Wireless Networks (2022) 28:3503–3524
https://doi.org/10.1007/s11276-022-03053-2

ORIGINAL PAPER

Machine learning-based approaches for user association and access point selection in heterogeneous fixed wireless networks

Zayan El Khaled[1] · Hamid Mcheick[1] · Wessam Ajib[2]

Accepted: 31 May 2022 / Published online: 23 July 2022

Abstract

Fixed Wireless Networks (FWNs) provide an alternative means for internet connectivity in rural and harsh propagation environments. Intensive technical expertise is therefore crucial for the planning and the optimization of such wireless links, as multiple difficult to estimate parameters are involved. It should be noted that a connection to a convenient Access Point (AP) is necessary to the establishment of efficient communication services in FWN. In this paper, we propose two algorithms that predict the success of a user association to FWN via the analysis of user feedback and subscription status, in combination with radio frequency parameters. These approaches are based on supervised machine learning where data is collected from a wide Canadian FWN. The first algorithm is based on the Nearest Neighbor concept, for which a new distance function is proposed. In the second algorithm, named Deep Nearest Neighbor, we extract the distance function with an artificial neural network. The accuracy of each of these algorithms is 86%. Additionally, we develop an AP selection scheme based on deep imitation learning to predict the success of a user-AP association. This model has a better accuracy of 94% since it combines a wide variety of parameters, use cases and conditions.

Keywords User association · Access point selection · Fixed wireless networks · Nearest neighbor · Artificial neural networks · Deep imitation learning

1 Introduction and context

Fixed Wireless Networks (FWNs) can effectively provide amenities to rural areas and facilitate essential services, such as health care, as well as economic activity. Advanced techniques have been developed to satisfy their tight requirements and allocate limited resources efficiently [1]. In addition, FWNs can be implemented easily under the difficult propagation conditions of rural areas as well as extreme circumstances and disasters [2, 3]. Due to its numerous advantages, many internet providers are shifting their focus toward FWN, such as Bell Canada with their Wireless Home Internet technology [4].

FWNs provide broadband internet via static wireless links, having almost stable performances, connecting fixed locations (mostly houses) to Access Points (APs). They rely on fixed receivers to transfer data at high speeds and lower latency with a reliable and cost-effective approach. The AP choice is fixed and maintained the first time during installation. Whereas in cellular networks the propagation environment and performances are variable due to receiver movement. The base station connecting the mobile device is almost changing since it is moving.

Efficient planning and deployment of FWNs is a complex and costly process in terms of time and resources. Regrettably, it is not well covered, as most studies within the field focus on mobile networks. Convenient planning allows APs to cover given areas for optimal resource allocation, while satisfying the required quality of service (QoS). Many pertinent parameters are then involved, such as AP location, antenna selection and power consumption. In mobile networks, these parameters are typically optimized on the AP side before deployment since few of them can be tuned later, and providers don't have access to the

✉ Zayan El Khaled
zayan.el-khaled1@uqac.ca

1 Computer Science Department, University of Quebec at Chicoutimi, Chicoutimi, QC G7H 2B1, Canada

2 Computer Science Department, University of Quebec in Montreal, Montreal, QC H2X 1Y4, Canada

receiver side [5]. Whereas, in FWN, these parameters can be tuned on both sides (i.e., subscriber and AP). Consequently, FWNs require additional investments on the user side to deploy convenient receiver equipment for a given user-AP association (UA) policy. However, if the required QoS is not respected, the transmission performance drops, the user ends up unsubscribing and investments are lost. Whereas in cellular networks, if the required QoS is not respected the receiver can either use his performances according to the best effort rule, or automatically change its base station or simply disconnects its link until finding another one during its movement. Therefore, a prediction method is required, not only for the AP side but also for the subscriber one to ensure the success of FWN access given a target QoS.

Such a prediction method must focus on subscribers whose signals are not powerful enough to easily confirm their subscriptions but not weak enough to unsubscribe them due to lack of QoS. In this situation, other parameters may come into play such as antenna height, line-of-sight obstruction, the number of connected users, the link distance and so forth. In addition, since the subscribers can best judge the quality of their service, their quantified feedback (e.g., amount of assistance calls, amount of maintenance or support requests, and so on) and subscription statuses are important indicators for evaluating QoS satisfaction. This study combines the well-known radio signal parameters (e.g., received signal power) with subscribers' information to obtain automated and accurate subscription success prediction.

When analyzing and improving the performances of wireless networks, Machine Learning (ML) is an invaluable asset because it can deal with complex issues, infer knowledge, and learn rules from input datasets without explicit programming, e.g., for resource management, mobility and UA policy [1]. Deep learning is also more accessible than ever due to advancements in data storage and computer architecture as well as the availability of convenient datasets. Artificial Neural Networks (ANNs) can provide approximate performances to highly complex algorithms. Furthermore, ANNs can mix user's quantified feedback with radio signal parameters in order to provide an accurate prediction of FWN subscription success. For an in-depth study of supervised, unsupervised and reinforcement ML algorithms in wireless networks, refer to Zappone et al. [6].

This paper proposes ML-based approaches for UA and AP selection by considering the subscription statuses of users, their feedback, and their pertinent information. The target is to predict whether a subscriber with a required QoS can be satisfied and associated with the FWN, then later select the most convenient AP. A new area can be outfitted with FWN by using the proposed algorithms in the selection of convenient configurations. The FWN used dataset is provided by a large-scale internet service provider for two rural areas in Canada. There are two main user classifications: satisfied and unsatisfied. Members of the former group maintain their subscriptions, contrarily to the members of the latter one. Two supervised UA algorithms are investigated. The first one relies on Nearest Neighbor whose distance function is based on the geographical distance between neighboring users. In the second algorithm, named Deep Nearest Neighbor, we extract the distance function with an ANN. Next, a new scheme based on deep imitation learning is proposed to select the most convenient AP for a given FWN subscriber. By using this scheme, the subscription success, which depends on several parameters and conditions, can be predicted accurately and efficiently.

Our main contributions are summarized as follows:

- We compare the accuracies of several ML-based UA algorithms within the context of our study.
- We propose two UA algorithms based on Nearest Neighbor and ANN. The first one is named the Weighted Geographical Radius K-Nearest Neighbor (WGRKNN) algorithm. The second algorithm is named Deep Nearest Neighbor (DNN).
- We describe a new deep imitation learning-based AP selection scheme that optimizes resource allocation in rural areas for FWNs.
- We extract the boundaries between connected and disconnected users according to a wide range of parameters with many performance indicators.
- To the best of our knowledge, we are the first to combine subscription status and user feedback information with traditional QoS indicators to optimize FWN UA and AP selection.

This paper is organized as follows: Sect. 2 introduces the state of the art. In Sect. 3, we provide details on FWNs, data collection and processing as well as the system model. In Sect. 4, we present the UA algorithms. Section 5 explains the deep imitation learning to be used for AP selection. Finally, conclusions are drawn in Sect. 6.

2 State of the art

Many UA algorithms have been proposed to efficiently distribute the load between APs covering the same area [1]. For instance, Zoppone et al. [6], analyzed the performance of a ANN transmitter–receiver UA assigner with six hidden layers (HLs). Maghsudi et al. [7] proposed a UA rule based on mean-field, multi-armed bandit games to solve the uplink cell association problem in ultra-dense small cells.

Li et al. [8] proposed a distributed UA algorithm using online reinforcement learning for balancing loads in vehicular networks. Zhao et al. [9] developed a distributed optimization framework based on multi-agent deep reinforcement learning to maximize the long-term overall cellular network utility while guaranteeing users' downlink QoS. Kim et al. [10] developed several UA policies for wireless networks by focusing on balancing flow-level cell loads under spatially heterogeneous traffic distributions. Attiah et al. [11] provided a survey of UA mechanisms and spectrum-sharing approaches for mmWave networks.

In Wi-Fi networks, UA is mostly used in densely deployed APs. Many scenarios have been studied according to specifications, services, applications or QoS [12, 13]. Xu et al. [14] studied UA in densely distributed Wi-Fi APs from an energy efficiency perspective with the goal of turning off as many APs as possible without compromising QoS. Kafi et al. [15] presented an online centralized UA scheme based on reinforcement learning to minimize subscriber dissatisfaction. Xu et al. [16] considered channel assignments and UA to balance the loads of APs operating on different channels. The authors of [17] and [18] made use of reinforcement Q-Learning for designing UA algorithms considering different scenarios. Lye et al. [19] concluded that the literature is limited regarding proposals for efficient large-scale AP deployment. They proposed an energy-efficient APs management (ON or OFF) by using the historical association records of a wide Wi-Fi system. Theoretical optimization framework and approximation algorithms of resource allocation in dense Wi-Fi networks are provided in [20] and [21]. The UA is also investigated in multiple radio access technologies (multi-RAT) networks such as in [22] and [23]. Bojovic et al. [24] outlined an AP selection scheme with an ANN trained on Wi-Fi measurements of a simple wireless router. Tang et al. [25] optimized the throughput of the Wi-Fi network through joint AP selection and bandwidth allocation. The authors of [26] proposed a novel approach that halts exploration when a suitable AP is found and resumes it after some unsatisfactory association periods. The authors of [27] optimized the AP association through a centralized potential game. Zheng et al. [28] formulated channel selection and UA as an adversarial problem, that captures the uncertainty of channel states, along with individual parameters of APs and stations.

Khalili et al. [29] studied UA and resource allocation in a HetNet to maximize the data rates of small cell users while protecting macro-cell users via a threshold on cross-tier interference. Chaieb et al. [30] formulated a joint optimization problem to maximize the number of associated users while minimizing the number of allocated time slots in a hybrid HetNet. Zhang et al. [31] focused on maximizing energy efficiency while respecting the QoS under interference and power limitations for an mmWave NOMA HetNet. Huang et al. [32] considered optimizing UA in a HetNet via utility maximization. An efficient solution using stochastic sampling is introduced to solve the learning problem.

A. Discussion and motivation

Existing UA and AP selection techniques are mainly investigated for mobile networks, densely deployed Wi-Fi local area networks, and multi-RAT networks, where an inappropriate configuration could result in AP overloading, resource underutilization or excessive interference. Since receiver terminals tend to connect traditionally to the highest received signal AP, many approaches have been proposed to optimize performances such as latency, throughput, power consumption, channel allocation, traffic load, etc. To the best of our knowledge, none of these approaches have considered all possible real-life constraints simultaneously as illustrated in the Table 1 where many research projects on UA are compared. Since the conventional procedure starts with an objective function to be optimized according to a unique scenario, the pertinent network parameters are tuned according to the QoS indicators to maximize the performance being considered. Objective functions include the optimization of radio resource allocations, latency, interference, power consumption, load balancing, RAT cohabitation and the number of APs. The network parameters and performance indicators are diversified and cannot be included together in one optimization problem. According to the Table 1, despite each research is focused toward improving one performance and is considering four pertinent parameters at best, resulting optimization problems are mostly intractable and many relaxation techniques and multi-objective optimizations have been used [20, 21, 30] [32] to facilitate the convergence toward convenient solution. The proposed algorithms have been validated by simulations only, and they are rarely used in real-life implementations.

UA is seldom used in FWNs since APs are manually selected during installation or on-site test. This operation requires the mobilization of several technicians, increasing the network deployment costs and wasting investments if tests do not lead to subscriptions. Furthermore, many doubtful test results complicate users' network subscription decisions. More precisely, if users fail to subscribe, then all investments are lost, while, if they do subscribe, they may be unsatisfied with the QoS. Whereas UA techniques can automate this decision and prevent useless operations. Therefore, mixing users' feedback with well-known radio parameters may avoid unfavorable outcomes as users are in the best position to judge the QoS of their wireless connections. The heterogeneous nature of this information

Table 1 Comparison between many research projects on UA

Project	Optimization target	ML usage	Involved parameters	Data	Application
Zappone et al. [6]	Maximize energy efficiency in interference-limited networks	Yes	Energy efficiency, maximum power, SNR	Simulation	Mobile networks
Anany et al. [22]	Balance and optimize uplink and downlink consumption and throughput respectively	No	Number of users, total throughput, consumption, probability of failure	Simulation	Multi-RAT
Elmosilhy et al. [23]	Maximize total network throughput	No	Data rate, number of users, probability of failure	Simulation	Multi-RAT
Tang et al. [25]	Optimize data rate	No	Data rate, number of users, AP occupancy rate	Simulation	Dense LAN
Carrascosa et al. [26]	Optimize AP selection	Yes	Data rate, number of APs, satisfaction criteria	Simulation	Dense LAN
Raschellà et al. [27]	Optimize the allocation of Wi-Fi hotspots according to interference	No	Flow of dissatisfaction, interference	Real implementation	Dense LAN
Khalili et al. [29]	Maximize data throughput of small cell users while macro cell users have interference threshold	No	Interference, small cell Average rate	Simulation	Mobile networks
Chaieb et al. [30]	Maximize associated users and the number of allocated time slots	Yes	Number of users, base stations and available time slots	Simulation	Mobile networks
Zhang et al. [31]	Maximize energy efficiency based on QoS under interference and power limitations	Yes	Sum rate, energy efficiency, SINR	Simulation	Mobile networks

makes ANN the obvious solution for mixing them to produce accurate predictions of UA.

The AP selection is often based on the received signal power or signal-to-noise ratio (SNR). Whereas, in real life, many other parameters are involved, such as the frequency band efficiency, number of connected users, line of sight, and synchronization rate. Unfortunately, it is difficult to consider these parameters simultaneously while trying to ensure the QoS of a radio link, especially in the case of a doubtful subscription decision. Furthermore, after customer-premises equipment (CPE) installation, some conditions may change. For instance, more users can be connected to the same AP, decreasing supplied bandwidths per user. Furthermore, leaves may decrease the received signal power during warm seasons, affecting the whole AP traffic, even in places having good radio signals. Moreover, interference may become a problem when many APs are deployed in the same area or when new radio devices, such as wireless thermometers and cameras, are installed by users. For mobile networks, advanced network optimizations or AP upgrades are used to face these issues since service providers do not control the user side. Whereas, in FWNs, CPEs can be modified or delocalized. However, if the user is still unsatisfied, its subscription may be canceled, resulting in the loss of all investments.

In order to overcome these challenges, we use ML techniques to predict whether a given user with a required QoS can be satisfied (and then subscribed). For this aim, two ML-based UA algorithms are proposed, and the scenario is simplified into a binary decision to connect or not a potential user based on his GPS coordinates and connection speed. Then an AP selection scheme is proposed through a deep imitation learning agent trained with a wide variety of APs, users, and service information from a wide FWN deployed in two different rural regions in Canada. Hence, by using supervised ML based on quantified users' feedback, real radio parameters measurements, pertinent feature selection among tens of parameters and a convenient tagging policy, a wide variety of practical implementation challenges and optimization scenarios can be considered simultaneously.

The principal target of this research paper is to overcome the drawbacks of traditional optimization techniques that lead mostly to intractable problems despite having a single objective and considering quite a few parameters. Contrarily to most of existing solution that have been validated by simulations only, our approach is using real-life network dataset to learn its optimal association policy through supervised ML while considering a wide range of parameters (about 15 parameters) without worrying about the problem of complexity and intractability. Hence, by mixing diversified users and wireless network information, a wide variety of practical implementation challenges and optimization scenarios can be considered easily and simultaneously. Considering our previous discussion, the motivations of this paper are summarized as follows:

- UA and AP selection techniques are not well covered in FWNs.
- The received signal power is less efficient for UA and AP selection when used alone.
- The network parameters, real-life constraints and QoS performance indicators are diversified and cannot be included simultaneously in one optimization problem.
- Existing solutions have been validated only by simulation, and they are rarely tested in real-life implementations.
- The mobilization of several technicians during field tests to predict users' network subscription decisions must be avoided to reduce invested resources in FWN.
- Many field tests produce doubtful results that complicate subscription decisions.
- Find methods to mix users' feedback with well-known radio parameters to increase the accuracy of the subscription decision especially for doubtful performance indicators or when network conditions change.

3 Preliminaries

A. Fixed Wireless Networks (FWNs)

A typical FWN's radio link is presented in Fig. 1. The AP supports long-range Wi-Fi, or LTE communication, with many CPEs providing broadband internet connectivity to fixed locations [3]. It can be installed in a high place, such as a communication tower, tree or house's roof, with a good line of sight (LOS) toward the area to be covered. A CPE can be installed on a high spot, such as a roof, or tree, then hardwired to users' inside equipment, such as wireless routers or computers. The lines of sight contain hills, plains, and lakes, and they are mostly covered with trees. The propagation environment varies between two extreme conditions: extremely cold and snowy weather with conifers during the winter, or hot and rainy weather with deciduous leaves in the summer.

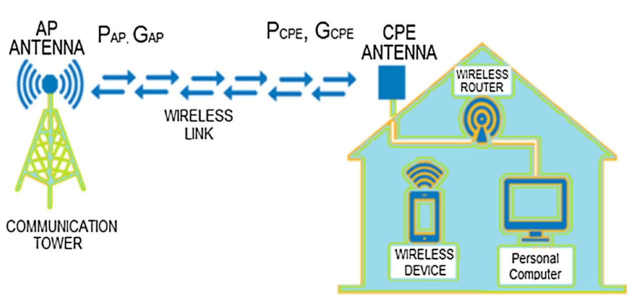


Fig. 1 Typical fixed wireless radio link between a given AP and a CPE

The heterogeneity of FWN is reflected through the frequency bands 915 MHz, 2.4 GHz and 5.8 GHz, in addition to the licensed 3.65 GHz band, for diverse radio signal penetration. Higher frequencies are mostly used for LOS links, whereas lower ones are used for non-LOS (NLOS) links, as they have better penetration. The bandwidth ranges between 5 and 80 MHz for diversified throughput options. Antennas' heights range from few feet to more than one hundred feet. Many AP and CPE antennas' gains, GAP and GCPE, respectively, and radio transmitted powers PAP and PCPE, respectively, are used for various radio link configurations. They have one or two streams for multiple input-multiple output (MIMO) diversity. The distance between radio links ranges from few meters up to 34 km. More details are provided in our previous works [3] and [33].

Figure 2 shows a typical FWN deployment in rural area. In this scenario, many APs are deployed to deliver broadband internet to all the houses in the target region. Since wireless links can be obstructed, some houses cannot be serviced, even if they are inside the coverage area. In some cases, subscribers can experience decreases in QoS due to a variety of reasons such as interference and path loss attenuation, that lead them to unsubscribe. In other cases, a given house might be situated in an overlapping coverage area, where many APs operate at the same time, or an area with weak received signal power. Consequently, prediction algorithms are required to decide if a given location can be subscribed to the FWN with a given QoS, then select a convenient AP or configuration that guarantees this QoS.

B. Data collection

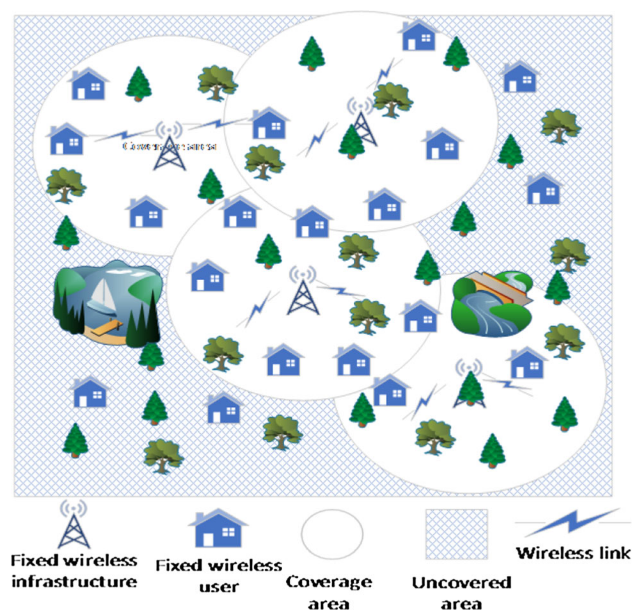


Fig. 2 Typical FWN deployment in a rural area

The collected data was extracted from the database and radio link measurements of an internet service provider for two rural regions in the province of Quebec in Canada: Saguenay-Lac-Saint-Jean (SLSJ) and Outaouais (OUT). Figures 3 and 4 show the distributions of thousands of wireless links in these two regions. The dataset describes user profiles, provided services and network deployment. Firstly, it was cleaned to remove inaccurate sample points and ensure the reliability of the collected data. As an example, validity intervals are verified, for instance, the height of an antenna installed on a house cannot exceed 50 feet. In addition, it is adapted to the used tools such as replacing special character. It is also transformed as needed, such as the transformation of distances from km to meters. Some missing information is inferred, such as the distance between two GPS coordinates, the season and the subscription duration. Next, the data is scaled with the Scikit learn library in Python to improve the ML algorithm's convergence [34].

Despite the advantages of using a real-life dataset, many challenges are present due to the recurrence of human operators' mistakes and the lack of certain information due to the fact that the industrial environment is constantly changing. Therefore, some pertinent information was not collected in the past because it was not relevant at that time. For example, the Google API was used to extract GPS coordinates from civic addresses since they had not been collected, even though they are necessary for the calculation of distances. In addition, the radio links are upgraded regularly, but sometimes their information is not updated due to human error; in such cases, the dataset is updated by inference. Likewise, suspicious sample points are eliminated. For instance, some radio links are established on the secondary radiation patterns of APs, so only the links included in their main lobes are retained. Eighty percent of the dataset is used to train the learning algorithms, while the remaining 20% is used for testing.

Fig. 3 Distribution of wireless links in the SLSJ region

Fig. 4 Distribution of wireless links in the OUT region

C. System model

Our system architecture consists of K users and B APs. The indexes $k \in \mathcal{K} = [1, \ldots, K]$ and $b \in \mathcal{B} = [1, \ldots, B]$ represent a given user and a given AP respectively. Each user is either subscribed and satisfied with the QoS or unsubscribed due to bad QoS, the subscription status of a given user k is given by y_k. In the UA scenario, each user is described by GPS coordinates and connection speed. Let r_k represents the connection speed of the k $_{\text{th}}$ user and $\mathcal{G} = (a_k, o_k)_{k=1}^{K}$ the set of GPS coordinates, where each GPS coordinate consists of latitude and longitude, i.e., $g = (a, o)$. Each potential user k has a set of ascending distances $\mathcal{D}_k$ with all neighboring users k, namely, $\mathcal{D}_k$ = $\{d_{1,k}, d_{2,k}, d_{3,k},\ldots,d_{k,k}\}$, that is computed based on the GPS coordinates set $\mathcal{G}$.

Most UA and resource allocation systems are based on the data rate and Signal to Interference plus Noise Ratio (SINR) [9, 29–32]. The data rate r of user k, in its most basic form, is given by:

$$r_k = W \log(1 + \mathrm{SINR}_k), \tag{1}$$

where W is the frequency bandwidth, its range may vary between 5 and 80 MHz for various throughput options. SINR is the signal to interference plus noise ratio. Its computation hides some complications, such as the estimation of the received signal power, which depends on the type of area (e.g., urban, rural); LOS obstruction; frequency band and so on. The noise and the interference can be generated by any radio signal that disturbs the main signal. An accurate assessment of this ratio is important when predicting the data rate, but it requires a detailed estimate of all possible interferences in addition to the propagation path loss. This assessment is difficult since the radio environment is "polluted" with many kinds of interfering signals, especially since the FWN frequency bands are mostly open and the path loss propagation is dependent on many difficult-to-predict parameters. To ensure that

QoS expectations are met despite the SINR complications, we use the measured data rates of potential users and then predict subscription classes according to UA and AP selection scenarios explained in the following sections.

D. Rule-based data tagging

The aim of rule-based data tagging is to support the dataset in providing the optimal UA results. Since the output classes for our proposed ML algorithms are based on measurements of users' subscription statuses, this information must reflect only QoS fulfillment. For instance, some satisfied users may unsubscribe for a better price or seasonal leave, which is irrelevant as user feedback regarding QoS. So, their subscription status must be modified from "*disconnect*" to "*connect*" since they have unsubscribed based on their personal choice rather than a bad QoS. Hence, the optimal results of ML algorithms can be obtained by paying special attention to the output classes (user subscription status) in order to remove irrelevant effects.

Rule-based tagging is presented in Fig. 5, where subscription status is corrected to reflect only the QoS. Therefore, users are split into three groups according to their connection speed: basic, medium and high. Users having medium or high speeds are tagged as "*Connect*" since they have a good QoS, the reason why they were offered such connection speed. Users with basic speed may have doubtful QoS since they were offered the lowest connection. For this aim their subscription statuses are corrected according to many criteria. For instance, if their received signal power is higher than -55 dBm, their subscription status is set as "*Connect*". If the signal power is between -55 and -60 dBm with a direct LOS, the tag is also "*Connect*". For a signal power between -60 and -65 dBm, a direct LOS and a subscription duration of more than two years, a "*Connect*" tag is assigned as well since these users have had time to unsubscribe if unsatisfied with the QoS. At the end, a minimal margin of 5 to 10 dB is provided according to the sensitivity of the highest data rate [3] and the seasonal propagation variation [35].

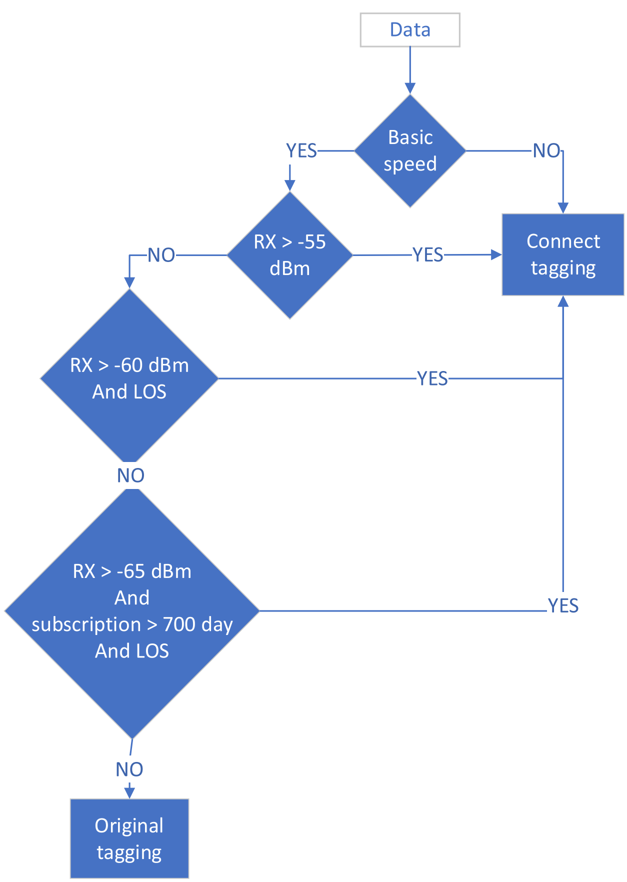


Fig. 5 Rule-based tagging scheme

4 User association

A. Prediction scenario

The UA prediction scenario is illustrated in Fig. 6; it aims to predict the subscription success to the FWN for a given connection speed at a given location. Traditionally, this scenario requires a wireless planning software to establish AP coverage; such software requires considerable resources to cover license costs, simulation times, computing complexity and storage. Furthermore, a manual field test is then needed, which involves simultaneously many qualified technicians in the field and in the head office. This scenario can be simplified greatly with supervised ML algorithms trained with the GPS coordinates and the connection speeds of all the sample points in the coverage area as input parameters. Output classes correspond to subscription statuses. After training, a new subscription status (connected or not) is predicted through the GPS coordinates and the required connection speed, which reflects the satisfaction of the potential user with the QoS of his wireless link. To do so, the algorithm uses distances to neighboring users. Generally, the smaller the distance, the greater the similarity between users, and two ML algorithms based on Nearest Neighbors and ANN are optimized to exploit this assumption.

B. Comparison of machine learning algorithms.

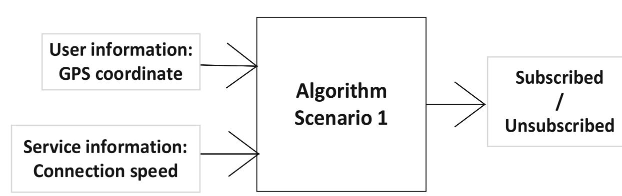


Fig. 6 UA prediction scenario

Since the No Free Lunch Theorem establishes that when all possible data-generated distributions are averaged, all ML algorithms will have the same performance [36]. Table 2 presents the accuracy of many ML algorithms before and after rule-based tagging. It identifies the most accurate algorithms to be improved later according to our dataset. We notice that the Nearest Neighbor and ANN algorithms are the most accurate. The accuracies of the Support Sector Machine (SVM) and Gaussian Radial Basis Function (RBF) algorithms are both 0.862 after rule-based tagging. The Quadratic Discriminant Analysis (QDA) algorithm has the lowest precision at 0.345.

C. Nearest Neighbors.

The Nearest Neighbor algorithm is characterized by its conceptual and implementation simplicity [37]. It basically identifies sample points with features similar to those of unlabeled data and then assign their tag to them by a majority vote. Many distance functions that assess this similarity are proposed [38], and many technics have also been inspired by this concept [39]. Its basic form requires $O(ktd)$ iterations, where k is the number of neighbors, t is the number training sample points and d is the data dimensionality.

Its accuracy for UA in FWNs is shown in Fig. 7 and was obtained by using the Scikit learn library in Python [34]. The maximum of 0.872 is achieved for k=9 in the K-Nearest Neighbor (KNN) classification scheme, which labels data according to a majority vote. For Radius Nearest Neighbor (RNN), only sample points having distance values within the specified radius participate in the vote; its accuracy is 0.865. RNN is better than KNN at detecting uncovered zones or unavailable connection speeds inside a given area since its radius can be tuned accordingly.

D. Weighted Geographical Radius K-Nearest Neighbor (WGRKNN)

Table 2 Accuracy comparison for best known ML algorithms

ML algorithm	Original tagging	Rule-based tagging
Nearest Neighbors	0.764	0.865
Artificial Neural Net	0.703	0.862
Linear SVM	0.703	0.862
RBF SVM	0.703	0.862
Gaussian Process	0.760	0.862
Random Forest	0.746	0.862
Decision Tree	0.759	0.861
AdaBoost	0.763	0.856
Naive Bayes	0.611	0.345
QDA	0.718	0.345

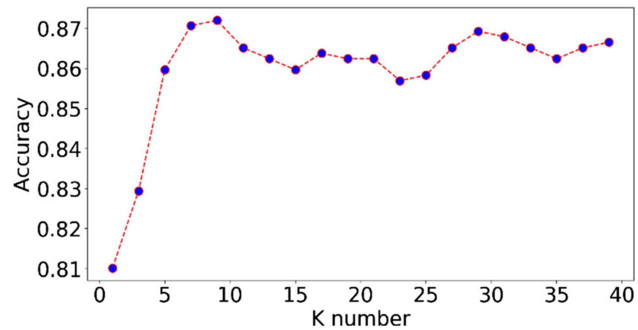


Fig. 7 Accuracy of KNN with Euclidean distance function

The Weighted Nearest Neighbor algorithm assigns larger weights to the nearest or most similar sample points. For example, weights inversely proportional to distance could be assigned. The geographical location is important in the context of our research and for context-aware applications, in general [40]. This proposed algorithm considers only the first K sample points with the smallest distances within a given radius. Their votes are weighted by their geographical distances and scaled so that the weights sum is equal to 1 to reflect the prediction certainty. The weighting equation is presented as follows:

$$W_{i,j} = \frac{\frac{1}{d_{i,j}}}{\sum_{j=1}^{j=K} \frac{1}{d_{i,j}}}, \quad (2)$$

where $W_{i,j}$ is the weight of the jth training sample point participating in the vote regarding the ith testing sample point (potential user) and $d_{i,j}$ is the geographical distance between them. The predicted class is presented according to the WGRKNN equation:

$$y_i = \sum_{j=1}^{j=k} \left(W_{i,j} \times Y_j \right), \quad (3)$$

where y_i is the predicted class (subscription status: *connect* or *disconnect*) of the ith testing sample point and Y_j is the subscription status of the jth training sample point.

This algorithm is presented in Fig. 8. It avoids the weaknesses of KNN and RNN, as it makes a better selection of neighbors, and it controls their number and their radius while maintaining the same asymptotic time complexity. First, it detects whether the potential user (testing sample point) is within the coverage area (see Fig. 2), then it predicts whether the required connection speed is available. If one of these conditions is not satisfied, the subscription status is directly labeled as *disconnect*. If the two conditions are satisfied, then the subscription status is predicted by the WGRKNN equation. Figure 9 presents its accuracy according to the radius and the number of neighbors (k). This accuracy increases with the number of neighbors until reaching its maximum of 0.868 for a radius

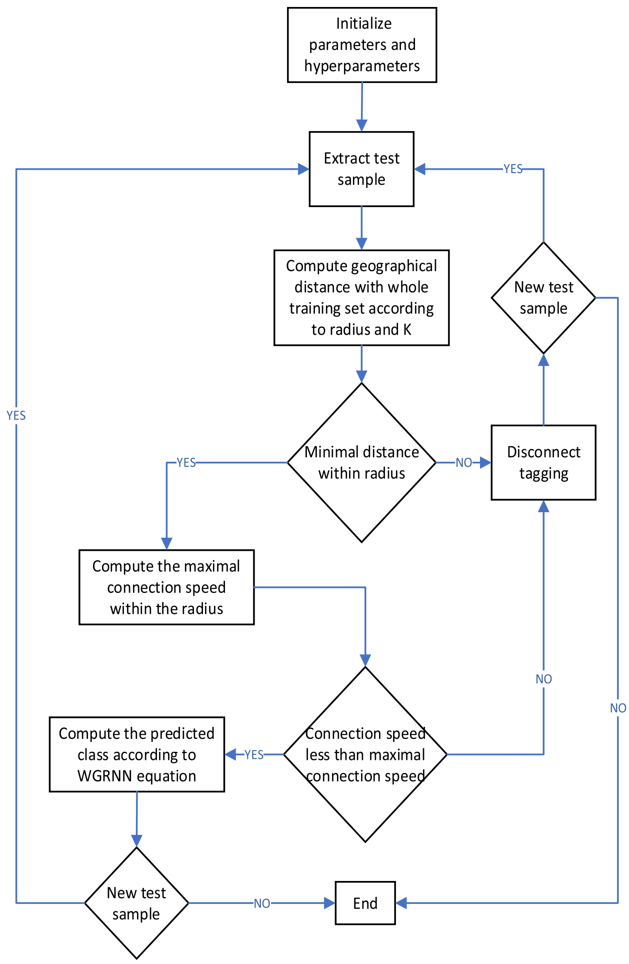


Fig. 8 WGRKNN scheme

of 9 km and k=20, while decreasing afterwards since a greater number of neighbors will not help. However, it remains within the range of the state-of-the-art, i.e., between 63 and 98%, depending on the used dataset [41, 42]. The hyperparameters (k and the radius) are optimized with a parameter sweep.

E. Neural network optimization.

ANN has been used as a universal approximator in many technical fields, such as wireless networks [1], to approximate the behavior of iterative and optimization algorithms for various classification problems [43]. Therefore, an ANN is optimized for UA prediction scenario (see Fig. 6) by using the TensorFlow library in Python. GPS coordinates and connection speeds are the input parameters. They are normalized and scaled before being injected into the input layer. Various HL configurations have been tried, and one HL with 10 neurons works fine in our case. The output layer is composed of one neuron for connection success prediction. Other hyperparameters, such as learning rate, number of neurons in the hidden layer, batch size, epoch number, are optimized with grid search according to Figs. 10, 11 and 12. The maximal accuracy is 0.862 for a learning rate of 0.05, a batch size of 4, and 20 epochs.

The neural networks used during our research are trained by using the learning rate decay which provides a better convergence [44]. At first, the learning rate is set with a large value to accelerate the training and prevent its blockage in a local bad optimum. Then, it decreases progressively to provide the convergence toward the optimal solution and prevent the oscillation. The convergence of

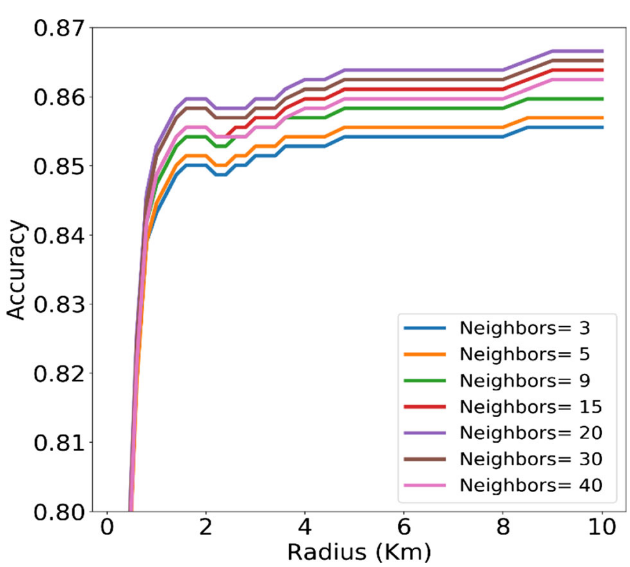


Fig. 9 WGRKNN algorithm accuracy

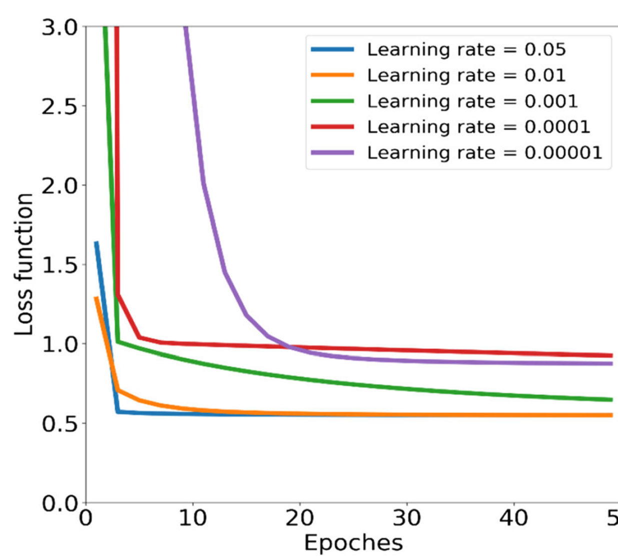


Fig. 10 Loss function according to the learning rate and number

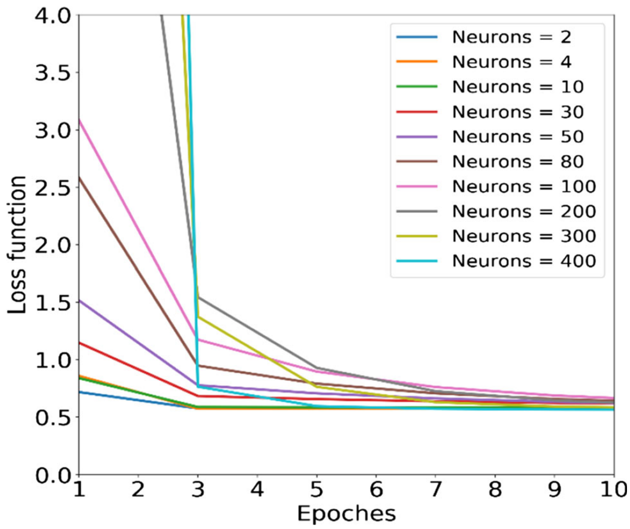


Fig. 11 Loss function according to HL's neurons and epochs

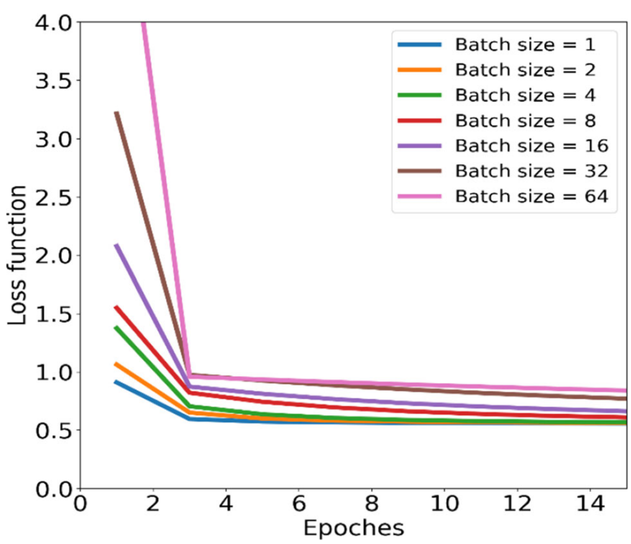


Fig. 12 Loss function according to batch size and number of epochs

ANN is analyzed earlier in many research papers such as [45] and [46]. It is also proved through simulations and results presented in the remaining of this paper. Bienstock et al. [47] studied the computational complexity of many types of ANN. Its minimum during offline training is of $O(t.w^l)$, where t is the training data size, l is the number of HLs and w is the number of neurons per HL [48]. When the training is repeated during e epochs, the time complexity is $O(e.t.w^l)$.

F. Deep Nearest Neighbors (DNN).

The Deep Nearest Neighbor algorithm extracts the distance function through an ANN trained with the geographical distances between the testing sample points and their neighbors. Hence, the input layer has as many neurons as there are neighbors. The output layer has one neuron for connect or disconnect tagging. The number of HLs and the number of neurons in each layer are optimized for better accuracy. The optimization is limited to two HLs, which is sufficient in general [49, 50]. Hyperparameters are optimized similarly as the previous section. The distance function is extracted and applied to the Nearest Neighbor algorithm, as shown in Fig. 13. First, the hyperparameters, number of HLs, number of neurons in each layer, number of neighbors and the radius are initialized. For each combination of parameters, the ANN is trained with the distance vectors $D=[d_1, d_2,..,d_k]$, where k is the number of neighbors. After training, the weights are extracted and applied to the neighbors' corresponding classes. The process is repeated for all testing sequences, then the accuracy is computed. Finally, the configuration that leads to the best accuracy is retained.

As justified previously, the ANN training requires a time complexity of $O(e.t.w^l)$. After the extraction of the distance function, it is applied to the KNN algorithm requiring $O(ktd)$ iterations. Hence, the training of a single

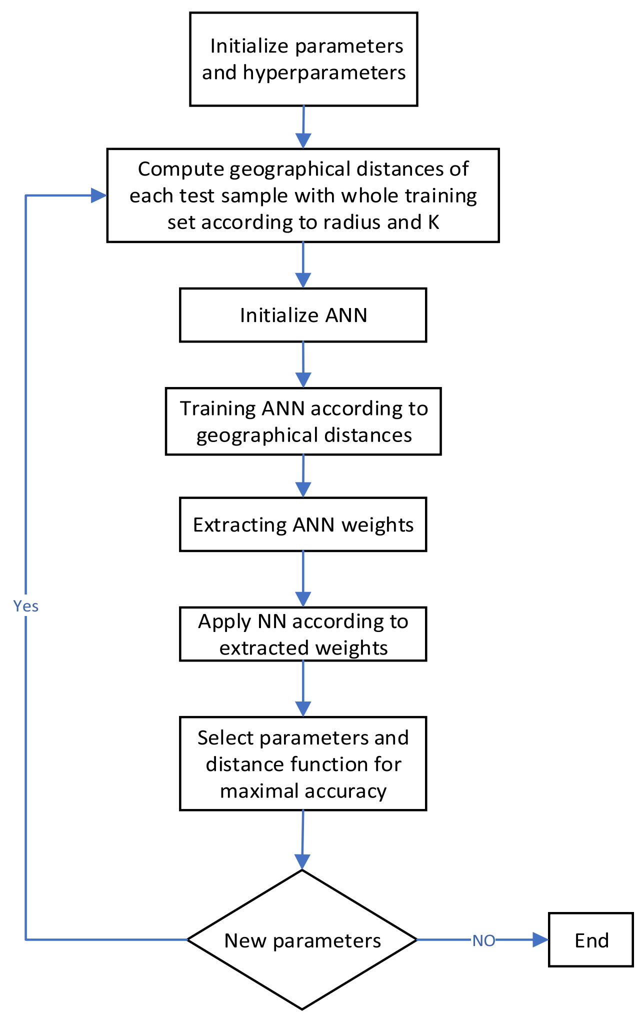


Fig. 13 The DNN algorithm's scheme

configuration of the algorithm requires $O(t(ew^l + kd))$ iterations. Complete optimization requires many iterations according to the hyperparameters' dimensionality. Therefore, the total time complexity is $O(ht(ew^l + kd))$, where h is the number of hyperparameter combinations used to find the optimal configuration.

Figures 14 and 15 show the improvement of the DNN algorithm over the ANN. They compare the accuracy of the DNN algorithm with ANNs having one HL and two HLs, respectively. The number of neurons in each layer is also optimized. The number of neighbors and the radius are set to the optimal values extracted previously (k=20 and the radius=9 km). The maximal accuracy is 0.842 for both one HL and two HLs. A DNN with three HLs is optimized, and the maximal accuracy remains the same. Figures 16 and 17 present the DNN accuracy for one and two HLs, respectively, where the number of neighbors and the radius are also tuned. Note that the optimal parameters have changed, and the maximum accuracy is improved to 0.863. The configuration of the one-HL DNN is k=10, radius=6, and 80 neurons in the HL. The configuration of the two-HL DNN is k=25 and radius=9, with one and 80 neurons in the first and second HLs, respectively. Note that the WGRKNN algorithm is a good alternative since it provides the same accuracy as the DNN for a greater simplicity and while requiring lower resources.

5 AP selection

When a user with a poor radio signal is connected to an AP, the data retransmission increases due to the bit error rate increase. Consequently, the data exchanges of users with

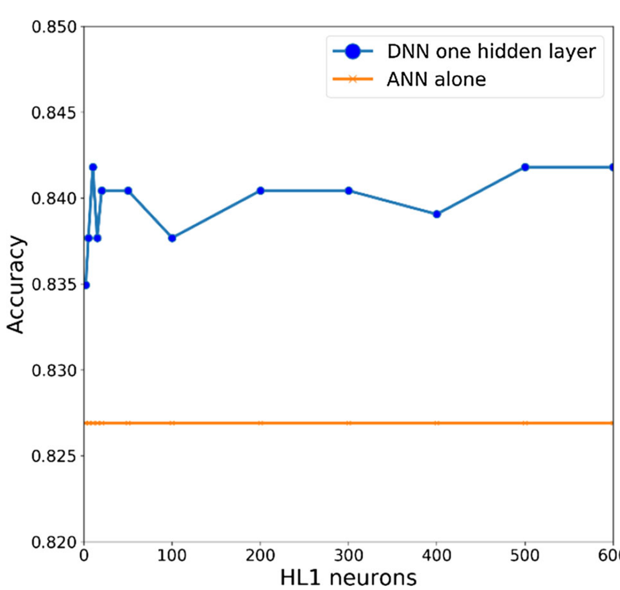


Fig. 14 Comparison between DNN and ANN for one HL

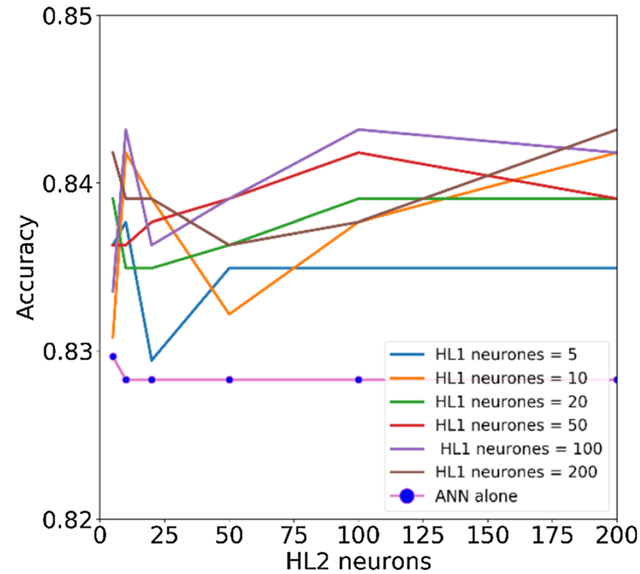


Fig. 15 Comparison between DNN and ANN for two HLs

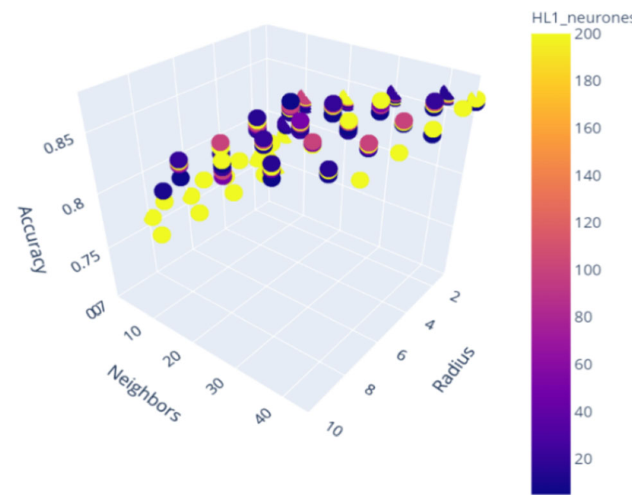


Fig. 16 DNN algorithm with one HL, all parameters tuned

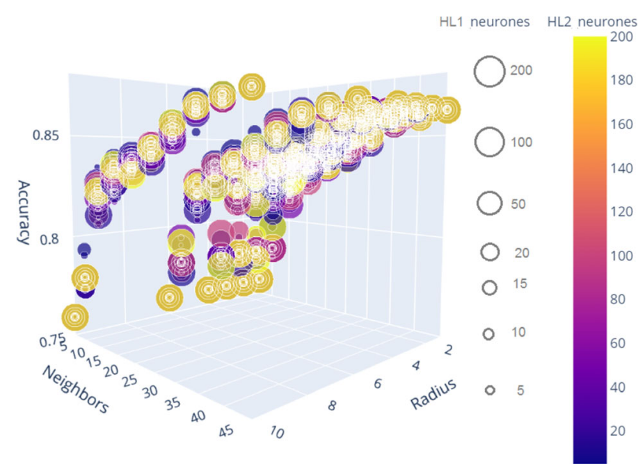


Fig. 17 DNN algorithm with two HLs, all parameters tuned

good radio signals can be delayed and the total available throughput of the concerned AP is reduced. The main challenge of AP selection is allowing only satisfiable users to access the FWN through a given AP to avoid decreasing its throughput. This technique protects the QoS of satisfied users against users with poor radio signals. Thus, service providers focus on satisfiable users to save time and costs.

A. Prediction scenario

The AP selection scenario (see Fig. 18) predicts whether a given AP leads to an acceptable QoS level for a given user. Traditionally, this prediction is made with the received signal power or the SNR of all APs covering the target location. Whereas, in real life, many parameters are involved in the QoS estimation, and many cases lead to doubtful QoS indicators, as explained previously. Hence, the ML algorithm must consider and mix real-life information with classical radio signal parameters to accurately predict the success of a given user-AP association. For this aim, it is trained with users, APs, service information as inputs in addition to subscription statuses as outputs. This approach has better accuracy than previous ones, as it includes user feedback. This quantified feedback is collected through users' interactions with support services (e.g., calls, support and maintenance requests). Consequently, good QoS ensures the satisfaction of users, who then maintain their subscriptions. Whereas bad QoS lead to unsatisfied users who unsubscribe and lost investments.

This prediction scenario can also be used as a recommendation system for the best AP equipment to cover new areas, as it is able to infer the probability of success for a given AP configuration. Thus, the AP leading to the highest probability of success is maintained. The same process can be used to infer the best CPE configuration leading to the highest subscription probability for a new user.

B. Deep imitation learning (DIL)

The deep imitation learning (DIL) concept is explained as an ANN trained to imitate the UA behavior of a FWN in order to predict the subscription success of a new radio link. For this aim, pertinent information concerning the AP, user and service is used as input, whereas the output classes consist of *Connect* and *Disconnect* labels. The target is to predict whether a given wireless link will have a good QoS satisfying its user, who will then remain as subscriber, or a bad QoS that will fail to satisfy this user, who will then unsubscribe.

Nowadays, DIL has attracted attention in many fields, such as robotics [51] and wireless resource allocation [52], where advanced skills are learned through the observation of expert demonstrations. During these demonstrations, a convenient policy (π), which maps a given action ($\mathcal{A}$) to an environment state ($\mathcal{S}$): $\pi(\mathcal{S}) \rightarrow \mathcal{A}$, is learned. The concept is similar to the supervised learning approach, where a machine can learn a wide variety of decisions and skills without explicit programming. The key element for the success of imitation learning is the availability of high-quality demonstrations that include all possible environmental states, which can be a difficult task. Once this condition is satisfied, DIL has remarkable advantages over traditional ML methods, including better accuracy, better performances for large datasets, fast inferences and easy deployment [53].

Figure 19 presents the DIL training and testing schemes, where the FWN radio link state $\mathcal{S}$ is fed into DIL policy π, which corresponds to an ANN, to predict the subscription success. The ANN is trained offline by comparing its output $\mathcal{A}^*$ with the output of the expert demonstration $\mathcal{A}$ of

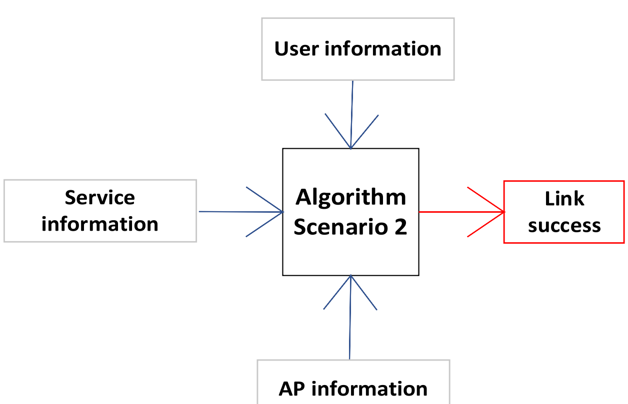


Fig. 18 AP selection prediction scenario

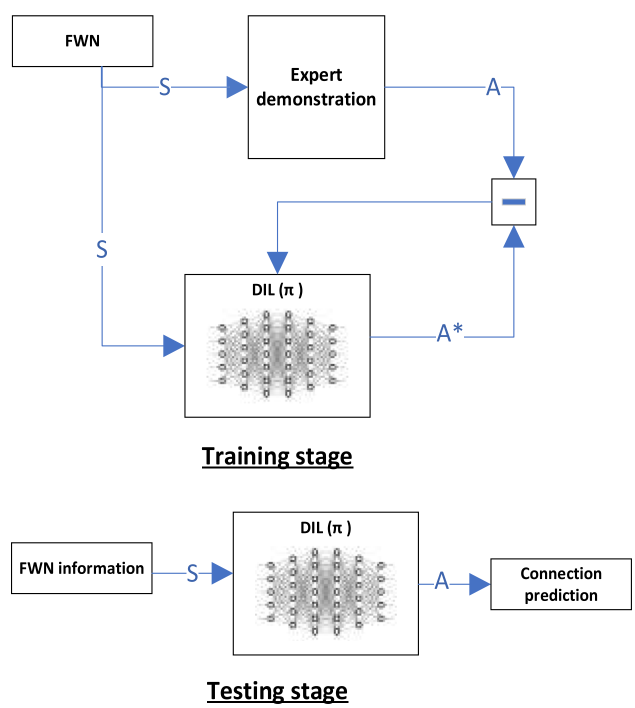


Fig. 19 DIL scheme for an FWN

the FWN subscription status. At the end of the training, the policy π is able to provide convenient action $\mathcal{A}$ for the corresponding link state $\mathcal{S}$ and the shift between $\mathcal{A}$ and $\mathcal{A}^*$ is minimized to ideally reach zero. Once trained, the DIL process can make a pertinent direct online subscription decision according to the current FWN state. The more faithfully the state is represented, the closer the action is to reality. Hence, the challenge lies on the selection of the most pertinent parameters for representing the FWN radio link state faithfully.

C. Used Parameters

Classically, the most frequently used parameters are the received signal power, data rate and SNR. Yet, in real life, many scenarios come into play simultaneously, meaning that more parameters must be considered. The pertinent parameters presented in Table 3 were selected through a correlation study of a wide variety of FWNs and user information with subscription status. Figure 20 presents the t-distributed stochastic neighbor embedding (TSNE) 2D dimensionality reduction of the AP selection dataset to ease its representation since it includes around twenty parameters. This technique aims to help with the visualization of high-dimension datasets, by reducing their dimensionalities while retaining only the most pertinent parameters [54]. The TSNE representation of the whole and test datasets are illustrated in Figs. 20 and 21, respectively. Figure 22 shows the TSNE after minority class up-sampling with the KMeansSMOTE up sampler to correct data imbalance [55]. During our research, in a test involving many up samplers, the KMeansSMOTE up sampler provided the best accuracy. Figure 23 presents the TSNE of the up-sampled test dataset after its transformation with the MaxAbsScaler of the Scikit learn library in Python [34].

D. Deployment.

The deployment flow is subdivided into three main phases: dataset generation for offline demonstration, offline training to optimize the imitation policy and finally, the online subscription success predictions for potential users. First, accurate and reliable demonstrations are generated through measuring thousands of radio links in addition to gathering other relevant information from the central database, as presented in Table 3. Then the dataset is pre-processed by removing irrelevant information and inferring missing one. It is also up sampled to correct minority class imbalance and then scaled for better ANN convergence.

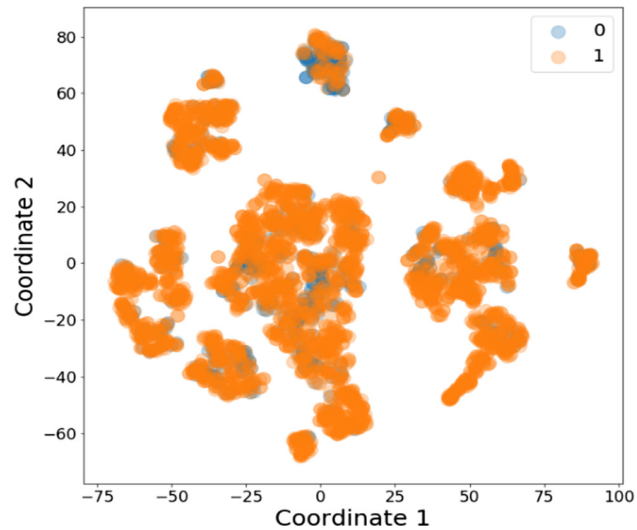


Fig. 20 The dataset for AP selection with TSNE 2D dimensionality reduction

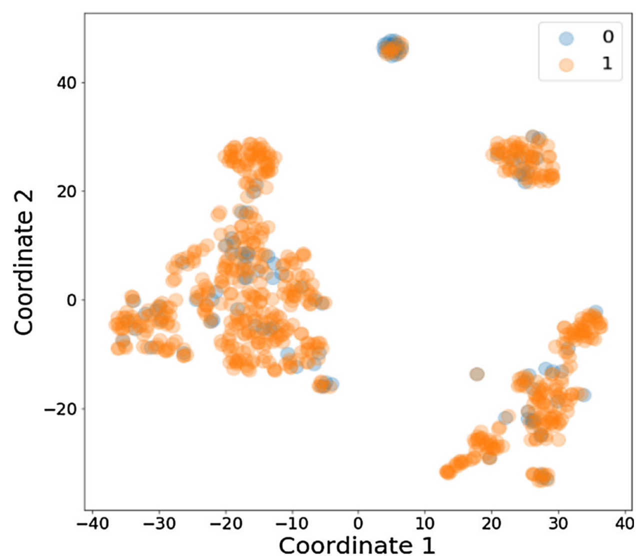


Fig. 21 The test dataset with TSNE 2D dimensionality reduction corresponding to 20% of the complete dataset

Table 3 Pertinent parameters used for AP selection

FWN radio link state $\mathcal{S}$ AP	Action $\mathcal{A}$ User	ANN Output
Frequency band, AP height, connected users, antenna streams, received signal strength, client connection quality (CCQ)	Received signal strength, LOS, link distance, data rate, antenna streams, support requests for levels 1 and 2, maintenance requests, calls, synchronization rate, SNR, CCQ, path loss	*Connect* and *Disconnect* labels

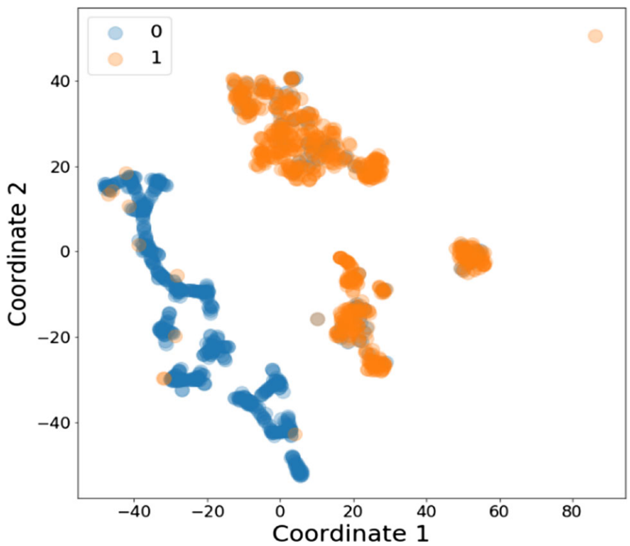


Fig. 22 TSNE 2D dimensionality reduction of the test dataset after minority class up-sampling with the KMeansSMOTE up sampler to correct the dataset imbalance

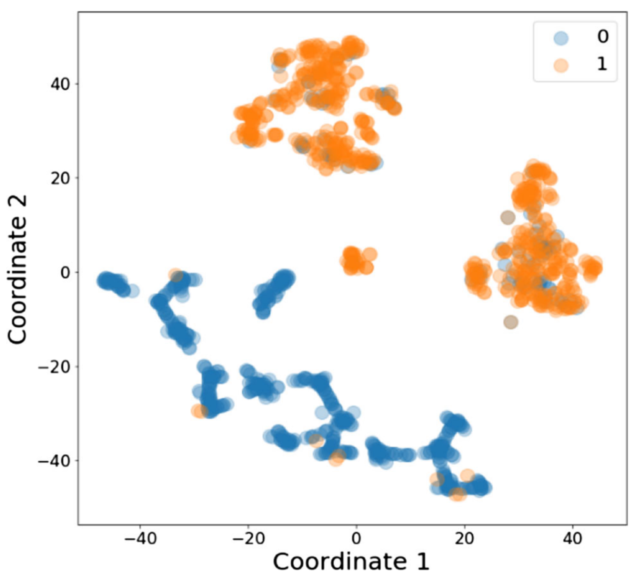


Fig. 23 TSNE 2D dimensionality reduction of the up-sampled test dataset after its transformation with the MaxAbsScaler

This information (Table 3) is transformed into an input vector of states $\mathcal{S} = [\mathcal{S}_{1,i}, \mathcal{S}_{2,i}, \mathcal{S}_{3,i}, \ldots, \mathcal{S}_{n,i}]$, where the ith user has his own radio link state vector and subscription decision Y_i for the DIL output class. The DIL ANN policy model is trained offline to imitate the subscription patterns for the practical FWN deployment, as shown in Fig. 24. During training, its coefficients are tuned through the backpropagation of the difference between the annotated output class (or action space $\mathcal{A}$) and its prediction $\mathcal{A}^*$. The rectified linear unit (ReLU) is used conventionally as the activation function for the HLs, whereas the sigmoid function is used for the output layer. Binary cross entropy serves as the loss function and stochastic gradient descent (SGD) is used to optimize the ANN coefficients. The input layer has n neurons, each one represents a state parameter, and the output layer has one neuron to make the binary decision. Since the DIL training process is essentially based on ANN, its time complexity is the same as that studied previously.

At the end of the training, the predicted actions $\mathcal{A}^*$ converge toward the true action space $\mathcal{A}$, allowing the ANN policy model to infer the subscription decisions for new FWN users (Y_i) by providing their state vectors $\mathcal{S}$. AP selection is presented in Fig. 25. After initializing the optimal parameters and hyperparameters (e.g., batch size, learning rate and so forth), neighboring users of the location to be predicted are extracted in order to deduce information regarding potential APs. During the data inference step and for each extracted AP, the corresponding new state vector is applied to the trained ANN. When there is more than one successful prediction, the AP with the highest probability is selected since the DIL outputs probabilities of success for wireless links. The extraction of available APs is made with the nearest neighbor algorithm requiring O(kKd) iterations. The AP selection process injects possible APs into the trained ANN, then selects the AP with the maximal output. The ANN requires $O(k.w^l + k)$ iterations. The total time complexity of the online deployment is $O(kK\text{d} + k.w^l + k)$.

E. Optimization

Since AP selection depends mostly on the ANN's accuracy, many ANNs with various HLs have been tested. The maximal accuracies according to the number of HLs are listed in Table 4. The two HLs configuration is the best and has an accuracy of 0.94. Figure 26 presents the optimization of this configuration with the Scikit learn library in Python [34]. The X and Y axes are used for the number of neurons in HL1 and HL2, respectively. The sizes and the colors of circles also differ for HL1 and HL2 for better clarity. The optimal layer configuration assigns 90 and 4 neurons to HL1 and HL2, respectively. The learning rate is 0.05 and the batch size is 8.

The receiver operating characteristic (ROC) curve is presented in Fig. 27. It is annotated with the binary classifier thresholds so that a convenient threshold for a fair balance between True and False positive rates can be chosen. The area under the curve (AUC) is 0.97. Note that the classifier has good classification skills since it differentiates well between the two subscription actions. Figure 28 presents the precision recall curve, which is annotated with the decision thresholds. Its AUC is 0.972, which also proves the high classification capability of the designed classifier.

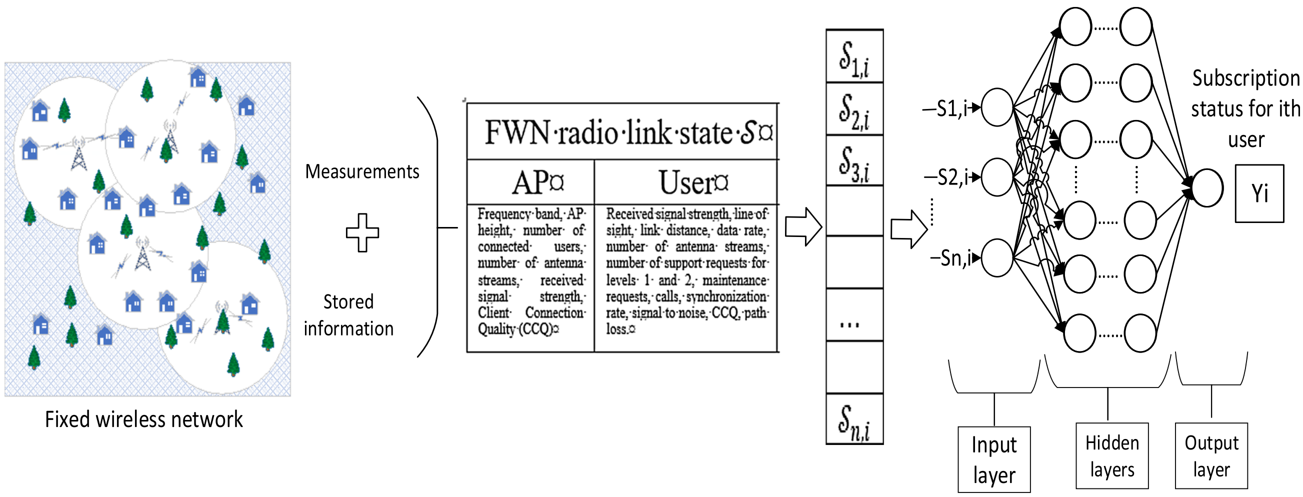


Fig. 24 Offline training of deep imitation learning

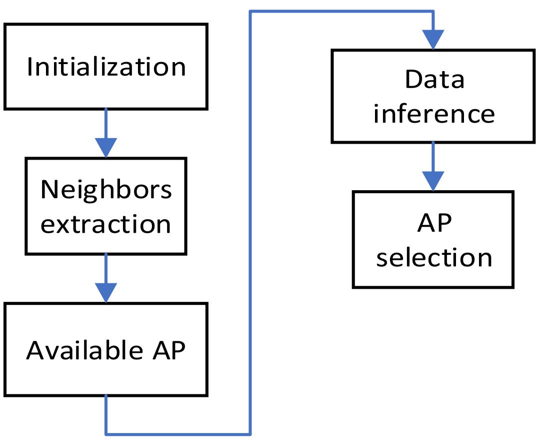


Fig. 25 Proposed AP selection online deployment

Figures 29 and 30 show its loss function and accuracy during the training and testing cycles, respectively, via the Keras ML framework. Its accuracy is still 0.94, which is close to the accuracy obtained with the Scikit learn library, thus proving the performance reproducibility over different frameworks. The convergence is reached within 80 epochs when training and testing performances are the closest. After this limit, the ANN is memorizing training data and the difference between predictions and measurements is decreasing for the training dataset. Whereas, it is increasing for the testing dataset since its generalizing capability is degraded.

Figure 31 summarizes the normalized confusion matrix. It predicts 97% of the users who subscribe and 92% of users who unsubscribe, which means that only 3% of users that can be subscribed are not predicted conveniently, representing lost opportunities. Whereas 92% of users who will unsubscribe are predicted, preventing unprofitable investments, which means that equipment costs, workforce time, bandwidth transfers and QoS degradation will be avoided and efforts can be redirected to successful network deployments in new areas.

Figure 32 shows the classifier threshold tuning according to the accuracy, geometric mean (G-Mean), Youden's J statistic and the F score. Its optimal value is 0.49, so it provides a fair balance between the sensitivity and specificity or precision and recall. It also maximizes all previously listed performances. Figure 33 presents the TSNE representation of the test dataset and the classifier contour. Note that, classes are distributed into several collections with similar characteristics, such as frequency bands and LOS. Thus, the classifier succeeded in detecting and classifying these data patterns.

F. FWN performance prediction.

In this section, the algorithm's predictions are compared to measurements for many FWN QoS parameters. The

Table 4 Maximal accuracy according to the number of HLs

Number of HLs	Maximal accuracy	ROC AUC score	Precision-recall AUC score
1	0.939	0.939	0.95
2	0.943	0.943	0.952
3	0.939	0.939	0.953
4	0.939	0.939	0.951
5	0.937	0.937	0.951

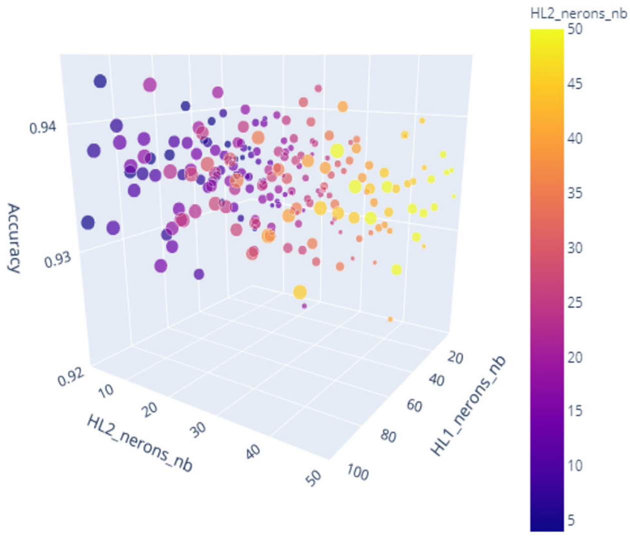


Fig. 26 Accuracy optimization of the two-HL ANN used for AP selection

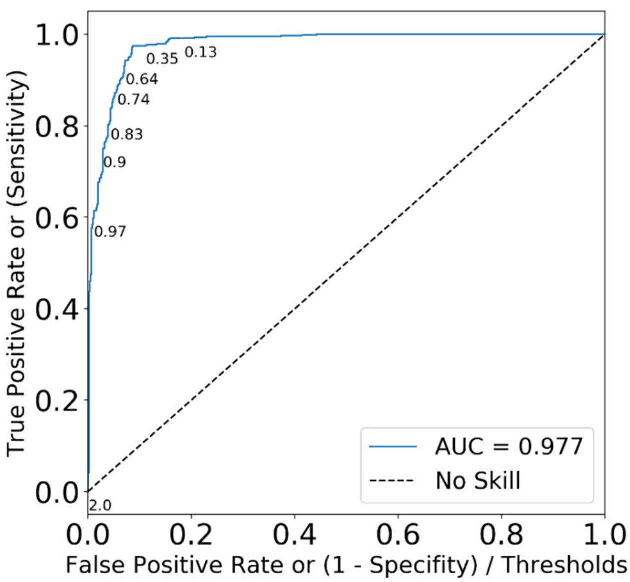


Fig. 27 ROC curve for the two-HL ANN used for AP selection

cumulative distribution functions (CDFs) are similar for the measurements and predictions, which proves the accuracy of the proposed algorithm. Figure 34 shows that the path loss (PL) predictions are close to the measurements for the Connect and Disconnect sample points. This figure also exhibits the PL boundary between the Connect and Disconnect users. For low PL, the CDFs of the Connect sample points are higher; as soon as the PL increase above 125 dB, the Disconnect users exceed the Connect ones since the radio signal performances decrease. Figure 35 presents the CDFs of the received signal powers of the Connect and Disconnect sample points. It is obvious that the Connect users have higher signals. This figure shows

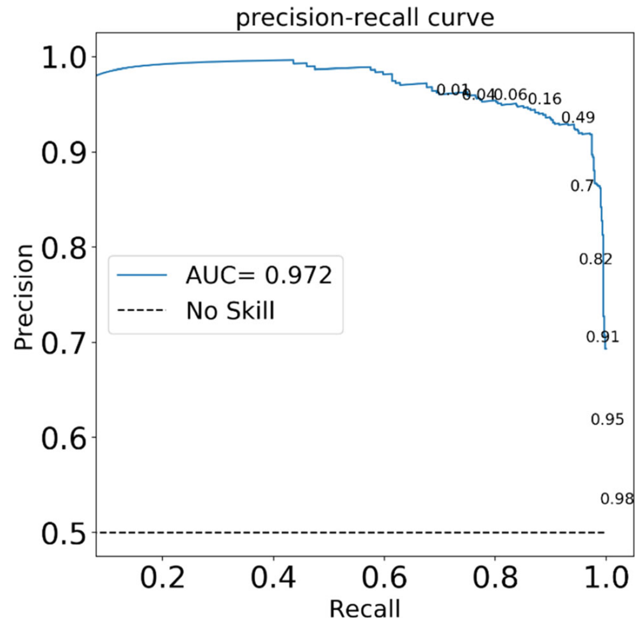


Fig. 28 Precision recall curve for the two-HL ANN used for AP selection

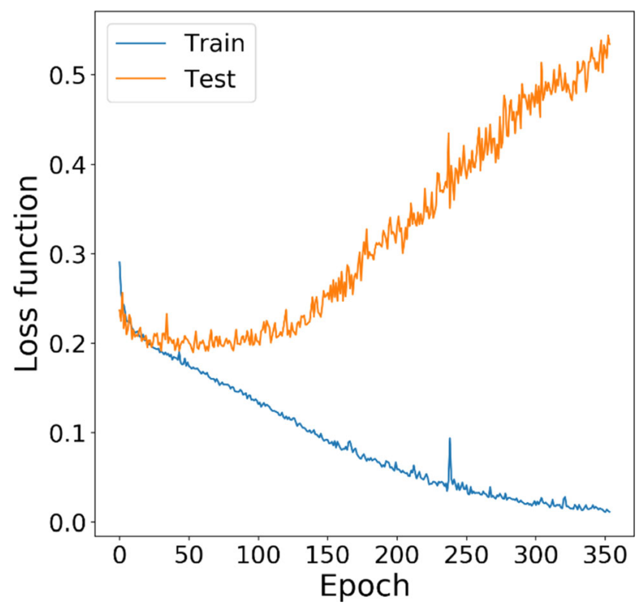


Fig. 29 Loss function for the optimal ANN with two HLs

that the signal power experienced by the Connect users is almost 10 dB higher for 50% of these users. The same conclusion is inferred from Figs. 36 and 37, which proves that the SINR and data rate synchronization of connected users are almost 15 dB and 40 Mbps higher than those for disconnect ones. Figure 38 presents the client connection quality (CCQ) comparison between the connected and disconnected users. For CCQs above 85%, the connected users outnumber the disconnected ones. Figure 39 compares the CDFs of the link distances between the Connect

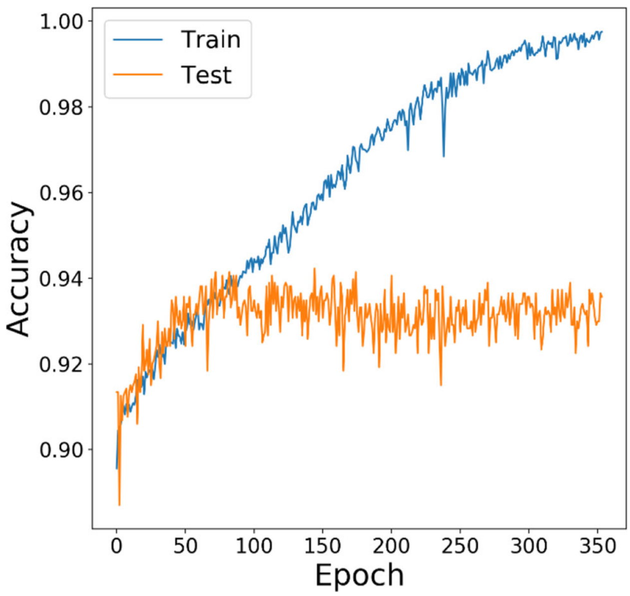


Fig. 30 Accuracy for the optimal ANN with two HLs

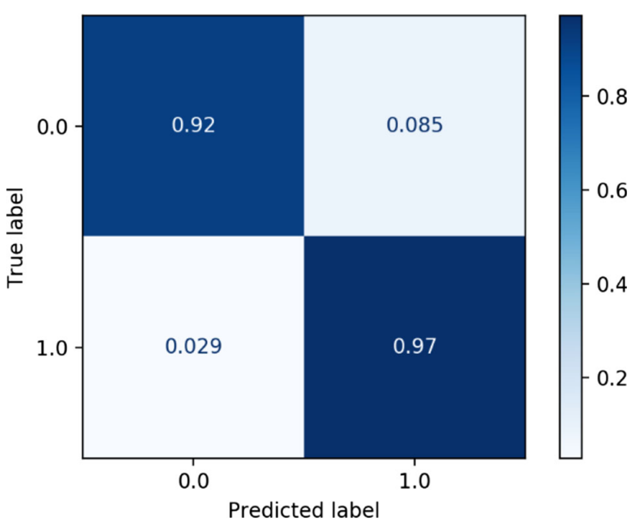


Fig. 31 Normalized confusion matrix for the two-HL ANN

and Disconnect sample points. For radio links greater than 7 km, disconnected users outnumber connected ones. For shorter links, connected users are greater in number since the PL attenuation is lower.

Figuers 40, 41, 42, 43, 44, 45, 46, 47, 48, 49 presents the comparison between prediction success and failure for other pertinent parameters. Note that the predictions are close to the measurements for all listed parameters. These parameters are respectively: allocated throughput per user, frequency band, LOS obstruction, the number of streams for AP and CPE equipment, the number of users connected to an AP, AP height, the numbers of calls to support services as well as the number of level 1, level 2 and maintenance requests per user.

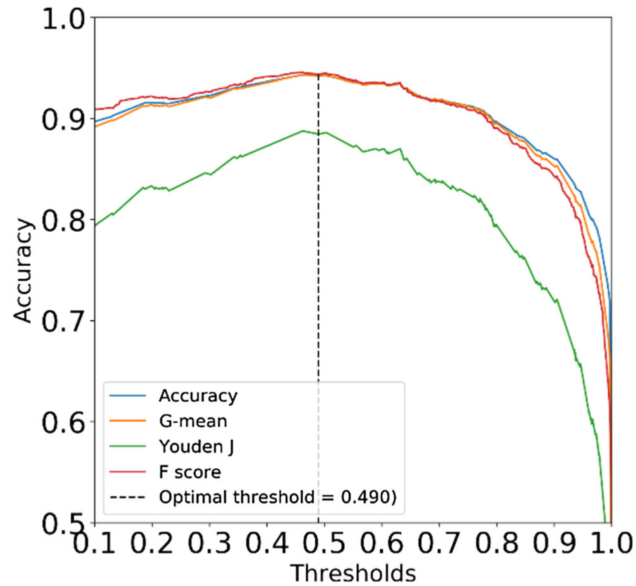


Fig. 32 Threshold tuning

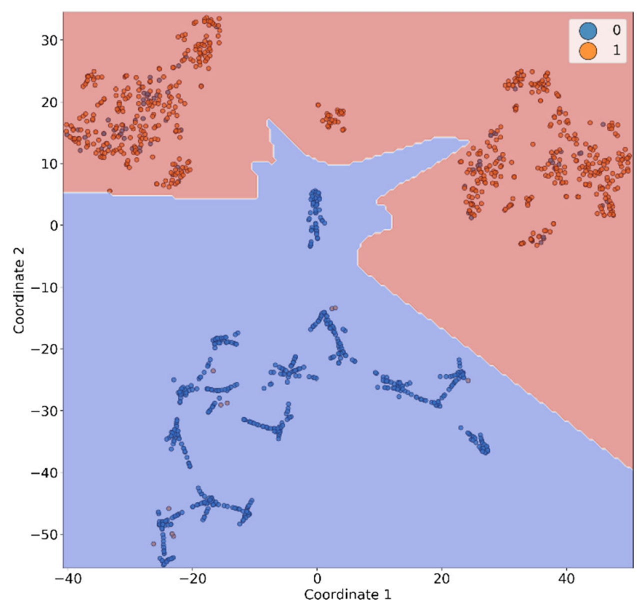


Fig. 33 TSNE representation of the test dataset and classifier contour for the two-HL neural network

6 Conclusion

In this paper, we were inspired by UA techniques to predict the users' subscription decisions to prevent useless operations and avoid wasting resources in FWN. The main idea involves integrating and mixing users' feedback with well-known radio parameters in addition to many other pertinent parameters through ML algorithms to improve the prediction confidence. We also propose two UA algorithms to control the access to FWNs. The first one uses geographical proximity in the nearest neighbor algorithm, while the

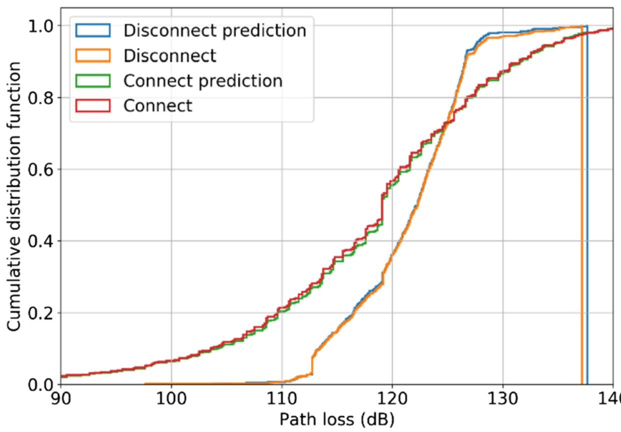


Fig. 34 The CDFs of the path losses for the Connect and Disconnect sample points; predictions are compared to measurements

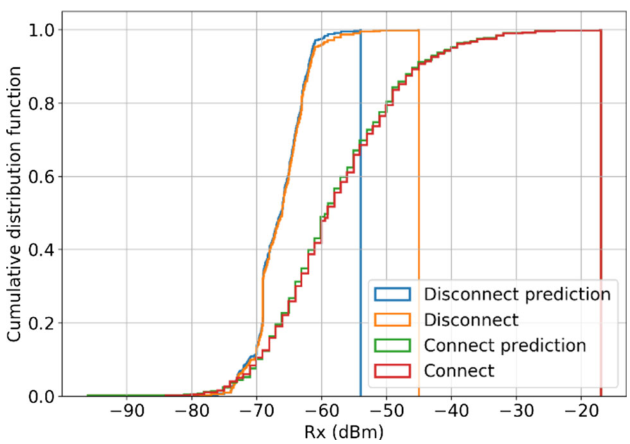


Fig. 35 The CDFs for received signal power for the Connect and Disconnect sample points; predictions are compared to measurements

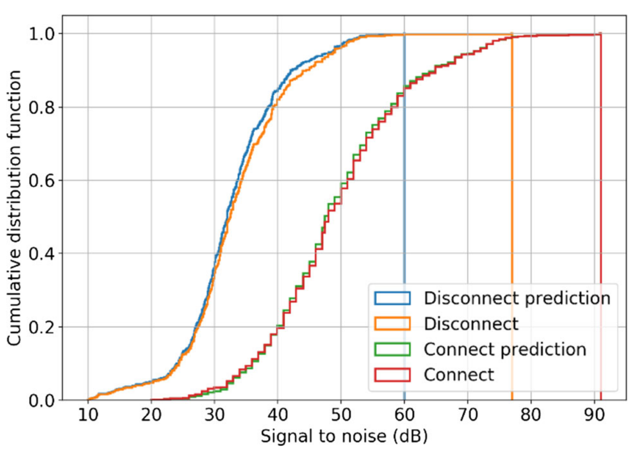


Fig. 36 The CDFs of the signal-to-noise ratios for the Connect and Disconnect sample points; predictions are compared to measurements

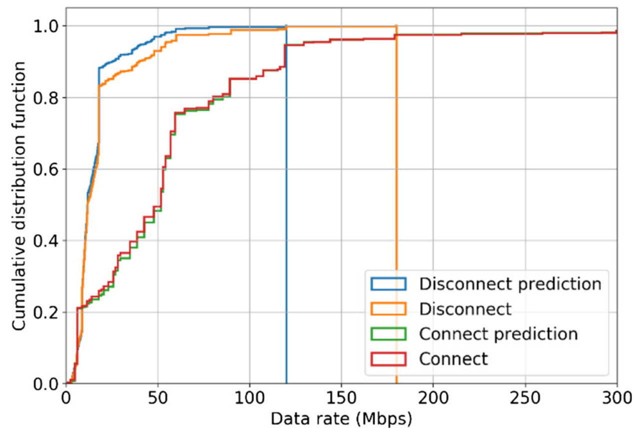


Fig. 37 The CDFs of the data rates for the Connect and Disconnect sample points; predictions are compared to measurements

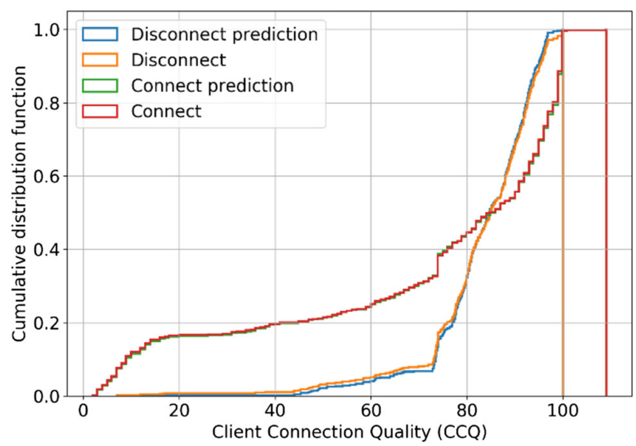


Fig. 38 The CDFs for client connection quality for the Connect and Disconnect sample points; predictions are compared to measurements

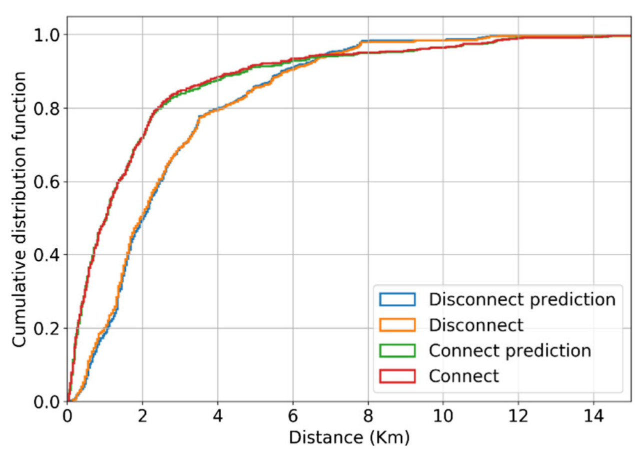


Fig. 39 The CDFs of the radio link distances for the Connect and Disconnect sample points; predictions are compared to measurements

second one, deep nearest neighbor, extracts the distance function with an ANN. Finally, we propose an AP selection technique based on deep imitation learning. The target is to predict whether a given location can be successfully added to the FWN through a given AP while maintaining QoS. Most resource allocation techniques are hard optimization

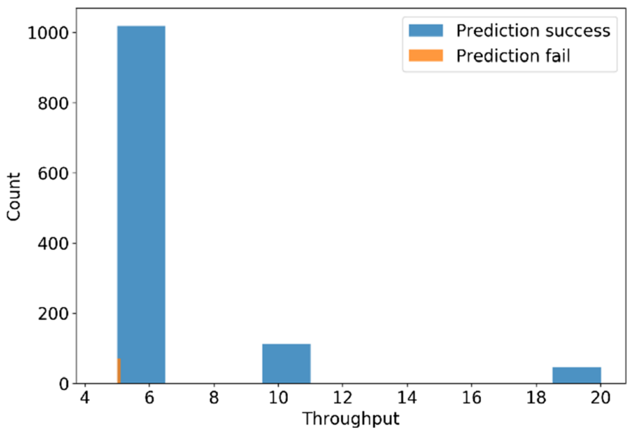


Fig. 40 Comparison between prediction success and failure according to the allocated throughput for each user

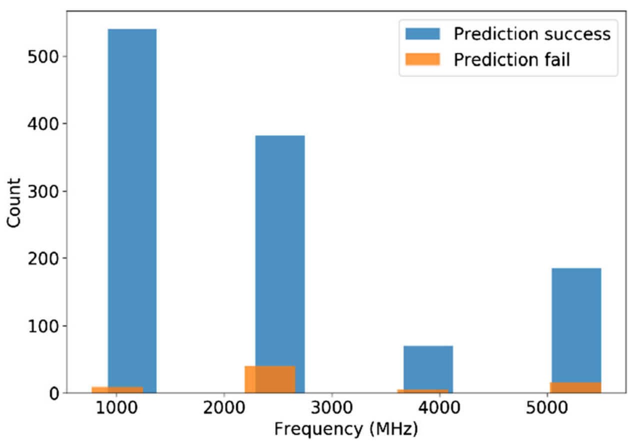


Fig. 41 Comparison between prediction success and failure according to the frequency band

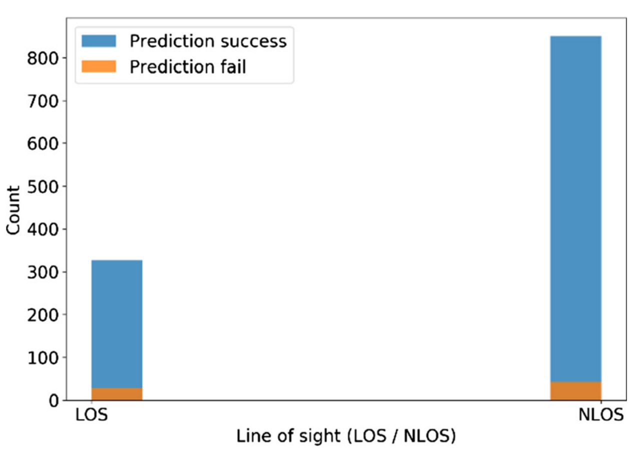


Fig. 42 Comparison between prediction success and failure according to the line-of-sight obstruction

problems covering single scenarios at a time. The proposed technique can cover many practical use cases with good simplicity and computing capability at an accuracy of 0.94.

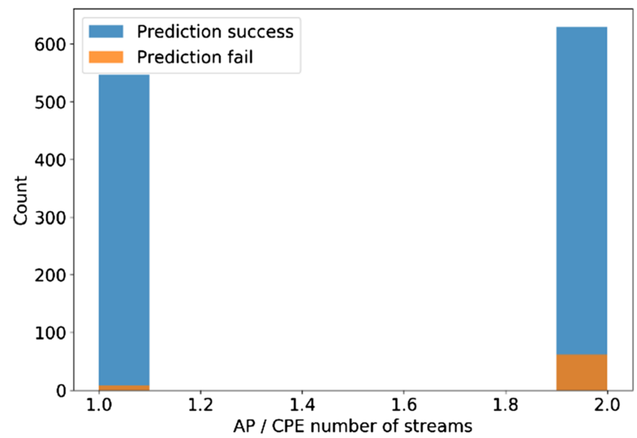


Fig. 43 Comparison between prediction success and failure according to the number of streams of AP and CPE equipment

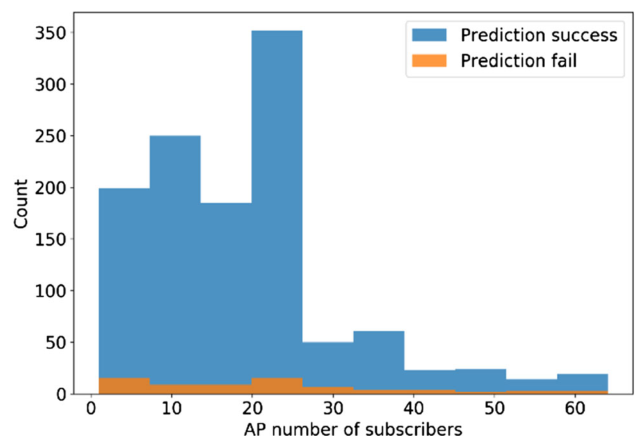


Fig. 44 Comparison between prediction success and failure according to the number of users connected to an AP

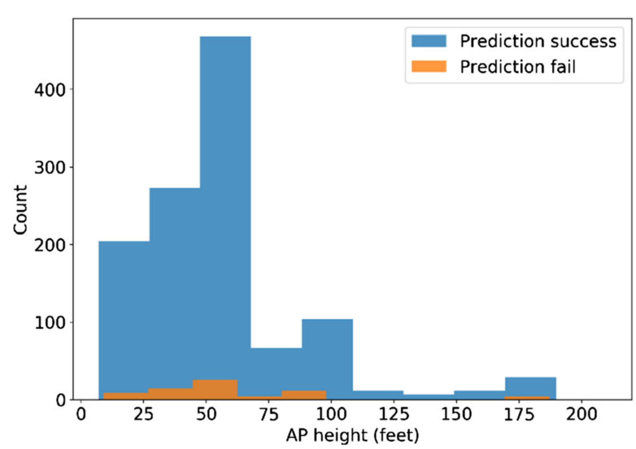


Fig. 45 Comparison between prediction success and failure according to AP height

Using this technique, the boundaries between connected and disconnected users are extracted according to a wide

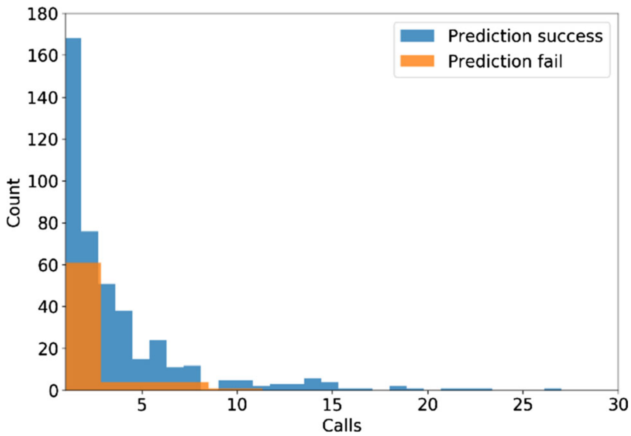


Fig. 46 Comparison between prediction success and failure according to the number of user calls to support services

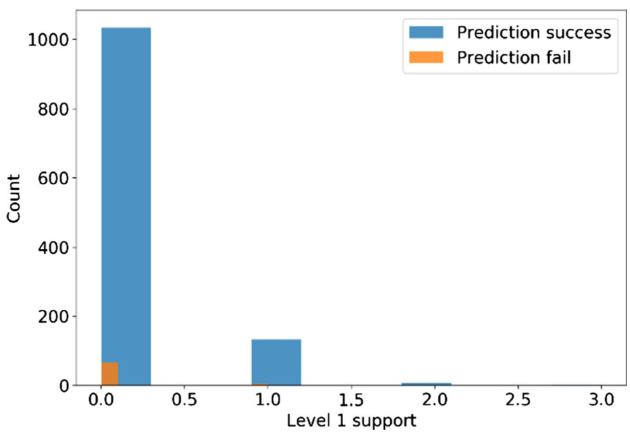


Fig. 47 Comparison between prediction success and failure according to the number of level 1 support requests from a user

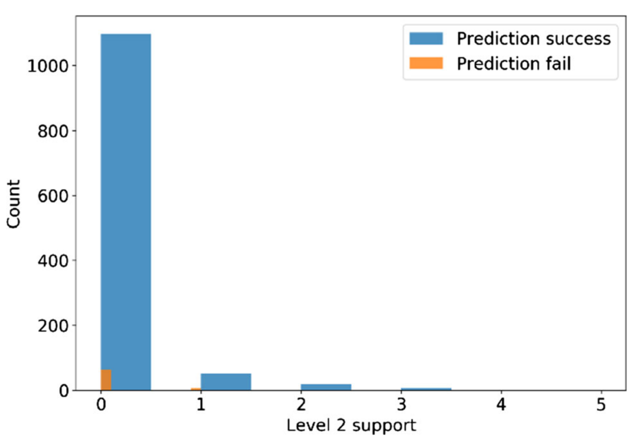


Fig. 48 Comparison between prediction success and failure according to the number of level 2 support requests from a user

range of network, service and environmental parameters and many performance indicators.

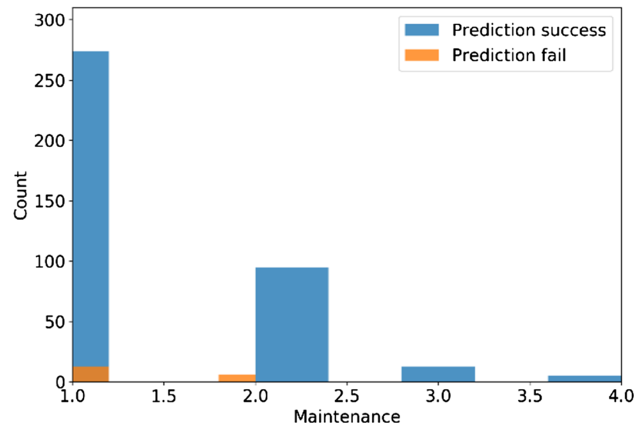


Fig. 49 Comparison between prediction success and failure according to the number of maintenance operations performed for a user

Acknowledgements We would like to thank Digicom Technologies Inc. for providing the datasets.

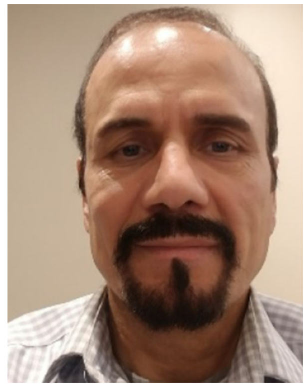
References

1. Sun, Y., Peng, M., Y. Zhou, Huang, Y. & Mao S. (Fourth quarter 2019). Application of machine learning in wireless networks: Key techniques and open issues, *IEEE Communications Surveys and Tutorials, 21*(4), 3072–3108.
2. El Khaled, Z. & Mcheick, H. (2019). Case studies of communications systems during harsh environments: A review of approaches, weaknesses, and limitations to improve quality of service. *International Journal of Distributed Sensor Networks, 15* (2).
3. El Khaled, Z., Mcheick, H., & Ajami, H. (2016). Wide range WiFi network formal design model for ubiquitous emergency events, *In Proc. of Int. Conf. on Future Nets. and Comm.*, Montreal, Canada.
4. Bell Canada (2021). https://www.bell.ca/Bell_Internet/promotions/wireless-home-internet.
5. Mishra, A. (2018). Radio network planning and optimization. In *Fundamentals of network planning and optimization: 2G/3G/4G... Evolution to 5G*, Wiley, 2nd ed.
6. Zappone, A., Di Renzo, M., & Debbah, M. (2019). Wireless networks design in the era of deep learning: Model-Based, AI-Based, or Both? *IEEE Transactions on Communications, 67*(10), 7331–7376.
7. Maghsudi, S., & Hossain, E. (2017). Distributed UA in energy harvesting dense small cell networks: A mean-field multi-armed bandit approach. *IEEE Access, 5*, 3513–3523.
8. Li, Z., Wang, C., & Jiang, C. J. (2017). UA for load balancing in vehicular networks: An online reinforcement learning approach. *IEEE Transactions on Intelligent Transportation Systems, 18*(8), 2217–2228.
9. Zhao, N., Liang, Y., Niyato, D., Pei, Y., Wu, M., & Jiang, Y. (2019). Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks. *IEEE Transactions on Wireless Communications, 18*(11), 5141–5152. https://doi.org/10.1109/TWC.2019.2933417
10. Kim, H., de Veciana, G., Yang, X., & Venkatachalam, M. (2012). Distributed α-Optimal UA and cell load balancing in wireless networks. *IEEE/ACM Transactions on Networking, 20*(1), 177–190.

11. Attiah, M. L., Isa, A. A. M., Zakaria, Z., et al. (2020). A survey of mmWave UA mechanisms and spectrum sharing approaches: An overview, open issues and challenges, future research trends. *Wireless Networks, 26*, 2487–2514.
12. Carrascosa, M., Bellalta, B. Multi-Armed Bandits for Decentralized AP selection in Enterprise WLANs, *arXiv* preprint arXiv:2001.00392
13. Tan, J., Xiao, S., Han, S., Liang, Y., & Leung, V. C. M. (2019). QoS-Aware UA and resource allocation in LAA-LTE/WiFi coexistence systems. *IEEE Transactions on Wireless Communications, 18*(4), 2415–2430.
14. Xu, C., Wang, J., Zhu, Z., & Niyato, D. (2019). Energy-efficient WLANs with resource and re-association scheduling optimization. *IEEE Transactions on Network and Service Management, 16*(2), 563–577.
15. Kafi, M. A., Mouradian, A., & Vèque, V.: Online Client Association Scheme Based on Reinforcement Learning for WLAN Networks" *In Proc. of IEEE Wireless Comm. and Net. Conf. (WCNC)*, Marrakesh, Morocco, 2019, pp. 1–7.
16. Xu, W., Hua, C., & Huang, A. (2011). Channel Assignment and UA Game in Dense 802.11 Wireless Networks, *In proc. of IEEE Int. Conf. on Comm. (ICC)*, Kyoto, pp. 1–5
17. Klaine, P. V., Jaber, M., Souza, R. D., & Imran, M. A. (2019). Backhaul aware user-specific cell association using Q-learning. *IEEE Transactions on Wireless Communications, 18*(7), 3528–3541.
18. Ha, D. T. (2019). Optimized resource allocation and user/cell association for future dense networks, *Networking and Internet Architecture [cs.NI]*, Université Paris-Saclay. English. NNT: 2019SACLS290, Tel-02341472
19. Lyu, F., Fang, L., Xue, G., Xue, H., & Li, M. (2019). Large-scale full wifi coverage: Deployment and management strategy based on user Spatio-Temporal association analytics. *IEEE Internet of Things Journal, 6*(6), 9386–9398.
20. Kim, S., Lee, K., Kim, Y., Shin, J., Shin, S., & Chong, S. (2020). Dynamic control for on-demand interference-managed WLAN infrastructures. *IEEE/ACM Transactions on Networking, 28*(1), 84–97.
21. Dwijaksara, M. H., Jeon, W. S., & Jeong, D. G. (2019). UA for load balancing and energy saving in enterprise WLANs. *IEEE Systems Journal, 13*(3), 2700–2711.
22. Anany, M., Elmesalawy, M. M., & Abd El-Haleem, A. M. (2019). Matching game-based cell association in Multi-RAT HetNet considering device requirements, In *IEEE Internet of Things Journal*, *6*(6), 9774–9782
23. Elmosilhy, N. A., Elmesalawy, M. M., & Abd Elhaleem, A. M. (2019). UA With Mode Selection in LWA-Based Multi-RAT HetNet, In *IEEE Access*, *7*, 158623–158633.
24. Bojovic, B., Baldo, N., Nin-Guerrero, J., Dini, P. (2011). A supervised learning approach to cognitive access point selection, *In Proceedings of IEEE GLOBECOM Workshops (GC Wkshps)*, Houston, TX, pp. 1100–1105
25. Tang, H., Yang, L., Dong, J., et al. (2016). Throughput optimization via association control in wireless LANs. *Mobile Networks and Applications, 21*, 453–466.
26. Carrascosa, M., & Bellalta, B. (2019). Decentralized APS using Multi-Armed Bandits: Opportunistic ε-Greedy with Stickiness, *arXiv*, arXiv:1903.00281
27. Raschellà, A., Bouhafs, F., Mackay, M., Shi, Q., Ortin, J., Gallego, J. R., Canales, M. (2020). A dynamic access point allocation algorithm for dense wireless lans using potential game. *Computer Networks*, *167*.
28. Zheng, R., Hua, C. (2016). Channel Selection and UA in WiFi Networks. In: Sequential Learning and Decision-Making in Wireless Resource Management. *Wireless Networks*. Springer
29. Khalili, A., Akhlaghi, S., Tabassum, H., & Ng, D. W. K. (2020). Joint user association and resource allocation in the uplink of heterogeneous networks. *IEEE Wireless Communications Letters, 9*(6), 804–808. https://doi.org/10.1109/LWC.2020.2970696
30. Chaieb, C., Mlika, Z., Abdelkefi, F. & Ajib, W. On the optimization of user association and resource allocation in HetNets with mm-wave base stations, In *IEEE Systems Journal*, https://doi.org/10.1109/JSYST.2020.2984596
31. Zhang, H., Zhang, H., Long, K., & Karagiannidis, G. Deep learning based radio resource management in NOMA Networks: User Association, Subchannel and Power Allocation, In IEEE Transactions on Network Science and Engineering, https://doi.org/10.1109/TNSE.2020.3004333
32. Huang, X., Xu, W., Xie, G., Jin, S., & You, X. (2018). Learning oriented cross-entropy approach to user association in load-balanced HetNet. *IEEE Wireless Communications Letters, 7*(6), 1014–1017. https://doi.org/10.1109/LWC.2018.2846610
33. El Khaled, Z., Mcheick, H., & Petrillo, F., (2019). Wifi coverage range characterization for smart space applications. In Proceedings of the 1st International Workshop on Software Engineering Research and Practices for the Internet of Things (SERP4IoT'19). IEEE Press, 61–68. https://doi.org/10.1109/SERP4IoT.2019.00018.
34. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.
35. Khaled, Z. E., Ajib, W., & Mcheick, H. (2020). An accurate empirical path loss model for heterogeneous fixed wireless networks below 5.8 GHz frequencies. *IEEE Access, 8*, 182755–182775. https://doi.org/10.1109/ACCESS.2020.3023141
36. Wolpert, D. H., & Macready, W. G. (2005). Coevolutionary free lunches. *IEEE Transactions on Evolutionary Computation, 9*(6), 721–735.
37. Cover, T. M., & Hart, P. E. (1967). Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory, IT-13*, 21–27.
38. Walters-Williams, J., & Li, Y. (2010). Comparative study of distance functions for nearest neighbors, Advanced Techniques in Computing Sciences and Soft. Engineering, K. Elleithy, Springer, pp. 79–84
39. Bhatia, N. Survey of nearest neighbor techniques, arXiv 2010, arXiv:1007.0085.
40. Choudhury, T., Quigley, A., Strang, T., Suginuma, K. (2009). Location and Context Awareness in proceedings of International Symposium, LoCA 2009, Tokyo, Japan.
41. Gagn, C., & Parizeau, M. (2007). Coevolution of nearest neighbor classifiers. *International Journal of Pattern Recognition and Artificial Intelligence, 21*(5), 921–946.
42. Gou, J., Du, L., Zhang, Y., & Xiaong, T. (2012). A new distance-weighted k-nearest neighbor classifier. *Journal of Information and Computational Science, 9*(6), 1429–1436.
43. Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks, 2*, 359–366.
44. You, K., Long, M., Jordan, M. I., How does learning rate decay help modern neural networks, arXiv:1908.01878
45. Zhang, N., Wei, Wu., & Zheng, G. (2006). Convergence of gradient method with momentum for two-Layer feedforward neural networks. *IEEE Transactions on Neural Networks, 17*(2), 522–525. https://doi.org/10.1109/TNN.2005.863460
46. Polycarpou, M. M., & Ioannou, P. A. (1992). Learning and convergence analysis of neural-type structured networks. *IEEE*

Transactions on Neural Networks, 3(1), 39–50. https://doi.org/10.1109/72.105416
47. Bienstock, D., Muñoz and Sebastian Pokutta, G. (2018). "Principled Deep Neural Network Training through Linear Programming." ArXiv abs/1810.03218 n. pag.
48. Mismar, F. B., & Evans, B. L. (2019). Deep learning in downlink coordinated multipoint in new radio heterogeneous networks. *IEEE Wireless Communications Letters, 8*(4), 1040–1043. https://doi.org/10.1109/LWC.2019.2904686
49. Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford University Press.
50. Sontag, E. D. (1992). Feedback stabilization using two-hidden-layer nets. *IEEE Transactions on Neural Networks, 3*(6), 981–990.
51. Zhang T. et al. (2018). Deep Imitation Learning for Complex Manipulation Tasks From Virtual Reality Teleoperation, Proceedings 2018 IEEE Int'l. Conference Robotics and Automation, pp. 1–8.
52. Lee, M., Yu, G., Li, G. Y. (2019). Accelerating resource allocation for D2D communications using imitation learning, 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall), Honolulu, HI, USA, pp. 1–5, https://doi.org/10.1109/VTCFall.2019.8891075.
53. Yu, S., Chen, X., Yang, L., Wu, D., Bennis, M., & Zhang, J. (2020). Intelligent edge: Leveraging deep imitation learning for mobile edge computation offloading. *IEEE Wireless Communications, 27*(1), 92–99. https://doi.org/10.1109/MWC.001.1900232
54. van der Maaten, L. J. P., & Hinton, G. E. (2008). Visualizing high-dimensional data using t-SNE. *Journal of Machine Learning Research, 9*, 2579–2605.
55. Last, E., Douzas, G., Bacao, F., "Oversampling for Imbalanced Learning Based on K-Means and SMOTE" https://arxiv.org/abs/1711.00

Zayan El Khaled received an Engineer diploma from Ecole Nationale Supérieure d' Ingénieurs de Limoges, France in 2007 in electronics and telecommunications with specialization in embedded systems and the M.S. degree in electronics, optics and telecommunications from the University of Limoges, France, 2007. He then gets his Ph.D. Degree in Computer Sciences and Computer Networks from the University of Quebec in Chicoutimi, Canada in 2021. From 2007 to 2016 he occupied many industrial positions such as research and development engineer, IT and wireless networks consultant, wireless network administrator. His research interests include ubiquitous computing, wireless networks and communications, wireless sensor networks, IoT, the application of AI in wireless networks. He is also a senior engineer in the order of engineers of Quebec Canada.



Hamid Mcheick is a full professor in Computer Science department at the University of Québec at Chicoutimi, Canada. He has more than 20 years of experience in both academic and industrial area. He has done his Ph.D. in Software Engineering and Distributed System in the University of Montreal, Canada. He is working on design and adaptation of distributed and smart software applications. He has supervised many post-doctorate, Ph.D., master and bachelor students. He has nine book chapters, more than 50 research papers in international journals and more than 130 research papers in international/national conference and workshop proceedings in his credit. Dr. Mcheick has given many keynote speeches and tutorials in his research area, particularly in Healthcare systems, Pervasive and Ubiquitous computing, Distributed Middleware Architectures, Software Connectors, Service Oriented Computing, Internet of Things (IoT), Mobile Edge Computing, Fog Computing, and Cloud Computing. Dr. Mcheick has gotten many grants from governments, industrials and academics. He is a chief in editor, chair, co-chair, reviewer, member in many organizations (such as IEEE, ACM, Springer, Elsevier, Inderscience) around the world.

Wessam Ajib received an Engineer diploma from Institut National Polytechnique de Grenoble, France (INPG-ENSPG) in Physical Instruments, in 1996, a DEA (Diplôme d'Études Approfondies) from École Nationale Supérieure des Télécommunication (ENST), Paris, France in 1997 in Digital Communication Systems, and a Ph.D. degree in Computer Sciences and Computer Networks from ENST, Paris, France, in 2000. He had been an architect and radio network designer at Nortel Networks, Ottawa, ON, Canada between October 2000 and June 2004. He had conducted many projects and introduced different innovative solutions for UMTS system, the third generation of wireless cellular networks. He had a post-doc fellowship at Electrical Engineering department of École Polytechnique de Montréal, QC, Canada between June 2004 and June 2005. Since June 2005, He had been with the Department of Computer Sciences, Université du Québec à Montréal, QC, Canada, where he was promoted in june 2014 to become a full professor of computer networks. His research interests include wireless communications and wireless networks, multiple access and MAC design, traffic scheduling error control coding, MIMO systems, cooperative communications and cognitive radio networks. He is the author or co-author of many journal papers and conferences papers in these areas.