IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 October 2024, accepted 11 November 2024, date of publication 22 November 2024, date of current version 9 December 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3504729

RESEARCH ARTICLE

A Hybrid Approach to Reliable Jamming Identification in UAV Communications Using Combined DNNs and ML Algorithms

HAMED FARKHARI[1,*], (Member, IEEE), JOSEANNE VIANA[2,*], (Member, IEEE), SARANG KAHVAZADEH[3], (Member, IEEE), PEDRO SEBASTIÃO[1,4], (Member, IEEE), VICTOR P. GIL JIMENEZ[2], (Senior Member, IEEE), AND RUI DINIS[4,5], (Senior Member, IEEE)

[1]ISCTE–Instituto Universitário de Lisboa, 1649-026 Lisbon, Portugal
[2]Departamento de Teoría de la Señal y Comunicaciones, Universidad Carlos III de Madrid (UC3M), 28903 Madrid, Spain
[3]Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), CERCA, 08860 Barcelona, Spain
[4]Instituto de Telecomunicações (IT), 1049-001 Lisbon, Portugal
[5]FCT, Universidade Nova de Lisboa, Monte da Caparica, 2829-516 Caparica, Portugal

Corresponding author: Hamed Farkhari (hamed_farkhari@iscte-iul.pt)

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Programme under the Marie Sklodowska-Curie under Project 813391; in part by the Ministerio de Asuntos Económicos y Transformación Digital (MINECO), in part by the European Union (EU)-NextGenerationEU in the Frameworks of the ''Plan de Recuperación, Transformación y Resiliencia'' and of the ''Mecanismo de Recuperación y Resiliencia'' under Grant TSI-063000-2021-55 and Grant PID2021-126431OB-I00; in part by the Ministerio de Ciencia, Innovación y Universidades (MCIN)/Agencia Española de Investigación (AEI)/10.13039/501100011033; in part by the ''European Regional Development Fund (ERDF) A way of making Europe'' and Generalitat de Catalunya under Grant 2021 SGR 00770; in part by Project ''SOFIA-AIR'' PID2023-147305OB-C31, Ministerio de Ciencia, Innovación y Universidades (MICIU)/AEI/10.13039/501100011033/Ministerio de Ciencia, Innovación y Universidades (FEDER) EU, in part by FCT—Fundação para a Ciência e Tecnologia, I.P., and Instituto de Telecomunicações under Project UIDB/50008/2020, with DOI identifier https://doi.org/10.54499/UIDB/50008/2020; and in part by the Distributed Access Design for Cell-less Smart 6G Networks (CELL-LESS6G) project, 2022. 08786.PTDC with DOI identifier https://doi.org/10.54499/2022.08786.PTDC.

*Hamed Farkhari and Joseanne Viana contributed equally to this work.

ABSTRACT Deep Neural Networks (DNNs) have gained prominence due to their remarkable accomplishments across various domains, including telecommunications and security. Their integration into decision-making processes within 5G telecommunication systems and UAV security is noteworthy. However, the iterative nature of DNN data processing can introduce uncertainties in classification decisions, impacting their reliability. This paper presents novel combined preprocessing and post-processing techniques designed to enhance the accuracy and reliability of binary classification DNNs by managing uncertainty levels. The study evaluates these methods through calibration error metrics, confidence values, and the Reliability Score (RS), which quantifies the disparity between Mean Accuracy (MA) and Mean Confidence (MC). Additionally, the effectiveness of these methods is demonstrated by applying them to simulated real-world scenarios to improve jamming detection reliability in UAV communications. The proposed algorithms' impact is compared against baseline DNNs and DNNs augmented with the eXtreme Gradient Boosting (XGB) classifier, as well as the latest research to validate our approach. This paper comprehensively overviews the experimental setup, dataset, deep network architecture, preprocessing and post-processing techniques, evaluation metrics, and results. By addressing uncertainty in XGB and DNN outputs, this study improves the trustworthiness of ML-DNN-based decision-making processes in 5G UAV security scenarios.

INDEX TERMS Unmanned aerial vehicle, deep neural networks, machine learning, uncertainty, reliability, jamming identification, eXtreme gradient boosting (XGB) classifier, 5G, 6G.

The associate editor coordinating the review of this manuscript and approving it for publication was Xiao-Sheng Si.

I. INTRODUCTION

Deep Neural Networks (DNNs) have gained significant prominence due to their remarkable achievements across various domains, including telecommunications and security [1], [2], [3]. These models are increasingly integrated into decision-making processes within 5G telecommunication systems and Unmanned Aerial Vehicle (UAV) [4] security. Notably, Machine Learning (ML) mechanisms, including DNNs, are anticipated to be incorporated into the standards of 6G telecommunication systems [1]. Extensive research has also focused on leveraging deep learning for decision-making in the physical layer [2]. In the 5G UAV security realm, DNNs offer capabilities such as universal function approximation, exceptional logic for complex time series modeling challenges, and potential for parallel data processing, contingent upon their design [5], [6]. However, the iterative nature of DNNs' data processing during classification tasks can lead to output probabilities accompanied by uncertainties, raising concerns regarding the reliability of classification decisions. Addressing the calibration of DNNs to ensure high accuracy and reliable output decisions is critical, as discussed in [7]. The authors present various calibration techniques that enhance these parameters by leveraging well-known datasets such as CIFAR-10 and ImageNet and pre-trained DNNs such as ResNet, WideNet, and LeNet. As augmentation techniques are integrated into the original data preprocessing stage, understanding concepts such as risk, uncertainty, and trust in a model's output becomes increasingly vital. In [8], the authors propose that preprocessing and post-processing techniques can enhance DNNs' performance. Furthermore, they introduce methods that improve classification accuracy while reducing uncertainty, accompanied by mathematical approaches to compute metrics like Expected Calibration Error (ECE) and Maximum Calibration Error (MCE). In this paper, we present novel combined preprocessing and post-processing techniques to enhance the accuracy and reliability of binary classification DNNs by managing uncertainty levels. Our main contributions are as follows:

- We introduce combined preprocessing and post-processing algorithms that improve the reliability and accuracy of ML-DNN outputs for jamming identification in 5G UAV scenarios.
- We provide a comprehensive overview of the experimental setup, dataset, deep network architecture, preprocessing and post-processing techniques, evaluation metrics, and results, highlighting improvements in trustworthiness for DNN-based decision-making processes in 5G UAV security scenarios.
- We propose a Time-Series Augmentation (TSA) technique as part of the preprocessing phase, generating diverse versions of each sample to provide diversity for post-processing techniques.
- We evaluate the effectiveness of our proposed methods through calibration error metrics, confidence values, and the Reliability Score (RS), quantifying the disparity between Mean Accuracy (MA) and Mean Confidence (MC).
- We demonstrate that the proposed methods can be directly applied to real-world scenarios to enhance the reliability of jamming detection in UAV communications. A comparative analysis is performed against baseline DNN and Enhanced ML-DNN using the eXtreme Gradient Boosting (XGB) classifier.

The structure of the paper is as follows: Section II provides a comprehensive description of all the components involved in the experiment. The dataset is explained in Subsection II-A. In Subsection II-B, the deep network used in the study is discussed. Subsection III delves into the combined preprocessing and post-processing techniques and elucidates how they enhance reliability and accuracy. Subsection III-B presents the metrics employed to evaluate the reliability of each method. Section IV presents the results for the proposed system. Finally, Section V summarizes the main conclusions drawn from the study and presents some topics for future work.

II. SYSTEM MODEL

We consider the DNN jamming classification system based on the Received Signal Strength Indicator (RSSI) and Signal to Interference-plus-Noise Ratio (SINR) signals as described in [9]. The scenario depicted in Figure 1 involves up to four attackers randomly positioned within a designated area, while the UAV maintains a connection to the base station via 5G.

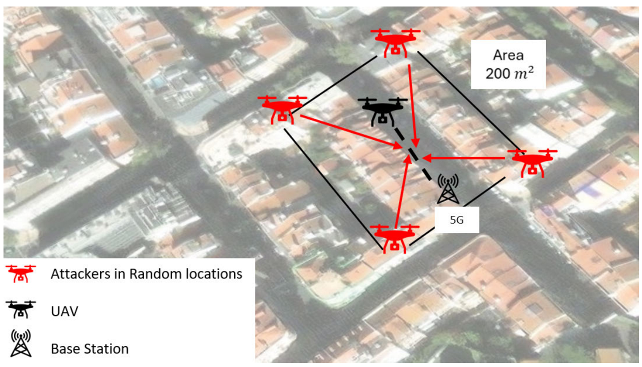


FIGURE 1. Jamming scenario.

The dataset utilized for this research is sourced from the same reference, and the classification is conducted using either a designed DNN or an XGB classifier algorithm. This system enhances accuracy and reliability by integrating preprocessing and post-processing algorithms, as illustrated in Figure 2. The proposed system comprises four main components, arranged from the upper left to right in the figure: the preprocessing algorithm, the primary classifier, the post-processing algorithms, and the auxiliary classifier. The preprocessing algorithm processes the input sample $sample_i$ and augmented samples generated via the TSA technique. The post-processing algorithm incorporates Methods 1, 2, and 3, each with their respective auxiliary algorithms. The system includes the "No Method" block for comparative purposes. At the end of the post-processing algorithm, we perform feature classification and independently evaluate the accuracy and reliability of each algorithm by analyzing the classification results from the primary and auxiliary

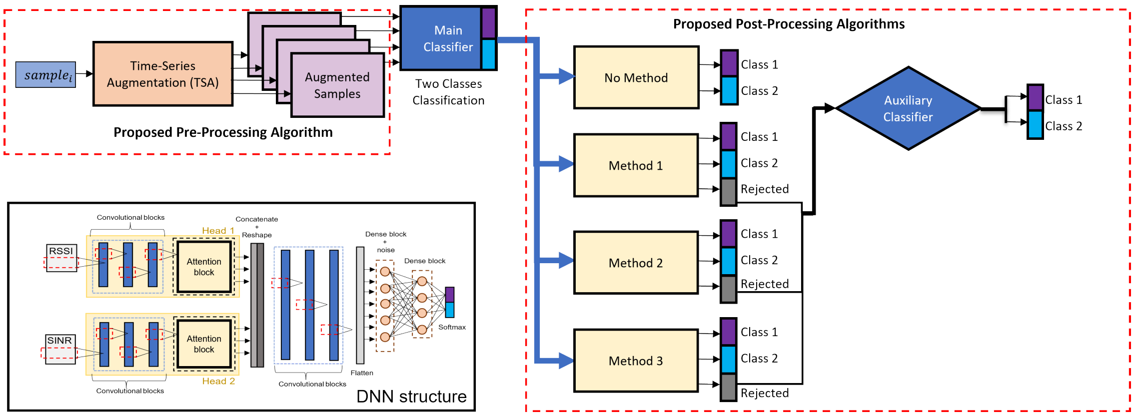


FIGURE 2. Schematic representation of the integrated preprocessing and post-processing algorithms, in which the DNN is the primary algorithm, and ML is the supporting algorithm. The DNN architecture is based on the work by [9].

classifiers. In our study, we employ the XGB classifier algorithm due to its superior performance in detecting the presence or absence of attacks across various scenarios and configurations, as detailed in the dataset discussed in Section II-A and described in [9] and [10].

A. DATASET

Our dataset consists of two signal parameters: RSSI and SINR measurements. These measurements are collected when an authenticated UAV connects to a small cell through the 5G communication system while being subjected to power attacks from other UAVs. Both values were collected from NS3 5G-Lena Simulator in [11]. We transform these two time series variables into supervised samples by employing a rolling window of 300 values. Additionally, other terrestrial users are connected to the network. Consequently, the measured parameters in the authenticated UAV exhibit variations as interference from other devices fluctuate. The dataset includes up to four attackers and 30 terrestrial users connected simultaneously. As part of our research, we are also investigating and analyzing other open-source datasets, including WSN-DS [12] and [13], to obtain a diverse dataset with mature data. Further details regarding the dataset construction and its potential applications can be found in [9] and [14].

B. DEEP NETWORK ARCHITECTURE

In this paper, our primary focus is to conduct a comprehensive analysis of the confidence values within the XGB and deep network, particularly assessing its reliability. Detailed discussions regarding the design and parameters of the DNN utilized in this study are available in previous works [9], [10].

III. DESIGNED SOLUTION

This section presents an overview of the proposed preprocessing and post-processing algorithms, precisely Method 1, Method 2, and Method 3, as well as the implementation of the overall solution.

1) PREPROCESSING TECHNIQUE

In the TSA technique, the time series sequence of each sample is inverted to generate a new augmented sample. For example, the RSSI and SINR will include the original sequence and its inverted counterpart within an appropriate rolling window. During the preprocessing phase, each variable's original and inverted sequences are combined to create four new samples. Generally, the technique can generate $2^{N_{variables}}$ augmented samples, where $N_{variables}$ is the number of variables the DNN uses as input. All original and augmented samples are used to train our ML and DNN, as depicted in Figure 2. After the primary classifier processes the augmented version of each sample, the results are utilized as inputs in the post-processing algorithms. Table 1 provides an example of generating the four new augmented samples using the preprocessing algorithm.

TABLE 1. Output of the TSA.

Sample	Sequence 1 (RSSI)	Sequence 2 (SINR)
Sample 1	Same	Same
Sample 2	Same	Flipped
Sample 3	Flipped	Same
Sample 4	Flipped	Flipped

2) POST-PROCESSING METHODS

In the preprocessing step of the proposed method, the TSA technique is applied to each instance, generating all augmented samples as outlined in Table 1. Subsequently, the primary classifier produces output for each sample, including its augmented versions. This paper introduces three post-processing techniques for the proposed system designated Methods 1, 2, and 3 (M1, M2, and M3). Each method leverages the main classifier outputs of all augmented samples generated in the preprocessing phase, as described in subsection III-1, to evaluate the classification output. These post-processing techniques aim to enhance the primary classifier's classification accuracy and reliability, providing greater flexibility in selecting the optimal approach for a given classification task. Method 1, Method 2, and Method 3 utilize the outputs from the primary classifier to classify each augmented sample into Class 1, Class 2, or Rejected. For samples rejected by Methods 1, 2, or 3, an auxiliary classifier is employed to refine these classifications further, thereby improving overall accuracy and reliability. The auxiliary classifier processes only the rejected results to produce the final classification output. Each classification occurs inside an interval range defined in Section III-3.

3) FILTERS INTERVAL

The primary classifier produces probability outputs for N classes. A filter range encompassing all N probability values is defined for post-processing analysis. In binary classification, a sample is considered accepted if its confidence value exceeds the upper limit of the filter interval and the probability of other classes is below the lower limit of the filter interval. Conversely, a sample is rejected if at least one of the output vector probabilities falls within the filter range. Figure 3a shows the proportion of accepted and rejected samples per filter interval, while Figure 3b illustrates the accuracy of accepted samples after applying the filter. Adopting this filtering approach, the primary classifier can produce more refined outputs and ensure high-quality classification results. This method also facilitates the identification of samples that meet specific criteria, offering a more nuanced evaluation of model performance. In all figures, utilizing a filter range of 0.5-0.5 where the rejected samples are 0% signifies the exclusive application of methods on the main algorithm without conditions for rejecting samples. Consequently, no accepted samples are transmitted to the second auxiliary algorithm.

4) METHOD 1

In Method 1, the initial step identifies whether a sample is accepted or rejected and assigns the corresponding class label if accepted. This step involves checking the filter interval conditions for all four samples. If all four augmented versions are rejected, the sample is classified as rejected, and the process terminates. The output of Method 1 is the average probability per class for the accepted samples. The filter interval condition for this output is then re-evaluated, and the result is categorized as accepted or rejected. Finally, the accepted samples are classified into their respective classes, while the rejected samples are forwarded to an auxiliary classifier for final classification as in algorithm 1.

Algorithm 1 Method 1

Require: $0 \leq yp_{ij} \leq 1$ for $j \in \{1, 2, 3, 4\}, i \in 1, \ldots, N$
Ensure: $Accepted \mid\mid Rejected \leftarrow$ Assign yp_{ij}
  $(\beta_1, \beta_2) \leftarrow \beta$ filter range
  if $yp_{ij} \leq \beta_1 \| \beta_2 \leq yp_{ij}, \forall j \in \{1, 2, 3, 4\}$ **then**
    $yp_i \leftarrow Average_j(yp_{ij})$
  end if
  if $yp_i \leq \beta_1 \| \beta_2 \leq yp_i$ **then**
    $Accepted \leftarrow yp_i$
  else
    $Rejected \leftarrow yp_i$
  end if

5) METHOD 2

Method 2's filter interval conditions for each augmented output are not individually checked. Instead, the average probability per class is computed across all four results. This average is then evaluated against the filter interval conditions to determine whether the sample should be accepted or rejected. This approach contrasts with Method 1, in which the filter interval conditions for each augmented output are matched before computing the average probability per class for the accepted samples. The detail of Method 2 is presented in algorithm 2.

Algorithm 2 Method 2

Require: $0 \leq yp_{ij} \leq 1$ for $j \in \{1, 2, 3, 4\}, i \in 1, \ldots, N$
Ensure: $Accepted \mid\mid Rejected \leftarrow$ Assign yp_{ij}
  $(\beta_1, \beta_2) \leftarrow \beta$ filter range
  $yp_i \leftarrow Average_j(yp_{ij}), \forall j \in \{1, 2, 3, 4\}$
  if $yp_i \leq \beta_1 \| \beta_2 \leq yp_i$ **then**
    $Accepted \leftarrow yp_i$
  else
    $Rejected \leftarrow yp_i$
  end if

6) METHOD 3

Method 3, as detailed in Algorithm 3, explores three distinct techniques. The first technique, M3-max, involves selecting the output with the highest confidence. The second strategy, M3-min, applies a minimum trust threshold to reject samples with low confidence when integrating this method into the primary algorithm. In M3-N, augmented samples with the highest and lowest confidence values are initially eliminated due to concerns of overconfidence and underconfidence. Subsequently, the remaining augmented versions are averaged. Finally, averaging is performed over all enriched samples without removal when no augmented outputs remain after excluding high and low-confidence results. All techniques are summarized in Table 2. Section IV presents the results of Method 3.

7) CONFIDENCE VALUES

In DNNs, the softmax layer is typically employed as the output layer to generate a probability distribution over a set of classes for each input sample. The resulting output is a one-hot encoded vector, where each element represents the probability of the corresponding category. To determine the correct class for each sample, only the maximum value in the vector is considered as one, indicating the most probable class, while the rest are rounded to zero. Our study posits that the confidence level of a sample's classification in N-class problems can be determined by considering the maximum probability value of its corresponding one-hot vector. This approach results in confidence values ranging from 0.5 to 1 in binary classification. A minor difference between the confidence score and its rounded

Algorithm 3 Method 3

Require: $0 \leq yp_{ij} \leq 1$ for $j \in \{1, 2, 3, 4\}, i \in 1, \ldots, N$
Ensure: *Accepted* || *Rejected* || *Output* $\leftarrow$ Assign yp_{ij}
  $(\beta_1, \beta_2) \leftarrow \beta$ filter range
  $conf_b \leftarrow$ confidence base
  if $conf_b$ is Max **then** ▷ Method 3-max
    $yp_i \leftarrow \arg\max confidence_j(yp_{ij}), \forall j \in \{1, 2, 3, 4\}$
  else if $conf_b$ is Min **then** ▷ Method 3-min
    $yp_i \leftarrow \arg\min confidence_j(yp_{ij}), \forall j \in \{1, 2, 3, 4\}$
  else ▷ Method 3-N
    $j_1 \leftarrow \arg\max confidence_j(yp_{ij})$
    $j_2 \leftarrow \arg\min confidence_j(yp_{ij})$
    $yp_i \leftarrow Average_j(yp_{ij}), \forall j \in \{1, 2, 3, 4\} - \{j_1, j_2\}$
    if yp_i is {} **then**
      $yp_i \leftarrow Average_j(yp_{ij}), \forall j \in \{1, 2, 3, 4\}$
    end if
  end if
  if β filter defined **then**
    if $yp_i \leq \beta_1 \| \beta_2 \leq yp_i$ **then**
      $Accepted \leftarrow yp_i$
    else
      $Rejected \leftarrow yp_i$
    end if
  else
    $Output \leftarrow yp_i$
  end if

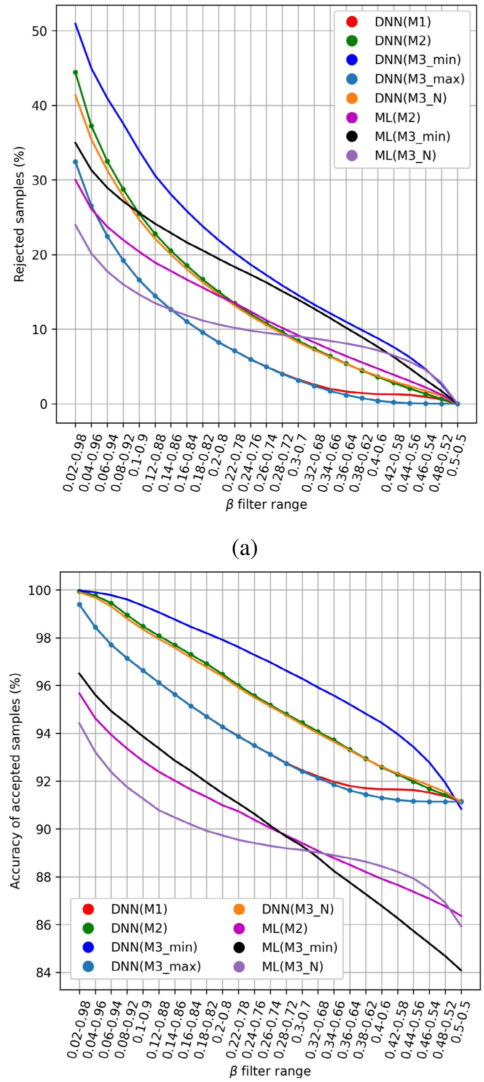
value indicates higher reliability for correctly classified samples. By leveraging this confidence value, one can assess the quality of the classification output and evaluate the classifier's performance.

8) OVERFITTING-UNDERFITTING OVER SAMPLES

One of the motivations for introducing M3-N is the observed impact of M3-Max and M3-Min on our experimental results. While M3-Max involves selecting samples with the highest confidence, it does not improve accuracy. Instead, it leads to unstable uncertainty when combined with different algorithms, as evidenced in Table 2. Conversely, M3-Min results in a significant decrease in the quality of the final results, suggesting overfitting on samples with the highest confidence and underfitting on instances with the lowest confidence. M3-N is an initial step in studying this effect and attempts to enhance reliability by mitigating this phenomenon. Further research is required on different datasets and deep networks with equitable augmentation on sampling. We propose using proper augmentation, where all augmented versions contain similar information, such as the flipping technique, and avoiding using lossy augmentation like cropping to study this phenomenon.

A. ALGORITHM COUPLING AND FINAL SETUP

The final classification results are obtained by integrating the outputs of Deep Neural Networks (DNN) and Machine Learning (ML) using the aforementioned methods. Two



(a)

(b)

FIGURE 3. Flexible ranges with a resolution of 0.02 between the accuracy of accepted samples as classes 1 or 2 and the portion of rejected samples based on the β filter ranges: (a) Samples versus β (b) Accuracy versus β. The results of ML(M1) and ML(M3_max) were removed due to the worst achievement.

distinct scenarios are considered. In the first scenario, a fast ML algorithm is the primary classifier, while the DNN is an auxiliary classifier. Conversely, in the second scenario, the DNN is the primary classifier, with the ML algorithm acting as an additional classifier. During our experiments, the second scenario neither yielded significant improvements nor justified the additional complexity introduced to the final design. After thorough analysis, this scenario was ultimately discarded due to its suboptimal results. The rationale for the first scenario is that a faster ML algorithm can efficiently handle simple samples. If the ML algorithm lacks confidence, the more precise DNN can manage the complex samples, ensuring a balance between speed and accuracy. Various metrics are evaluated in the context of the following hybrid combinations:

- The output of only the main classifier as a baseline;
- The output of the primary classifier after applying either Method 1, Method 2, or Method 3, followed by using a secondary classifier for the rejected samples.

We have implemented the trained machine learning algorithms within the UAV to meet the fast-processing requirements essential for enabling the UAV to make timely and effective decisions while accomplishing its mission.

B. EVALUATION METRICS

We employ widely recognized metrics proposed by [7] to assess the model's uncertainty, accuracy, and quality, enabling a comparison of the improvements made by different methods. The following section provides a detailed explanation of these metrics:

1) ACCURACY PER CONFIDENCE

The visual representation of this metric is utilized to analyze the calibration and uncertainty characteristics of the DNN model. This chart, commonly called the "reliability diagram" by the authors in [15] and [16], evaluates the model's reliability. The metric is computed by partitioning the samples into groups based on their confidence values within specified interval ranges and subsequently estimating the accuracy of each group. Our deep network architecture employs a one-hot encoding output scheme with a softmax activation function and binary cross-entropy loss function. Given that the DNN under study produces results in one-hot encoding probabilities, the maximum probability value among the predicted output classes is assigned as the confidence score. The confidence values are then grouped within interval ranges from 0.5 to 1, with each interval defined by the user.

2) MEAN CONFIDENCE AND MEAN ACCURACY

These metrics, Mean Confidence (MC) and Mean Accuracy (MA), represent the total weighted average of confidence and accuracy for the number of samples within each confidence interval. In a fair scenario concerning reliability and accuracy, these two values should be equal. However, in DNN architectures, it is often observed that these values tend to exhibit biases towards one extreme or the other. Over-confidence arises when the Mean Confidence surpasses the Mean Accuracy, indicating excessive confidence in the model's predictions. Conversely, Under-confidence occurs when the Mean Accuracy exceeds the Mean Confidence, indicating a lack of confidence in the model's predictions. Calibrating uncertainty can bring the model's probabilistic outputs closer to the desired levels. This calibration process aims to minimize or eliminate any loss in accuracy values while achieving optimal confidence levels.

3) RELIABILITY SCORE

The distinction between the MC and MA values is defined in this paper by a metric referred to as the Reliability Score (RS). When the RS equals zero, the DNN achieves an optimal balance of accuracy and reliability. Over-confidence arises when the MC surpasses the MA, while Under-confidence occurs when the reverse condition holds. Previous research conducted by the authors in [7] demonstrates that DNNs with N classes and M inputs tend to exhibit overconfidence. Our study proposes that the implementation of simple preprocessing and post-processing algorithms has the potential to alter this behavior.

4) EXPECTED AND MAXIMUM CALIBRATION ERRORS

In the context of this paper, the error per confidence interval is determined by measuring the accuracy deviation from the center of the interval. The Expected Calibration Error quantifies the weighted error across all intervals [17], while the Maximum Calibration Error represents the maximum error observed among all intervals. In an ideal scenario, both errors would be zero.

5) NORMALIZED NEGATIVE LOG LIKELIHOOD LOSS (NLL)

The metric referred to as cross-entropy loss is employed as a loss function for DNNs [18], [19]. Furthermore, it serves as a metric for evaluating the efficacy of probabilistic models [20]. Initially, for each sample output from the DNN, the negative logarithm of the predicted probability of the ground truth class is computed. Then, these values are normalized per sample and summed together.

6) BRIER SCORE LOSS (BSL)

This metric is formulated as the mean squared error between the predicted probability, which ranges from zero to one, and the actual outcome, restricted to values of 0 or 1. The primary objective is to minimize this metric, aiming for a value that approaches zero as closely as possible [7]. While the metric inherently falls within the zero to one range, it is presented as a percentage to enhance the comparability of our research findings. To expand the applicability of the BSL and accommodate multiclass classification scenarios, we employ a computation method that compares the predicted output, represented as probabilities using one-hot encoding, with the corresponding ground truth output. The ground truth output is also encoded using a one-hot representation comprising zeros and ones. By calculating the squared difference between these two sets of values, we obtain the BSL for multiclass classification, averaged across all samples [21]. The computation of the BSL follows the formulation presented in Eq. (1), where M signifies the total number of samples. The variable N denotes the number of classes involved in the classification task. Furthermore, yp indicates the predicted probability for each class, while yg represents the ground truth encoded in one-hot form, with values of either zero or one for each class.

$$BSL = \frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{N} (yp_{ij} - yg_{ij})^2. \tag{1}$$

TABLE 2. Performance evaluation of Methods 2 and 3 with varied output selection approaches in Validation and Test Pairs without considering Filter Range. Key performance parameters for various metrics are compared. M2 uses the average of augmented samples, M3-Max selects the output with maximum confidence, M3-Min selects the output with minimum confidence, and M3-N performs averaging while excluding the highest and lowest confidence values. The best test values in each column for DNN and ML are bold.

Val., Test	Accuracy (%)	AUC	ECE (%)	MCE (%)	BSL (%)	NLL	RS=MA-MC
DNN	94.58, 90.98	0.95, 0.91	**4.61, 3.48**	**7.8, 6.81**	7.95, 12.64	0.13, 0.2	**4.61, 1.21**
DNN (M2)	**94.85, 91.15**	**0.95, 0.92**	4.95, 3.77	10.3, 7.59	**7.58, 12.27**	**0.13, 0.19**	4.91, 1.4
DNN (M3-Min)	94.34, 90.84	0.94, 0.91	6.79, 4.04	18.25, 4.99	8.81, 12.4	0.15, 0.2	6.8, 3.76
DNN (M3-Max)	94.86, 91.14	**0.95, 0.92**	4.13, 3.75	8.93, 16.28	7.58, 13.37	0.13, 0.22	8.93, 16.28
DNN (M3-N)	94.84, 91.12	**0.95, 0.92**	4.81, 3.79	9.52, 7.93	7.57, 12.31	**0.13, 0.19**	4.79, 1.25
ML	87.2, 85.5	0.89, 0.87	5.17, 6.91	23.71, 27.37	21.06, 24.4	0.5, 0.6	-5.17, -6.91
ML (M2)	**88.48, 86.36**	0.89, 0.88	**1.61, 3.77**	**14.54, 18.89**	**17.43, 21.04**	0.35, 0.42	**-1.6, -3.77**
ML (M3-Min)	85.7, 84.08	0.87, 0.86	3.14, 4.26	17.07, 22.16	19.58, 22.35	**0.35, 0.41**	-2.61, -4.26
ML (M3-Max)	88.53, 86.28	**0.9, 0.88**	6.24, 8.52	32.25, 33.21	21.88, 26.38	0.69, 0.85	-6.24, -8.52
ML (M3-N)	88.03, 85.93	0.89, 0.88	2.99, 4.93	23.16, 27.87	18.68, 22.3	0.4, 0.5	-2.84, -4.93

Considering the utilization of the softmax function in our deep network for binary classification, our output predictions were binary but represented in the form of one-hot encoding. As a result, we have employed Eq. (1) as the method for computing the BSL.

IV. EXPERIMENTAL RESULTS

This section presents the outcomes of the two hybrid combinations and the scenario that uses fast ML as a primary classifier and DNN as the secondary classifier described in Section III-A. For each combination of algorithms and methods, results are produced for various filter choices in the test sets. Given that the filter variable is a hyperparameter, an appropriate filter is selected based on the validation results. These results are then utilized to assess the extent of divergence and disparity between the validation and test sets. Since relying on a single metric is insufficient for selecting the optimal filter choice, a comprehensive analysis is provided using all metrics outlined in Section III-B. Variations in the filters are determined by comparing these metrics and evaluating each result. The average deep neural network prediction from 10 runs is employed in this analysis.

A. FILTERING EFFECT

Despite combining different methods and algorithms, one of the most straightforward strategies involves altering the number of classes by introducing an additional category for rejected samples. This approach allows us to focus on classifying only highly reliable instances while categorizing the remaining samples into a separate neutral class. As a result, the percentage of rejected samples in both the validation and test sets remains nearly identical. Similarly, the achieved accuracy for accepted samples exhibits a similar behavior between the validation and test sets, albeit with slight variations in percentage. The percentage of rejected samples for the test set is illustrated in Figure 3a, and the accuracy for accepted samples for the test set is indicated in Figure 3b. Applying Methods 1 and 3-max on the DNN in this context yields similar results, while Methods 3-min and 2 produce more substantial changes. When comparing various methods on DNN using the narrowest filter range of 0.02 to 0.98, the utilization of Method 1 or 3-max results in a notable enhancement in accuracy, increasing from 90.98% to 99.39%. This improvement is achieved by discarding approximately 32.5% of the test set's data. Conversely, Method 2 attains an accuracy of 99.94%; however, it necessitates a higher proportion of data rejection in the test set, approximately 44.5%. Moreover, when the same filter range is applied to other methods, such as M3-N and M3-min, the rejection of 41% and 51% of the test set data leads to achieved accuracies of 99.9% and 99.97%. The suboptimal results obtained from the application of Method 1 on ML necessitate its removal from Figure 3. Consequently, the Method 1 approach is not recommended for the extra class strategy when ML is used as the primary classifier. Moreover, Method 2 significantly improves ML outputs, raising the accuracy from 86% to 95.67% when the same filter range of 0.02 to 0.98 is employed. In this case, the rejected sample rate is approximately 30%. We recommend the neutral class strategy tasks where AI's decision-making can be bypassed in hazardous situations, deferring to human experts or alternative algorithms. Consequently, the subsequent analysis will focus on applying another method or algorithm to the neutral class (rejected samples) to ensure the classification of all samples and examine its impact on reliability and accuracy, considering various calibration metrics.

B. ACCURACY ANALYSIS

Table 2 demonstrates that analyzing the combination of techniques and algorithms while considering their impact on uncertainty can lead to a slight increase in overall accuracy but negatively affects most calibration metrics (specifically in ML) or, to a lesser extent, in DNN. Given these observations, the necessity of conducting a comprehensive metrics analysis beyond accuracy becomes evident when comparing the effects of different methods. This approach allows for a more precise selection of combination methods and filter ranges to enhance most metrics or, at the very

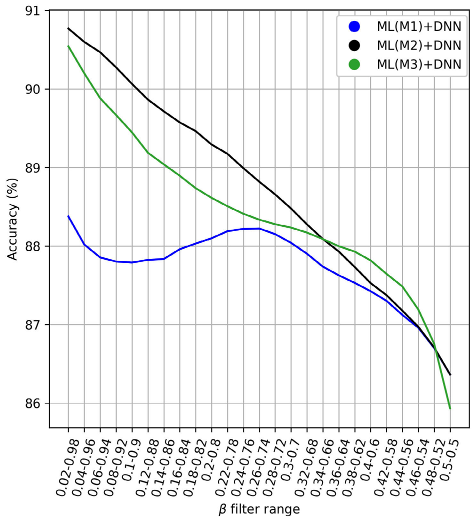
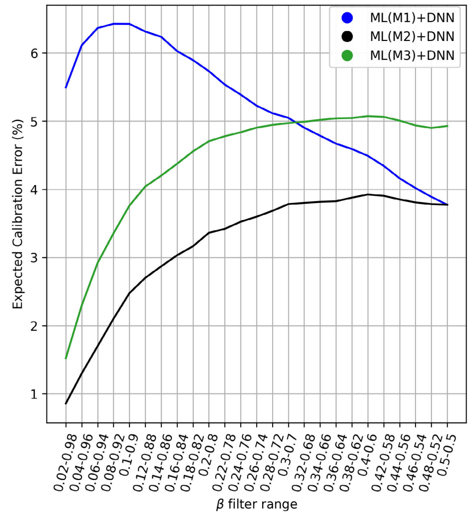


FIGURE 4. **Accuracy per filters for the combination of algorithms and methods, using fast ML (XGB) as the primary classifier and DNN as the complementary algorithm.**

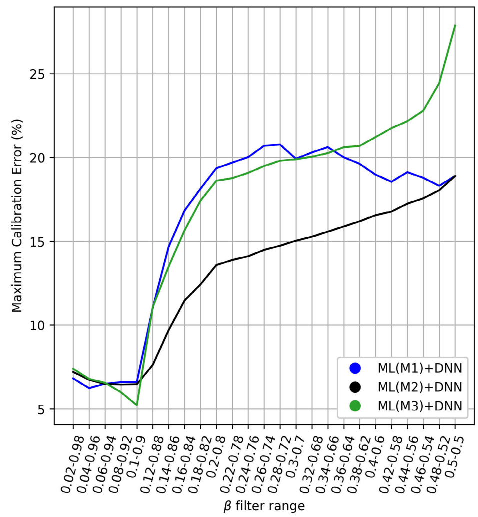
least, improve accuracy while minimizing sacrifices in other metrics. In Figure 4, three combinations of methods and algorithms are presented for various filters, highlighting fast ML as the primary algorithm. Adjusting certain filters enables achieving test accuracy above 90% for two combinations. The impact of this combination on uncertainty analysis for a potentially favorable filter range will be further examined. However, according to Table 2, if Method 3-Max is applied, no improvement in uncertainty is expected, as Method 3-Max itself adversely affects uncertainty. Based on the findings from Table 2, the detrimental effect of M3-Max on uncertainty metrics prompts the exclusive use of the M3-N version for subsequent analysis to mitigate the decline in reliability. The selection of the M3-N version over M3-Max is driven by the need to minimize adverse effects on uncertainty metrics. This decision is informed by observed outcomes and the aim to uphold a higher level of reliability throughout the analysis. Using the M3-N version ensures that the analysis prioritizes reducing negative impacts on reliability while maintaining consistency in evaluating uncertainty metrics. When filters demand higher confidence, more samples are transferred to DNN, increasing accuracy. As anticipated from Figure 3, the combination of ML (Method 2) yields the most significant changes in accuracy. The combination of ML (Method 2) + DNN indicates a slightly superior effect on accuracy compared to employing ML (Method 3) + DNN.

C. ECE AND MCE ANALYSIS

The analysis of ECE behavior, where ML assumes a primary role while employing DNN as a secondary algorithm, is depicted in Figure 5a. The most effective approaches identified within hybrid algorithms and methodologies are ML(M2)+DNN and ML(M3)+DNN. When comparing different hybrid algorithms that integrate ML(M2) or ML(M3)



(a) ECE per filters.

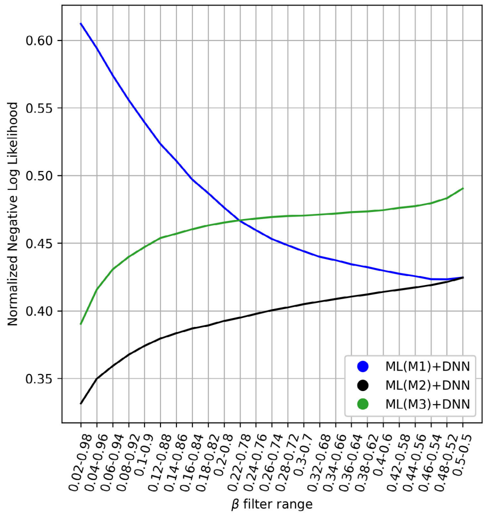


(b) MCE per filters.

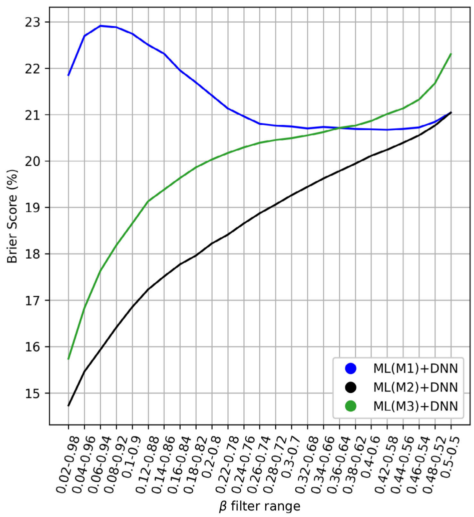
FIGURE 5. **ECE and MCE per filters for the combination of algorithms and methods, using fast ML (XGB) as the main algorithm and DNN as the complementary algorithm.**

as the main component, it is evident that employing ML(M2) yields superior outcomes.

The MCE analysis offers valuable insights when considering ECE analysis, leading to more comprehensive conclusions. As depicted in Figure 5b, it is advisable to narrow the filter range from 0.2-0.8 to 0.02-0.98 to restrict the MCE values while simultaneously achieving improvements in this metric. The intersection of the new filter range is determined by considering ECE, MCE, and accuracy transitions. Based on Figure 5b, the analysis of MCE indicates that for all hybrid algorithms combining ML with DNN using various methods, it is feasible to obtain reliable MCE values by adjusting the range from 0.02-0.98 to 0.1-0.9. This adjustment is based on the validation results, which are also valid for the test set. This new tightened range handles



(a) NLL per filters.



(b) Brier Score per filters.

FIGURE 6. **NLL and Brier Score per filters for the combination of algorithms and methods, using fast ML (XGB) as the main algorithm and DNN as the complementary algorithm.**

around 70 to 80% of the data by ML. It transfers only around 20 to 30% of samples to DNN for classification as demonstrated in 5b, leading to a total accuracy of 89.5 to 91%.

D. NLL AND BSL ANALYSIS

The NLL and BSL are reliable indicators of model quality. Figures 6a and 6b illustrate their comparable behavior across different algorithms and methods applied per filter. Lower values of these metrics correspond to higher model quality. When XGB assumes the primary role and the filtered samples are fed into the DNN, an improvement in model quality is expected with an increase in the number of samples analyzed by the DNN. However, contrary to this expectation, the results indicate that the ML(M1) + DNN combination does not yield satisfactory performance based on the overall metrics. Therefore, Method 1 is not recommended for implementation when ML is the main algorithm. Different combinations utilizing Method 2 or 3 for XGB significantly demonstrate improved model quality. Specifically, when the filter range is closer to 0.02-0.98, the ML(M2)+DNN combination exhibits enhanced model quality.

E. FILTER RANGE SELECTION

Based on various metrics, when selecting an appropriate range for beta filters with ML playing the principal role, it is preferable to choose the left side of the filter range, spanning from 0.02-0.98 to 0.1-0.9. If a particular use case assigns greater importance to one of these metrics over the others, these recommendations may need to be adjusted accordingly.

V. CONCLUSION

In the context of 5G and 6G UAV communication systems, integrating fast Machine Learning (ML) and Deep Neural Networks (DNN) mechanisms is highly anticipated. Consequently, it is essential to comprehend the uncertainties associated with their utilization and assess their reliability when used individually or in combination. This paper explores the uncertainties of ML and DNN algorithms in these systems, considering different filtering ranges based on uncertainty metrics. By examining the reliability and total accuracy, we demonstrate that combining a DNN algorithm with ML can enhance overall accuracy. However, to ensure reliability is not compromised, limiting the contribution of the ML algorithm is crucial. As ML algorithms exhibit faster processing than DNN, employing a fast ML algorithm as the primary algorithm and DNN as an auxiliary can mitigate the negative impact of DNN prediction latency in 5G/6G networks while improving overall accuracy and reliability.

We study and analyze various probabilistic-based combinations to identify the most reliable and accurate combination, considering uncertainty metrics. We define a probability filtering range and introduce different methods for combining outputs within this reliable range. Furthermore, we demonstrate that augmentation techniques can enhance reliability in this combination approach. Our study proposes three main combined methods to concurrently increase accuracy and reliability in binary classification applied to UAV security scenarios. We aim to identify the optimal range and the most reliable combination based on analyzing reliability metrics. By implementing suitable preprocessing techniques, such as Time-Series Augmentation (TSA), for classification tasks, we demonstrate the generation of diverse versions of each sample, providing diversity for post-processing techniques.

Ultimately, while the proposed methods successfully improve accuracy, not all enhance reliability. Therefore, network engineers and developers must exercise caution when designing DNN architectures and thoroughly analyze them for accuracy and reliability. Our study focuses on the collaborative utilization of DNN and ML algorithms, such as when the ML algorithm serves as the main algorithm and the DNN as an auxiliary. The objective of this study does not encompass simultaneous or parallel combinations of both DNN and ML algorithms or the usage of different DNN or ML algorithms for parallel predictions. These topics may be considered for future research, requiring the development of appropriate methods and algorithms to explore their effects.

REFERENCES

[1] X. Lin, "An overview of 5G advanced evolution in 3GPP release 18," *IEEE Commun. Standards Mag.*, vol. 6, no. 3, pp. 77–83, Sep. 2022, doi: 10.1109/MCOMSTD.0001.2200001.

[2] Z. Qin, H. Ye, G. Y. Li, and B. F. Juang, "Deep learning in physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 93–99, Apr. 2019, doi: 10.1109/MWC.2019.1800601.

[3] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, "Machine learning and deep learning methods for cybersecurity," *IEEE Access*, vol. 6, pp. 35365–35381, 2018, doi: 10.1109/ACCESS.2018.2836950.

[4] T. Taleb, N. Sehad, Z. Nadir, and J. Song, "VR-based immersive service management in B5G mobile systems: A UAV command and control use case," *IEEE Internet Things J.*, vol. 10, no. 6, pp. 5349–5363, Mar. 2023, doi: 10.1109/JIOT.2022.3222282.

[5] Y. Li, J. Pawlak, J. Price, K. Al Shamaileh, Q. Niyaz, S. Paheding, and V. Devabhaktuni, "Jamming detection and classification in OFDM-based UAVs via feature- and spectrogram-tailored machine learning," *IEEE Access*, vol. 10, pp. 16859–16870, 2022, doi: 10.1109/ACCESS.2022.3150020.

[6] A. Krayani, A. S. Alam, L. Marcenaro, A. Nallanathan, and C. Regazzoni, "Automatic jamming signal classification in cognitive UAV radios," *IEEE Trans. Veh. Technol.*, vol. 71, no. 12, pp. 12972–12988, Dec. 2022, doi: 10.1109/TVT.2022.3199038.

[7] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. 34th Int. Conf. Mach. Learn.* (Proceedings of Machine Learning Research), vol. 70, D. Precup and Y. W. Teh, Eds. PMLR, Aug. 2017, pp. 1321–1330. [Online]. Available: http://proceedings.mlr.press/v70/guo17a/guo17a.pdf

[8] D. Hafner, D. Tran, T. Lillicrap, A. Irpan, and J. Davidson, "Noise contrastive priors for functional uncertainty," in *Proc. 35th Uncertainty Artif. Intell. Conf.*, vol. 115, R. P. Adams and V. Gogate, Eds., Jul. 2020, pp. 905–914. [Online]. Available: https://proceedings.mlr.press/v115/hafner20a.html

[9] J. Viana, H. Farkhari, P. Sebastião, L. M. Campos, K. Koutlia, B. Bojovic, S. Lagén, and R. Dinis, "Deep attention recognition for attack identification in 5G UAV scenarios: Novel architecture and end-to-end evaluation," *IEEE Trans. Veh. Technol.*, vol. 73, no. 1, pp. 131–146, Jan. 2024, doi: 10.1109/TVT.2023.3302814.

[10] J. Viana, H. Farkhari, L. M. Campos, P. Sebastião, K. Koutlia, S. Lagén, L. Bernardo, and R. Dinis, "A convolutional attention based deep learning solution for 5G UAV network attack recognition over fading channels and interference," in *Proc. IEEE 96th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2022, pp. 1–5, doi: 10.1109/VTC2022-Fall57202.2022.10012726.

[11] N. Patriciello, S. Lagen, B. Bojovic, and L. Giupponi, "An E2E simulator for 5G NR networks," *Simul. Model. Pract. Theory*, vol. 96, Nov. 2019, Art. no. 101933, doi: 10.1016/j.simpat.2019.101933. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569190X19300589

[12] I. Almomani, B. Al-Kasasbeh, and M. AL-Akhras, "WSN-DS: A dataset for intrusion detection systems in wireless sensor networks," *J. Sensors*, vol. 2016, pp. 1–16, Jan. 2016, doi: 10.1155/2016/4731953.

[13] O. Puñal, C. Pereira, A. Aguiar, and J. Gross. (May 2014). *CRAWDAD Dataset Uportorwthaachen/Vanetjamming2012 (v. 2014-05-12)*. Downloaded From. [Online]. Available: https://crawdad.org/uportorwthaachen/vanetjamming2012/20140512

[14] J. Viana, H. Farkhari, P. Sebastiao, S. Lagen, K. Koutlia, B. Bojovic, and R. Dinis, "A synthetic dataset for 5G UAV attacks based on observable network parameters," 2022, *arXiv:2211.09706*.

[15] M. H. DeGroot and S. E. Fienberg, "The comparison and evaluation of forecasters," *Statistician*, vol. 32, no. 1, p. 12, Mar. 1983. [Online]. Available: http://www.jstor.org/stable/2987588

[16] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *Proc. 22nd Int. Conf. Mach. Learn. (ICML)*. New York, NY, USA: Association for Computing Machinery, 2005, pp. 625–632, doi: 10.1145/1102351.1102430.

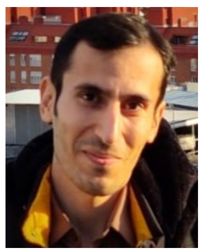
[17] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, and J. Snoek, "Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift," in *Proc. Adv. Neural Inf. Process. Syst.*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Red Hook, NY, USA: Curran Associates, 2019, pp. 1–11. [Online]. Available: https://proceedings.neurips.cc/paperfiles/paper/2019/file/8558cb408c1d76621371888657d2eb1d-Paper.pdf

[18] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[19] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (Springer Series in Statistics). Cham, Switzerland: Springer, 2009, doi: 10.1007/978-0-387-84858-7.

[20] M. P. Naeini, F. G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using Bayesian binning," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 2901–2907, doi: 10.1609/aaai.v29i1.9602.

[21] G. W. Brier, "Verification of forecasts expressed in terms of probability," *Monthly Weather Rev.*, vol. 78, no. 1, pp. 1–3, Jan. 1950, doi: 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.



HAMED FARKHARI (Member, IEEE) is currently pursuing the Ph.D. degree with the ISCTE–University Institute of Lisbon. He is an Early-Stage Researcher with the TeamUp5G Group, a European Training Network, under the Marie Skłodowska-Curie Actions (MSCA ITN) of the European Commission's Horizon 2020 Program. His research interests include cybersecurity, machine learning, deep learning, data science, meta-heuristic techniques, and optimization algorithms.

JOSEANNE VIANA (Member, IEEE) received the bachelor's degree in telecommunication engineering from the University of Campinas, Brazil. She is currently pursuing the Ph.D. degree with the Universidad Carlos III de Madrid (UC3M). She is an Early-Stage Researcher in the project TeamUp5G, a European Training Network under the MSCA ITN of the European Commission's Horizon 2020. Her research interests include wireless communications applied to interconnected systems, such as UAVs, aerial vehicles, and non-terrestrial devices.

SARANG KAHVAZADEH (Member, IEEE) received the B.S. degree in industrial engineering from Olom va Fonon Mazandaran, Iran, in 2010, the M.S. degree in software information technology from Debrecen University, Hungary, in 2014, and the Ph.D. degree from the Department of Computer Architecture, Universitat Politècnica de Catalunya Barcelona Tech–UPC, Barcelona, Spain, in 2019. He is currently a Researcher with the Services as NetworkS (SaS) Research Unit, Center Tecnologic de Telecommunications de Catalunya (CTTC), Barcelona. He has been involved in numerous European projects and co-authored many articles in international journals and conferences. His research interests include security, 5G NFVI, Kubernetes, OSM, cloud-native, microservices, testbed as a service, NFV, and distributed security mechanisms in the 5G environment.

PEDRO SEBASTIÃO (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from IST. He is currently a Professor with the Information Science and Technology Department, ISCTE-IUL. He is also the Board Director of the AUDAX-ISCTE's Entrepreneurship and Innovation Center, ISCTE. He is responsible for the LABS LISBOA Incubator. He is a Researcher with the Institute of Telecommunications. He has served as an Expert and an Evaluator for over 100 national and international civil and defense research and development projects. He has supervised several master's dissertations and Ph.D. theses. Furthermore, he has led several national and international research and development projects. He has received several awards in scientific, engineering, and pedagogical fields. He has organized or co-organized more than 55 national and international scientific conferences. In addition, he has planned and developed several postgraduate courses in technologies and management, entrepreneurship and innovation, and technology and innovation transfer. He has supported numerous projects involving technology transfer and the creation of startups and spinoffs that contribute value to society and the market. His professional experience includes work in the National Defense Industries, initially in the Office of Studies and later as the Board Director of the Quality Department for the Production of New Products and Technologies. He was also responsible for communication technology systems in the Nokia-Siemens business area. He is the author or the co-author of over 200 scientific articles. His research interests include monitoring, control, and communications of drones and unmanned vehicles, planning tools, stochastic processes (modeling and efficient simulations), the Internet of Things, and efficient communication systems.

VICTOR P. GIL JIMENEZ (Senior Member, IEEE) received the B.S. degree (Hons.) in telecommunications from the University of Alcala, in 1998, and the M.S. and Ph.D. degrees (Hons.) in telecommunications from the Universidad Carlos III de Madrid, in 2001 and 2005, respectively. In 1999, he was a Communications Staff Member with the Spanish Antarctica Base. He visited the University of Leeds, U.K., in 2003; the Chalmers Technical University, Sweden, in 2004; and the Instituto de Telecomunicações, Portugal, from 2008 to 2010. He is currently an Associate Professor with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid. He has led several private and national Spanish projects and participated in various European and international projects. He holds one patent and has published over 80 journal articles/conference papers and nine book chapters. His research interests include advanced multicarrier systems for wireless radio, satellite, and visible light communications. He received the Master's Thesis Award and the Ph.D. Thesis Award from the Professional Association of Telecommunication Engineers of Spain, in 1998 and 2006, respectively. He was the IEEE Spanish Communications and Signal Processing Joint Chapter Chair, from 2015 to 2023.

RUI DINIS (Senior Member, IEEE) received the Ph.D. degree from the Instituto Superior Técnico (IST), Technical University of Lisbon, Portugal, in 2001, and the Habilitation degree in telecommunications from the Faculdade de Ciências e Tecnologia (FCT), Universidade Nova de Lisboa (UNL), in 2010. He was a Researcher with the Centro de Análise e Processamento de Sinal (CAPS), IST, from 1992 to 2005. From 2001 to 2008, he was a Professor with IST. In 2003, he was an Invited Professor with Carleton University, Ottawa, Canada. He conducted research with the Instituto de Sistemas e Robótica (ISR), from 2005 to 2008. Since 2009, he has been a Researcher with the Instituto de Telecomunicações (IT). He is currently an Associate Professor with FCT, Universidade Nova de Lisboa (UNL). He actively participates in several national and international research projects in the broadband wireless communications area. His research interests include transmission, estimation, and detection techniques. He serves as an Editor for IEEE Transactions on Wireless Communications, IEEE Transactions on Communications, IEEE Transactions on Vehicular Technology, IEEE Open Journal of Communications, and *Physical Communication* (Elsevier). He also served as the Guest Editor for *Physical Communication* (Elsevier) on the Special Issue on Broadband Single-Carrier Transmission Techniques. He is recognized as a VTS Distinguished Lecturer.