Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 222 (2023) 81–93

Procedia Computer Science

www.elsevier.com/locate/procedia

International Neural Network Society Workshop on Deep Learning Innovations and Applications (INNS DLIA 2023)

Machine Learning Applied to Anomaly Detection on 5G O-RAN Architecture

Pedro V. A. Alves[a], Mateus A. S. S. Goldbarg[a], Wysterlânya K. P. Barros[a], Iago D. Rego[a], Vinícius J. M. T. Filho[a], Allan M. Martins[a], Vicente A. de Sousa Jr.[a,*], Ramon dos R. Fontes[a,*], Eduardo H. da S. Aranha[a,*], Augusto V. Neto[a,*], Marcelo A. C. Fernandes[a,*]

[a] *Federal University of Rio Grande do Norte (UFRN), Natal, 59072-970, RN, Brazil*

Abstract

This article presents a study with feasibility and performance analysis of machine learning (ML) techniques using supervised techniques for anomaly detection problems in a 5G communication network. The proposed ML models (Multilayer Perceptron, Decision Tree, and Support Vector Machine) were used to classify data into anomaly or non-anomaly based on two 5G Open Radio Access Network (O-RAN) datasets with various key performance indicators (KPIs). Furthermore, we propose a strategy that devotes to labeling anomalous situations, leveraging the t-Distributed Stochastic Neighbor Embedding (tSNE) technique atop datasets enclosing multiple KPIs. The results were significant, with an accuracy above 90% for all use cases considered.

Keywords: KPIs; anomaly detection; machine learning; O-RAN

1. Introduction

To support new 5G cellular network requirements (e.g., data rates exceeding 10 Gbps, network latency under 1 ms, capacity expansion by a factor of 1,000, and energy efficiency gains), vendors have begun investigating new radio access network (RAN) architectures [17, 13, 25, 5, 23].

Open Radio Access Network (O-RAN), suggested by the O-RAN Alliance [15], stands as a promising radio technology that has gained worldwide acceptance. O-RAN is a worldwide community of operators, manufacturers, and academic institutes [18, 1]. The vision is to rewrite the RAN industry towards establishing an open, adaptable, and intelligent RAN [15]. Artificial intelligence (AI) in machine learning (ML) will play a crucial role in the 5G network,

* Corresponding author.
E-mail address: allan@dca.ufrn.br, vicente.sousa@ufrn.br, ramon.fontes@imd.ufrn.br, eduardo.aranha@ufrn.br, augusto@dimap.ufrn.br, mfernandes@dca.ufrn.br

1877-0509
10.1016/j.procs.2023.08.146

with particular emphasis on the O-RAN. For example, ML use can drive more efficient enhancements in 5G network planning, automation of network operations (e.g., provisioning, optimization, and fault prediction), network slicing, service quality prediction, and other applications and services [8, 3, 14, 20].

As presented in [4] and [16], O-RAN defines five different ML-based application deployment scenarios characterized by the ML task type and application latency requirements. As for the learning paradigm, the O-RAN architecture supports three main types of ML tasks: supervised learning, unsupervised learning, and reinforcement learning, and three levels of latency requirements: high latency, low latency, and ultra-low latency. Supervised learning algorithms can deal with regression and prediction problems [4, 16], whereas unsupervised learning techniques best suit data classification and clustering [4, 16]. In contrast, reinforcement learning algorithms train agents to operate in environments to maximize their utility in reaching some goals [4, 16, 10].

The O-RAN architecture is composed of several interworking components, and from the ML point of view, we highlighted both the non-real-time (non-RT) RAN intelligent controllers (RIC) and the near-real-time (near-RT) RIC. The non-RT RIC and the near-RT RIC support ML applications (training and inference) in non-RT and near-RT, respectively. Non-RT applications are called rApps, whereas the near-RT applications are xApps [16, 10, 6, 21, 28].

An anomaly can be understood to as a pattern that does not conform to expected behavior. Therefore, it appears infrequently and is closely linked to normality. All network traffic that does not fall into the typical class can be considered anomalous. Consequently, anomaly detection systems should not be limited by any predefined set of anomalies; instead, they must be flexible enough to adapt to any strange events that affect the network [10, 9, 27].

Machine learning-based anomaly detection techniques act as classifiers and can operate either supervised or unsupervised. A dataset with traffic labeled as normal or anomalous in the supervised case must be available. In the unsupervised case, no labeled training set is needed, and the objective is to discover groups of similar examples within the data (clustering) [10, 9, 27].

Several works in literature have proposed ML for anomaly detection as presented in [9, 29, 30, 12, 27, 2]. However, most works focus on using time series prediction, such as works presented in [29, 30, 24] and applied to cybersecurity defence systems as shown in [12, 2].

Thus, differently from the existing literature, this paper aims to present a study with feasibility analysis and performance of AI-ML using supervised techniques for anomaly detection problems in a 5G communication network. The proposed ML models (Multilayer Perceptron, Decision Tree, and Support Vector Machine) were used to classify data in anomaly or non-anomaly. The results were based on three 5G datasets with various Key Performance Indicators (KPIs) such as Reference Signal Received Power (RSRP), Reference Signal Received Quality (RSRQ), and others.

This paper is organized as follows. Section 2 introduces our dataset, including some preliminary processing and the proposed methodology for labeling. Then, Section 3 discusses anomaly detection results while Section 4 presents the final remarks.

2. Dataset

2.1. Dataset description

This work used three 5G datasets called here DS-1, DS-2, and DS-3. The DS-1 is characterized by the dataset presented in anomaly detection O-RAN's use case [22], and the datasets DS-2 and DS-3 are described in [19].

DS-1 has about 10000 rows in a range of about 33 s, and it is composed of several KPIs, as showed in Table 1. DS-2 and DS-3 are characterized by the streaming data for playing the *Adventure Time* and *Rick and Morty* cartoons on Amazon Prime and Netflix, respectively. DS-2 and DS-3 datasets are composed of several files with about 3800 rows per file, and the range time per file is about 38 s. The KPIs of the datasets DS-2 and DS-3 are presented in Table 2.

2.2. Dataset preprocessing

Some KPIs in the DS-2 and DS-3 have a considerable amount of missing data. We used the roulette wheel selection for each KPI to cope with this problem. Widely used in genetic algorithms [26, 7], roulette selection is a stochastic selection method where the probability of selecting a given value is proportional to a previously designed evaluation

Table 1: Description of the dataset DS-1.

KPI name	Description
x	position
y	position
rsrp	reference signal received power
rsrq	reference signal received quality
rssinr	received signal strength indication
nbCellIdentity_0	neighbor 0 cell identifier
nbCellIdentity_1	neighbor 1 cell identifier
nbCellIdentity_2	neighbor 2 cell identifier
nbCellIdentity_3	neighbor 3 cell identifier
nbCellIdentity_4	neighbor 4 cell identifier
rsrp_nb0	reference signal received power for neighbor 0
rsrq_nb0	reference signal received quality for neighbor 0
rssinr_nb0	received signal strength indication for neighbor 0
rsrp_nb1	reference signal received power for neighbor 1
rsrq_nb1	reference signal received quality for neighbor 1
rssinr_nb1	received signal strength indication for neighbor 1
rsrp_nb2	reference signal received power for neighbor 2
rsrq_nb2	reference signal received quality for neighbor 2
rssinr_nb2	received signal strength indication for neighbor 2
rsrp_nb3	reference signal received power for neighbor 3
rsrq_nb3	reference signal received quality for neighbor 3
rssinr_nb3	received signal strength indication for neighbor 3
rsrp_nb4	reference signal received power for neighbor 4
rsrq_nb4	reference signal received quality for neighbor 4
rssinr_nb4	received signal strength indication for neighbor 4

Table 2: Description of the datasets DS-2 and DS-3.

KPI name	Description
Longitude	longitude GPS coordinate
Latitude	latitude GPS coordinate
CellID	cell identifier
RSRP	reference signal received power
RSRQ	reference signal received quality
SNR	signal to noise ratio
CQI	continuous quality improvement
RSSI	received signal strength indication
DL_bitrate	Download rate measured at device in kbit/s
UL_bitrate	Uplink rate measured at the device in kbit/s
RAWCELLID	cell Identifier
NRxRSRP	neighboring cell signal received power
NRxRSRQ	neighboring cell signal received quality

function. The KPI values associated with the DS-2 and DS-3 datasets are discrete values belonging to a finite set of possibilities. Therefore, the evaluation function was characterized as the number of occurrences of each value. In other words, the greater the value occurrence, the greater its slice on the sheet.

Thus, histograms were calculated for each KPI of the DS-2 and DS-3 datasets. Then, a roulette wheel was set up based on the histogram of each dataset's KPI. This way, the roulette wheel was performed for each missing value in a given KPI, filling the value in. Figs. 1 and 2 show an example of the missing data strategy used. Fig. 1 presents a histogram with all values for the KPI NRxRSRQ (see Table 2) before treating missing values, and Fig. 2 shows the same KPI after being treated.

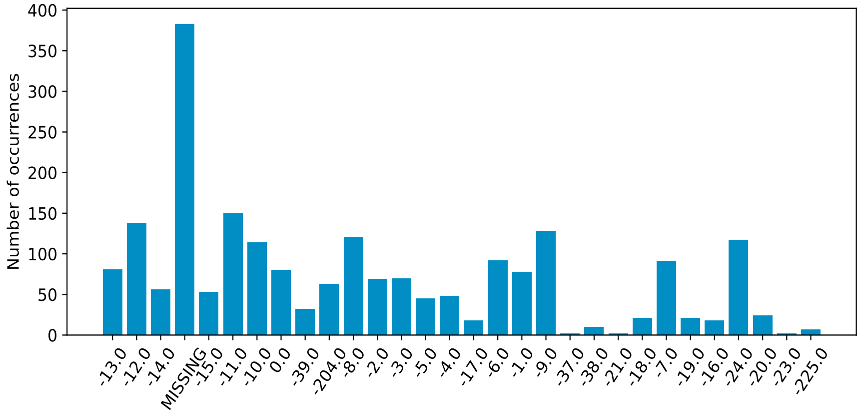

Fig. 1: NRxRSRQ KPI before treating missing values. The roulette wheel slices are based on each value's proportional number of occurrences.

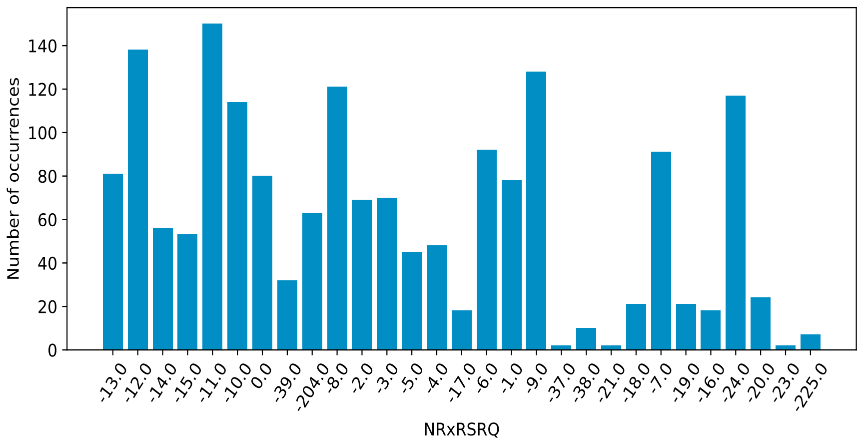

Fig. 2: NRxRSRQ KPI after treating missing values.

2.3. Dataset labelling

In wireless communications, as in the case of 5G systems, network traffic situations that are out of a regular pattern can be considered anomalies. For example, abrupt throughput reduction, latency increase or significant signal loss can be observed as anomalous situations. It is essential to understand that anomaly detection in 5G systems is a multifactorial problem that must be treated based on several variables, based on several KPIs [12, 9, 11]. On the other hand, anomaly detection systems should not be limited to a specific group of anomalies and should be able to act in any situation [12, 9, 11].

Thus, aiming to create distinct anomaly profiles that may be caused by multiple KPIs and, at the same time, validate the performance of ML techniques, this article proposes a dataset labelling scheme composed of three strategies described as follows.

All data were labelled as an anomaly (A) and not-anomaly (NA). The three labelling strategies used t-Distributed Stochastic Neighbor Embedding (tSNE). This technique reduces data dimension so that we visualize data from DS-2 and DS-3 (with 13 dimensions) and DS-1 (with 23 dimensions) in just two dimensions. In the first labelling strategy, called here balanced strategy, the data was divided approximately half for each class (A and NA). In the unbalanced strategy's second labelling approach, the data was split with more elements of the A class than the NA class. Finally, in the third one, called alternated strategy, the data was divided with approximately the same size for each class but spread in the tSNE space.

Next figures show the labelling process for:

- the dataset DS-1: Figs. 3, 4, and 5;
- the dataset DS-2: Figs. 6, 7, and 8; and
- the dataset DS-3: Figs. 9, 10, and 11.

The blue points represent class A, and the red is class NA.

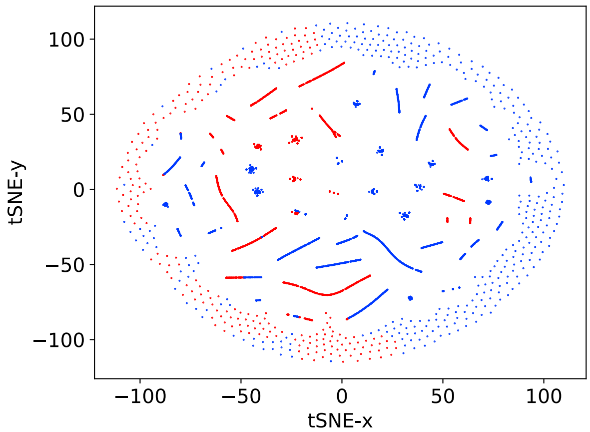

Fig. 3: Labelling processing for DS-1 in balanced strategy. The blue points represent class A, and red points class NA.

3. Results

The results were obtained for three ML techniques: the Multilayer Perceptron (MLP), Decision Tree (DT) and the Support Vector Machine (SVM). As Python is a supported language for developing xAPPs and rAPPs on RIC [6, 21, 28], all ML techniques were developed and executed in Python.

The MLP architecture used 1000 neurons in a hidden layer and trained for 25 epochs. The DT strategy was based on the Gini criterion. Finally, the SVM architecture was based on the radial basis function as kernel and regularization parameter set as 1.0. Each ML model used 25 inputs for DS-1 and 13 for DS-2 and DS-3. The ML training step was based on k-fold cross-validation approach with $k = 5$.

Tables 3, 4, and 5 depict the confusion matrix for DS-1 associated with MLP, DT, and SVM, respectively, while Table 6 summarizes the accuracy results for the DS-1.

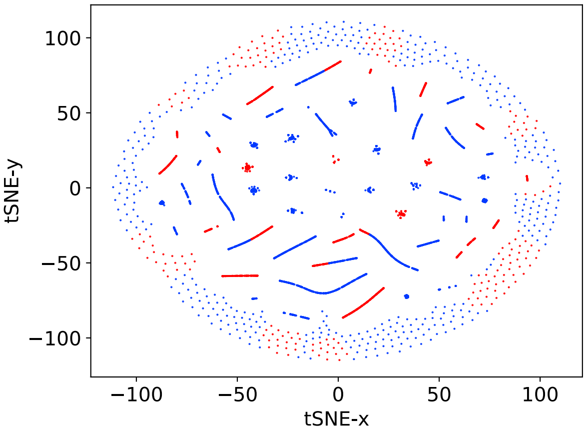

Fig. 4: Labelling processing for DS-1 in alternated strategy. The blue points represent class A, and red points class NA.

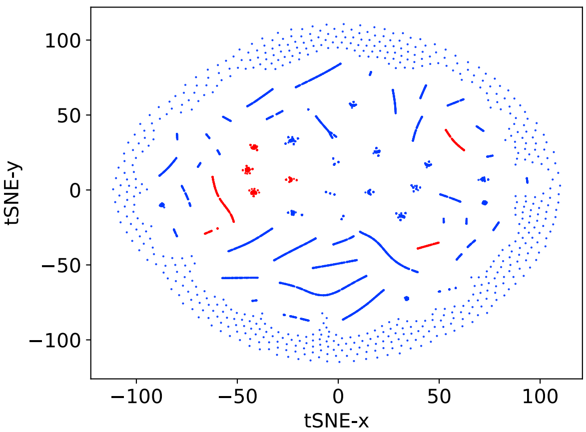

Fig. 5: Labelling processing for DS-1 in unbalanced strategy. The blue points represent class A, and red points class NA.

Table 3: Confusion matrices associated for the MLP model with the DS-1 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

		Predicted Class					
		Balanced		Alternated		Unbalanced	
		A	NA	A	NA	A	NA
Actual Class	A	99.8%	0.2%	99.0%	1.4%	99.8%	1.3%
	NA	0.3%	99.7%	0.9%	98.8%	0.1%	99.1%

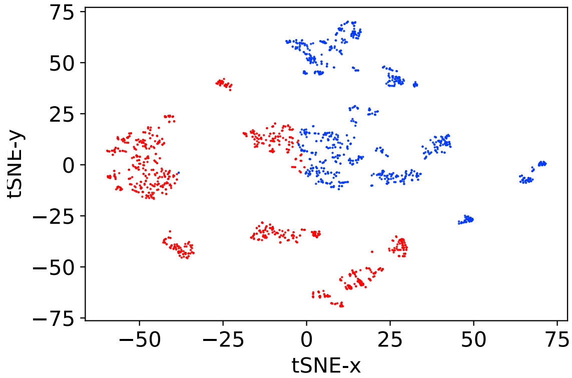

Fig. 6: Labelling processing for DS-2 in balanced strategy. The blue points represent class A, and red points class NA.

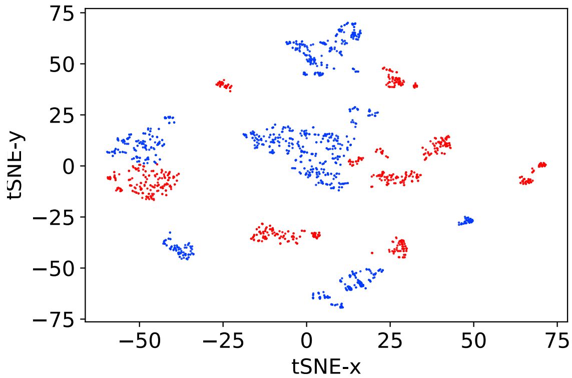

Fig. 7: Labelling processing for DS-2 in alternated strategy. The blue points represent class A, and red points class NA.

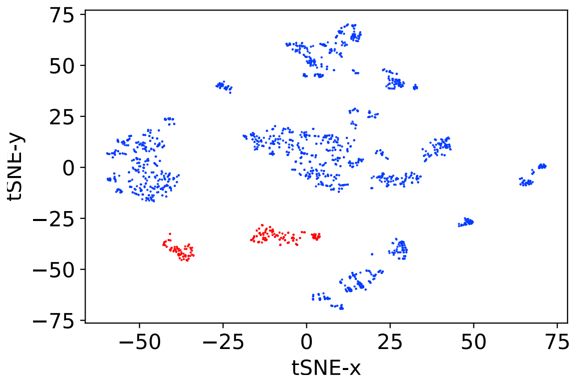

Fig. 8: Labelling processing for DS-2 in unbalanced strategy. The blue points represent class A, and red points class NA.

Table 4: Confusion matrices associated for the DT model with the DS-1 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

		Predicted Class					
		Balanced		Alternated		Unbalanced	
		A	NA	A	NA	A	NA
Actual Class	A	99.4%	0.6%	97.9%	2.6%	99.6%	1.4%
	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

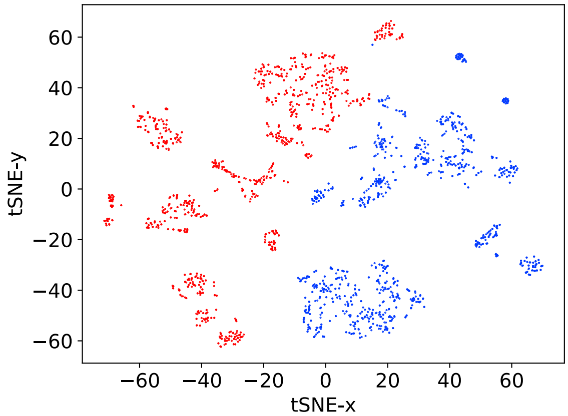

Fig. 9: Labelling processing for DS-3 in balanced strategy. The blue points represent class A, and red points class NA.

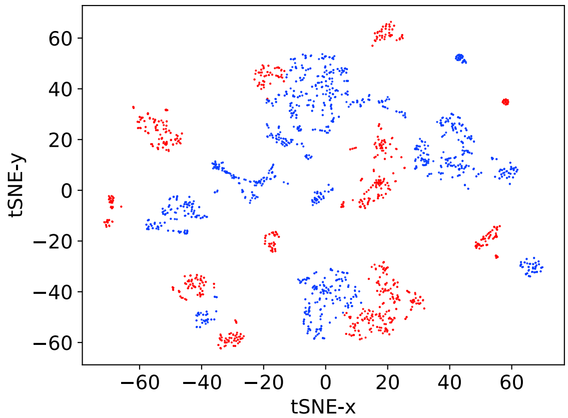

Fig. 10: Labelling processing for DS-3 in alternated strategy. The blue points represent class A, and red points class NA.

Table 5: Confusion matrices associated for the SVM model with the DS-1 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

		Predicted Class					
		Balanced		Alternated		Unbalanced	
		A	NA	A	NA	A	NA
Actual Class	A	99.0%	0.9%	95.7%	10.2%	99.9%	0.9%
	NA	0.9%	99.1%	4.3%	89.8%	0.1%	99.1%

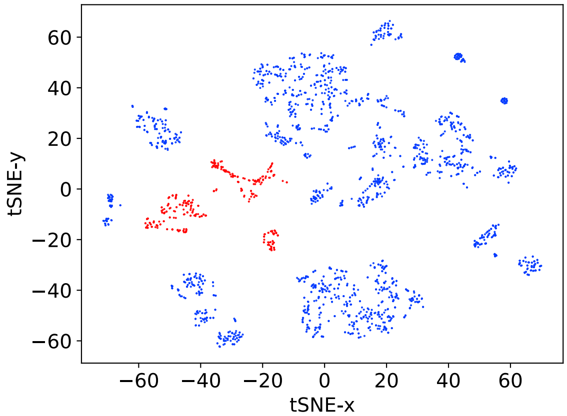

Fig. 11: Labelling processing for DS-3 in unbalanced strategy. The blue points represent class A, and red points class NA.

Table 6: Accuracy results for the MLP, DT, and SVM models with the DS-1 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

Model	**Balanced**	**Alternated**	**Unbalanced**
MLP	99.76%	98.90%	99.72%
DT	99.29%	97.44%	99.45%
SVM	99.09%	93.92%	99.76%

The best accuracy for the DS-1 dataset was using the MLP algorithm that reached 99.76%, 98.90%, and 99.72% for Balanced, Alternated, and Unbalanced strategies, respectively (see Table 6). Although the other techniques (DT and SVM) did not achieve the same result as MLP, their results were quite promising (see Tables 3 and 5).

For DS-2 and DS-3 the confusion matrix are presented in Tables 7, 8, 9, 11, 12, and 13. Tables 10 and 14 summarize the accuracy results of the DS-2 and DS-3, respectively.

Table 7: Confusion matrices associated for the MLP model with the DS-2 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

		Predicted Class					
		Balanced		Alternated		Unbalanced	
		A	NA	A	NA	A	NA
Actual Class	A	99.5%	0.4%	99.0%	1.4%	100%	0.0%
	NA	0.8%	99.3%	0.8%	98.9%	0.1%	99.5%

Table 8: Confusion matrices associated for the DT model with the DS-2 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

		Predicted Class					
		Balanced		Alternated		Unbalanced	
		A	NA	A	NA	A	NA
Actual Class	A	99.3%	0.6%	98.3%	2.2%	99.9%	0.9%
	NA	1.8%	98.4%	1.9%	97.5%	0.1%	99.1%

Table 9: Confusion associated obtained for the SVM model with the DS-2 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

		Predicted Class					
		Balanced		Alternated		Unbalanced	
		A	NA	A	NA	A	NA
Actual Class	A	97.4%	2.5%	97.8%	3.0%	99.9%	0.5%
	NA	0.3%	99.7%	1.5%	98.0%	0.0%	100%

Table 10: Accuracy results for the MLP, DT, and SVM models with the DS-2 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

Model	**Balanced**	**Alternated**	**Unbalanced**
MLP	99.42%	98.95%	99.95%
DT	98.84%	98.00%	99.79%
SVM	98.58%	97.89%	99.94%

Table 11: Confusion matrices associated for the MLP model with the DS-3 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

		Predicted Class					
		Balanced		Alternated		Unbalanced	
		A	NA	A	NA	A	NA
Actual Class	A	99.8%	0.2%	98.9%	1.5%	99.8%	1.3%
	NA	0.3%	99.7%	0.5%	99.4%	0.2%	98.8%

Table 12: Confusion matrices associated for the DT model with the DS-3 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

		Predicted Class					
		Balanced		Alternated		Unbalanced	
		A	NA	A	NA	A	NA
Actual Class	A	99.4%	0.6%	97.9%	2.7%	99.7%	1.9%
	NA	0.8%	99.2%	2.4%	96.8%	0.4%	97.6%

Table 13: Confusion matrices associated for the SVM model with the DS-3 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

		Predicted Class					
		Balanced		Alternated		Unbalanced	
		A	NA	A	NA	A	NA
Actual Class	A	98.9%	1.1%	91.7%	10.1%	99.8%	1.7%
	NA	1.0%	99.0%	2.2%	96.6%	0.1%	99.4%

Table 14: Accuracy results for the MLP, DT, and SVM models with the DS-3 dataset. Results associated with three anomaly labeling strategies: balanced, alternated, and unbalanced.

Model	Balanced	Alternated	Unbalanced
MLP	99.76%	99.09%	99.68%
DT	99.29%	97.44%	99.29%
SVM	98.93%	94.88%	99.72%

Similar to the DS-1, in DS-2 and DS-3 datasets, the accuracy of MLP (see Table 10) was better than the DT and SVM (see Table 14). The best accuracy for the DS-2 was about 99.42%, 98.95%, and 99.95% for Balanced, Alternated, and Unbalanced strategies, respectively (see Table 10). For the DS-3, it was about 99.76%, 99.09%, and 99.68% for Balanced, Alternated, and Unbalanced strategies, respectively (see Table 14). In the same situation of DS-1, other techniques (DT and SVM) did not achieve the same result as MLP, but their results were quite promising (see Tables 8, 9, 12, and 13).

The results showed that the anomaly detection performance depends on the scenario (DS-1, DS-2, and DS-3), the degree of data balancing (Balanced, Alternated, and Unbalanced), and the ML technique. Although the MLP achieved the best overall results, the accuracy values differed in each scenario and type of data balancing. Furthermore, the accuracy results indicated that the SVM achieved the best results for the unbalanced data in the DS-1 and DS-3 scenarios, with slightly better performance than MLP. This association between the ML technique and the scenario may imply the need to use a committee of ML techniques to choose the best technique for each experienced scenario.

Finally, the results show that anomaly detection based on multiple variables (i.e., multiple KPIs) is feasible for several anomaly scenarios (balanced, alternated, and unbalanced strategies). From the ML point of view, regardless of the KPIs values associated with signal quality (for example, RSRQ and RSSI) or the data stream quality (DL_bitrate, UL_bitrate), the anomaly generates a signature that ML can detect after training. In addition, the O-RAN architecture through Non-RT RIC and Near-RT RIC provides protocols and interfaces for model training, enabling models to adapt to new anomaly profiles.

4. Conclusion

In this work, we present results and analyses of three ML/AI supervised approaches applied to anomaly detection: Multilayer Perceptron, Decision Tree, and Support Vector Machine. The tests were conducted on an emulation testbed concerning a network environment dataset. The unsupervised ML/AI strategy, based on t-Distributed Stochastic Neighbor Embedding (tSNE), was used to create data labels. Results associated with the accuracy of the ML/AI algorithms were obtained, suggesting an excellent performance with an accuracy of above 90% for all cases.

Acknowledgements

This research was partially funded by Lenovo, as part of its R&D investment under Brazilian Informatics Law, and by the *Coordenacao de Aperfeicoamento de Pessoal de Nivel Superior - Brasil* (CAPES) - Finance Code 001.

References

[1] Abdalla, A.S., Upadhyaya, P.S., Shah, V.K., Marojevic, V., 2022. Toward next generation open radio access networks: What o-ran can and cannot do! IEEE Network 36, 206–213.

[2] Alrashdi, I., Alqazzaz, A., Aloufi, E., Alharthi, R., Zohdy, M., Ming, H., 2019. Ad-IoT: Anomaly detection of IoT cyberattacks in smart city using machine learning, in: 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), IEEE. pp. 0305–0310.

[3] Bitton, R., Avraham, D., Klevansky, E., Mimran, D., Brodt, O., Lehmann, H., Elovici, Y., Shabtai, A., 2022. Adversarial machine learning threat analysis in open radio access networks. arXiv preprint arXiv:2201.06093 .

[4] Bonati, L., D'Oro, S., Polese, M., Basagni, S., Melodia, T., 2021. Intelligence and learning in O-RAN for data-driven NextG cellular networks. IEEE Communications Magazine 59, 21–27.

[5] Dangi, R., Lalwani, P., Choudhary, G., You, I., Pau, G., 2022. Study and investigation on 5g technology: A systematic review. Sensors 22, 26.

[6] Iturria-Rivera, P.E., Zhang, H., Zhou, H., Mollahasani, S., Erol-Kantarci, M., 2022. Multi-agent team learning in virtualized open radio access networks (o-ran). Sensors 22, 5375.

[7] Katoch, S., Chauhan, S.S., Kumar, V., 2021. A review on genetic algorithm: past, present, and future. Multimedia Tools and Applications 80, 8091–8126.

[8] Kaur, J., Khan, M.A., Iftikhar, M., Imran, M., Haq, Q.E.U., 2021. Machine learning techniques for 5G and beyond. IEEE Access 9, 23472–23488.

[9] Lam, J., Abbas, R., 2020. Machine learning based anomaly detection for 5G networks. arXiv preprint arXiv:2003.03474 .

[10] Ma, B., Guo, W., Zhang, J., 2020. A survey of online data-driven proactive 5G network optimisation using machine learning. IEEE access 8, 35606–35637.

[11] Maimó, L.F., Clemente, F.J.G., Pérez, M.G., Pérez, G.M., 2017. On the performance of a deep learning-based anomaly detection system for 5g networks, in: 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), IEEE. pp. 1–8.

[12] Maimó, L.F., Gómez, Á.L.P., Clemente, F.J.G., Pérez, M.G., Pérez, G.M., 2018. A self-adaptive deep learning-based system for anomaly detection in 5G networks. Ieee Access 6, 7700–7712.

[13] Mattisson, S., 2018. An overview of 5G requirements and future wireless networks: Accommodating scaling technology. IEEE Solid-State Circuits Magazine 10, 54–60.

[14] Minovski, D., Ogren, N., Ahlund, C., Mitra, K., 2021. Throughput prediction using machine learning in LTE and 5G networks. IEEE Transactions on Mobile Computing .

[15] O-RAN, 2022. O-RAN ALLIANCE. URL: https://www.o-ran.org/.

[16] O-RAN Working Group 2, 2021. AI/ML workflow description and requirements. Technical Report. O-RAN ALLIANCE.

[17] Pirinen, P., 2014. A brief overview of 5g research activities, in: 1st International Conference on 5G for Ubiquitous Connectivity, IEEE. pp. 17–22.

[18] Polese, M., Bonati, L., D'Oro, S., Basagni, S., Melodia, T., 2023. Understanding o-ran: Architecture, interfaces, algorithms, security, and research challenges. IEEE Communications Surveys & Tutorials .

[19] Raca, D., Leahy, D., Sreenan, C.J., Quinlan, J.J., 2020. Beyond throughput, the next generation: a 5G dataset with channel and context metrics, in: Proceedings of the 11th ACM multimedia systems conference, pp. 303–308.

[20] Ramezanpour, K., Jagannath, J., 2022. Intelligent zero trust architecture for 5g/6g networks: Principles, challenges, and the role of machine learning in the context of o-ran. Computer Networks , 109358.

[21] Rego, I., Medeiros, L., Lopes, V., Filho, V., Sousa, V., Aranha, E., Martins, A., Fernandes, M., Fontes, R., Neto, A., Pupio, G., 2022. Frequency scanning xapps: O-ran ric and gnu radio with rtl-sdr use case, in: XL BRAZILIAN TELECOMMUNICATIONS AND SIGNAL PROCESSING SYMPOSIUM, SBrT.

[22] Rimmer, M., 2022. Anomaly detection use case. URL: https://wiki.o-ran-sc.org/display/RICP/Anomaly+Detection+Use+Case.

[23] Salahdine, F., Han, T., Zhang, N., 2023. 5g, 6g, and beyond: Recent advances and future challenges. Annals of Telecommunications , 1–25.

[24] Sevgican, S., Turan, M., Gökarslan, K., Yilmaz, H.B., Tugcu, T., 2020. Intelligent network data analytics function in 5g cellular networks using machine learning. Journal of Communications and Networks 22, 269–280.

[25] Siddiqui, M.U.A., Qamar, F., Ahmed, F., Nguyen, Q.N., Hassan, R., 2021. Interference management in 5G and beyond network: requirements, challenges and future directions. IEEE Access 9, 68932–68965.

[26] Singh, R.K., Panchal, V., Singh, B.K., 2018. A review on genetic algorithm and its applications, in: 2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT), IEEE. pp. 376–380.

[27] Toshniwal, A., Mahesh, K., Jayashree, R., 2020. Overview of anomaly detection techniques in machine learning, in: 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), IEEE. pp. 808–815.

[28] Zhang, H., Zhou, H., Erol-Kantarci, M., 2022. Team learning-based resource allocation for open radio access network (o-ran). arXiv preprint arXiv:2201.07385 .

[29] Zhao, Y., Zhang, X., Shang, Z., Cao, Z., 2021a. A Novel Hybrid Method for KPI Anomaly Detection Based on VAE and SVDD. Symmetry 13, 2104.

[30] Zhao, Z., Xu, C., Li, B., 2021b. A LSTM-Based Anomaly Detection Model for Log Analysis. Journal of Signal Processing Systems 93, 745–751.