPIs in our dataset, including other KPIs like uplink/downlink throughput and RACH success rate. This is naturally a multivariate time series with 82 features in each sample of the input time series, again due to the feature extraction step.

These single and multi-KPI models allow us to understand the added benefit of considering the various different KPIs beyond just the uplink RSSI.

D. Adaptive Thresholding

The two anomaly detection models described in the previous subsection produce a reconstruction/prediction error for each new sample in a cell KPI time series. But to detect whether that sample is an anomaly (due to jammer activity or other such

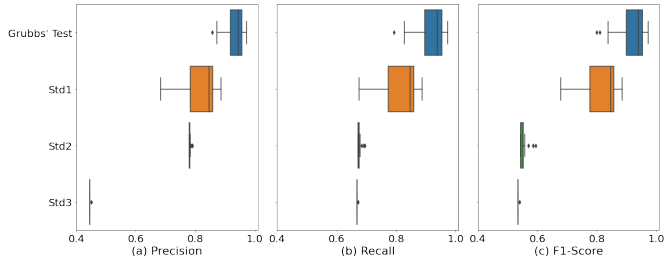

Fig. 6: Representative result showing the performance gain with our Grubb's test based adaptive thresholding approach compared to the state-of-the-art $n\sigma$ approach.

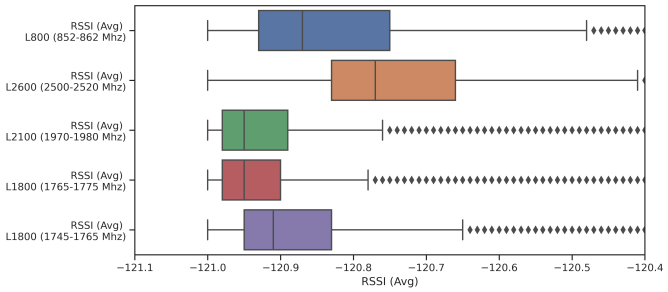

Fig. 7: Average RSSI distribution across cells using different uplink frequency bands.

interference), we need a threshold (error level) representing the boundary between normal and anomalous samples. Correctly setting this threshold is equally key to effective jammer detection. It can be seen as deciding the tradeoff between FP and FN probabilities.

Different approaches are taken for this thresholding in the literature. Some works do this manually, relying on human expert feedback to set/adapt thresholds (e.g., [27]–[29]) but this is unviable in our setting. Feng et al [30] simply use the median reconstruction error as the threshold, which is again not robust. Most recent wireless anomaly detection works, however, approach this thresholding by assuming that errors are generated from a Gaussian distribution. Some of these works [31]–[33] set the threshold based on a desired FP probability, which is not appropriate in our setting as we also would like to have equally good precision and recall performance.

Other works [34]–[36] adopt a $n\sigma$ thresholding approach for some small value of n where σ is the standard deviation of the error distribution. This essentially means that a sample is considered an anomaly if its error (reconstruction/prediction error in our case) is more than $n\sigma$ away from the mean of the error distribution, obtained using (normal) training data. While the reconstruction/prediction errors across all samples in the training data also follow a Gaussian distribution in our case (results not shown due to space limit), we find the $n\sigma$ thresholding approach is not robust in our setting, as we show shortly. We observe that this is because of the diversity among cells and so the impact of a jammer on those cells is also different. There also exist other works (e.g., [37]) that empirically obtain a fixed threshold.

We instead take a tailored adaptive thresholding approach on a per cell basis by examining the time series of (reconstruction/prediction) errors in each cell to detect anomalies. Our proposed approach to this issue can be seen as an adaptation of Grubb's Test [38] for single outlier detection in univariate data. Note that the data for thresholding purposes refers to either reconstruction or prediction errors, depending on the anomaly detection model used, and it is therefore univariate. In the following, we describe our proposed thresholding method.

We start by defining two hypotheses: H_0: There are no outliers in the data; and H_1: There is exactly one outlier in the data. We also define Grubb's test statistic to be calculated for each new data (error) sample Y_i:

$$G\_calc = \frac{max|Y_i - \overline{Y}|}{s} \tag{4}$$

where $\overline{Y}$ and s, respectively, represent the mean and standard deviation of the error data samples considered for outlier detection. Based on the above, we detect an outlier (Y_i) or equivalently, reject the null hypothesis (H_0 hypothesis of no outliers) if the calculated test statistic is greater than a critical value (as defined on the right hand side of the equation below):

$$G\_calc > \frac{(N-1)}{\sqrt{N}} \sqrt{\frac{\left(t_{\alpha/(2N),N-2}\right)^2}{N-2+\left(t_{\alpha/(2N),N-2}\right)^2}} \tag{5}$$

where N is the number of training error samples considered initially, t refers to the t-distribution and α is the significance level (related to the desired confidence interval). If, on the other hand, G_calc is less than or equal to the critical value then we conclude there is no outlier in the set of N samples.

We bootstrap the above statistic calculation with a series of N error samples (ϵ) from the training data and view it as a window. Then when we apply an anomaly detection model (one of the two from the previous subsection) to each new sample in the radio network KPI time series for a cell, we slide the window and include the new error sample ϵ_{new} to recalculate the statistic. If it is greater than the critical value, the new error sample corresponds to an outlier (anomaly) and so we undo the window sliding to ignore ϵ_{new}. Otherwise, ϵ_{new} is now part of the set of error samples considered for outlier detection. In this work, we set α to 0.05 (equivalent to 95% confidence interval) and empirically set N to 25.

Fig. 6 demonstrates the effectiveness of our above described Grubb's test based thresholding approach with the $n\sigma$ approach for different typical values of n (1, 2 and 3) for the J16 UL-1800 (1745-1765 MHz) test dataset and using the multi-KPI RSSI prediction based anomaly detection model. Box plots reflect the distribution of each metric across all cells in this dataset. These results clearly any single static threshold is not effective generally, while our adaptive approach always yields the best performance. We have observed similar performance improvement with our approach with the other test datasets (omitted due to space limitations).

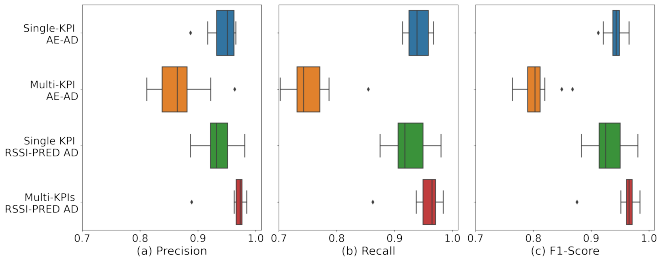

Fig. 8: Relative performance of AD models in JADE on the J16 UL1800 (1745-1765 MHz) test dataset.

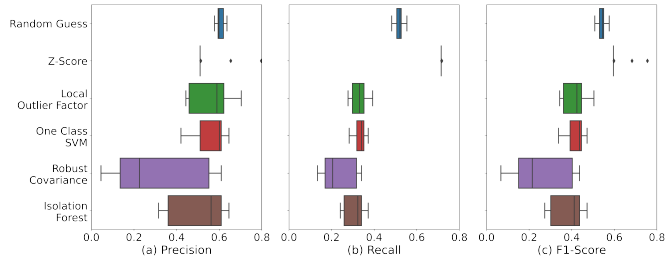

Fig. 9: Performance of baseline AD methods on the J16 UL-1800 (1745-1765 MHz) test dataset.

E. Transfer Learning

Our proposed solution for detecting jammer activity via anomaly detection model with adaptive thresholding approach thus far implicitly considered a single cell scenario. In practice, however, each cell tower site hosts multiple cells, possibly configured on different frequency bands. Moreover, an operator's network deployment may have thousands of such cell sites. But training and maintaining a per cell or even per frequency band anomaly detection models is not scalable from a deployment perspective.

We therefore aim at a single cell and frequency agnostic anomaly detection model, one per each alternative discussed in §III-C. We cannot, however, base such a model by training it on single cell or even single frequency band as different cells and frequency bands have diverse behavior in terms of radio network KPIs and jammer effect. For example, the RSSI distributions across cells on different uplink frequency bands shown in Fig. 7 clearly highlight such diversity. So we train our target cell and frequency agnostic model using 'normal' data from different cells and frequency bands. To make this training efficient, we leverage transfer learning (TL) [16].

Specifically, we train the cell and frequency agnostic model as follows. We start with a frequency band and a cell within that band. Once the model is trained with data for that cell, we treat that as the start point for training on a different cell from the same frequency band, reusing the already trained model's weights as opposed to starting from scratch. Once the model is trained across all cells of a frequency band, then it is used as the base model for training on cells for another frequency band. We repeat this process until we cover all frequency bands and cells in the training data, which ultimately results in the frequency and cell agnostic model.

IV. Evaluation

In this section, we evaluate the performance of the proposed JADE framework using the operator provided 4G RAN datasets described in §II-A in terms of the precision, recall and F1-score metrics defined in §II-B.

A. Comparative Evaluation of Anomaly Detection Methods

In the previous section, we have already presented evaluation results that show the effectiveness of the adaptive thresholding mechanism in JADE. Here we evaluate the different anomaly detection (AD) model alternatives in JADE relative to commonly used AD methods [13], [39].

Recall from §III-C that the JADE framework offers four different AD models: AE and RSSI prediction based models, each with single and multi-KPI versions. We train these models using the training dataset (§II-A). We evaluate using the J16 UL-1800 (1745-1765 MHz) part of the testing dataset. For comparison, we consider five diverse and commonly used AD methods: Z-Score, Local Outlier Factor, One-Class SVM, Robust Covariance and Isolation Forest. To make these baseline methods work with our multi-dimensional KPI data, we use Principal Component Analysis (PCA) [40] based dimensionality reduction to represent the dataset in two-dimensional space.

Fig. 8 shows the performance comparison between the four JADE AD models. We observe that the multi-KPI version of AE based AD model performs worse than the other three models, especially in terms of recall. However the single KPI version (specifically RSSI) of the AE model relatively performs much better. We attribute this to the characteristics of KPIs other than RSSI that allow reconstruction even in jammer presence, resulting in some jammer activations going undetected. RSSI prediction based AD models both perform well with higher than 0.9 values in all three metrics. The multi-KPI version of RSSI prediction based model offers the best performance overall, which suggests that considering all KPIs is beneficial although marginally. Relatively, the baseline methods perform quite poorly with values for all metrics less than 0.6, which is no better than the random guess based on the known probability of jammer activation events in the test dataset as prior (Fig. 9). These results provide a convincing justification for developing tailored AD methods for KPI based jammer detection as we do in JADE.

B. Robustness across Diverse Frequencies & Jammer Types

So far we have considered the JADE performance on one uplink frequency band (J16 UL-1800) and with jammer type (J16). Here we evaluate across different frequency bands and jammer types to assess its robustness. For this study, we consider the best performing model from the previous experiment as the JADE AD model – the multi-KPI version of RSSI prediction based AD model.

We first compare the jammer detection performance between frequency-specific and frequency-agnostic versions of the chosen JADE AD model on different frequencies that J16

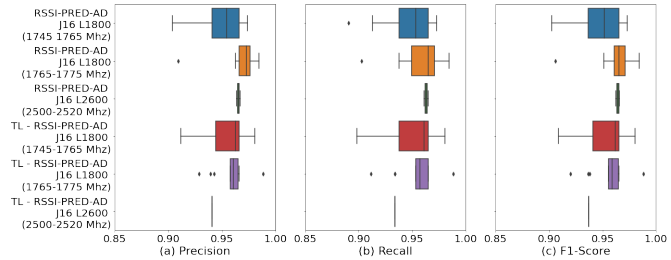

Fig. 10: The performance of multi-KPI version of RSSI prediction based AD model in JADE on different J16 frequency bands, comparing TL based frequency-agnostic model with frequency-specific models.

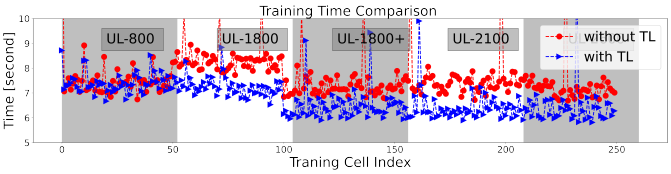

Fig. 11: Training time gain from using TL to train the frequency-agnostic multi-KPI RSSI prediction based AD model.

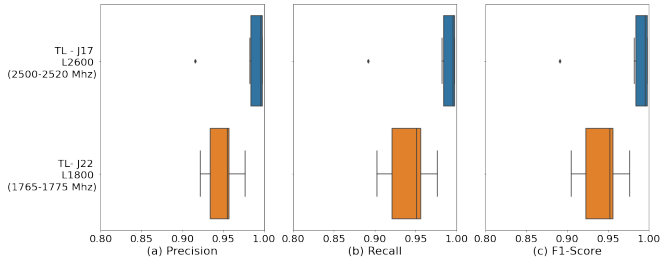

Fig. 12: The performance of TL based frequency-agnostic JADE AD model on J17 and J22 part of the testing dataset.

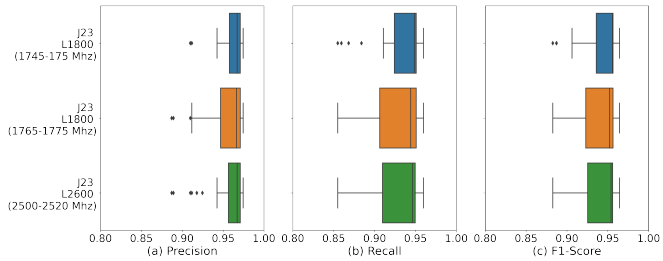

Fig. 13: Operator-side field validation with J23 jammer.

jammer operates on. The frequency-agnostic version is obtained with TL based training as described in §III-E. Results shown in Fig. 10 indicate that the frequency-agnostic model, though marginally worse than its frequency-specific counterparts, yields greater than 0.9 value for all metrics in more than 90% of the cells. The training time from using TL relative to not using it is shown in Fig. 11, which clearly demonstrates the training efficiency gain that TL provides.

To study robustness of JADE jammer detection performance, we apply the TL based frequency-agnostic model as in the previous experiment on J17 and J22 parts of the testing dataset. Results shown in Fig. 12 clearly confirm the effectiveness of JADE for these other jammer types. The root of the robustness property of JADE lies in its design choice to rely on semi-supervised form of anomaly detection, training only on 'normal' data.

C. Field Validation

So far our evaluation of JADE performance was based on operator provided data labeled with ground-truth on jammer activity (i.e., the testing dataset in §II-A). We now present results validating JADE performance in the field at the operator side. For this purpose, we provided the radio network engineers at the operator with the implementation of JADE's TL based frequency-agnostic AD model, as in the last experiment. It was used to reliably detect a different type of jammer (J23) with discontinuous activation pattern as shown in Fig. 13.

JADE was also used at the operator side to detect a military grade jammer (J19) targeting different uplink frequencies. Due to the complex nature of this jammer activity, it was not practical, like with J23, to manually label the ground-truth by the operator's radio engineers. Nevertheless, we visually demonstrate in Fig. 14 how JADE is able to detect jammer activity on a sample cell affected by this jammer. During the field trial period, JADE was also used to detect jammer-like activity that was eventually diagnosed to be due to a hardware related problem (see Fig. 15), which shows the versatility of our JADE approach to detect both intentional and unintentional interference behavior affecting mobile network operations.

D. Jammer Localization Case Study

Here we briefly discuss a case study for JADE on jammer localization. The idea is to use the jammer detection results with JADE in conjunction with cell site location data to estimate a jammer's location. Jammer localization is a kind of transmitter localization problem as cell sites surrounding a jammer detect its activity (with JADE) as receivers and it can be localized based on sensed signals at those sites.

As our purpose here is not on jammer localization algorithm design per se but rather on showing the usefulness of JADE for such downstream task, we consider three most commonly used transmitter localization algorithms [41]: max RSSI, centroid and weighted centroid. With max RSSI, the location of the cell site where jammer is detected with max RSSI is taken as jammer's location estimate. With (weighted) centroid, (RSSI weighted) geometric center of cell site locations that detect the jammer is estimated as the jammer location. We use the ground-truth jammer locations provided by the operator to calculate location estimation errors as Euclidean distance (between ground-truth and estimated locations).

Fig. 16a) shows the obtained results for J16, J17 and J22 jammers in our testing dataset while Fig. 16b) zooms in on the results for J17 case. Even with these commonly used localization algorithms, we find that jammers can be localized within a few hundred meters of the ground-truth, which is sufficient in practice for radio engineers to pinpoint the source of jamming activity. Between the three jammers considered, J16 is located in the country side with sparser mobile network

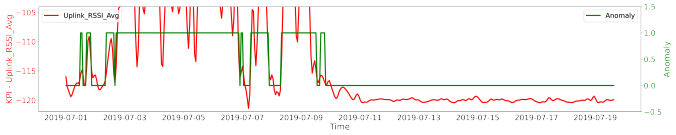

Fig. 14: Detection of J19 military jammer activity on a sample cell with JADE.

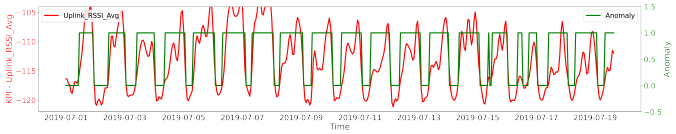

Fig. 15: Detection of hardware induced interference with JADE during the field trial period.

infrastructure, explaining the relatively higher localization errors.

V. Related work

In the last decade, wireless jamming has received considerable attention in the research literature. The earlier work, surveyed in [2], [42], mainly focused on jamming in the context of ad hoc and sensor networks; and 802.11 based wireless LANs and multihop networks. A significant early work [1] considers jamming in sensor networks. Considering 802.11 networks, [43] examines local and collaborative detection methods while [44] focus on distinguishing different types of jamming attacks. More recent works in this line of research focus on machine learning-based jammer detection [18]–[20], [22], [45] with most of them taking a supervised classification approach and concluding that the random forest method performs the best.

Closer to our target setting, jamming in mobile networks (specifically 4G/LTE) is considered in [11], [46], [47]. In contrast to our focus on jammer detection in operational mobile networks, these works concentrate on highlighting vulnerabilities of the LTE system to jamming attacks and proposing mitigation methods. [11] also presents a threat assessment of the LTE system that identifies the weakest points in its physical (PHY) layer while [46], [47] observe that jamming the uplink is more effective for an attacker than the downlink due to the relatively lower transmission power limit for LTE UEs in the former.

We are unaware of any previous work that leverages KPI data to detect jammers in mobile networks (by treating them as anomalies or otherwise). But there exist works in mobile networks and beyond that use KPI data for anomaly detection [27], [28], [35]. While [27], [35] are focused on mobile networks augmented with self-organising network (SON) features, OPPRENTICE [28] targets anomaly detection in the context of Internet-based service delivery. [27], [28] advocate the use of supervised classification for anomaly detection based on ensemble methods (specifically random forest in OPPRENTICE [28]), and manual labeling to address the ground-truth issue. On the other hand, [35] uses a simple Z-Score [48] like statistical method to detect anomalies at the

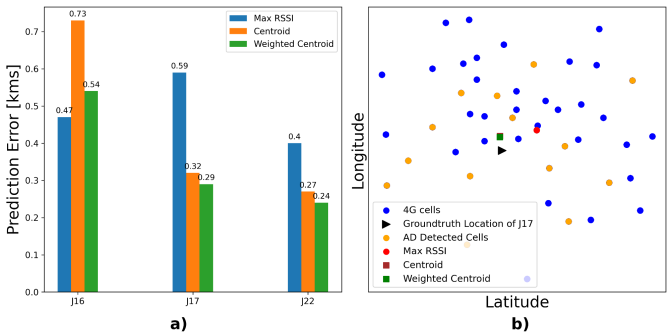

Fig. 16: a) Jammer localization errors with different algorithms; b) Location estimates for J17 jammer.

KPI level and then uses correlation among KPI anomalies to determine cell-level anomalies. ABSENCE [34] is another broadly related work that detects service disruptions in mobile networks using anonymized call detail record (CDR) data. The essential idea here is to monitor spatio-temporal customer usage based on anonymized CDRs and identifying deviations from historical usage as anomalies.

Also broadly related is the recent work on deep learning based RF/spectrum anomaly detection from the spectrum monitoring perspective [30]–[33], [36] to detect unauthorized transmissions, misconfigured transmitters, etc. In contrast to these works, the anomaly detection methods we develop rely on cell-level KPI data and are tailored for automated and scalable jammer detection in operational mobile networks. Our work is also unique due to the use of an operator provided 4G network dataset for evaluations and real-world validations.

VI. Conclusions

We have presented JADE, an online framework for jammer activity detection in operational mobile networks. At its core, the JADE framework consists of deep learning based semi-supervised anomaly detection models that solely rely on 'normal' training data. Also, JADE incorporates an adaptive mechanism for addressing the thresholding issue for cell-level anomaly detection. Moreover, the JADE framework utilizes transfer learning to enable itself to scalably work across many cells and multiple frequency bands. We have evaluated the JADE framework on a 4G RAN dataset provided by a multinational mobile network operator with jammer activation events labeled for different types of jammers. Field validation is also conducted, and it shows the effectiveness of the JADE framework in the wild. Lastly, we also provide promising results on the use of jammer detections from JADE for localization of jammers along with cell site location data.

References

[1] W. Xu, W. Trappe, Y. Zhang, and T. Wood. The feasibility of launching and detecting jamming attacks in wireless networks. In *Proc. ACM MobiHoc*, 2005.

[2] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy. Denial of service attacks in wireless networks: The case of jammers. *IEEE Communications Surveys & Tutorials*, 13(2), 2011.

[3] FCC. Jammer Enforcement. https://www.fcc.gov/general/jammer-enforcement, April 2020.

[4] Ofcom. Radio frequency jammers. https://www.ofcom.org.uk/spectrum/interference-enforcement/spectrum-offences/jammers.

[5] J. Kastrenakes. FCC issues largest fine in history to company selling signal jammers. https://www.theverge.com/2014/6/19/5824344/fcc-issues-signal-jammer-seller-largest-fine-ever-34-9-million, June 2014.

[6] P. McNamara. FCC formalizes massive fines for selling, using cell-phone jammers. https://www.networkworld.com/article/3075024/fcc-formalizes-massive-fines-for-selling-using-cell-phone-jammers.html, May 2016.

[7] 4G Cell Phone Jammers. https://www.jammer-shop.com/4g-lojack-xm-jammers.html.

[8] RF Jammers. http://www.digitalrf.net/products/rf-jammers/.

[9] GSM Jammers. https://www.selcomsecurity.com/en/products/counter-espionage-devices/gsm-jammers.

[10] D. Talbot. One Simple Trick Could Disable a City's 4G Phone Network. https://www.technologyreview.com/2012/11/14/84825/one-simple-trick-could-disable-a-citys-4g-phone-network/, Nov 2012.

[11] M. Lichtman, R. P. Jover, M. Labib, R. Rao, V. Marojevic, and J. H. Reed. LTE/LTE-A jamming, spoofing, and sniffing: threat assessment and mitigation. *IEEE Communications Magazine*, 54(4), 2016.

[12] S. Dudek. Modmobjam: Jam tomorrow, jam yesterday, but also jam today. https://www.synacktiv.com/ressources/sstic_rump_2018_modmobjam.pdf, June 2018.

[13] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3), July 2009.

[14] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han. Outlier detection for temporal data: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 26(9), 2014.

[15] A. Blázquez-García, A. Conde, U. Mori, and J. Lozano. A review on outlier/anomaly detection in time series data. *CoRR*, abs/2002.04236, 2020.

[16] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 2020.

[17] D. T Shipmon et al. Time Series Anomaly Detection: Detection of Anomalous Drops with Limited Features and Sparse Examples in Noisy Highly Periodic Data. *arXiv preprint arXiv:1708.03665*, 2017.

[18] O. Puñal, I. Aktaş, C. Schnelke, G. Abidin, K. Wehrle, and J. Gross. Machine learning-based jamming detection for IEEE 802.11: Design and experimental evaluation. In *Proceedings of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2014.

[19] Z. Feng and C. Hua. Machine learning-based RF jamming detection in wireless networks. In *2018 Third International Conference on Security of Smart Cities, Industrial Control System and Communications (SSIC)*, 2018.

[20] B. Upadhyaya, S. Sun, and B. Sikdar. Machine learning-based jamming detection in wireless IoT networks. In *2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, 2019.

[21] G. Selen, G. Caner, and K. Karabulut. Jammer detection based on artificial neural networks: A measurement study. In *Proceedings of the ACM Workshop on Wireless Security and Machine Learning*, WiseML 2019, 2019.

[22] Y. Arjoune, F. Salahdine, M. S. Islam, E. Ghribi, and N. Kaabouch. A novel jamming attacks detection approach based on machine learning for wireless communication. In *2020 International Conference on Information Networking (ICOIN)*, 2020.

[23] L. Breiman. Random Forests. *Machine Learning*, 45:5–32, 2001.

[24] F. Pedregosa et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[25] MA. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2):233–243, 1991.

[26] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[27] G. Ciocarlie, U. Lindqvist, K. Nitz, S. Nováczki, and H. Sanneck. On the feasibility of deploying cell anomaly detection in operational cellular networks. In *2014 IEEE Network Operations and Management Symposium (NOMS)*, 2014.

[28] D. Liu, Y. Zhao, H. Xu, Y. Sun, D. Pei, J. Luo, X. Jing, and M. Feng. Opprentice: Towards practical and automatic anomaly detection through machine learning. In *Proceedings of the 2015 Internet Measurement Conference*, IMC '15, 2015.

[29] R. Mijumbi, A. Asthana, M. Koivunen, F. Haiyong, and Q. Zhu. Design, implementation, and evaluation of learning algorithms for dynamic real-time network monitoring. *International Journal of Network Management*, page e2108, 2020.

[30] Q. Feng, Y. Zhang, C. Li, Z. Dou, and J. Wang. Anomaly detection of spectrum in wireless communication via deep auto-encoders. *The Journal of Supercomputing*, 73(7), July 2017.

[31] T. O'Shea, T. C. Clancy, and R. W. McGwier. Recurrent neural radio anomaly detection. *CoRR*, abs/1611.00301, 2016.

[32] N. Tandiya, A. Jauhar, V. Marojevic, and J. H. Reed. Deep predictive coding neural network for RF anomaly detection in wireless networks. In *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2018.

[33] Z. Li, Z. Xiao, B. Wang, B. Y. Zhao, and H. Zheng. Scaling deep learning models for spectrum anomaly detection. In *Proc. ACM MobiHoc*, 2019.

[34] B. Nguyen, Z. Ge, J. Van der Merwe, H. Yan, and J. Yates. Absence: Usage-based failure detection in mobile networks. In *Proc. ACM MobiCom*, 2015.

[35] L. Bodrog, M. Kajo, S. Kocsis, and B. Schultz. A robust algorithm for anomaly detection in mobile networks. In *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2016.

[36] S. Rajendran, W. Meert, V. Lenders, and S. Pollin. Unsupervised wireless spectrum anomaly detection with interpretable features. *IEEE Transactions on Cognitive Communications and Networking*, 5(3), 2019.

[37] C. Zhang et al. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1409–1416, 2019.

[38] Grubbs' Test for Outliers. https://www.itl.nist.gov/div898/handbook/eda/section3/eda35h1.htm.

[39] N. Butakov. How to build robust anomaly detectors with machine learning. https://www.ericsson.com/en/blog/2020/4/anomaly-detection-with-machine-learning, Apr 2020.

[40] H. Abdi and L. Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[41] Z. Li et al. Identifying value in crowdsourced wireless signal measurements. In *Proc. of 26th ACM Int'l Conf. on World Wide Web (WWW)*, 2017.

[42] K. Grover, A. Lim, and Q. Yang. Jamming and anti-jamming techniques in wireless networks: A survey. *Int. J. Ad Hoc Ubiquitous Comput.*, 17(4), December 2014.

[43] A. G. Fragkiadakis, V. A. Siris, and A. P. Traganitis. Effective and robust detection of jamming attacks. In *Future Network Mobile Summit*, 2010.

[44] L. Wang and A. M. Wyglinski. A combined approach for distinguishing different types of jamming attacks against wireless networks. In *Proceedings of 2011 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, 2011.

[45] D. Karagiannis and A. Argyriou. Jamming attack detection in a pair of RF communicating vehicles using unsupervised machine learning. *Vehicular Communications*, 13, July 2018.

[46] R.P. Jover, J. Lackey, and A. Raghavan. Enhancing the security of LTE networks against jamming attacks. *EURASIP Journal on Information Security*, 2014, April 2014.

[47] M. Lichtman, T. Czauski, S. Ha, P. David, and J. H. Reed. Detection and mitigation of uplink control channel jamming in LTE. In *2014 IEEE Military Communications Conference*, 2014.

[48] NIST/SEMATECH e-Handbook of Statistical Methods. https://www.itl.nist.gov/div898/handbook/eda/section3/eda35h.htm, April 2012.