M. Milani, D. Bega, M. Gramaglia and C. Mannweiler, "Optimizing Predictive Analytics in 5G Networks Through Zero-Trust Operator-Customer Cooperation," 2023 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Dresden, Germany, 2023, pp. 123-128.

DOI: 10.1109/NFV-SDN59219.2023.10329622.

Optimizing Predictive Analytics in 5G Networks through Zero-Trust Operator-Customer Cooperation.

Mattia Milani
Nokia, Germany
mattia.milani@nokia.com

Dario Bega
Nokia Bell Labs, Germany
dario.bega@nokia-bell-labs.com

Marco Gramaglia
Universidad Carlos III de Madrid, Spain
mgramagl@it.uc3m.es

Christian Mannweiler
Nokia, Germany
christian.mannweiler@nokia.com

***Abstract*—Data availability in softwarized networks plays a fundamental role in various operations, including network function control, management, and orchestration. Despite early trends of designing domain-specific architectures in isolation, interactions between network operators and their customers have often resulted in limited data exchange, and only recently, standardization bodies have addressed this challenge. In this paper, we advocate for a more robust collaboration between operators and customers by introducing a zero-trust analytics service. This service enables the creation of tailored models for the different customers of network operators. We outline the necessary procedures to support such analytics and present a use case that demonstrates how specific operator-provided analytics (network flow detection) can be enhanced through the incorporation of external signals from a customer.**

***Index Terms*—Network Analytics, NWDAF, Machine Learning**

I. Introduction

The benefits of network softwarization have been observed across all domains of mobile networks. In the domain of Access Networks, the O-RAN [1] architecture has played a crucial role in completely opening up the implementation architecture. This paradigm shift has allowed for greater flexibility and integration of third-party applications. Similarly, the management domain, as defined by 3GPP SA5, has also embraced this trend by adopting a fully API-based description and a compound information model [2] for e.g., the Network Slice as a Service Paradigm. Also, from their early stages [3], orchestration tools have provided the capability of API-based lifecycle management for network deployment. And finally, the 5G Core has seamlessly integrated the Service Based paradigm into its architecture since its initial design [4].

However, despite these advancements, designing such network architectures in isolation may fall short in enabling fully data-driven network management, as we proposed in our work [5]. Although these domains, which were traditionally designed separately in mobile networks, have been designed with softwarization in mind, they often lack extensive data exchange between them. For instance, in the current 3GPP architecture, there remains limited interaction between service providers (SPs) utilizing slices made available by network operators (NOPs) [6]. Furthermore, even within the boundaries of NOPs, there can be indirect access to data that can be used for its intelligent operation, as seen in the case of the Access network and the Core, which can only exchange data for network analytics purposes through the management system [7]. Thus, there is a need to further enhance data exchange capabilities to foster seamless collaboration between different domains within mobile networks.

That is, network analytics could benefit from a broader interaction between domains, involving the exchange of training data but also, as we discuss in this paper, the interaction between different models can involve the exchange of *steering signals* that allow the different models to *cooperatively learn* and improve their metrics when they act on the same data or when a model has access to different and better quality data. Hence, in this paper, we propose a third-party-supported operation for the network analytics system proposed by 3GPP, where they interact to achieve a common goal: improved network operation.

More specifically, we target the scenario where the third party is *customer* of the NOP, hence with even more stringent requirements on the kind of data that shall be exchanged among them. For this reason, our proposed system is especially useful for the creation of *zero-trust analytics*.

The contributions of this paper are as follows. We first investigate how the network operation can be improved by the interaction between network operators and their customers, discussing the recent proposals from standards, drawing the *zero-trust analytics* use case (Section II). Then we focus on the main application for our scenario, the Network Data Analytics Function (NWDAF) discussing how the standard procedures can be adapted for our specific use case (Section III). Then, we validate our approach by applying it to specific analytics (network flow detection), showing how the performance can be improved by using an external *steering* signal (Section III). Finally, we draw conclusions in Section V.

II. Enabling operator-customer cooperation

In this section, we make the case for a tighter cooperation between operators and their customers through the design of an enhanced exposure layer for the creation of tailored analytics.

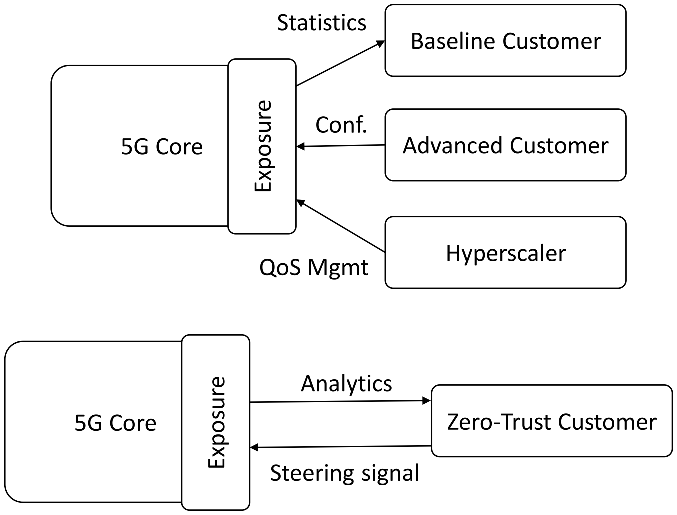


Fig. 1: The current view as studied by 3GPP (top), the additional scenario proposed in this paper (bottom).

A. Enhanced exposure layer customer cooperation

The current trends in the mobile network architectural design are envisioning more direct interactions between different players in the ecosystem. For instance, the 3GPP study items reported in TR 28.824 [8] define who, what, and how management services can be exposed to third parties. This issue primarily applies to the B2B/B2B2C market, specifically in the context of the Network Slice as a Service (NSaaS) model [9]. The report defines different types of consumers according to their characteristics. The first type is the baseline vertical customer, which is primarily concerned with monitoring the network slice to ensure it operates according to the Service Level Agreement (SLA). This type of consumer typically lacks telco experience and is associated with a network slice deployed for a given service. The network can expose capabilities to this consumer such as the network slice's status (active or inactive) and subscribed management data (e.g., KPIs, events/logs, trace data).

The second type is the advanced vertical customer, which contributes to the network operation as a portion of the network slice is deployed within the consumer premises, while other portions reside in the network operator infrastructure. Unlike the baseline vertical customer, this profile has some telco knowledge and wants to have more control over the running network slice. Hence the exposure capabilities may include monitoring functionalities and device configuration options, as well as edge discovery/selection for deploying workloads on the telco edge cloud.

The third type is the hyperscaler, which requests a dedicated network slice from the operator to establish a service-tailored connectivity pipeline for their customers. In this case, the network operator also has to set up network slice continuity between the network and the hyperscaler premises. In this case, the offered capabilities include monitoring functionalities, quality on demand (dynamic QoS and bandwidth management), and policy control.

For all of these roles, 3GPP identifies the specific issues that need to be overcome, including the specific core and management modules that need to take care of the exchange, and the API to be used. Three candidates are envisioned: CAMARA [10], GSMA Open Gateway [11], and TM Forum [12]. Still, the interactions are based mostly on the control and re-orchestration of the network elements, which require full trust between the different parties and, in general, do not allow correlation between the network status and the action to be taken.

Hence, building on top of the 3GPP framework, we propose an additional scenario, depicted in the bottom part of Fig. 1. In this case, the customer is a Zero-Trust one, that has to exchange information with the operator without exchanging sensitive data, for instance, data with confidentiality restrictions. In this view, both the operator and the customer exchange information as two *closed boxes*, each of them acting independently from the other, possibly driven by two intelligent algorithms: one in the NOP that is optimizing the production of analytics and one in the customer, that has the goal of maximizing the customer provided service business metrics. The main use case in this context is the one of *tailored network analytics*, as we discuss next.

The Network Data Analytics Framework specified by 3GPP [7] already envisions a closed loop with the NWDAF as the central hub. However, such analytics are computed with data that is solely available to the network operator, gathered from the other network functions in the 5G Core, or from the access network through the Management plane. One possible issue with this approach is the lack of synchronization between the NOP vantage point (that only analyzes network related Key Performance Indicators (KPIs)) and the customer, which has a full view of the overall operation (but it misses the view of the network internals). Thus, the analytics obtained from these QoS metrics can deviate from the ones that the customer really needs, as they are not tailored to the specific service that is offered in the network. Yet, specific information about the service may not be disclosed by the customer, due to confidentiality reasons.

A relevant example for this case is the one of a video provider that wants to acquire analytics for the network to improve internal business metrics (e.g., reducing the users' churn rate). In this case, revealing this metric to a network operator can also be a competitor like in the case of *triple-play*. Thus, the customer may want to *steer* the analytics, without exchanging business-related data, improving it to match the internal metrics.

B. Zero-trust analytics

As depicted in Fig. 2, the modules involved in this procedure are, the NWDAF and the customer. NWDAF in the 5GC, specified by 3GPP in TS 23.288[7], utilizes the mechanisms and interfaces defined for 5GC and the Management System for data collection and analytics production. Starting from

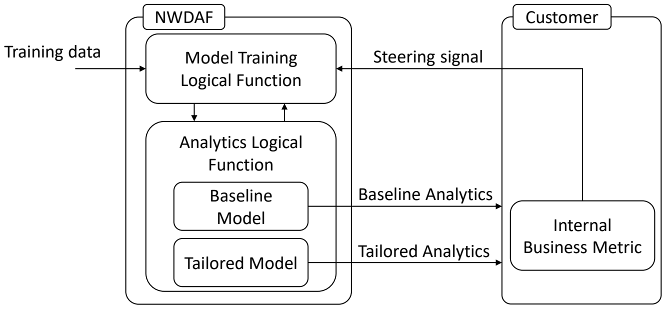


Fig. 2: The zero-trust analytics use case.

Rel. 17, NWDAF (that has been split into two logical functions, i.e., Analytics Logical Function or AnLF, and Model Training Logical Function or MTLF) may leverage AI/ML models for providing requested analytics. When a consumer (including Application Function (AF), which is relevant for the interaction with the customer) subscribes/requests analytics predictions, the NWDAF collects the required data input, trains the AI/ML model (if no trained AI/ML models are already available for the requested analytics) and produces the requested analytics.

In the beginning, the required analytics is the one provided by the baseline model, trained only with data available to the Network Operator (NOP). Then the customer evaluates the analytics exposed by the NOP against its business metric, hence measuring how the analytics fits the need of the business metrics. For instance, in a video content distribution service, how the QoS video analytics are matching user satisfaction.

This translates into a steering signal that is used to improve and tailor the model to the specific service. After several iterations, the model is re-trained and eventually improved. This will have two effects: i) an increased accuracy / precision on the analytics and ii) a better alignment to the (unknown to the NOP) customer metric.

III. Cooperation procedures in the 5G Core

As discussed in Section II, the AI/ML models, and so the analytics produced, are trained only using information available at the NOP in an isolated fashion. This lack of interaction between Zero-Trust customer, i.e., AF, and NOP, i.e., NWDAF, leads to the production of analytics that optimize network-related metrics instead of optimizing the customer's metric. To overcome this issue, we propose new procedures that allow NOP and Zero-Trust customer to exchange *steering signals* to drive analytics production ensuring customer metric optimization without disclosing any sensitive information.

In the following, the proposed procedure is detailed highlighting (see the boxes in Fig. 3) the main novelties compared with the current 3GPP standardized analytics production mechanism [7].

1) NWDAF updates its profile (stored at the Network Repository Function (NRF)) with the list of available AI/ML models and their configurable parameters. Configurable parameters are AI/ML model elements that could be updated or tuned for steering the model training phase. Examples of configurable parameters are variables of a parametrized loss function (as the one employed in [13]), or filters offset in a deformable convolutional layer or optimizer learning rate. Note that AI/ML models hyperparameters as model architecture cannot be modified during the training phase and are not exposed to Zero-Trust customers since they represent sensitive and proprietary information. Furthermore, AI/ML model weights are not considered configurable parameters since they are automatically updated by the optimization function to optimize the desired objective metric.

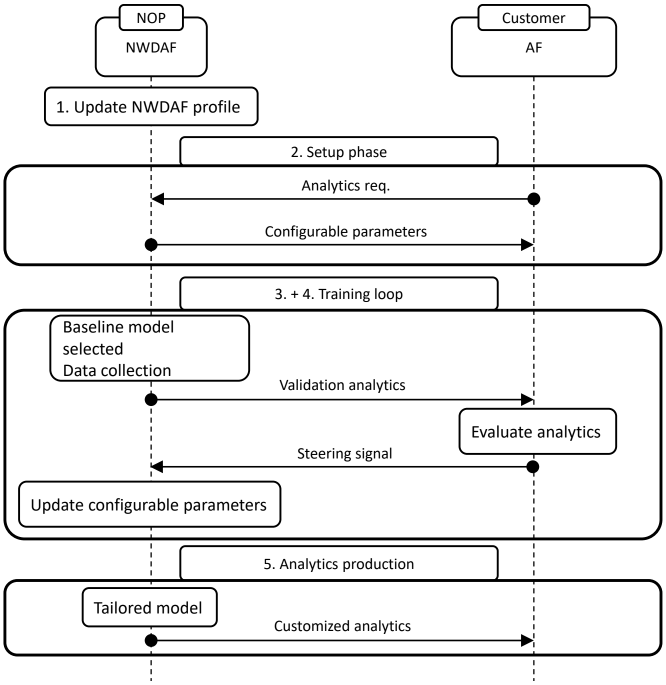


Fig. 3: The call flow for the zero-trust analytics.

2) Initially a setup phase is required between the Zero-Trust customer, achieved through the AF, interested in receiving customized analytics services, and the NOP, i.e., the NWDAF, to settle the analytics framework by specifying the analytics type, the configurable parameters that will be tuned during the training phase following AF *steering signals*, and the framework settings as for example the periodicity with which the configurable parameters should be updated.
3) After the setup phase, the NWDAF collects the required training data and starts the AI/ML model training phase. With the periodicity settled during the setup phase, the NWDAF provides the actual analytics to the AF obtained over the validation dataset.
The AF evaluating the obtained analytics can assess the Zero-Trust customer's metric and accordingly informs the NWDAF on how to update the configurable parameters. The *steering signal* could be in the form

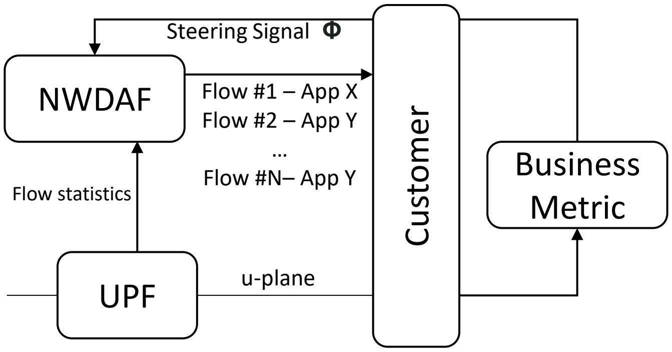


Fig. 4: The validated zero-trust analytics scenario.

of a high-level intent (i.e., informing the NWDAF if the analytics produced so far are assessed as "good" or "bad") appointing the NWDAF as responsible for changing the tunable parameters values, or directly the updated configurable parameters values (or delta to be added/subtracted) in case the AF is capable of understanding their impact over the AI/ML model performance. Note, the AF does not disclose any information about the Zero-Trust customer's metric preserving its privacy and security.

4) The NWDAF upon receiving the *steering signal*, updates the configurable parameters and continues the AI/ML model training. The loop composed of steps 3 and 4 is repeated until the AI/ML model training phase is concluded.
5) When the training phase ends, the AI/ML model is ready to be utilized for producing customized analytics that optimizes the Zero-Trust customer metric. Thus, the NWDAF starts producing the requested analytics and provides them to the customer.

The proposed procedure enables the *cooperative learning* between NOP and Zero-Trust customer in a safe manner: the Zero-Trust customer by receiving analytics during the training phase, is able to learn the training analytics status and drive their production to optimize customer's metric instead of mobile network's metric. In the meanwhile, the NWDAF thanks to the received *steering signals* can learn the goodness of the analytics produced during the learning phase leading to final analytics optimized for the customer's metric even without knowing any information of it.

IV. An application case

We now present and discuss a preliminary study about the possible performance improvements that can be achieved by leveraging the introduced procedure.

A. Scenario

We focus on a scenario depicted in Fig. 4 where the analytics provided to the customer is a flow type detection. This kind of analytics is obtained directly by the User Plane Function (UPF) thanks to the usage of a model that is trained using flow statistics.

This analytics is then passed to the customer which can elaborate its own business metric based on this report (e.g., profiling end users given the kind of traffic they generate) and help the operator to refine the model with the steering signal.

For this purpose, we used a dataset of traffic data available as open source, that contains 2.6 millions of network flow statistics with the associated label describing the associated service type [14]. This dataset contains 43 network associated KPIs (*e.g.*, packet size, flow duration, packet inter-arrival time, ...) that we used to train the model running in the NWDAF with the objective of predicting the associated service.

This task is then naturally associated with a classification problem where the goal is to correctly map the network statistics x to the service class $y \in \mathcal{Y}$ through a function $\mathcal{F} : x \to \mathbb{R}^{|\mathcal{Y}|}$. A Neural Network is used to mimic $\mathcal{F}$ where the output is a probability distribution over the $|\mathcal{Y}|$ possible classes.

To assess the benefits of employing the procedure described in Section III, we steer the Neural Network loss function configurable parameters during the training phase. The default loss function and its enhanced version are presented in Section IV-B. The results presented in Section IV-C show the difference in performances with and without the steering the training of the model based on Zero-Trust customer signals confirming that the collaboration between NOP and customer improves produced analytics.

B. Methodology

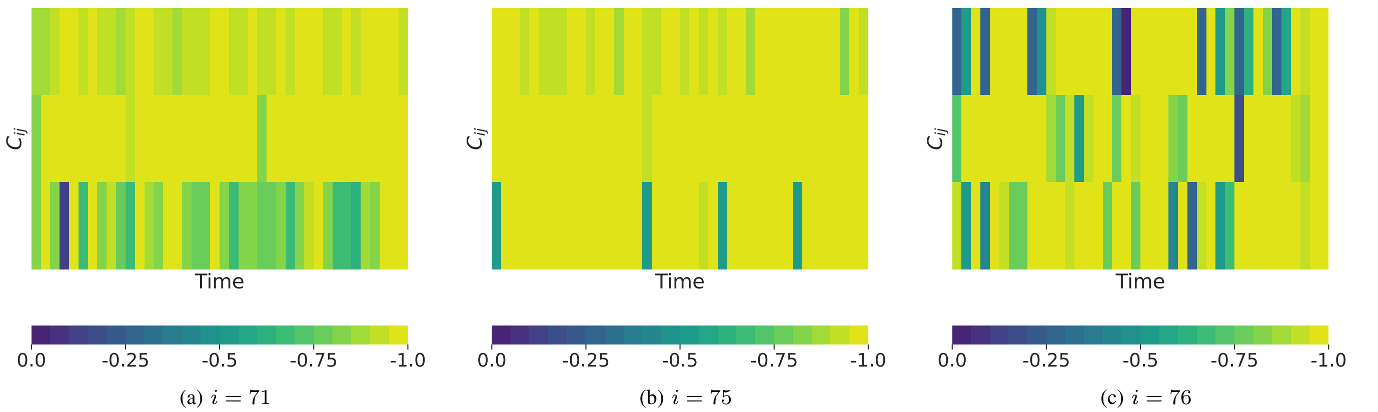
The classifier is enhanced through the dynamic tuning of the loss function configurable parameters, inspired by the work presented in [15]. The most common approach to deal with classification problems is to utilize the Categorical CrossEntropy (CCE) as a loss function. To make the loss function configurable, we leverage Augmented CCE (ACCE), a generalized version of the CCE presented in [15], that allows the steering of the behavior of the model.

$$l_{acce} = -y^T \Phi_t \log f_w(y|x) \tag{1}$$

The matrix $\Phi_t \in \mathbb{R}^{|\mathcal{Y}| \times |\mathcal{Y}|}$ in Eq. (1) encodes time-varying class correlations, a positive value of $\Phi_t(i,j)$ incentivizes the model to predict class j when the expected class label is i, while negative value spreads further apart the classes. The effect is that similar classes at the beginning of the training will be treated as a single class, and as Φ_t changes over time, discriminated during the training phase. This is similar to implementing a *curriculum learning* approach [16]. In our system, y is the one hot encoding that describes the expected class for the sample x, while, f_w is the classifier.

The matrix Φ_t is computed every K epochs. First, we calculate a matrix C that captures when the network mistakenly classifies a sample as picked from class j instead of class i. This matrix is computed as described in Eq. (2)

$$C_{ij} = \begin{cases} 1 & \text{if } i = j \\ \frac{\sum_{\forall x \in D_{val}} I(y_x, i) log(f_w^j x)}{\sum_{\forall x \in D_{val}} I(y_x, i)} & \text{Othrw.} \end{cases} \tag{2}$$



(a) $i = 71$

(b) $i = 75$

(c) $i = 76$

Fig. 5: Evolution of some classes tuple from matrix C through time. Fig. 4-a depicts three $C_{71,j}$ for class $j = \{4, 5, 6\}$. Fig. 4-b shows three classes for $C_{75,j}$ where $j = \{1, 2, 3\}$ while Fig. 4-c for $C_{76,j}$ with $j = \{1, 2, 3\}$. The x-axis represents all the iterations every 10 epochs from the beginning of the training to epoch 400

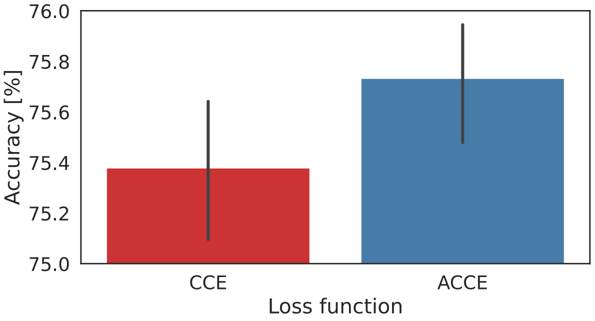


Fig. 6: Test dataset accuracy comparison, 10 experiments average, the error bars represent the 95% confidence interval

Where I is a function that retrieves 1 if the ground truth label of the validation sample y_x belongs to the i-th class. This function retrieves $C \in \mathbb{R}^{|\mathcal{Y}| \times |\mathcal{Y}|}$ that can be used to compute Φ_t, and provides an overview of the most probable mistakes made by f_w. We set Φ_t as an identity matrix, and then for each class i the class j with higher C_{ij} (normalized between -1 and 0 to avoid higher negative values of C_{ij}) is set to -1. In this way, during the training, the class that with higher probability leads to a mistake when evaluating i, is further penalized with a higher loss. By driving the training through Φ_t, the algorithm accuracy improves towards the correct solution.

In total, we repeated 10 experiments with the default CCE loss function l_{cce} and 10 with the ACCE loss function using the same RNG seeds. The dataset is preprocessed before the training cycle, removing missing values and duplicates. It's then normalized and separated into three sub-sets, the training, validation, and test datasets with respectively 60%, 20%, and the remaining 20% of the samples, sampled randomly.

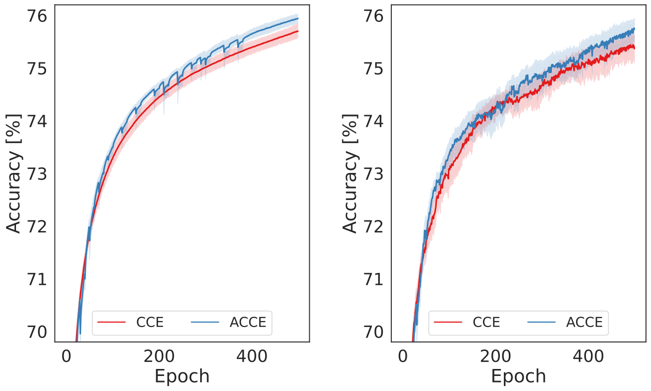
C. Results

Figure 5 shows a graphical representation of a sample of the matrix C for three classes i and how it evolves through time during a training cycle. Each sample contains 3 rows, representing 3 classes that were incorrectly classified by the neural network as class i. Different colors follow values obtained through Eq. (2). C_{ij} are normalized between -1 and 0 every epoch only considering the values set by Eq. (2) (i.e., we only consider class i mistakenly classified as class j). Colors closer to 0 (blue) represent more difficult couples, i.e., class i classified as class j by the neural network model with high probability.

As depicted in Fig. 5, at the first iteration C_{ij} can be close to 0, meaning that the network has wrongly classified validation data inputs related to class i as classes j with high probability and so corrective actions should be taken. As the training time goes on, is possible to see how all the C_{ij} elements in Fig. 5, reach a value closer to -1, meaning that the neural network will unlikely classify a sample from class i to one of the shown classes j. As can be noticed, during the training the network may learn wrong class associations, that are mitigated by properly setting Φ as can be verified in Fig. 5 later stages. The result is that the training ends with an overall accuracy improvement from the initial situation.

Overall, Fig. 5 shows that internally this approach is gradually moving toward a better solution to the problem, shifting the wrongly placed probabilities and enforcing the correct ones.

The overall performances of the two approaches presented in this paper are presented in Fig. 6 where the accuracy over the test dataset is compared over 10 different experiments. The results show that the average accuracy obtained by employing the cooperative procedure described in Section III is higher than the one achieved by the standardized mechanism. Also, it's important to note that the higher point of the 95 percentile error bars achieved by CCE is below the average accuracy obtained by ACCE loss function. This is a confirmation of the gain resulting from applying a cooperative learning approach, i.e., by optimizing Zero Trust's customer metric through *steering signal* and dynamic loss function instead of optimizing NOP's metric utilizing default static loss.



(a) Training Dataset

(b) Validation Dataset

Fig. 7: Accuracy evolution during the training cycle, 10 experiments, the shadow represents the 95%

Figure 7 shows the average accuracy performance during the training cycle of 10 experiments for both approaches. On the left side, Fig. 7a presents the evolution of the average accuracy over the training dataset. Figure 7b, on the other hand, shows the neural network performance over the validation dataset. The x-axis for both figures represents the number of epochs. The overall evolution is similar for both figures with an increase in the variance for the validation dataset as expected. The temporary average accuracy drops depicted in Fig. 7a for ACCE algorithm are caused by a change in the matrix Φ_t that is updated each 10 epochs. For both figures, starting at around 50 epochs, the average accuracy obtained by ACCE approach overcomes the CCE one. This corroborates the fact that a dynamic tuning of the training cycle provides positive effects confirming our initial assumptions.

V. Conclusion and Future Work

In this paper, we make the case for a closer collaboration between network operators and their customers, leveraging an enhanced exposure layer. More specifically, we discuss a zero-trust analytics service, enabling tailored models for different customers. By delivering specific error signals, customers enhance the operator-provided analytics, obtaining a better model that is also more properly aligned to the service business metric. We showcase the advantages of our approach by taking as an example flow detection analytics, obtaining an improved model thanks to the feedback received from the customer utilizing it. The next steps include the design of an autonomous algorithm for the calculation of the steering signal.

Acknowledgement

The work of the University Carlos III of Madrid has been funded by the H2020 Projects TrialsNet (Grant Agreement 101095871) and also partially supported by the Spanish Ministry of Economic Affairs and Digital Transformation and the European Union-NextGenerationEU through the UNICO 5G I+D 6G-CLARION project.

References

[1] A. Garcia-Saavedra and X. Costa-Pérez, “O-RAN: Disrupting the Virtualized RAN Ecosystem,” *IEEE Communications Standards Magazine*, vol. 5, no. 4, pp. 96–103, 2021.

[2] M. Chahbar, G. Diaz, A. Dandoush, C. Cérin, and K. Ghoumid, “A Comprehensive Survey on the E2E 5G Network Slicing Model,” *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 49–62, 2021.

[3] I. Afolabi, M. Bagaa, W. Boumezer, and T. Taleb, “Toward a Real Deployment of Network Services Orchestration and Configuration Convergence Framework for 5G Network Slices,” *IEEE Network*, vol. 35, no. 1, pp. 242–250, 2021.

[4] C. Zhang, X. Wen, L. Wang, Z. Lu, and L. Ma, “Performance Evaluation of Candidate Protocol Stack for Service-Based Interfaces in 5G Core Network,” in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2018, pp. 1–6.

[5] M. Gramaglia, M. Kajo, C. Mannweiler, O. Bulakci, and Q. Wei, “A unified service-based capability exposure framework for closed-loop network automation,” *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 11, p. e4598, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.4598

[6] 3GPP, “Management and orchestration; Concepts, use cases and requirements,” 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 28.530, March 2023, version 17.4.0. [Online]. Available: https://www.3gpp.org/DynaReport/28530.htm

[7] ——, “Architecture enhancements for 5G System (5GS) to support network data analytics services,” 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 23.288, March 2023, version 18.1.0. [Online]. Available: https://www.3gpp.org/DynaReport/23288.htm

[8] ——, “Study on network slice management capability exposure,” 3rd Generation Partnership Project (3GPP), Technical Report (TR) 28.824, July 2023, version 18.0.1. [Online]. Available: https://www.3gpp.org/DynaReport/28824.htm

[9] J. Ordonez-Lucena, C. Tranoris, and J. Rodrigues, “Modeling Network Slice as a Service in a Multi-Vendor 5G Experimentation Ecosystem,” in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2020, pp. 1–6.

[10] Linux Foundation, “Camara Project,” https://camaraproject.org/, [Accessed 27-Jun-2023].

[11] GSMA, “Gsma open gateway,” https://www.gsma.com/futurenetworks/gsma-open-gateway/, [Accessed 21-July-2023].

[12] TMForum, “Open digital architecture ODA,” https://www.tmforum.org/, [Accessed 21-July-2023].

[13] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Perez, “DeepCog: Optimizing resource provisioning in network slicing with AI-based capacity forecasting,” *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 2, pp. 361–376, 2019.

[14] J. S. Rojas, A. Pekar, Á. Rendón, and J. C. Corrales, “Smart user consumption profiling: Incremental learning-based ott service degradation,” *IEEE access*, vol. 8, pp. 207 426–207 442, 2020.

[15] C. Huang, S. Zhai, W. Talbott, M. B. Martin, S.-Y. Sun, C. Guestrin, and J. Susskind, “Addressing the loss-metric mismatch with adaptive loss alignment,” in *International conference on machine learning*. PMLR, 2019, pp. 2891–2900.

[16] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, “Curriculum learning,” in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 41–48.