IMPLEMENTATION OF BAYESIAN APPROACHES IN 5G/6G FOR CELLULAR COMMUNICATION USING MULTIPLE TIME SERIES MODELS

Omer Mohammed Khodayer AL-DULAIMI[1], Aymen Mohammed Khodayer AL-DULAIMI[2], Alexandra MAIDUC OSICEANU[3], Mohammed Khodayer AL-DULAIMI[4]

This research investigates the time-series dynamics of the Earnings per share (EPS) Radio Bearer Setup Failure Rate and the applicability of certain commonly used Time-Series prediction models. The regular part-time series prediction and the outliers' prediction are two major issues of proactive network management that have been explored. We have utilized Holt-Winters Exponential Smoothing, extreme Gradient Boosting (XGBoost), Support Vector Regression (SVR), Python Bayesian dynamic linear mode (PyDLM), and Seasonal Auto-Regressive Integrated Moving Average with exogenous factors, (SARIMAX) to predict the regular component. Median Absolute Error, Mean Absolute Error, Mean Square Error, and Root Mean Square Error were used to examine the error performance. The prediction of outliers has been suggested as a two-stage process.

Keywords: Support Vector Regression (SVR), Bayesian Dynamic Linear Mode (PyDLM), Operation Support System (OSS), Hidden Markov Model (HMM), Seasonal Auto-Regressive Integrated Moving Average with Exogenous Factors (SARIMAX).

1. Introduction

The rapid increase in user-plane and control-plane traffic, together with the ongoing progress in technology, has made the implementation easier and has led to the development of crucial new capabilities based on predictive data analysis. The Operation Support System (OSS) is the most important component of mobile networks. The following entities [1] demonstrate the interconnected nature of an OSS for cellular communication systems up to Long Term Evolution Advance (LTE-A) [2], a group or organization responsible for compiling quantitative data on characteristics of network deployment such as performance, fault tolerance, incident rate, alternatives for network management (cellular

[1] Ph.D. student, Dept. of Telecommunication Engineering, University POLITEHNICA of Bucharest, Romania, e-mail: omer_mohammed.al@stud.etti.upb.ro
[2] PhD eng., Dept. of Technical Communication Engineering, Al- Farahidi University College; Iraq, e-mail: aymenaldulaimi@uoalfarahidi.edu.iq
[3] PhD, Director for management of scientific research, University POLITEHNICA of Bucharest, Romania, e-mail: sanda.maiduc@upb,ro
[4] PhD eng., Dept. of Computer Engineering, Al-Rafidain University of college, Iraq, e-mail: mohammed.khudhaer.elc@ruc.edu.iq

network element interaction), database management system (DBMS), Graphical User Interface (GUI) for Admins and Techs in the Networking Industry. Fig. 1 is a graphic representation of the aforementioned system design. Opportunities to expand OSS capabilities through the predictive function that can provide the capacity to predict different mobile network Key Performance Indicators (KPIs) are the focus of this study [3].

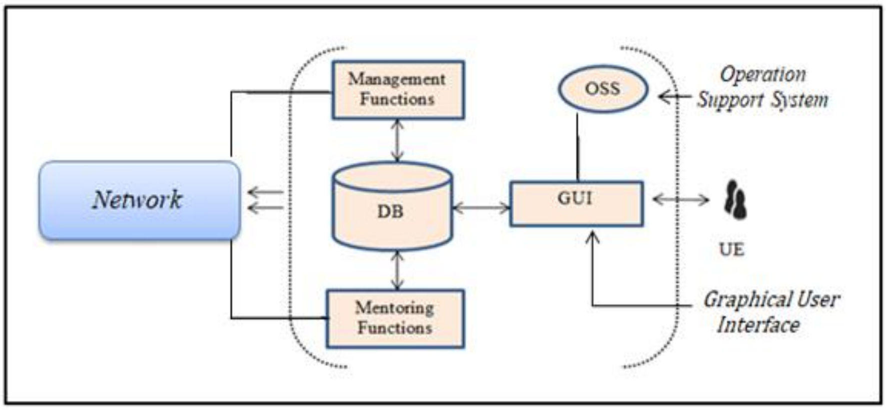

Fig. 1. Long term evolution and long-term evolution advance operation support system (OSS) architecture

Network performance indicators are typically collected by cellular network operators as time series, with aggregated data from the Returns Authorization Number (RAN) and Core Network (CN) often having their own unique periodicity. Third-order arrays can be used to express key performance indicators, with each direction corresponding to either a different Key Performance Indicators (KPI) value, a different time instant [4], or a different data aggregation. By simplifying the data structure into a matrix, we may integrate KPIs over aggregation objects to achieve network-level performance [5].

2. Methodologies and Research Materials

2.1 Models with Continuous State Space

The continuous state-space model and its partial application, the dynamic linear model (DLM) [6], are two of the most abstract methods for characterizing time-dependent time series. DLM is a special case of Hidden Markov Mode, (HMM) [7], which is also a definition of DLM.The following is how the model of time series data is expressed due to the presence of strict linearity constraints:

$$y_t = x_t + v_t \tag{1}$$

In this formula, random fluctuations are denoted by $v_t \sim N(0.v_t)$, is a component of current state represented by x_t, [8]:

$$x_t = \theta_t^T f \tag{2}$$

Using the following definitions for $\boldsymbol{f}$ (a regression vector) and θ_t (a vector of the states),

$$\theta_t = G_t \theta_{t-1} + w_t \tag{3}$$

Where G_t is a matrix representing transitions between states. and S is a vector representing random changes in state. $w_t \sim N(0.W_t)$. By combining the regression vectors into a matrix and the transition matrices into a block-diagonal form, the DLMs allow for the superposition of numerous simple models into a complex [9].

$$\begin{pmatrix} f_1(t) \\ f_2(t) \\ f_k(t) \end{pmatrix} \tag{4}$$

$$G_t = \begin{pmatrix} G_1(t) & 0 & 0 \\ 0 & G_2(t) & 0 \\ 0 & 0 & G_k(t) \end{pmatrix} \tag{5}$$

where k is the total amount of unique DLMs the following parts are often represented by separate DLMs, as illustrated in Fig 2.[10]:

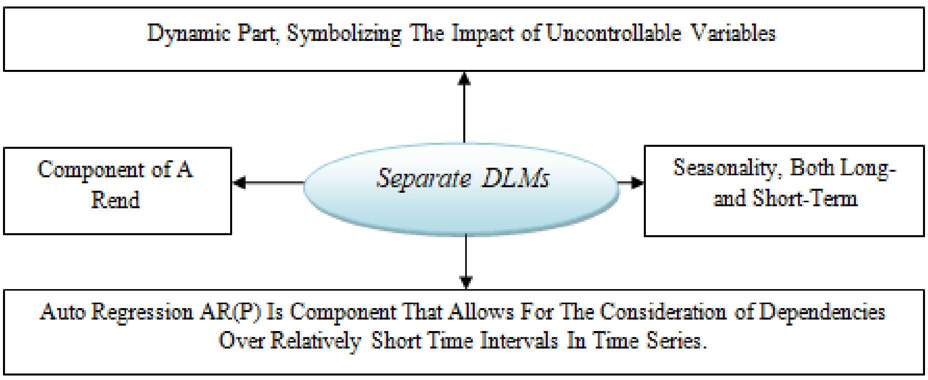

Fig. 2. Separate Bayesian dynamic linear mode (DLM)

Two phases make up DLM model training,as divided in below.[11]:

1) First step: filtering step, kalman filter coefficients, and gernerated independently of any previously predicted states.

2) Second step : smoothing step,raunch-tung-striebel algorithm, and prevoius estimates of states are refined in light of the latest computations.

These models, which are grounded in Bayesian theory, serve as a standard against which other time-series models with Markovian dynamics can be evaluated[12].

2.2 A Holt-Winter Technique

Exponential smoothing is an example of a traditional linear European Telecommunication Standard (ETS) technique. Due to the needs for seasonal component analysis, we explore the triple exponential smoothing, commonly known as Holt- Winter's during this study. An additive decomposition is preferred here, as stated by [13]. At each given instant t, the value of the time series can be expressed as:

$$\hat{y}_{t+h} = l_t(\varphi + \varphi^2 + \cdots + \varphi^h)b_t + S_{t+h-m(k+1)} \tag{6}$$

Where$0 < \varphi < 1$ is the trend damping parameter (for undamped trends, this parameter is equal to 1), S_t seasonal component, b_t the slope of the trend is; *h* is the integer value that shows, how many samples ahead the forecast should be done; m is the number of samples per one period (seasonality); and *k* is the integer part of (*h* - 1) / *m*, that guarantees usage of the belonging to one year only seasonal indexes estimations [14]. At instant *t*, the time series level is:

$$l_t = \alpha(y_t + s_{t-1}) + (1-\alpha)(l_{t-1} + b_{l-1}) \tag{7}$$

Where $0 \leq \alpha \leq 1$ is a smoothing parameter of the baseline:

$$b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1} \tag{8}$$

The seasonal component can be expressed as: where $0 \leq \beta \leq 1$ is a trend smoothing parameter [15].

$$S_t = \gamma = (y_t - l_{t-1} - b_{t-1}) + (1-\gamma)S_{t-m} \tag{9}$$

Where$0 \leq \gamma \leq 1$ is the seasonal component smoothing. During the model training phase, one of the maximum likelihood estimation methods (MLE) [16] can be utilized to determine the unknown coefficients,$\alpha.\beta$ and γ similarly, this model can be viewed as an illustration of DLM [17], suggesting that it was trained using Bayesian filtering and smoothing techniques.

2.3 Calculation SARIMAX Scheme

The seasonal autoregressive integrated moving average (SARIMAX) is another popular linear method, described by the (*p.d.q*) X (*P.D.Q.s*) form, where *p* and *P* are the necessary number of backward samples and periods of the non-seasonal and seasonal components of the time series, *d* and *D* are the orders of differentiation necessary to reduce to a stationary form of observation and seasonal component, and q and Q are the necessary number of backward samples The following equation describes the connection between the observation value and the approximation error:

$$\varphi_p(B^s)\varphi_p(B)\nabla_s^D\nabla^d y_t = \theta_q(B)\theta_Q(B^s)\varepsilon_t \quad (10)$$

Where is the seasonal $\varphi_p(B^s)$=$(1-\varphi_1 B^s - \cdots - \varphi_p B^{sP})$ part of the autoregressive (AR) model's component of the order $P. \varphi_p(B)$=(1-$\varphi_1 B - \cdots - \varphi_P B^P$) is the non-seasonal part of the AR component of the order P,$\nabla_s^D = (1-B^s)^D$ and $\nabla^d = (1-B)^d$ is nabla operators for seasonal and non-seasonal components of the orders D and d respectively, $\theta_Q(B^{s)} = (1-\theta_1 B^s - \cdots - \theta_Q B^{Qs})$ is the seasonal component of the moving average (MA) of the order Q, $\theta_q(B^{s)} = (1-\theta_1 B - \cdots - \theta_q B^q)$ is the non-seasonal component of the MA of the order q, and B is the lag operator. φ is the trend damping parameter. Each iteration of the algorithm described in [18], sets of unknown coefficients of the AR and MA polynomials are calculated using one of the maximum likelihood estimation (MLE) optimization algorithms, allowing for automatic selection of the parameters *p, d.q.P.D Q*. This technique is an example of a DLM model, and it can be expanded to include more exogenous variables (the SARIMAX model) [19].

2.4 Traditional Methods of Machine Learning Prediction

Instead of using a predetermined statistical model, like in the aforementioned method, a regression tree can be used as a more heuristic approach to anomaly detection. Extreme Gradient Boosting (XGBoost) [20] is a popular technique for classification and regression applications, and we employ it in our study. The approach relies on CART, which is an implementation of a tree-based model for classifying and predicting data. The following benefits can be derived from this analysis, a lightning-fast execution, there is no requirement for normalizing data, and handling non-linear dependencies. The CART's drawbacks stem from the fact that a fixed number of dependent variables must be used in order to do a regression analysis, as illustrated in Fig.3.

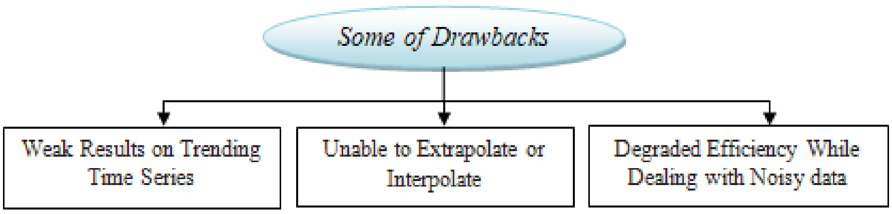

Fig. 3. The CART's drawbacks stem

The Support Vector Regression (SVR) technique can be utilized. SVR's fundamental notion is to find the minimum possible error bound for a regression. From what we can gather in the literature [21], we might gain as shown in Fig 4:

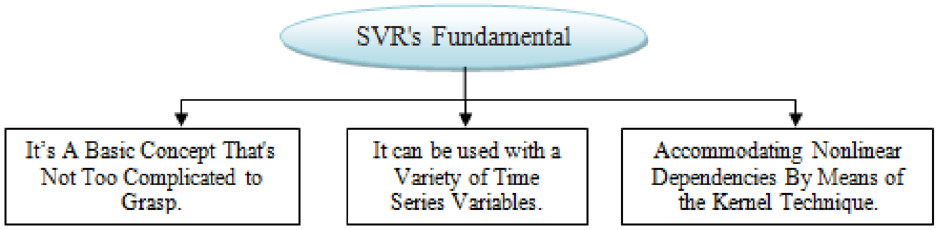

Fig. 4. Support vector regression (SVR) technique

This approach, however, is found to have the following drawbacks as illustrate in Fig.5:

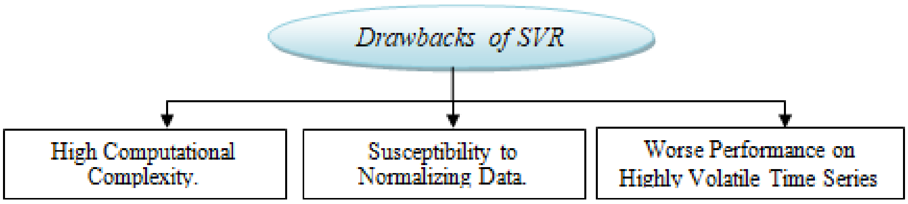

Fig. 5. Drawbacks of support vector regression (SVR)

2.5 Techniques for Identifying and Predicting Abnormal Events

Predicting potential events based on KPI is a significant difficulty in cellular network performance analysis. The available literature provides a wide variety of methods for accomplishing this. Defining internal dependencies and patterns in the data is one such thing. Here are some examples of such methods: XGBoost's non-linear models based on regressive trees [22], long short-term memory (LSTM) networks, Trans- formers [23], and artificial neural network (ANN) auto encoders [24]. In this research, we present a two-pronged strategy for anticipating anomalies as shown in Fig.6:

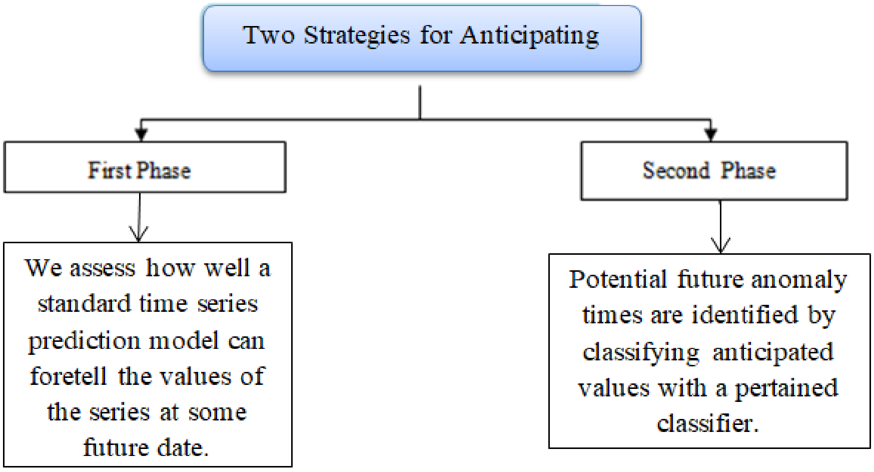

Fig. 6. Forecasting strategy for different situations

So, rather of trying to foretell the future shape of the time series, we're more interested in pinpointing critical junctures when accidents are more likely to occur, as illustrate in Fig. 7.

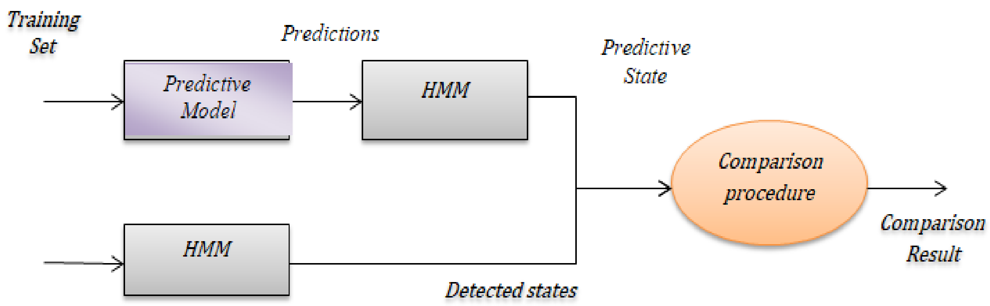

Fig. 7. The block scheme of outliers' investigation approach

The notion that statistical properties of data of any kind may be represented by a Gaussian Mixture motivates the use of Gaussian Mixture Hidden Markov Model (HMM) in the implementation of the pretrained classifier. Therefore, the Gaussian Hidden Markov Model (HMM) is the best option, with its parameters determined by the Baum-Welsh Expectation-Maximization (EM) technique [25].

3 Prediction of the Regular Distribution Based on the Results of Developing Methodological

Table 1 display the outcomes of regular-part prediction experiments using the models presented above. We use the following free and open-source Python 3 librairies in our study, module (Holt-Winter's model), module (Hidden Markov

Model), Pmdarima (SARIMAX), X boost (XGBoost), PDLM (Continuous State-Space Model). XGBoost, Bayesian dynamic linear mode (PyDLM), Seasonal Auto-Regressive Integrated Moving Average with eXogenous factors SARIMAX, and SVR all take into account the current time of day and year as exogenous variables. Auto ARIMA is used to get the SARIMA model's super parameters [26]. Some common measures of efficiency include the Mean Absolute Error, the Mean Squared Error (MSE), and the Root Mean Squared Error (RMSE). The cross-validation procedure is divided into 12 sub-steps. There are 2019 values in the data collection.

Table 1

Results of the predictions of the regular part

Model name	*Median Absolute Error*	*Mean Absolute Error*	*MSE*	*E*
Holt-Winter's (additive trend, additive seasonality)	0.040	0.074	0.055	**0.173**
SVR	0	0.087	0.057	**0.179**
XGboost	-	0.043	0.052	**0.163**
PyDLM	0.074	0.105	0.068	**0.198**
SARIMAX (2, 0, 1) x (1, 0, 2, 24)	**0.064**	**0.098**	**0.060**	**0.189**

The outlier prediction issue makes use of the same features, and the dataset has been updated to include 2017. A true positive rate is used as a measure of the algorithms' efficacy [27]. The outcomes are shown in tables 2 and 3. Because no adequately detectable outliers were found, SARIMAX model results are not presented. When implementing a discrete-state HMM, we turned to the pomegranate module for Python 3.

Table 2

Results of predictions of day and hour of outliers' occurrence

Modelname	No. detected outliers	No. predicted outliers	True positive alarms	No. false alarms	No. missed alarms
Train years: 2017, 2018; test year: 2019.					
Holt-Winter's	551	524	523	1	23
SVR	551	2504	551	1953	0
XGboost	551	392	392	0	1
DLM	551	3174	325	2849	226
Train years: 2017, 2019; test year: 2018.					

Holt-Winter's	1760	2157	1759	398	1
SVR	1760	4186	283	3903	1477
XGboo t		254	1759	695	1
PyDLM	1760	539	38	501	1722
Train years: 2018, 2019; test year: 2017.					
Holt-Winter's	1419	1015	1014	1	405
SVR	1419	4228	0	4228	1419
XGboost	1419	1342	1341	1	78
PyDLM	1419	855	2	853	1417

Table 3

Results of predictions of the day of outliers' occurrence

Mode lname	No. detected outliers	No. predicted outliers	True positive alarms	No. false alarms	No. missed alarms
Train years: 2017, 2018; test year: 2019.					
Holt- Winter's	82			0	4
SVR	82	2	82	155	0
XGboost	82			0	20
PyDLM	82	3	81	235	1
Train years: 2017, 2019; test year: 2018.					
Holt-Winter's	170	195	170	25	0
SVR	170	351	164	187	6
XGboost	170	220	170	50	0
PyDLM	170	90	44	46	126
Train years: 2018, 2019; test year: 2017.					
Holt-Winter's	134	111	110	1	24
SVR	134	363	132	231	2
XGboost	134	130	129	1	5
PyDLM	134	152	57	95	77

Table 4 displays the average cross-validation set results, rounded to the nearest integer. To begin, it is generally agreed that the regular component prediction results are comparable among models. Second, Holt-model Winter's and the XGBoost algorithm can be seen as the most suitable options for the task of outliers prediction since they produce the highest number of correctly anticipated alarms while generating the fewest false alarms.

Table 4

Summarized results of outliers prediction

Model name	No.detected outliers	No. predicted outliers	True positive alarms	No. false alarms	No. missed alarms
Holt-Winter's	1243	1232	1099	133	145
SVR	1	3639	278	3361	965
XGboost		1396	1164	232	79
PyDLM	1243	1523	121	1401	1122
Holt-Winter's	127	128	119	9	9
SVR	127	317	126	191	3
XGboost		137	120	17	9
PyDLM	127	186	60		68

4. Predictive Functionality in Communication Systems: A Discussion of Results, Development, and Implementation

As we've seen, there are two possible routes for incorporating predictive capability into existing LTE/LTE-A networks.

- First possible, adding new features and entities into the preexisting OSS framework. New tendencies in the development of Database Management System (DBMS) engines justify this strategy (distributed storage, insertion of artificial intelligence AI) [28]. Fig 8 provides a graphic representation of this method.

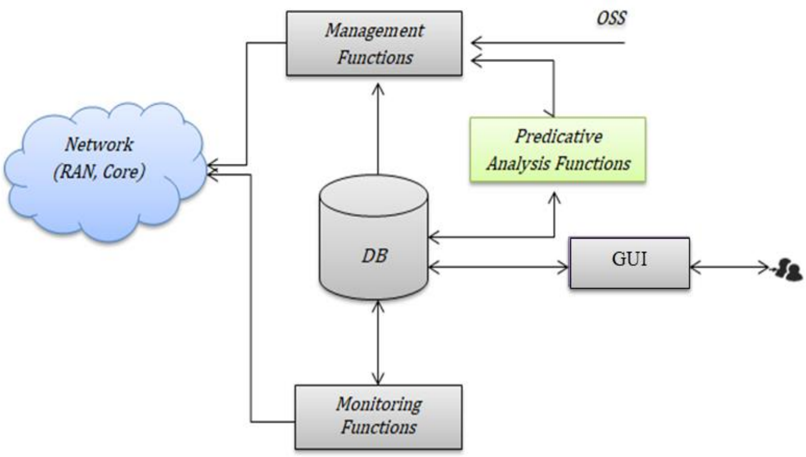

Fig. 8. System Architecture of OSS with Embedded PredictiveAnalysis Tool

- Second Possible, making new, separate systems on top of existing open source ones Fig. 9. The implementation of OSS would benefit more from this method. In addition, the progress being made in network function virtualization makes such an implementation attractive.

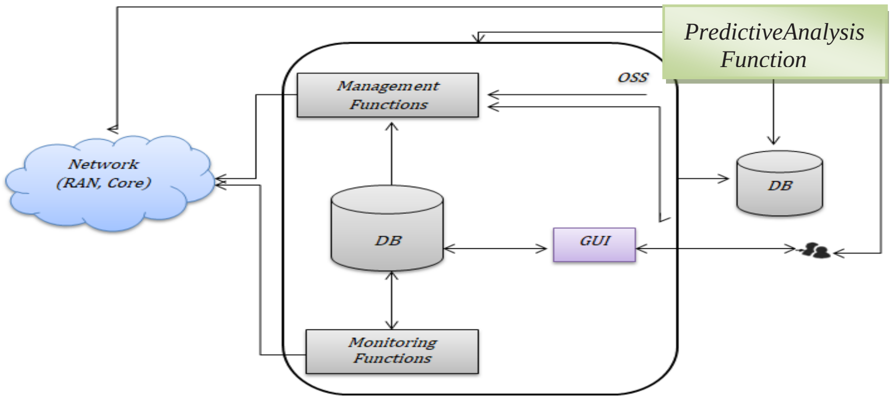

Fig.9. System architecture of OSS with additional predictiveanalysis subsystem

All of these approaches aren't limited to standalone 5G systems (New Radio and LTE core). 3GPP [29] has already advocated including such features as part of the system architecture for standalone 5G. Because of the need for increased data rates and lower latency for essential 5G applications, this feature, known as Network Data Analytics Function (NWDAF) Fig. 10, is crucial (industrial IoT, telec, smart homes and cities).

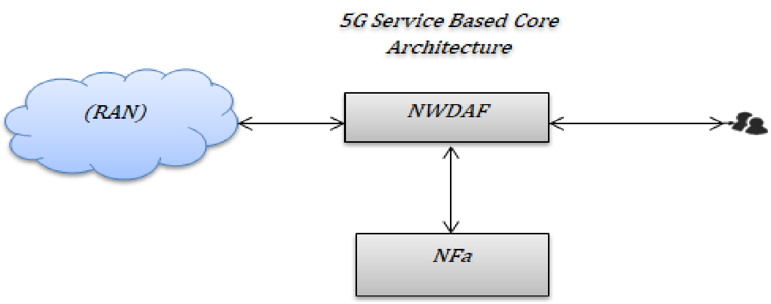

Fig. 10. Interaction of NWDAF with other 5G network elements

Options for the following types of data analysis and forecasting are included in NWDAF, as illustrate in Fig. 11. [30]:

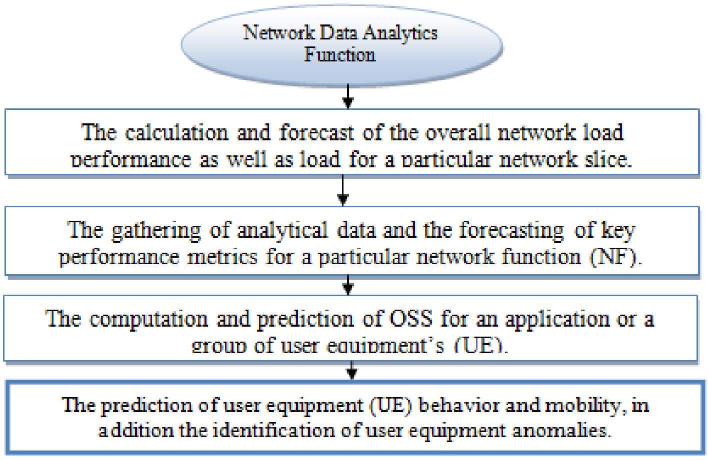

Fig. 11. Data analysis and forecasting are included in NWDAF

A collection of information regarding network overloading, both current and forecast for a particular place, and enforcement of the quality of service's stability, together with relevant reports It is also important to point out that the 3GPP recommendations indicate a very flexible choice of the data analysis tools, in addition to an increase in the utilization of open-source software that is provided by third parties for these purposes

4.1 Partitioning Time Series Over Periods of Time

Disentangling fixed and variable time periods is a crucial part of any time series analysis. Since some indicators correspond to human life cycles, the time series described in the previous sentence may exhibit harmonic components (seasonality) (working hours and days, weekends, holidays). A rise or fall in subscription numbers may also reveal trending elements. Therefore, the data in question can be effectively decomposed using an Error Trend Seasonality (ETS) [31] model. Decompositions can be either additive or multiplicative. An additive model can be defined as follows:

$$y_t = S_t + T_t + R_t \tag{11}$$

where $y_t -$ is the value of y at time instant t, $S_t -$ is the seasonal component, $T_t -$ is the trend component, and $R_t -$ is a residual that cannot be characterized in terms of the first two components due to the presence of random fluctuations (noise, spikes etc). Instead, [32] we say the following about a multiplicative decomposition:

$$y_t = S_t \text{ x } T_t \text{ x } R_t \tag{12}$$

Or, equivalently as:

$$ln(y_t) = ln(S_t) + ln(T_t) + ln(R_t) \qquad 13)$$

We also suggest that further ETS-family linear decompositions can be derived from this type of decomposition. Heuristically developed components, such as the Facebook Prophet model, which also encounters public holidays component [33], can be used to extend the models (11) and (12). Number of failed E-UTRAN Radio Access Bearer (E-RAB) connections is a key performance indicator (KPI) considered in this article (E-RAB Setup Failures). The failure to send the "E-RAB SETUP REQUEST" message to the Mobility Management Entity (MME) on the network of the considered operator is the primary cause of the aforementioned errors. The MME initiates the procedure by sending a location reporting control message. On receipt of a location reporting control message the eNodeB shall perform the requested location reporting control action for the user equipment (UE). When there are issues with the communication routes between the base stations (eNodeB) and CN, the checksum of a particular message can become corrupted, leading to these occurrences as illustrate in Fig.12. [34].

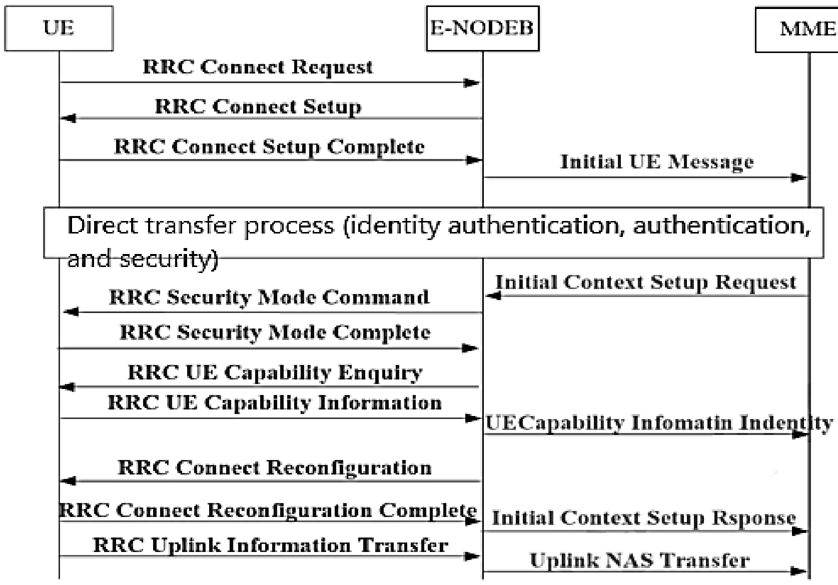

Fig. 12. Step-by-step establishment of the E-RAB protocol

Fig 13 and 14 display the results of using formula(11) to decompose the key performance indicator readings for E-RAB setup failures.

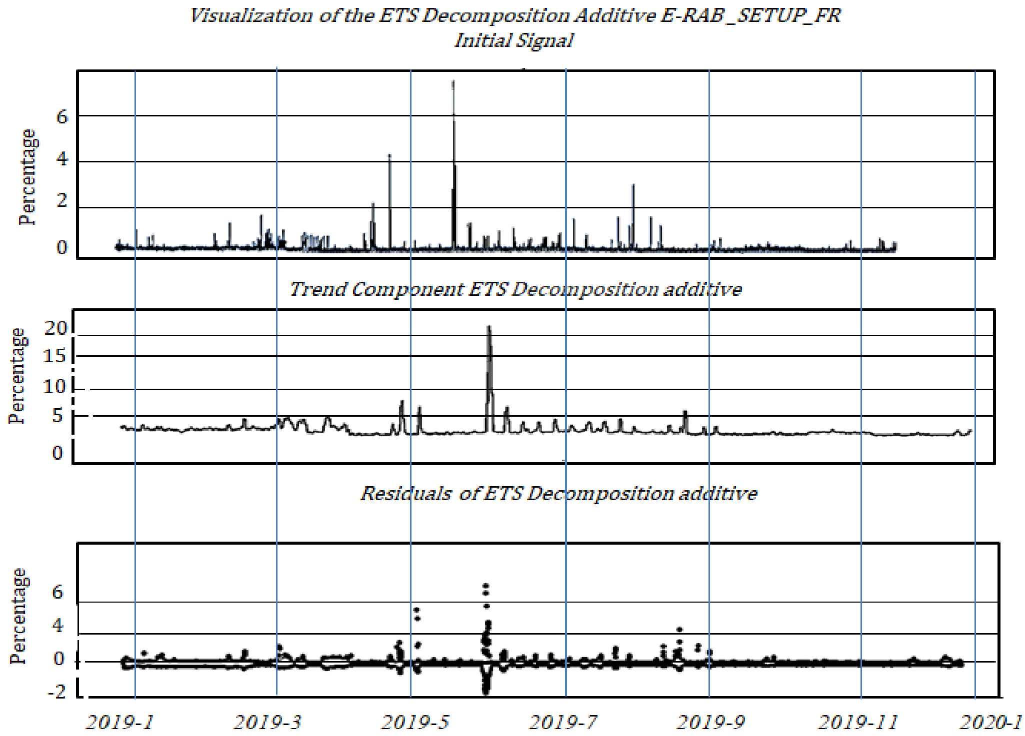

Fig. 13. Trend and residual components of E-RAB setup failures KPI readings

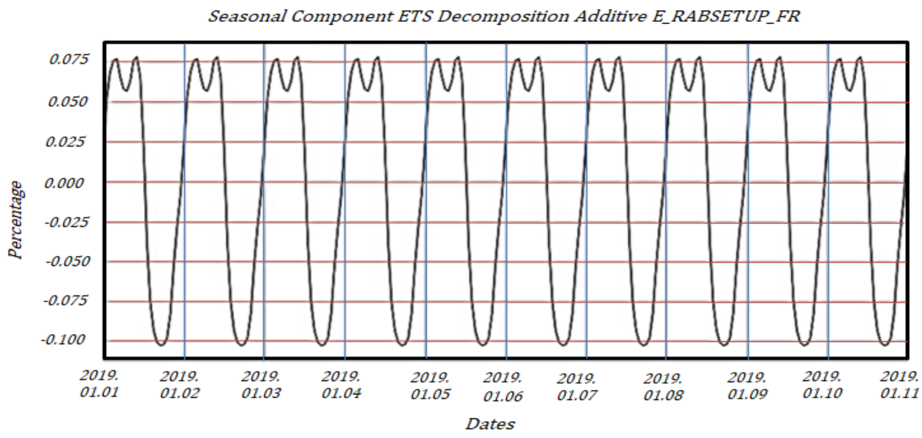

Fig. 14. Seasonal breakdown of E-RAB setup failures key performance indicators

Fig 15 and 16 depict the decomposition of the pondered KPI measures according to formula (13).

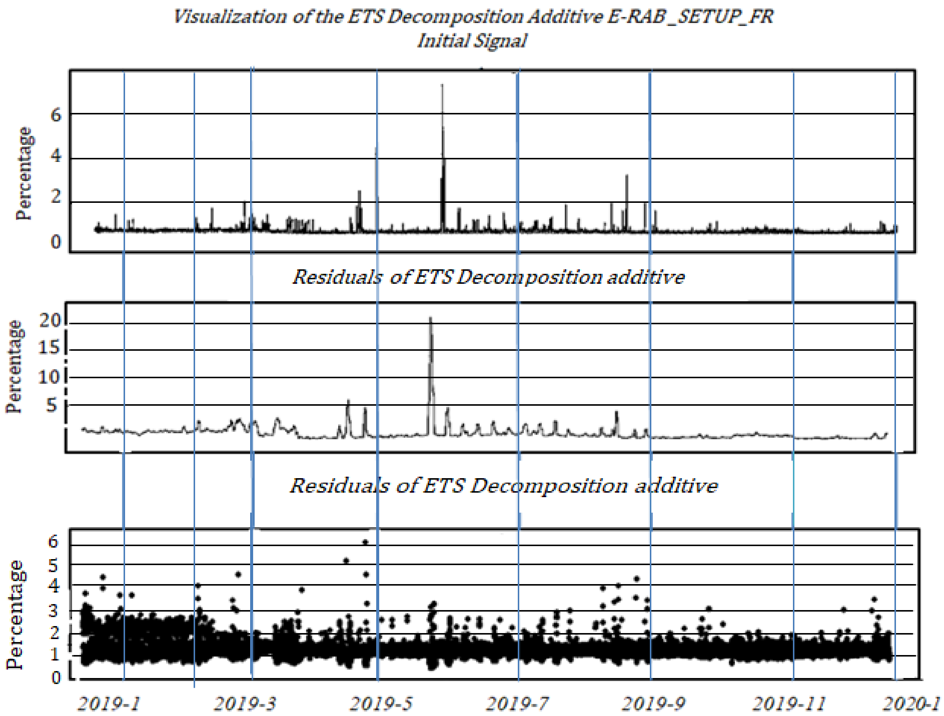

Fig. 15. A Graphical representation of the trend and residual factors in E-RAB setup failures key performance indicators.

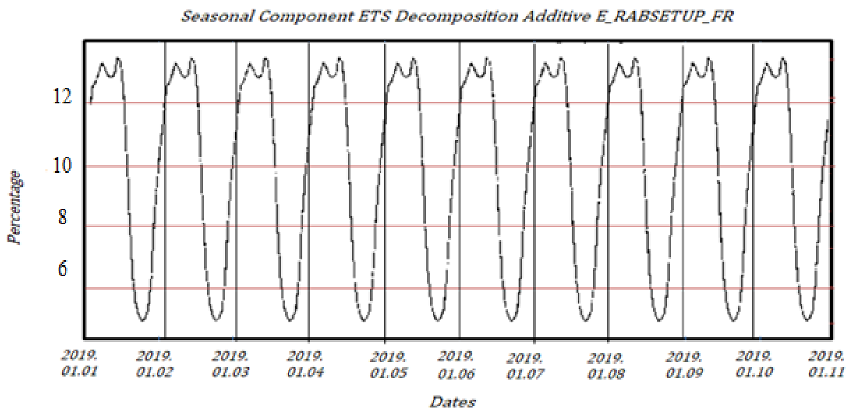

Fig. 16. Shows How season affects KPI for E-RAB installation problems (multiplicative ETS model)

The additive model is considered to be correct because of what is known about the residual component (Figs. 13, 15).

4.2 The Decomposition of Time Series Using Statistics

Determining the statistical distribution of the data being analyzed is another crucial part of time series analysis. What's more, real-world data may have multiple statistically distinct elements [35]. Since our key performance indictors are affected by a wide range of variables (time of day, total number of subscribers, potential incidents) that are not highly correlated with one another, [36] it is reasonable to assume that KPI value can be decomposed into several statistically independent values:

$$f(x\ \alpha_t\ \theta) = \sum_{x=1}^{x} af(x\theta_t) \quad (14)$$

Here, $a(a > 0. \sum_{x=1}^{x} a = 1)$ are probabilistic mixture weights, [37] is the probability density function of the i-th mixed component θ_i is the set of distribution parameters, and K is the number of components. Here, we assume both the presence of a regular (repetitive) statistical component in our data, and the presence of an outliers component (Fig. 17).

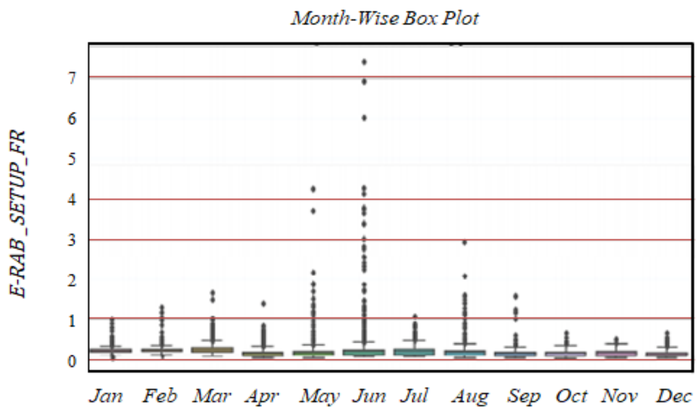

Fig. 17. A box plot representing the data for E-RAB setup failures

Numerous outliers is a driving factor in the need for an outliers prediction task [38].

5. Conclusions

We can draw the following conclusion from table 1: XGboost and Holt-Winters perform better than other algorithms when compared to the metrics that were chosen for the regular part prediction problem. In addition, XGboost has a lower error value spread in comparison to every other algorithm that was taken into consideration. The outcomes of the outlier's prediction problem, which are presented in table 4, are comparable to those that were calculated for the regular part prediction problem. When compared to SVR and PyDLM, XGboost and Holt-Winters have significantly lower rates of both false alarms and missed alarms, and this is true for both two- and single-featured issues. It is also important to point out that SVR has a high rate of false alarms, whereas PyDLM has a high rate of missed alarms. It's possible that the smoothing nature of the respective methods is to blame for the latter two observations. The next stage in this research could be a study into whether or not Deep Learning models can be applied to the many tasks that are being explored.

REFERENCES

[1]. Yu, J., Hutson, A. D., Siddiqui, A. H., & Kedron, M. A. (2016). Group sequential control of overall toxicity incidents in clinical trials–non-Bayesian and Bayesian approaches. Statistical methods in medical research, 25(1), 64-80.
[2]. R. J. Hyndman and G. Athanasopoulos, Forecasting: principles and practice. OTexts, 2018.
[3]. T. Hong and S. Fan, “Probabilistic electric load forecasting: A tutorial review,” Int. J. Forecast., vol. 32, no. 3, pp. 914–938, 2016.
[4]. A. C. Harvey and S. Peters, “Estimation procedures for structural time series models,” J. Forecast., vol. 9, no. 2, pp. 89–108, 1990.
[5]. S. J. Taylor and B. Letham, “Forecasting at scale,” Am. Stat., vol. 72, no. 1, pp. 37–45, 2018.
[6]. N. J. Gordon, D. J. Salmond, and A. F. M. Smith, “Novel approach to nonlinear/non-Gaussian Bayesian state estimation,” in IEE proceedings F (radar and signal processing), 1993, vol. 140, no. 2, pp. 107–113.
[7]. M. J. Baxter, “Encyclopedia of Statistical Sciences.” JSTOR, 1991.
[8]. Z. Que and Z. Xu, “A data-driven health prognostics approach for steam turbines based on XGBoost and DTW,” IEEE Access, vol. 7, pp. 93131–93138, 2019.
[9]. A. B. Koehler, R. D. Snyder, J. K. Ord, and A. Beaumont, “A study of outliers in the exponential smoothing approach to forecasting,” Int. J. Forecast., vol. 28, no. 2, pp. 477–484, 2012.
[10]. U. Kumar and V. K. Jain, “Time series models (Grey-Markov, Grey Model with rolling mechanism and singular spectrum analysis) to forecast energy consumption in India,” Energy, vol. 35, no. 4, pp. 1709–1716, 2010.
[11]. M. Geldenhuys, K. Taba, and C. M. Venter, “Meaningful work, work engagement and organisational commitment,” SA J. Ind. Psychol., vol. 40, no. 1, pp. 1–10, 2014.
[12]. A. V. Bers, F. Momo, I. R. Schloss, and D. Abele, “Analysis of trends and sudden changes in long-term environmental data from King George Island (Antarctica): relationships between global climatic oscillations and local system response,” Clim. Change, vol. 116, no. 3, pp. 789–803, 2013.
[13]. D. Barrow and N. Kourentzes, “The impact of special days in call arrivals forecasting: A neural network approach to modelling special days,” Eur. J. Oper. Res., vol. 264, no. 3, pp. 967–977, 2018.
[14]. R.-C. Chen et al., “An end to end of scalable tree boosting system,” Sylwan, vol. 165, no. 1, pp. 1–11, 2020.
[15]. C. Lee and S. Lee, “Exploring the Contributions by Transportation Features to Urban Economy: An Experiment of a Scalable Tree-Boosting Algorithm with Big Data,” Land, vol. 11, no. 4, p. 577, 2022.
[16]. T.-L. Le, D. S. Nguyen, V.-T. Le Nhat Hoang Tran, and T. Khanh, “Building a multi-output hybrid model for interval-valued time series forecasting,” Tạp chí Khoa học và Công nghệ-Đại học Đà Nẵng, pp. 51–54, 2020.
[17]. M. R. Abonazel, “A practical guide for creating Monte Carlo simulation studies using R,” Int. J. Math. Comput. Sci., vol. 4, no. 1, pp. 18–33, 2018.
[18]. M. A. A. da Cruz, L. R. Abbade, P. Lorenz, S. B. Mafra, and J. J. P. C. Rodrigues, “Detecting Compromised IoT Devices Through XGBoost,” IEEE Trans. Intell. Transp. Syst., 2022.
[19]. C. Su, L. Li, and Z. Wen, “Remaining useful life prediction via a variational autoencoder and a time-window-based sequence neural network,” Qual. Reliab. Eng. Int., vol. 36, no. 5, pp. 1639–1656, 2020.
[20]. Y. Himeur, A. Alsalemi, F. Bensaali, and A. Amira, “Smart non-intrusive appliance identification

using a novel local power histogramming descriptor with an improved k-nearest neighbors classifier," Sustain. Cities Soc., vol. 67, p. 102764, 2021.

[21]. T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. K. A. A. Khan, "Performance analysis of machine learning algorithms in intrusion detection system: a review," Procedia Comput. Sci., vol. 171, pp. 1251–1260, 2020.

[22]. H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," arXiv Prepr. arXiv1402.1128, 2014.

[23]. O. Press, N. A. Smith, and M. Lewis, "Shortformer: Better language modeling using shorter inputs," arXiv Prepr. arXiv2012.15832, 2020.

[24]. H. Zhou et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting," in Proceedings of the AAAI Conference on Artificial Intelligence, 2021, vol. 35, no. 12, pp. 11106–11115.

[25]. Z. Liang et al., "EEGFuseNet: Hybrid unsupervised deep feature characterization and fusion for high-dimensional EEG with an application to emotion recognition," IEEE Trans. Neural Syst. Rehabil. Eng., vol. 29, pp. 1913–1925, 2021.

[26]. N. Ueda, R. Nakano, Z. Ghahramani, and G. E. Hinton, "Split and merge EM algorithm for improving Gaussian mixture density estimates," J. VLSI signal Process. Syst. signal, image video Technol., vol. 26, no. 1, pp. 133–140, 2000.

[27]. S. Goli, H. Mahjub, J. Faradmal, H. Mashayekhi, and A.-R. Soltanian, "Survival prediction and feature selection in patients with breast cancer using support vector regression," Comput. Math. Methods Med., vol. 2016, 2016.

[28]. Z. qasim El-Ezzi, A. M. Al-Dulaimi, and A. A. Ibrahim, "Personalized Quality of Experience (QOE) Management using Data Driven Architecture in 5G Wireless Networks," 4th Int. Symp. Multidiscip. Stud. Innov. Technol. ISMSIT 2020 - Proc., 2020, doi: 10.1109/ISMSIT50672.2020.9254863.

[29]. A. Mikhaylov, N. S. Mazyavkina, M. Salnikov, I. Trofimov, F. Qiang, and E. Burnaev, "Learned Query Optimizers: Evaluation and Improvement," IEEE Access, vol. 10, pp. 75205–75218, 2022.

[30]. V. A. Fadeev, A. K. Gaysin, and A. F. Nadeev, "Investigation of statistical algorithms for prediction of unsuccessful LTE signaling connections," in 2018Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO), 2018, pp. 1–6.

[31]. G. Li and X. Zhou, "Machine learning for data management: A system view," 2022.

[32]. J. Cao et al., "A survey on security aspects for 3GPP 5G networks," IEEE Commun. Surv. tutorials, vol. 22, no. 1, pp. 170–195, 2019.

[33]. F. Kaltenberger, A. P. Silva, A. Gosain, L. Wang, and T.-T. Nguyen, "OpenAirInterface: Democratizing innovation in the 5G Era," Comput. Networks, vol. 176, p. 107284, 2020.

[34]. L. Martenvormfelde, A. Neumann, L. Wisniewski, and J. Jasperneite, "A simulation model for integrating 5g into time sensitive networking as a transparent bridge," in 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2020, vol. 1, pp. 1103–1106.

[35]. S. Sevgican, M. Turan, K. Gökarslan, H. B. Yilmaz, and T. Tugcu, "Intelligent network data analytics function in 5G cellular networks using machine learning," J. Commun. Networks, vol. 22, no. 3, pp. 269–280, 2020.

[36]. L.-V. Le, D. Sinh, B.-S. P. Lin, and L.-P. Tung, "Applying big data, machine learning, and SDN/NFV to 5G traffic clustering, forecasting, and management," in 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft), 2018, pp. 168–176.

[37]. I. Hadj-Kacem, S. Ben Jemaa, S. Allio, and Y. Ben Slimen, "Anomaly prediction in mobile networks: A data driven approach for machine learning algorithm selection," in NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium, 2020, pp. 1–7.

[38]. 3GPP, "System architecture for the 5G system (5GS)," 3rd Gener. Partnersh. Proj. (3GPP), Tech. Specif. 23.501, 2020.