

## Article

# Federated Learning-Based Spectrum Occupancy Detection

Łukasz Kułacz <sup>1</sup> and Adrian Kliks <sup>1,2,\*</sup><sup>1</sup> Institute of Radiocommunications, Poznan University of Technology, 60-965 Poznan, Poland; lukasz.kulacz@put.poznan.pl<sup>2</sup> Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, 971 87 Lulea, Sweden

\* Correspondence: adrian.kliks@put.poznan.pl

**Abstract:** Dynamic access to the spectrum is essential for radiocommunication and its limited spectrum resources. The key element of dynamic spectrum access systems is most often effective spectrum occupancy detection. In many cases, machine learning algorithms improve this detection's effectiveness. Given the recent trend of using federated learning, we present a federated learning algorithm for distributed spectrum occupancy detection. This idea improves overall spectrum-detection effectiveness, simultaneously keeping a low amount of data that needs to be exchanged between sensors. The proposed solution achieves a higher accuracy score than separate and autonomous models used without federated learning. Additionally, the proposed solution shows some sort of resistance to faulty sensors encountered in the system. The results of the work presented in the article are based on actual signal samples collected in the laboratory. The proposed algorithm is effective (in terms of spectrum occupancy detection and amount of exchanged data), especially in the context of a set of sensors in which there are faulty sensors.

**Keywords:** federated learning; machine learning; spectrum occupancy detection

**Citation:** Kułacz, Ł.; Kliks, A.Federated Learning-Based Spectrum Occupancy Detection. *Sensors* **2023**, *23*, 6436. <https://doi.org/10.3390/s23146436>

Academic Editor: Dima Bykhovsky

Received: 16 June 2023

Revised: 13 July 2023

Accepted: 13 July 2023

Published: 16 July 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Recently, the popularity of radiocommunication system utilization has been growing. Many new and more demanding services are expected to be delivered wirelessly with guaranteed and typically high quality. Each smartphone, laptop and even smartwatch, as well as many IoT devices, cars, etc., require access to the wireless network, which leads to the very challenging problem of limited spectrum resources. For many reasons, the statically and exclusively allocated spectrum is only partially utilized. It is common that the spectrum resources are not used all the time or even are not used at all, as certain services are not delivered in the given location. In view of the problem of insufficient spectrum resources, dynamic spectrum access (DSA) is a very promising solution to meet the growing requirements for radiocommunication systems [1–3]. Contrary to the static-allocation approach, DSA can use the unoccupied spectrum for purposes other than those for which it was initially intended and reserved. Although conceptually simple, practical realization of the DSA is not a trivial task. By using the appropriate spectrum-sensing algorithms, unlicensed users (also known as secondary users, SUs) should have access to new spectral resources, which are detected as unused at the given time and location [1,4]. DSA may be realized in the wide-range form, but also in the small-scale variant [5]. At the same time, care should be taken to ensure that the primary user's (PU) transmission is provided with the best possible quality of service, whether through appropriate separation of resources between users, control of the transmission power of system users or even disabling the possibility of transmission for unlicensed users. The key aspect here is to gain or collect information about the presence of primary users' transmissions. Once the system has this information, it can take appropriate action to protect the PU's transmission. It is necessary to have a well-established understanding of the spectrum hole [6]. The straightforward

and most popular way of collecting information about spectrum occupancy is spectrum sensing [7,8]. Many solutions have been proposed in the rich literature in this respect, exploring various opportunities in the process of determining PU signal presence. One may mention such approaches as energy detection, feature detection and cyclostationary-based spectrum sensing, to name a few. However, single-node spectrum sensing cannot guarantee the required level of reliability of the spectrum, thus cooperative variants are widely considered, where collaborating nodes exchange information (raw measurements, rough decisions, final decisions, etc.) to improve sensing performance. In such an approach, both distributed and centralized (i.e., with the presence of a fusion center) approaches are possible [9,10]. On a big scale, the term crowd sensing is used [11], where multiple sensing nodes collaborate. However, one of the key problems related to cooperative spectrum sensing (CSS) is the huge amount of data to be exchanged between the cooperating nodes. It is also necessary to identify the spatial correlation between the cooperating nodes as it should be clear if the sensing information from a specific node is also valid for another node.

The paper is organized as follows. In the next subsection, we describe related work and then in Section 2 we revise the foundations of the FL and describe the overall FL-based spectrum-sensing scheme. In Section 3, we present the experimentation setup, discuss the features detected during the FL spectrum-sensing process and present the reference scenario. In Sections 4 and 5, we analyze the results achieved for the spectrum occupancy detection in one and over multiple channels. All results presented in the paper are outlined in the conclusion and discussed in Section 6.

#### *Related Work*

Furthermore, machine learning (ML) algorithms are also successfully used to improve the quality of detection, which supports the detection process by extracting important but not always noticeable details of the transmission [12]. An example of such details may be the periodicity of (or within) the transmission, different traffic volumes depending on the time of day or year or the characteristics of the transmission system used. One of the solutions for spectrum-sensing efficiency improvement is to benefit from various predictions in time, space and frequency domains, which can be characterized by, e.g., daily traffic demand trends. In such an approach, some ML techniques could be used to predict such trends, like recurrent neural networks (RNN) [13] or support vector regression (SVR) [14]. Various traffic patterns can be detected in different domains and they appear very often simultaneously using advanced AI/ML solutions [15–17]. However, the application of ML towards better spectrum sensing has some limitations; for example, the usage of supervised learning entails the need for labeling reference signals and this may be problematic. In addition, as we mentioned above, the knowledge from a single sensor can often be distorted due to the sensor's specific location or its unreliability. Recently, distributed learning (in its various variants) has gained attention in wireless communications domains. Thus, cooperative spectrum-sensing algorithms have been considered an improvement, yet they lead to increased complexity in ML algorithms [18].

One of the promising solutions to the above problems is applying the federated learning (FL) concept in the spectrum-sensing procedure. Here, regardless of whether the FL variant is used (i.e., centralized or distributed), the nodes share the artificial intelligence (AI) model instead of raw data to detect the presence of PU better. Such an approach has many advantages—the overall control traffic is reduced while benefiting from the cooperation between the nodes [19,20]. In particular, there are multiple works related to FL in radiocommunication systems; they, however, consider aspects other than spectrum occupancy detection. Such FL algorithm could be utilized, e.g., in resource allocation process [21]. The summary of FL algorithm application for wireless networks can be found in [22]. In terms of the clue of this paper, there are not that many publications that utilize FL for spectrum sensing. In [23] authors were focused on waveform detection in the citizens broadband radio service (CBRS) system and they showed that FL can improve detection efficiency. Then in [24], the authors presented the FL framework which could be used in

cooperative spectrum sensing (CSS). In our work, we focus more on the FL application in the real environment. In the as-found signal received by the sensors, we propose an algorithm for spectrum occupancy detection, which achieves comparable accuracy to classical approaches and is resistant to the presence of faulty sensors. In [25], a very interesting, exhaustive and complete work describing long-term research on the FL framework can be found. The authors performed the whole FL process with many adjustments, however, the authors worked on an air quality dataset, which cannot be directly compared to the spectrum-sensing challenge. A crucial part of FL and ML algorithms is a clustering algorithm, especially in the context of distributed spectrum sensing. In another interesting work [26] FL algorithm is proposed mainly for the classification of road objects in the vehicular to everything (V2X) network.

Motivated by the above observation, we wanted to address the important problem of reliable distributed spectrum sensing realized by resource-limited devices (sensors). The idea was to propose a solution that, to some extent, benefits from the experience of the neighboring nodes and minimizes the amount of shared data between them. The advantage of FL in that context lies in the fact that sensing nodes may in some sense share the gained knowledge with other nodes about the environment in the form of the model coefficients.

In this paper we propose the utilization of centralized FL for spectrum sensing in the scenario with three sensors. The proposed solution assumes machine learning model coefficients exchange between these sensors—in a central server and the creation of a common model that is further used by each sensor. A couple of machine learning models were examined and a comparison with the reference (without FL) scenario was done. Additionally, a case with a faulty sensor was verified. The proposed solution also focuses on the main bottleneck of spectrum sensing in commercial networks—rare access to verified information—which could be used as training data. This paper presents the results of the hardware experiment where FL has been implemented in the laboratory, where universal software radio peripheral (USRP) boards have acted as sensing nodes.

## 2. Federated Learning Spectrum Sensing (FLSS)

Reliable spectrum sensing, especially in a harsh environment (like operation in very low signal-to-interference-plus-noise, SINR, regime), usually requires a very long sensing time or the cooperation of numerous nodes [27–29]. Consequently, there is an immediate problem of either a long sensing time or a vast amount of data that must be transferred between the sensors. In both cases, the time available for the actual transmission of user data is reduced and so is the system's overall efficiency. It should be remembered that sensors should also be very simple devices, which is an additional limitation regarding the number of data transferred between them. Another critical challenge is the transparency of the transmitted data, i.e., the transmission of the collected measurement data by sensors can be relatively easily intercepted and analyzed and, worse, replaced.

One of the promising ideas for solving these problems that is gaining popularity recently and is widely considered in many aspects and applications is federated learning (FL) [30]. By assumption, instead of sharing raw data, the local nodes share between themselves only the parameters specifying the ML models (such as coefficients of the neural networks). In the classical approach, the central node trains the global model and shares it with the underlying nodes, which receive the global model and adjust it locally based on its local observations. Once the ML models residing in the local nodes are trained (and thus adjusted to the local scenario), they are sent back to the central node to update the global model and spread the updates within the whole cooperating network. The whole process is then repeated, allowing continuous learning and adaptation of the ML models (both global and local). The above procedure may be summarized as follows. First, one has to specify the set of cooperating nodes and define the learning parameters such as ML model details, learning rate, number of communication rounds, the local batch size for updating local models or an algorithm used for combining old and new models. Let us assume that there are  $K$  cooperating nodes indexed by  $k = 1, \dots, K$  and each node  $k$

updates the weights of the local ML model, denoted as  $\mathbf{x}_k$ . Then, the ultimate optimization function will be denoted as  $f(\mathbf{x}_1, \dots, \mathbf{x}_K)$  and may be calculated as the average of local objective functions  $f_i(\mathbf{x}_i)$ :

$$f(\mathbf{x}_1, \dots, \mathbf{x}_K) = \frac{1}{K} \sum_{i=1}^K f_i(\mathbf{x}_i). \quad (1)$$

This simple averaging approach may be modified to reflect the typical situation where local nodes may operate over different sets of samples (of cardinalities denoted as  $n_k$ ).

Inspired by [19,25], in our work, we concentrate on applying the FL in a cooperative sensing network, where a set of nodes exchange model coefficients to sense the wireless spectrum better and more accurately. Hereafter, we refer to FL spectrum sensing (FLSS). As stated before, FLSS assumes the operation of individual spectrum sensors in a specific area of the network. They collect data by observing the spectrum in their location and use these data to train their ML model. Then, the SS model coefficients of individual sensors are exchanged, as opposed to the samples collected in traditional cooperative spectrum sensing and an aggregated model is created. To this end, various merging approaches may be considered, such as the simplest Federated Averaging (FedAvg) [31] scenario—created by averaging all coefficients received from the sensors—or Federated Stochastic Gradient Descent (FedSDG), which is based on gradient computation over a random subset of the whole dataset and uses them to make one step of the gradient descent. Once created, the coefficients of the collective model are transferred to all sensors. In the simplest scenario, the sensors use a collective model and may update it based on the newly collected spectrum samples. The procedure described above is repeated many times. As a consequence, individual sensors have access to a common SS model, which is at least indirectly based on many more samples than those collected by a single sensor. One of the benefits of using FLSS is the ability to train a single sensor model much faster in the event of its replacement or simply adding a new one. Therefore, the speed of adaptation of such a system should be its advantage—especially in the context of the aforementioned general challenge related to spectrum occupancy detection (rare access to verified information—training data). Similarly, the ability to detect a faulty sensor (i.e., by detecting unexpected reports from it) should be a valuable safeguard for the system leading to higher sensing reliability. Obviously, FLSS used in this way cannot offer higher detection efficiency than cooperative spectrum occupancy detection algorithms due to limited access to training data (only indirectly). However, at the expense of slightly lower detection efficiency, it is possible to significantly reduce the amount of sent control data and improve system security.

In this work, we aim to verify the efficiency of hardware-implemented centralized FLSS, where the sensing nodes observe the environment in their vicinity, update the local model and share the model coefficients with the central node. The central node applies the FedAvg algorithm and distributes the models among the cooperative nodes. Once the local nodes receive the new global model, they average it with its local copy and continue sensing with the updated one. As the ML technique used in the FL procedure, mainly the neural network has been selected. The considered solution is presented graphically in Figure 1. Additionally, we compare the results with the non-FL approach, where nodes apply locally some of the ML techniques (mainly support vector machine, K-nearest neighbors and random forest), but do not share the models' coefficients with the surrounding nodes. In the following, we present the setup of the conducted experiment.

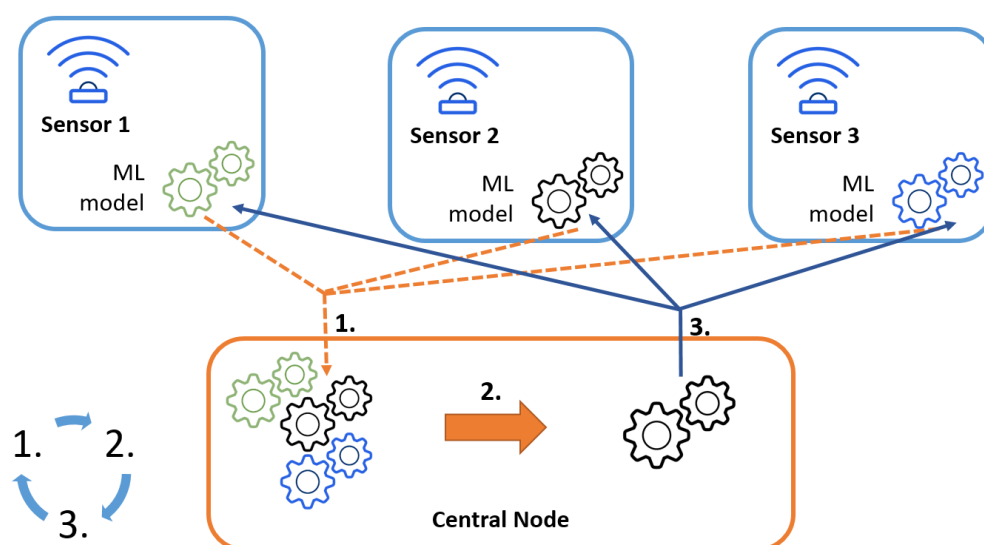


Figure 1. Considered system solution.

### 3. Experimentation Setup

Let us now discuss the experimentation setup used in the laboratory to verify the practical performance of the FL-based spectrum sensing. First, we present the overall hardware setup and present the process of data collection. Next, we discuss the implemented ML tool for FLSS, as well as the whole process of data preparation for reliable reasoning. After that, we present the reference scenario.

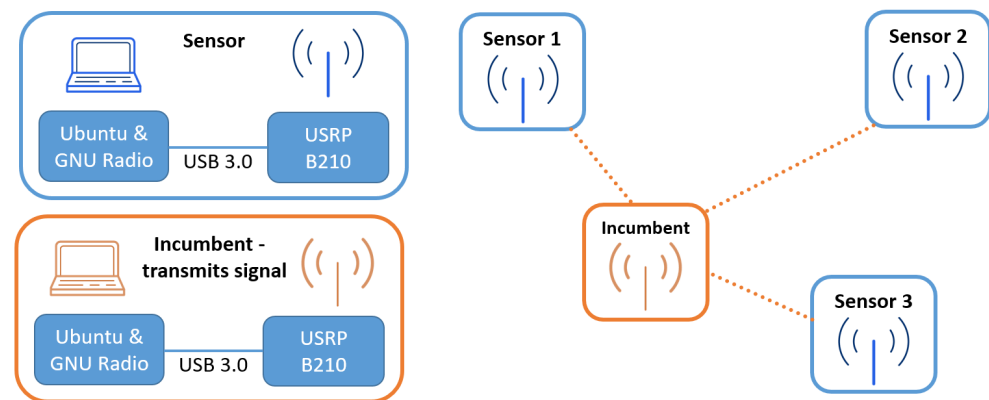
#### 3.1. Measurement Setup

As we concentrated on the evaluation of the FLSS performance in the real environment, it was necessary to gather a rich set of real samples and work on them. To achieve this, the true measurement data were collected.

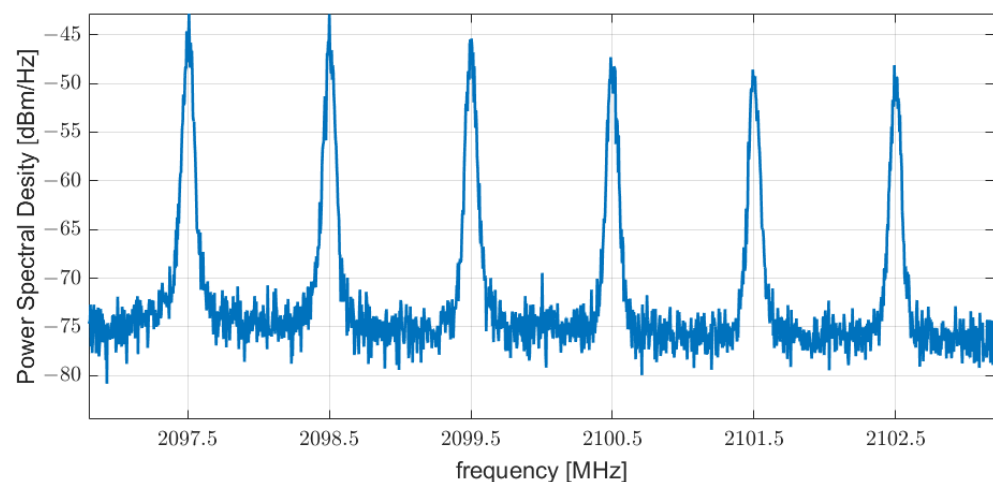
As illustrated in Figure 2, three measurement series were made, each of which was carried out for a different position of the signal receiver (spectrum sensor). Each of the analyzed receiver positions varied between 2 and 5 m from the transmitter, at a similar height (difference less than 15 cm) and with direct visibility between the transmitter and receiver antennas. During all three measurements, the position of the signal transmitter remained constant. The signal transmitter and receiver were implemented with the help of a laptop with Ubuntu and GNU Radio software 3.8 and USRP connected to the computer via USB 3.0 interface. The signal generated by the transmitter was in the form of six adjacent Gaussian Minimum Shift Keying (GMSK) signals with a bandwidth of 350 kHz each. Each of the signals was placed in the frequency domain in a separate channel with a width of 1 MHz. The center frequency of the transmitter is 2100 MHz; hence, the next carriers of the GMSK signal are 2097.5, 2098.5, 2099.5, 2100.5, 2101.5 and 2102.5 MHz, respectively. The spectrum of the transmitted GMSK comb signal is shown in Figure 3.

For each sensor location, in-phase and quadrature samples of the received signal were collected—first with the transmitter turned off (3 times 10M samples) and then with the transmitter turned on (31 times 1M samples), each time changing the gain setting of the transmitter amplifier. The transmit amplifier gain settings used were from 35 dB to 80 dB. The receiver was set to a center frequency of 2100 MHz, a bandwidth of 40 MHz. The receiver was prepared to record samples from 40 adjacent channels, 1 MHz each. However, taking into account the non-linear nature of the filters and amplifiers in the physical receiver, additional processing had to be performed. First, the local oscillator frequency shift was set to 10 MHz and second, the extreme channels (12 MHz on each side) were omitted in the analysis process. The local oscillator frequency shift was intended to minimize the constant power observed at the carrier frequency of the receiver. The extreme channels were omitted due to the imperfections of the physical receiver.





**Figure 2.** Measurement setup.



**Figure 3.** The spectrum of transmitted GMSK comb signal.

### 3.2. Data Analysis and Preparation

Once the raw data are collected (i.e., the samples representing the observed environment), it has to be decided how they will be processed for reliable decision-making. In the case of FLSS, the ultimate goal is to detect the presence or absence of the PU signal in the observed spectrum. As in classic spectrum sensing, the simplest solution is calculating the observed signal power. However, the experiments have shown that relying only on the received signal power in the ML algorithm would not be sufficient to obtain better results than the classical energy-detection scheme. To this end, the collected data were subjected to the following analysis. For each analyzed channel, a portion of samples equal to 10k was processed, based on which three values were computed: average power, kurtosis of the autocorrelation function and skewness of the autocorrelation function. In this way, it was noticed that the data obtained for these two cases (i.e., presence and absence of PU transmission) are clearly distinguishable for all three features; the signal strength is obviously higher when there is an active PU transmission, but kurtosis and skewness are lower. This observation confirms the correctness of selecting these three properties of the observed signal for use in ML algorithms. Obviously, the conclusion drawn above is valid and the distinction between the presence and absence of PU signal is easy when one operates in the high signal-to-noise regime (i.e., the transmitted signal is much stronger than the noise). However, for a low signal-to-noise ratio case, this task is not trivial and it is expected that also, in that case, the proposed FLSS should work efficiently. Let us also note that in data processing, the collected data for all three sensors were also normalized depending on the requirements of the ML algorithm; scaling to a specific range was the most common approach. All data used for ML were published in this public repository [32].

### 3.3. Reference Scenario

To evaluate the performance of the proposed solution, we compared it with traditional spectrum sensing and with non-FL approaches. In the former case, the energy-detection algorithm, which consists of several simple steps, was adopted as the reference scenario. In the latter case, we compare the FLSS with the non-FL logistic regression model and neural networks (with two hidden layers containing four neurons each). In the energy-detection case, the estimation of the noise power  $\sigma^2$  is required for the collected data (for each sensor separately). Second, for the given false alarm probability  $p_{fa}$  (in this paper,  $p_{fa} = 0.01$ ), the detection threshold  $\Gamma$  was calculated according to the following formula [8]:

$$\Gamma = \sigma * \left( Q(p_{fa}) \sqrt{\frac{2}{N}} + 1 \right), \quad (2)$$

where  $Q$  is the tail distribution function of the standard normal distribution and  $N$  is the number of samples ( $N = 10,000$  in this paper). The last step is to decide about the presence  $\Psi$  of the PU signal by comparing the received power with the calculated detection threshold:

$$\Psi = \begin{cases} 1 & \text{if } P_{rx} \geq \Gamma \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where  $P_{rx}$  is power of the received signal. The correctness of the collected results was determined by the model's accuracy. Mainly, it was verified by the number of data entries that were predicted (by the model) correctly in terms of its label. As a label in this case, we used information about the presence or lack of the transmitted signal. Data were prepared using the *StratifiedKFold* method and thus divided into five parts. This results in an even division in each of the parts between the data with and without the presence of the transmitted signal. In this case, the training data were used for noise estimation and the verification data for the actual signal detection. For the three sensors used in the experiment, we obtained an average accuracy of 0.7846, 0.8621 and 0.6564; the average accuracy among all sensors was 0.7677. A difference between accuracy score achieved for particular sensors is mainly due to the different distances between the transmitter and receiver (i.e., the source of the signal and the sensor). It results in different SNRs and in more "samples" with higher received power by some sensors (which is easier to classify as a transmitted signal—not noise).

Similarly, non-FL solutions were analyzed in terms of their accuracy. The accuracy scores achieved by all sensors are presented in detail in Figure 4. The verification of the logistic regression model and the neural network (carried out in exactly the same way as for classic energy detection) showed average accuracy of 0.8, 0.9968 and 0.7029 (with an average between sensors equal to 0.8332) for logistic regression and 0.799, 0.9991 and 0.7178 (with mean between sensors equal to 0.8386) for the neural network. This shows that using additional information, such as kurtosis and skewness of the autocorrelation function, allows for an average accuracy improvement over the energy-detection scheme. In particular, it improved by 6.55 percentage points when using the logistic regression algorithm and 7.09 percentage points when using the neural network.

Once the accuracy of the reference scenarios has been checked, let us now evaluate the performance of the proposed FLSS scheme. Thus, in the next section, we will focus on the analysis of a single channel (of the six analyzed) to facilitate the presentation of concepts and algorithms. The following sections will describe how the cumulative algorithm works with all six channels.

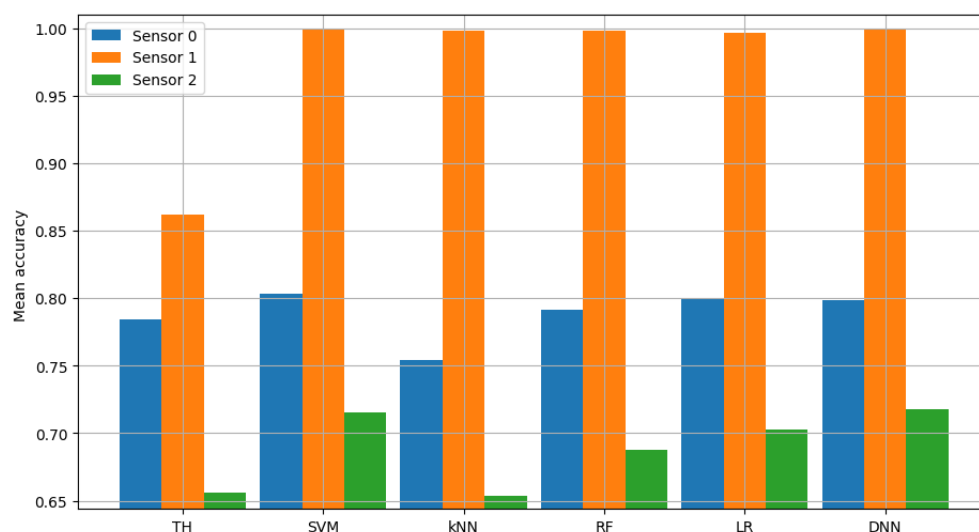


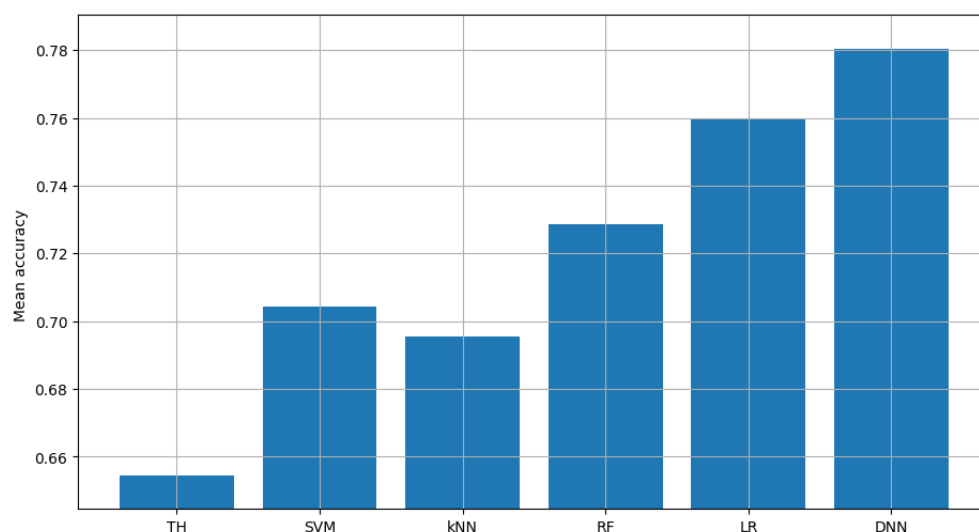
Figure 4. Validation of single channel model.

#### 4. Discussion—Single Channel Spectrum Sensing

The federated learning algorithm proposed in this paper assumes the independent operation of all three sensors. I.e., although they share model coefficients between themselves, they collect their own data and learn their own machine learning model, and make independent decisions solely. Immediately before deciding to detect the spectrum, each sensor sends its model coefficient to the central node. In the central node, a common model is created, which in the following step is distributed among all sensors. Next, the sensors make a decision based on a common machine-learning model. The joint model is created in the central node as a weighted average of the coefficients of each model obtained from the sensors. The weight for each sensor is based on its last accuracy. Values less than 0.4 are ignored. This is to reject the model's coefficients, which in the previous trial gave poor results. The experiment was carried out in the following way. First, we treat all data from all sensors separately. The datasets were shuffled; next, the first 100 datasets were sent to the sensors for initialization of the local ML models and for noise estimation required in the reference model. The simulator then runs iteratively; during each iteration, the sensors processed 30 sets of data of  $N = 10,000$  samples. Each sensor fitted the model to the received data and sent the trained model to the central node, creating a common model; once computed, the global model was distributed. Once the training phase was finished, the testing phase was initiated. Thus, each sensor receives a set of test data (the remaining data of a specific sensor not used during the learning process) and verifies the accuracy of its own model.

As a result, each sensor obtains slightly worse results than if the model coefficients were not exchanged. This is very logical behavior because the data from neighboring sensors are different and a sensor tailored to its own data would achieve better results than when it also has to consider data from other sensors. However, in the other part of the verification, the models were checked by all the data that were used in the simulation (regardless of whether they come from measurements from a given sensor or another) as shown in Figure 5. In this case, much better results were obtained than with single sensor data. This indicates that a created common model is better suited to the general data emerging in this environment. It allows, for example, for easier replacement of sensors and faster adaptation in the event of failure of one of them. This is particularly important in the context of the main problem of spectrum occupancy detection, i.e., access to reliable information about the presence of a signal (later used as a data label in the machine learning process).





**Figure 5.** Accuracy score of single channel model (federated learning).

It is worth taking a closer look at the results obtained when one of the sensors deliberately reports wrong decisions in the process of FLSS. In this experiment, we assumed these would be arbitrary decisions about spectrum occupancy. Indeed, the data used to train individual models are slightly different for different sensors. This may be due to the specific location of the sensor or even the use of another device. When exchanging model coefficients between sensors, it is necessary to compensate for these differences in the inputs of the machine learning algorithms. Otherwise, the averaging (or other mathematical operation) of the coefficients of models with different input data characteristics will diminish the models' effectiveness.

## 5. Discussion—Multi-Channel Spectrum Sensing

In the previous section, we concentrated on an evaluation of the results of the single-channel FLSS; we would like now to extend it to the multi-channel case. Thus, in the next simulation stage, all steps (from data preparation through model training, validation and federated learning simulations as described) were repeated for six adjacent channels of the GMSK-comb signal presented in Figure 3. Each channel was treated independently, i.e., each channel has its own FLSS model and its own data and in the process of federated learning, the coefficients of independent models were exchanged separately for each channel. In this way, six separate common models were created. A fundamental assumption in the considered system is the specificity of the transmission, which assumes the use of all six channels at once or none of them. In the last phase, when the decision on the spectrum occupancy is made, the decision criterion of at least four out of six were used, i.e., at least four channels must be considered occupied to consider that the entire analyzed band (six adjacent channels) is occupied. Referring to classical cooperative spectrum sensing, such an approach is known as majority voting [33].

The result of model validation for the reference scenario (presented in Figures 6–8) was the average accuracy score (average between trials and between analyzed channels) for all involved sensors: 0.7891, 0.8595 and 0.7364 (mean between sensors 0.795). Three indicated figures show the mean accuracy score of each sensor in the scenario, where the detected signal is transmitted in all six channels at once. Bars described as "TH" represent the energy-detection algorithm; those denoted "SVM", "kNN" and "RF" show the accuracy score of support vector machines (SVM), K-nearest neighbors (kNN) and random forest (RF), respectively. The last two sets of bars present results for logistic regression (LR) and dense neural network (DNN). The reference scenario using logistic regressions in the machine learning process obtained average accuracy for the three involved sensors at the level of 0.8086, 0.9968 and 0.7715 (with an average between sensors equal to 0.859). In the

reference scenario using a neural network in the machine learning process, the following accuracy was achieved: 0.7912, 0.9661 and 0.7647 (with an average between sensors equal to 0.8406).

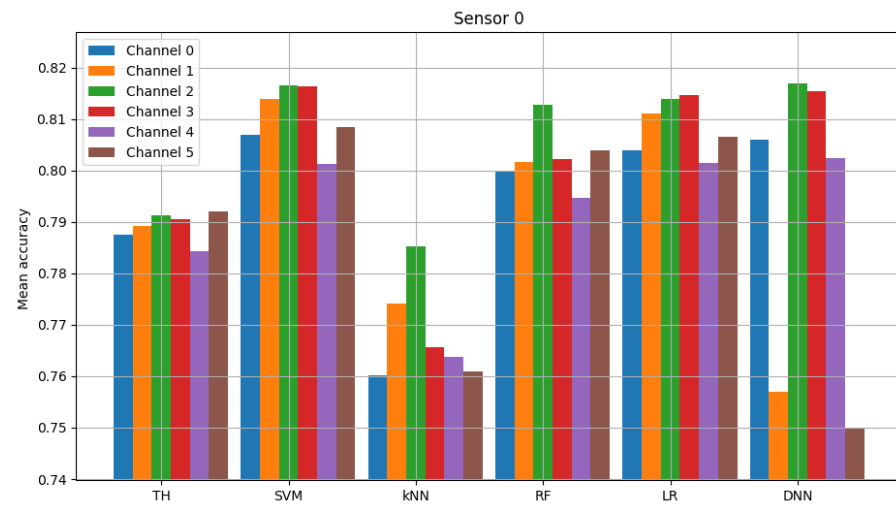


Figure 6. Validation of “sensor 0” model (multiple channels).

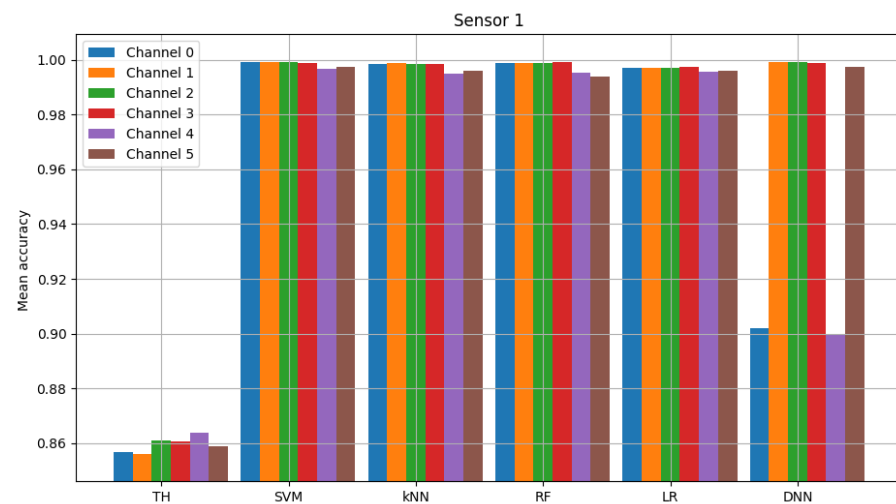


Figure 7. Validation of “sensor 1” model (multiple channels).

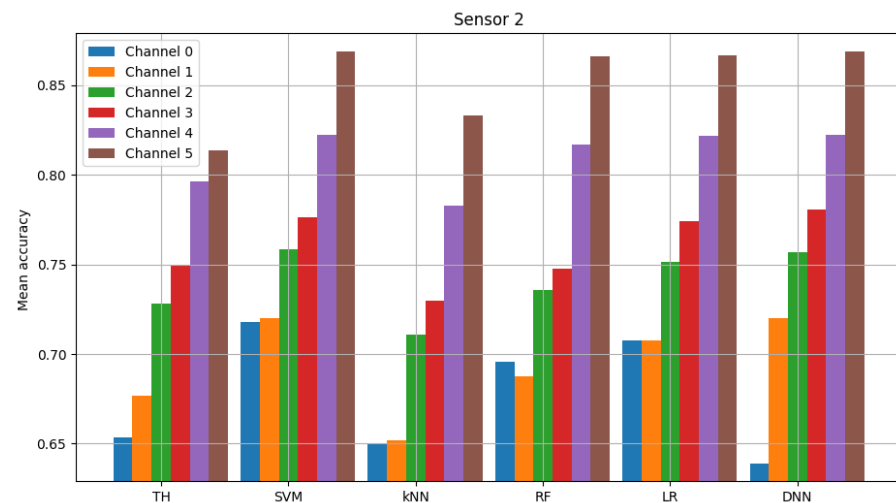


Figure 8. Validation of “sensor 2” model (multiple channels).

For the federated learning algorithm, the following average accuracy was obtained (average between trials) assuming the selected decision criterion (i.e., minimum four out of six occupied channels) and logistic regression for common model 0.7730. The average accuracy for the neural network is 0.8945. In the reference scenario where the decision is made only based on energy detection, the value is 0.8708. This set of results is presented in Figures 9–11. These figures show the mean accuracy score of each sensor in the scenario, where the detected signal is transmitted in all six channels at once. Models used in this simulation are the same as in previously described figures (and are denoted in the same way, as “TH”, “SVM”, “kNN”, “RF”, “LR” and “DNN”). Here, a group of six bars for each model does not present the accuracy score in each individual channel, but alternatively we take into account the knowledge that the signal was transmitted in each channel simultaneously. For this reason, we added majority voting ( $k$  of  $n$ , in our case we considered all variants, i.e., from 1 to 6 ending with from 6 to 6) and the accuracy score achieved for each voting method is presented in the form of bars.

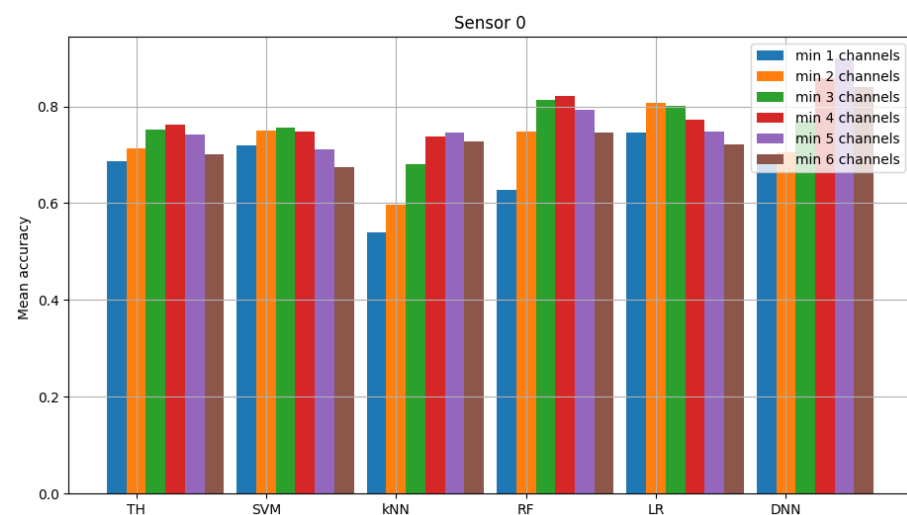


Figure 9. Accuracy score of sensor 0 model (multiple channels and federated learning).

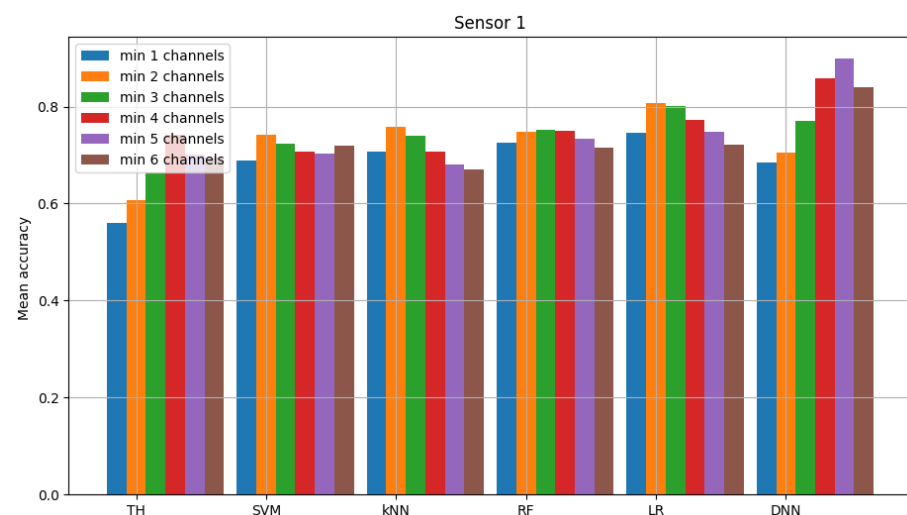
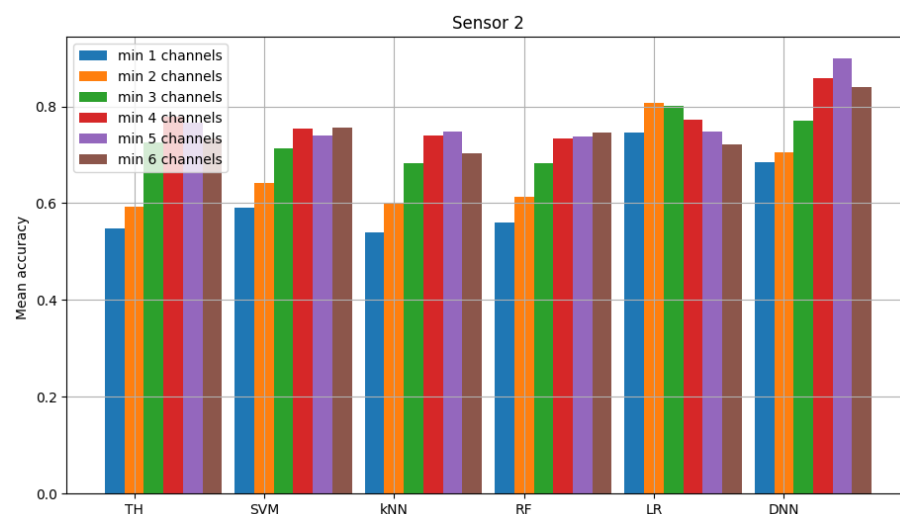


Figure 10. Accuracy score of sensor 1 model (multiple channels and federated learning).



**Figure 11.** Accuracy score of sensor 2 model (multiple channels and federated learning).

Assuming that one sensor is generating random decisions during the simulation, we will observe two important effects. First, the obtained average accuracy equals 0.7268. This was achieved while averaging results over all experimentation trials, assuming the selected majority decision-making criterion and logistic regression used for the common model. Analogously, the average accuracy for the neural network case was at the level of 0.8395. Second, for the same simulation scenario, where one sensor is faulty but model coefficients are not exchanged, we observe the average accuracy for logistic regression of 0.2787, 0.7504 and 0.7704 per each sensor with an average between sensors of 0.5998. The neural network's average accuracy was equal to 0.3095, 0.6998 and 0.8773 (average between sensors 0.6289). As a result, if logistic regression is utilized with federated learning, the average accuracy score can be improved by 12.7 percentage points in cases when one faulty sensor is observed. If a neural network is utilized, then the average accuracy score can be improved by around 21 percentage points.

## 6. Conclusions

The algorithms presented in the work show that there is a potential for using federated learning in spectrum occupancy-detection systems. In specific scenarios, they provide increased reliability of detection and, more importantly, ensure greater efficiency of decisions made compared to classic algorithms (such as energy detection) and comparable efficiency compared to algorithms using machine learning (compared to a model trained on a larger number of samples—such as in cooperative spectrum occupancy detection). It would be worth considering checking other model and result validation metrics, such as the F1 metric. Using different ML techniques could be beneficial in different scenarios. In the considered scenario, if logistic regression is utilized with federated learning, the average accuracy score was slightly improved in cases when one faulty sensor is observed. The amount of data (coefficients) that needs to be exchanged remains at a low level, as this ML model uses only a few coefficients. When a neural network is utilized, then the average accuracy score can be improved more than in the logistic regression case, however with the cost of around ten times more coefficients needed in the model coefficient exchange procedure. This leads us to the point where we can choose the most suitable ML model for the considered scenario and network. In future work, we plan to make a comparison between presented results and the distributed system, where the model's coefficients are exchanged only with the central node but between neighboring sensors. Such a distributed approach should be more efficient in terms of the data amount that needs to be exchanged and should have better local decisions—especially in the network places over huge areas. However, we suppose the algorithm of model creation will be the most critical. Due to this, the presence of faulty sensors will be more challenging.

**Author Contributions:** Conceptualization, Ł.K. and A.K.; methodology, Ł.K. and A.K.; software, Ł.K.; validation, Ł.K. and A.K.; formal analysis, Ł.K. and A.K.; investigation, Ł.K. and A.K.; writing—original draft preparation, review and editing, Ł.K. and A.K.; visualization, Ł.K.; supervision, A.K.; project administration, A.K.; funding acquisition, A.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** The presented work has been funded by the National Science Centre in Poland within the project (no. 2021/41/N/ST7/01298) of the PRELUDIUM program.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are openly available in Zenodo at 10.5281/zenodo.8046418.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
CSS	Cooperative Spectrum Sensing
DSA	Dynamic Spectrum Access
FL	Federated Learning
FLSS	Federated Learning Spectrum Sensing
ML	Machine Learning
PU	Primary User
SINR	Signal-to-Noise and Interference Ratio
SS	Spectrum Sensing
SU	Secondary User
USRP	Universal Software Radio Peripheral

## References

1. Arjoune, Y.; Kaabouch, N. A Comprehensive Survey on Spectrum Sensing in Cognitive Radio Networks: Recent Advances, New Challenges and Future Research Directions. *Sensors* **2019**, *19*, 126. [[CrossRef](#)] [[PubMed](#)]
2. Liang, Y.C.; Chen, K.C.; Li, G.Y.; Mahonen, P. Cognitive radio networking and communications: An overview. *IEEE Trans. Veh. Technol.* **2011**, *60*, 3386–3407. [[CrossRef](#)]
3. Zhao, Q.; Sadler, B.M. A Survey of Dynamic Spectrum Access. *IEEE Signal Process. Mag.* **2007**, *24*, 79–89. [[CrossRef](#)]
4. Molina-Tenorio, Y.; Prieto-Guerrero, A.; Aguilar-Gonzalez, R.; Lopez-Benitez, M. Cooperative Multiband Spectrum Sensing Using Radio Environment Maps and Neural Networks. *Sensors* **2023**, *23*, 5209. [[CrossRef](#)] [[PubMed](#)]
5. Kryszkiewicz, P.; Kliks, A.; Bogucka, H. Small-Scale Spectrum Aggregation and Sharing. *IEEE J. Sel. Areas Commun.* **2016**, *34*, 2630–2641. [[CrossRef](#)]
6. Tandra, R.; Mishra, S.M.; Sahai, A. What is a Spectrum Hole and What Does it Take to Recognize One? *Proc. IEEE* **2009**, *97*, 824–848. [[CrossRef](#)]
7. Haykin, S. Cognitive radio: Brain-empowered wireless communications. *IEEE J. Sel. Areas Commun.* **2005**, *23*, 201–220. [[CrossRef](#)]
8. Yucek, T.; Arslan, H. A survey of spectrum sensing algorithms for cognitive radio applications. *IEEE Commun. Surv. Tutorials* **2009**, *11*, 116–130. [[CrossRef](#)]
9. Cichoń, K.; Kliks, A.; Bogucka, H. Energy-Efficient Cooperative Spectrum Sensing: A Survey. *IEEE Commun. Surv. Tutorials* **2016**, *18*, 1861–1886. [[CrossRef](#)]
10. Chen, X.; Chen, H.H.; Meng, W. Cooperative Communications for Cognitive Radio Networks—From Theory to Applications. *IEEE Commun. Surv. Tutorials* **2014**, *16*, 1180–1192. [[CrossRef](#)]
11. Dasari, V.S.; Kantarci, B.; Pouryazdan, M.; Foschini, L.; Girolami, M. Game Theory in Mobile CrowdSensing: A Comprehensive Survey. *Sensors* **2020**, *20*, 2055. [[CrossRef](#)] [[PubMed](#)]
12. Gao, J.; Yi, X.; Zhong, C.; Chen, X.; Zhang, Z. Deep Learning for Spectrum Sensing. *IEEE Wirel. Commun. Lett.* **2019**, *8*, 1727–1730. [[CrossRef](#)]
13. Roy, D.; Mukherjee, T.; Chatterjee, M.; Pasiliao, E. Primary User Activity Prediction in DSA Networks using Recurrent Structures. In Proceedings of the 2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), Newark, NJ, USA, 11–14 November 2019; pp. 1–10. [[CrossRef](#)]

14. Zhang, Z.; Zhang, K.; Gao, F.; Zhang, S. Spectrum prediction and channel selection for sensing-based spectrum sharing scheme using online learning techniques. In Proceedings of the 2015 IEEE 26th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Hong Kong, China, 30 August–2 September 2015; pp. 355–359. [\[CrossRef\]](#)
15. Camelo, M.; Schepper, T.D.; Soto, P.; Marquez-Barja, J.; Famaey, J.; Latré, S. Detection of traffic patterns in the radio spectrum for cognitive wireless network management. In Proceedings of the ICC 2020—2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020; pp. 1–6. [\[CrossRef\]](#)
16. Yu, L.; Wang, Q.; Guo, Y.; Li, P. Spectrum availability prediction in cognitive aerospace communications: A deep learning perspective. In Proceedings of the 2017 Cognitive Communications for Aerospace Applications Workshop (CCAA), Cleveland, OH, USA, 27–28 June 2017; pp. 1–4. [\[CrossRef\]](#)
17. Wasilewska, M.; Bogucka, H. Space-Time-Frequency Machine Learning for Improved 4G/5G Energy Detection. *Int. J. Electron. Telecommun.* **2020**, *66*, 217–223. [\[CrossRef\]](#)
18. Chen, S.; Shen, B.; Wang, X.; Yoo, S.J. A Strong Machine Learning Classifier and Decision Stumps Based Hybrid AdaBoost Classification Algorithm for Cognitive Radios. *Sensors* **2019**, *19*, 5077. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Wasilewska, M.; Bogucka, H.; Kliks, A. Federated Learning for 5G Radio Spectrum Sensing. *Sensors* **2022**, *22*, 198. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Jiang, J.C.; Kantarci, B.; Oktug, S.; Soyata, T. Federated Learning in Smart City Sensing: Challenges and Opportunities. *Sensors* **2020**, *20*, 6230. [\[CrossRef\]](#)
21. Xu, J.; Wang, H. Client Selection and Bandwidth Allocation in Wireless Federated Learning Networks: A Long-Term Perspective. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 1188–1200. [\[CrossRef\]](#)
22. Bouzinis, P.S.; Diamantoulakis, P.D.; Karagiannidis, G.K. Wireless Federated Learning (WFL) for 6G Networks Part I: Research Challenges and Future Trends. *IEEE Commun. Lett.* **2022**, *26*, 3–7. [\[CrossRef\]](#)
23. Troglia, M.; Melcher, J.; Zheng, Y.; Anthony, D.; Yang, A.; Yang, T. FaIR: Federated Incumbent Detection in CBRS Band. In Proceedings of the 2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), Newark, NJ, USA, 11–14 November 2019; pp. 1–6. [\[CrossRef\]](#)
24. Chen, Z.; Xu, Y.Q.; Wang, H.; Guo, D. Federated Learning-Based Cooperative Spectrum Sensing in Cognitive Radio. *IEEE Commun. Lett.* **2022**, *26*, 330–334. [\[CrossRef\]](#)
25. Gao, Y.; Liu, L.; Zheng, X.; Zhang, C.; Ma, H. Federated Sensing: Edge-Cloud Elastic Collaborative Learning for Intelligent Sensing. *IEEE Internet Things J.* **2021**, *8*, 11100–11111. [\[CrossRef\]](#)
26. Barbieri, L.; Savazzi, S.; Brambilla, M.; Nicoli, M. Decentralized federated learning for extended sensing in 6G connected vehicles. *Veh. Commun.* **2022**, *33*, 100396. [\[CrossRef\]](#)
27. Tandra, R.; Sahai, A. SNR Walls for Signal Detection. *IEEE J. Sel. Top. Signal Proc.* **2008**, *2*, 4–17. [\[CrossRef\]](#)
28. Zeng, J.; Su, X. On SNR wall phenomenon under cooperative energy detection in spectrum sensing. In Proceedings of the 2015 10th International Conference on Communications and Networking in China (ChinaCom), Shanghai, China, 15–17 August 2015; pp. 53–60. [\[CrossRef\]](#)
29. Gohain, P.B.; Chaudhari, S.; Koivunen, V. Cooperative Energy Detection With Heterogeneous Sensors Under Noise Uncertainty: SNR Wall and Use of Evidence Theory. *IEEE Trans. Cogn. Commun. Netw.* **2018**, *4*, 473–485. [\[CrossRef\]](#)
30. Nguyen, D.C.; Ding, M.; Pathirana, P.N.; Seneviratne, A.; Li, J.; Vincent Poor, H. Federated Learning for Internet of Things: A Comprehensive Survey. *IEEE Commun. Surv. Tutorials* **2021**, *23*, 1622–1658. [\[CrossRef\]](#)
31. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A.Y. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Artificial Intelligence and Statistics, Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*; Singh, A., Zhu, J., Eds.; PMLR: Cambridge, MA, USA, 2017; Volume 54, pp. 1273–1282.
32. Kułacz, Ł. GMSK Comb Spectrum Sensing for Machine Learning. 2023. Available online: <https://zenodo.org/record/8046418> (accessed on 16 June 2023).
33. Xie, S.; Shen, L.; Liu, J. Optimal threshold of energy detection for spectrum sensing in cognitive radio. In Proceedings of the 2009 International Conference on Wireless Communications & Signal Processing, Perugia, Italy, 21–24 June 2009; pp. 1–5. [\[CrossRef\]](#)

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.