Performance Analysis of Deep Anomaly Detection Algorithms for Commercial Microwave Link Attenuation

Olof Engström*, Sahar Tahvili* ‡, Auwn Muhammad*, Forough Yaghoubi*, Lissy Pellaco†
*Global Artificial Intelligence Accelerator (GAIA), Ericsson AB
‡School of Innovation, Design and Engineering, Mälardalen University, Västerås, Sweden
†Division of Information Science and Engineering, KTH Royal Institute of Technology
{olof.engstrom, sahar.tahvili, auwn.muhammad, forough.yaghoubi}@ericsson.com
pellaco@kth.se

***Abstract*—Highly accurate weather classifiers have recently received a great deal of attention due to their promising applications. An alternative to conventional weather radars consists of using the measured attenuation data in commercial microwave links (CML) as input to a weather classifier. The design of an accurate weather classifier is challenging due to diverse weather conditions, the absence of predefined features, and specific domain requirements in terms of execution time and detection sensitivity. In addition to this, the quality of the data given as input to the classifier plays a crucial role as it directly impacts the classification output. However, the quality of the measured attenuation data in the CMLs poses a serious concern for different reasons, e.g. the nature of the data itself, the location of each link, and the geographical distance between the links. This mandates the adoption of a data preprocessing step before classification with the purpose to validate the quality of the input data. In this paper, we propose a data preprocessing framework which employs a deep learning model to (i) detect anomalies in the raw data and (ii) validate the measured CML attenuation data by adding quality flags. Moreover, the feasibility and possible generalizations of the proposed framework are studied by conducting an empirical case study performed on real data collected from CMLs at Ericsson AB in Sweden. The empirical evaluation indicates that the average area under the receiver operating characteristic curve exceeding 0.72 using the proposed data preprocessing framework.**

***Index Terms*—Microwave Link, Anomaly Detection, Artificial Intelligence, Time Series, Deep Learning, Data preprocessing**

I. Introduction

Many modern technological solutions rely upon an accurate estimate of current weather conditions. Examples include (i) the Internet of Things (IoT)-supported agriculture, in which accurate weather information is used to predict when to adjust irrigation systems, (ii) traffic management organizations, which use real-time weather information to determine whether to adjust speed limits and reroute traffic due to inclement weather and (iii) mapping services such as Google Maps which might incorporate current weather data to route vehicles in presence of stormy conditions. The solutions for rainfall detection used in industry can be categorized into three main approaches, i.e. rain gauges, weather radars, and commercial microwave links (CML). On one hand, rain gauges provide high accuracy but cover limited spaces. Although weather radars cover significantly larger areas but at the cost of a lower spatial resolution at the scale of 1 km^2. Moreover, both rain gauges and weather radars are costly due to extra equipment installation. On the contrary, using data from already installed CMLs for weather classification comes with the benefit of finer spatial resolution at zero additional costs [1]. Hitherto, several research results have shown the potential usage of CML attenuation data for hydro-meteorological applications [2]–[4]. With the recent development in the area of deep learning, many studies, such as [4], [5], focus on predicting the weather phenomena using deep neural networks. In particular, in [4] the authors use CML data as input to a deep learning model for weather classification. However, empirical evidence shows that CML data can be affected by anomalies in the raw data. Therefore, the CML data given as input to the weather classifier need to be preprocessed and validated before classification, otherwise, the output of the classifier is not to be trusted. In this paper, we propose, apply, and evaluate a framework that consists of a data preprocessing step which performs anomaly detection and data quality flagging, before weather classification. The proposed framework employs deep learning models and the main idea consists of predicting the normal behavior of the attenuation data and verifying whether the current attenuation data deviate from it or not. We consider both convolutional (CNN) and recurrent neural networks (RNN) as they are two efficient and suitable algorithms for learning temporal correlations in the data. For a better comparison of the mentioned models, we apply both approaches to learn normal CML's data behavior. The proposed framework is evaluated on real data collected at Ericsson. The results of our evaluation show that existing anomalies in attenuation data measured from CMLs can significantly impact the generalization performance of weather classification technologies based on CML data, such as the classifier proposed by Polz et al. [4]. In other words, this study shows the relationship between existing anomalies in the raw data and the number of wrong classifications (misclassification) made by the classifier. The performance improvement of the weather classifier after data prepossessing is left for future works. More specifically, this paper makes the following contributions:

- Employs two deep learning models that provide fast and accurate detection of anomalies in the measured attenuation data from CMLs.
- Adopts a dynamic threshold for anomaly detection, instead of relying upon static thresholds.
- Flags anomalous data as a quality warning before weather classification.
- Applies the proposed preprocessing framework to real data collected from CMLs.

II. Related Work

Data preprocessing has a significant impact on the generalization performance of supervised machine learning algorithms [6]. Data preprocessing can be performed in different forms of data cleaning, data integration, transformation, and data reduction [7]. Moreover, the data type, size, and quality can guide us to apply one or more of the mentioned data preprocessing methods. Hitherto, several anomaly detection techniques have been proposed, but as each technique is developed specifically based on the nature of the input data, providing a generic anomaly detection solution is challenging [8]. While anomaly detection approaches for independent data samples have been extensively studied, less work has been done on data samples with temporal correlation, e.g. time series problems. Existing anomaly detection algorithms that rely on distance-based measures, such as K-nearest neighbors, clustering, and density-based, have shown poor performance when applied to time series. Available anomaly detection techniques suitable for time series are divided into the following categories: similarity-based [9], clustering-based [10], classification-based [11], and model-based [12]. In the similarity-based algorithms, the resemblance between different time series is computed based on a distance function and an anomaly score is accordingly assigned [9]. In the cluster-based algorithms, time-series data are clustered using an appropriate technique and the resulting cluster centers are used to compute the anomaly scores of each data sample. For instance, Chandola et al. [10] use a k-medoids clustering method and use the inverse of a similarity measure to the closest medoid as the anomaly score. In the classification-based methods, a classifier is trained on an anomaly-free time series and its output is used to score the anomalous data sets. As an example, Hu et al. [11] compute the set of local measurements of a time series, such as kurtosis, variability, oscillation, and regularity, and use them as input features to a one-class support vector machine (OSVM), which provides a decision boundary to discriminate between anomalous and normal time series. In the model-based algorithms, time series models such as autoregressive (AR) and autoregressive integrated moving average (ARIMA) are used to estimate current values using previous time instances [12]. These methods are used to predict the normal behavior of time series and directly flag the data points as anomalous if the prediction error is higher than the threshold or report the error as the anomaly score. However, these traditional linear methods fail to capture the non-linearity in the data. Therefore, with the recent developments in deep learning, various neural network-based approaches started to be adopted with the purpose to learn non-linearity in the data, without the need to specify a model [13]–[16]. As a result, both CNNs and RNNs have shown to successfully learn the temporal correlation - whether linear or not - in the data. In particular, Munir et al. [17] proposed an unsupervised learning anomaly detection algorithm based on CNN. This approach employs CNNs to predict the next sample based on a time window and then compares the original value with the predicted one to detect the anomalies in real-time. In [18]–[20] a similar approach is used, but the learning algorithm is supervised and the prediction is performed utilizing a type of RNN, i.e. a long short-term memory (LSTM).

III. Proposed Approach

In this study, we propose a data preprocessing framework which employs deep learning models to (i) detect hidden anomalies and (ii) validate the measured CML attenuation data by adding quality flags before the weather classification.

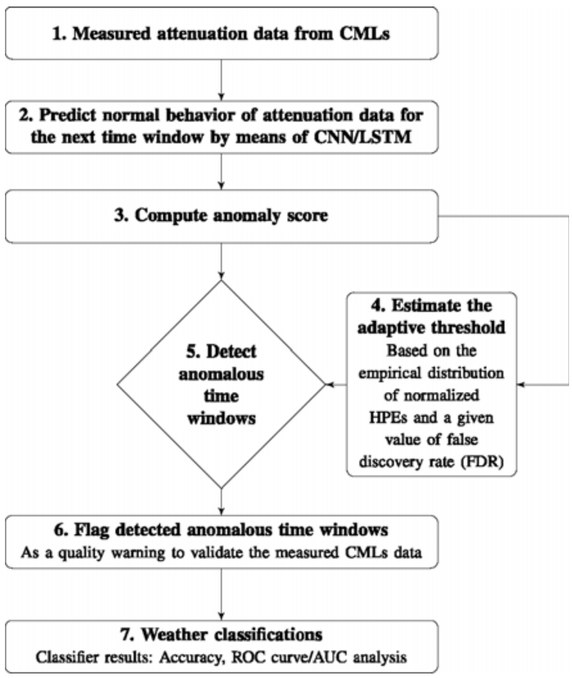


Fig. 1: Flowchart of the proposed framework.

The proposed framework is designed to be connected to the CMLs and to be used as a step prior to weather classification based on CMLs data [4]. Generally, the weather is classified on an hourly basis, therefore our purpose is to flag each hour as normal or anomalous before performing weather classification. However, our framework can be extended to time windows of any length by setting the desired prefixed time window in advance. Figure 1 gives an overview of the proposed framework by us. As can be seen, step 1 consists of the currently measured attenuation data from the CMLs, which is the input to the proposed framework. In step 2, a deep learning model (CNN or LSTM) is employed to predict the normal behavior of the measured attenuation data from the current data in the next time window. Step 3 consists of comparing, by means of a proper metric, the predicted normal behavior, and the observed behavior of the attenuation data in the time window. The output of this step consists of an anomaly score which indicates how close the prediction and the actual behavior are. Step 4 and 5 are focused on anomaly detection. In this regard, an adaptive threshold needs to be estimated. If the computed anomaly score in step 3 exceeds the threshold, then the data in the time window need to be flagged as anomalous in step 6, otherwise, the data is flagged as normal. In other words, exceeding the threshold means that in the time window the predicted normal behavior and the actual behavior of the data deviate. As a result, in the end, the entire data set is flagged as normal or anomalous. Then, the flagged data set enters the weather classification model, where the performance of the classifier can be measured using a different confusion matrix. It must be noted that such an approach guarantees robustness, as it relies on data over a time window, rather than on each single data sample. Moreover, the threshold used for anomaly detection is not fixed but dynamically adjusted, leading to higher flexibility and improved performance. Finally, the proposed framework can be easily adapted and added as a preprocessing step prior to weather data classification based on CML data, such as the one presented in [4]. As can be seen, the proposed framework serves as a validation process for the CML attenuation data by setting the data quality flag. In this study, we consider data points that do not follow normal behavior as anomalous. Thus, outliers, attacks, extreme observations, temperature anomaly[1] and obstacles[2] are considered as anomalies. In the upcoming section, we provide more details for each step illustrated in Figure 1.

A. The proposed solution for anomaly detection

We denote by $x^t \in \mathbb{R}$ the measured attenuation from CMLs at time t and we denote by X^t_{t-T} collection of measured attenuation data from time $t-T$ to t at a regular sampling time of Δ, i.e.,

$$X^t_{t-T} = \{x^{t-T}, x^{t-T+\Delta}, x^{t-T+2\Delta}, \ldots, x^t\} \tag{1}$$

Either a CNN or an LSTM is used to predict the attenuation data sample at time $t+\Delta$ under the assumption of normal behavior from a window of past attenuation values from $t-T$ to t, i.e. X^t_{t-T} (see Figure 2). This is achieved by training the CNN or the LSTM in a supervised fashion using anomaly-free data. The training set consists of fixed-size windows of length T of past attenuation measurements, paired with their immediately following measurement. Thus, a training sample consists of the pair $(X^t_{t-T}, x^{t+\Delta})$. The predicted attenuation data sample at time $t+\Delta$ is denoted by $\hat{x}^{t+\Delta}$. For the purpose of anomaly detection, the squared error between the predicted value and the actual value is computed, i.e.

$$e_{t+\Delta} = (x^{t+\Delta} - \hat{x}^{t+\Delta})^2 \tag{2}$$

Then, the squared errors are aggregated over a time window of length $\mathcal{T}$ and the mean square error (MSE), which we use as the anomaly score, is computed for each time window of length $\mathcal{T}$, i.e.

$$MSE = \frac{1}{N}\sum_{i=1}^{N} e_i \tag{3}$$

where N is the number of samples in each time window of length $\mathcal{T}$.

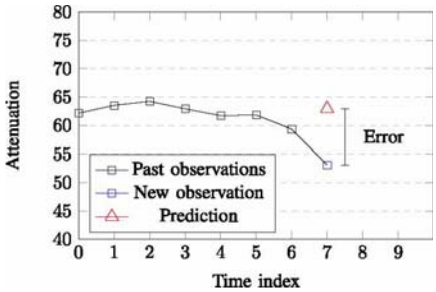


Fig. 2: Example of prediction performed by CNN. The past observations are used to predict the next data point, which is then compared to the actual captured value.

B. Adaptive threshold detection

The threshold on the prediction error (distance between the forecasted attenuation value and the observed attenuation value) is then used to flag observations as anomalous if the error exceeds the threshold. The prediction error from a deep learning model such as LSTM in anomaly detection follows a Gaussian distribution as proposed by Malhotra et al [21], where, they detected an optimal threshold in a supervised setting by maximizing the F_β measure on a calibration data set. In our case, we used an empirical approach Bayesian FDR [22] to estimate the threshold for a given false discovery rate parameter (i.e., alpha = 1%, 5%, and 10%). We first transformed the prediction errors into $Z-Score$ by standardizing the errors i.e.,

$$Z - Score = \frac{X-\mu}{\sigma} \tag{4}$$

The p-value for each data point i.e., its odds being anomalous, is calculated based on its Z-Statistics for a 2-sided Normal distribution test. The intuition behind this step is to define a measure that can be assigned as a p-value to test its significance.

To understand the intuition behind this, consider each data point as an experiment of tossing a coin and test-statistics ($Z-$

[1]Means a departure from a reference value or long-term average.

[2]When an obstacle stands in a signal's way, the signal may pass through the object or be absorbed by the object.

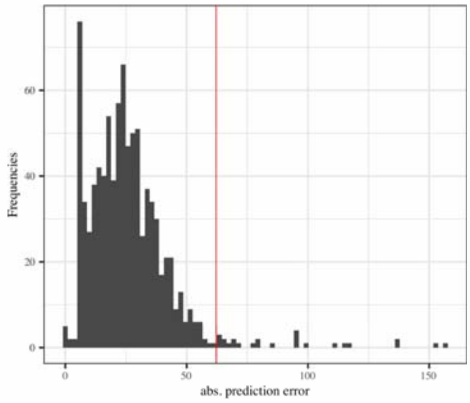


Fig. 3: A histogram of absolute prediction errors and a threshold (as a red vertical line) is detected by the Bayesian FDR approach for a pre-specified value of FDR $\alpha = 0.01$ in this case.

$Score$ in Equation 4) as a measure to decide a priori whether or not it will be anomalous (testing of hypothesis) depending on its position in unit normal. The success probability of the coin represents the unknown fraction of normal data points, and the outcome of each such experiment i.e., tossing a coin can be observed only through p-values of prediction error $Z - Score$ of the predicated data point. If the prediction error of a data point is high relative to its mean of observed error, its odds being anomalous will also be higher. To estimate the threshold on given probabilities and a pre-specified value of alpha, the Bayesian FDR sort the p-values and find a threshold that maximizes the fraction of null hypotheses (that data points are normal). In Figure 3, we provided a threshold value of 62.174 on normalized prediction error for a given alpha = 0.01 i.e., expected rate of falsely rejecting null hypothesis. More detail about the method can be found in [22]. Figure 3 shows a histogram of absolute prediction errors and the estimated threshold as a red vertical line. This approach can be used in the online system by updating the mean and standard deviation parameters used in $Z - Score$ test statistics depending on the model update policy of an online system.

IV. Industrial Case Study

As already stated, anomalies in the measured attenuation from CMLs might seriously affect the performance of the weather classifiers. In this section, we analyze the relationship between detected anomalies and the number of misclassifications.

1) Data: In order to analyze the feasibility of the proposed framework, we designed an industrial case study at Ericsson, by following the proposed guidelines in [23] and [24]. We use a data set consisting of real data collected from five CMLs over a period of ten months. We consider attenuation data measured every 10 second, i.e. $\Delta = 10$ s, and for training the forecasting models we consider time windows of 5 minutes, i.e. $T = 5$ min, which correspond to 30 measurements and scale the data to zero mean and unit variance. We aggregate the squared errors and compute the MSE over time windows of the length of one hour, i.e. $\mathcal{T} = 1$ hr.

For weather classification, we use the approach proposed in [4], which uses a classifier based on a one-dimensional (1D) CNN to distinguish between precipitation and non-precipitation events on an hourly base. Due to space limitations, we refer the reader to [4] for more detailed information regarding the architecture of the employed CNN. An overview of the data set after being classified into either precipitation or non-precipitation can be seen in Table I.

	Precipitation			Non-precipitation		
Link	Mean	Std	Count	Mean	Std	Count
1	48.34	0.80	6840	45.73	3.04	2736000
2	54.76	1.50	15480	53.41	1.30	2727360
3	50.37	1.05	12960	49.80	0.89	2729880
4	37.10	0.92	8280	35.69	0.69	2734560
5	52.39	0.59	5760	49.47	2.60	2737080

TABLE I: Statistics of the CML attenuation data set grouped by weather class (precipitation versus non-precipitation) as assigned by the weather classifier.

2) The ground truth (GT): To evaluate the performance of the proposed approach a ground truth has been derived by using information from the Swedish Meteorological and Hydrological Institute (SMHI). The SMHI data is used to determine the misclassifications of the weather classifier. SMHI data is available for two out of the ten months, therefore we use such two-month data as a test set, while we use the remaining eight-month data as a training set. The validation set size consists of 20% of the training data set.

3) Parameters: The following parameters and architectures are chosen for the CNN and LSTM.

- The CNN consists of two blocks of 1-D convolutions with rectified linear units (ReLU) activation functions, 32 feature maps, and a kernel size of 3. Both blocks are followed by a size of 2 max-pooling. Following the convolutional blocks are 16 fully connected units with ReLU, and finally a single linear output unit. No zero-padding of the inputs to the convolutional and max-pooling layers were used.
- The LSTM network consists of two stacked LSTM-layers with 35 cells each, where the entire hidden state of the first layer is propagated to the second layer. The two stacked LSTM-layers are then followed by a single linear output unit.

Both networks were trained using Adam optimizer [25] with a learning rate of 0.001 until MSE on a 20% validation split ceased to decrease.

V. Results and Discussion

Anomaly detection generally is categorized as an imbalanced problem, which means the ratio of the normal and anomalous data is not equal. In this context, the receiver

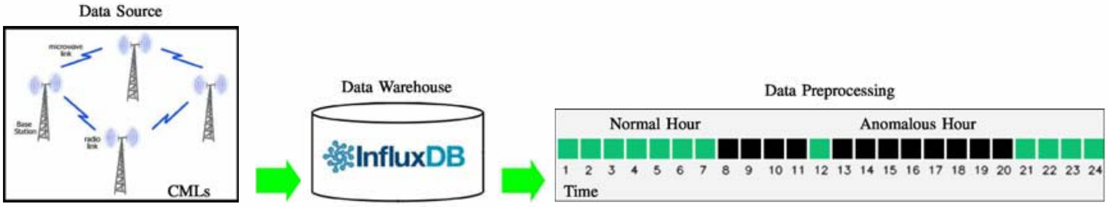


Fig. 4: Functional architecture of the proposed framework. Black blocks represent the flagged anomalous hours and green blocks represent normal hours.

operating characteristic (ROC) curve and the area under the ROC curve (AUC) is the most commonly used metrics for comparing classification models. As stated earlier, the meteorological observations to derive the GT for weather classification between precipitation and non-precipitation events were extracted using data from SMHI. The following confusion matrices can be interpreted as follows:

- True Positive: detected anomaly corresponds to a misclassification.
- False Positive: detected anomaly corresponds to the absence of misclassification.
- True Negative: the absence of anomaly corresponds to the absence of misclassification.
- False Negative: the absence of detected anomaly corresponds to a misclassification.

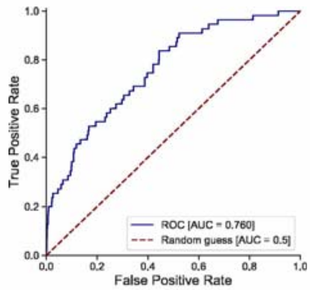
Table II shows the obtained results against the GT for the five considered CMLs in terms of MSE and AUC as performance metrics.

	MSE		AUC	
Link	CNN	LSTM	CNN	LSTM
1	0.0015	0.0018	0.760	0.832
2	0.0068	0.0129	0.729	0.698
3	0.0265	0.0225	0.725	0.706
4	0.0386	0.0449	0.720	0.797
5	0.0022	0.0045	0.699	0.621
Mean	0.0151	0.0173	0.726	0.731

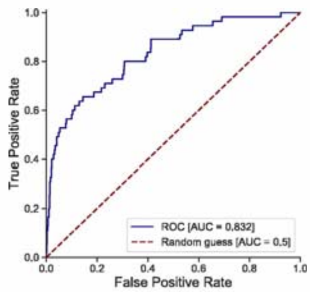
TABLE II: MSE and AUC values for anomaly detection, for five CMLs using CNN and LSTM.

As can be seen in Table II, the AUC values differ from link to link, which indicates that the location of the CMLs plays a key role in the appearance of anomalies. Moreover, the results show that on average the CNN has a lower MSE, but the LSTM has a slightly higher AUC, which indicates better detection capabilities. Moreover, the empirical evaluation presented in Table II indicates that the average AUC exceeding 0.72 using the proposed data preprocessing framework in this paper, which shows a strong effect on the performance of the weather classifier.

Figure 5 mirrors the evaluation of the proposed framework using both LSTM and CNN on link 1 of Table II. This link is characterized by the largest number of misclassifications and for this link, the highest AUC is achieved, both in case of prediction performed by the CNN (Figure 5a) and by the LSTM (Figure 5b). These results indicate that there is a direct relationship between the hidden anomalies in the data measured from CMLs and the performance of the 1D CNN weather classifier. Therefore, adopting this approach as a data preprocessing step prior to classification can have a substantial and direct impact on the generalization performance of the classifier. In this study, all abnormal behavior of the data is considered as an anomaly, therefore the detected anomalies need to be handled carefully. In fact, it might be the case that the detected anomalies represent a new weather class not embedded in the classifier or a combination of the weather classes already embedded, e.g., rain and sun simultaneously. Figure 4 shows an overview of the proposed framework which is connected to the Influx database at Ericsson that provided the CML attenuation data for the case study. Generally, for each CML the following information is stored in the



(a)



(b)

Fig. 5: ROC curve for link 1 obtained (a) with the CNN and (b) with the LSTM.

database: link ID, timestamp, received power, and transmitted power. Mostly, existing weather classification models classify the weather for each hour. Therefore, to enable immediate compatibility with existing weather classifiers, the presented framework provides data quality flagging on an hourly basis. This data quality flagging can tremendously help the network operators when deciding whether to amplify, damp, or reroute a signal on the basis of the weather classifier output. If the data given as input to the classifier is mostly flagged as anomalous, then the operator knows whether to trust the output of the classifier or not and can perform a manual inspection before acting on the signal. However, the proposed framework is not limited to these aspects and can be easily adjusted to different settings. Flagging the anomalous hours is one way to employ the proposed approach. The detected anomalies can be eliminated, interpolated, or replaced by a new value by means of smoothing.

VI. Future Work

A possible future direction consists of analyzing the correlation between the presence of anomalous data in a CML link and the geographical location of the link. By doing so, it would be possible to identify which geographical characteristics, e.g. vicinity to the sea, affect the quality of the attenuation data, and take proper countermeasures. The main threat to the validity of this study is the generalization of the proposed approach and findings. The proposed framework has been applied to just one industrial case study and it should be applied to other similar contexts using different classifiers. Furthermore, handling the detected anomalies automatically e.g. through applying smoothing methods, is another direction of this study.

VII. Conclusion

Data quality plays a critical role in the performance of machine learning models. We addressed the issue of data quality in the context of weather classification from CMLs attenuation data. The main goal of this study was to design, implement, and evaluate an automated framework that consists of a data preprocessing step before weather classification. The proposed framework employs deep learning models, performs anomaly detection, and validates the measured CMLs data by adding data quality flags. To this end, we made the following contributions: (i) detecting the hidden anomalies in the raw measured attenuation data from CMLs, (ii) allowing a dynamic threshold for anomaly detection, and (iii) flagging detected anomalous time windows (data quality warning) to validate the raw CMLs data before weather classification. The empirical evaluation indicates that the average area under the receiver operating characteristic curve exceeding 0.72 using the proposed data preprocessing framework.

References

[1] F. Yaghoubi, J. Chen, A. Rostami, and L. Wosinska, "Mitigation of rain impact on microwave backhaul networks," in *2016 IEEE International Conf. on Communs. Workshops (ICC)*. IEEE, 2016, pp. 134–139.

[2] A. Overeem, H. Leijnse, and R. Uijlenhoet, "Measuring urban rainfall using microwave links from commercial cellular communication networks," *Water Resources Research*, vol. 47, 12 2011.

[3] M. Graf, C. Chwala, J. Polz, and H. Kunstmann, "Rainfall estimation from a german-wide commercial microwave link network: Optimized processing and validation for one year of data," *Hydrology and Earth System Sciences Discussions*, vol. 2019, pp. 1–23, 2019.

[4] J. Polz, C. Chwala, M. Graf, and H. Kunstmann, "Rain event detection in commercial microwave link attenuation data using convolutional neural networks," *Atmospheric Measurement Techniques Discussions*, vol. 2019, pp. 1–22, 2019.

[5] A. G. Salman, B. Kanigoro, and Y. Heryadi, "Weather forecasting using deep learning techniques," in *2015 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2015, pp. 281–285.

[6] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Data preprocessing for supervised learning," *International Journal of Computer Science*, vol. 1, pp. 111–117, 01 2006.

[7] S. Tahvili, L. Hatvani, E. Ramentol, R. Pimentel, W. Afzal, and F. Herrera, "A novel methodology to classify test cases using natural language processing and imbalanced learning," *Engineering Applications of Artificial Intelligence*, vol. 95, pp. 1–13, August 2020.

[8] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 15:1–15:58, Jul. 2009.

[9] P. Protopapas, J. Giammarco, L. Faccioli, M. Struble, R. Dave, and C. Alcock, "Finding outlier light curves in catalogues of periodic variable stars," *Monthly Notices of the Royal Astronomical Society*, vol. 369, no. 2, pp. 677–696, 2006.

[10] V. Chandola, V. Mithal, and V. Kumar, "Comparative evaluation of anomaly detection techniques for sequence data," in *2008 Eighth IEEE international conference on data mining*. IEEE, 2008, pp. 743–748.

[11] M. Hu, Z. Ji, K. Yan, Y. Guo, X. Feng, J. Gong, X. Zhao, and L. Dong, "Detecting anomalies in time series data via a meta-feature based approach," *IEEE Access*, vol. 6, pp. 27 760–27 776, 2018.

[12] H. Izakian and W. Pedrycz, "Anomaly detection and characterization in spatial time series data: A cluster-centric approach," *IEEE Trans. on Fuzzy Systems*, vol. 22, no. 6, pp. 1612–1624, 2014.

[13] R. Frank, N. Davey, and S. Hunt, "Time series prediction and neural networks," *Journal of intelligent and robotic systems*, vol. 31, no. 1-3, pp. 91–103, 2001.

[14] P. Zhang, "Time series forecasting using a hybrid arima and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[15] A. Borovykh, S. Bohte, and C. Oosterlee, "Conditional time series forecasting with convolutional neural networks," *arXiv preprint arXiv:1703.04691*, 2017.

[16] J. Connor, D. Martin, and L. Atlas, "Recurrent neural networks and robust time series prediction," *IEEE trans. on neural networks*, vol. 5, no. 2, pp. 240–254, 1994.

[17] M. Munir, S.Siddiqui, A. Dengel, and S. Ahmed, "Deepant: A deep learning approach for unsupervised anomaly detection in time series," *IEEE Access*, vol. 7, pp. 1991–2005, 2019.

[18] T. Lee, J. Gottschlich, N. Tatbul, E. Metcalf, and S. Zdonik, "Greenhouse: A zero-positive machine learning system for time-series anomaly detection," 2018.

[19] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in *Proceedings*. Presses universitaires de Louvain, 04 2015.

[20] L. Pellaco, N. Singh, and J. Jaldén, "Spectrum Prediction and Interference Detection for Satellite Communications," *arXiv e-prints*, p. arXiv:1912.04716, Dec. 2019.

[21] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "Lstm-based encoder-decoder for multi-sensor anomaly detection," *arXiv preprint arXiv:1607.00148*, 2016.

[22] M. Newton, A. Noueiry, D. Sarkar, and P. Ahlquist, "Detecting differential gene expression with a semiparametric hierarchical mixture method," *Biostatistics*, vol. 5, no. 2, pp. 155–176, 2004.

[23] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Software Engineering*, vol. 14, no. 2, p. 131, 12 2008.

[24] S. Tahvili, "Multi-criteria optimization of system integration testing," Ph.D. dissertation, Malardalen University, 12 2018.

[25] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv e-prints*, p. arXiv:1412.6980, Dec. 2014.