Article

NetSel-RF: A Model for Network Selection Based on Multi-Criteria and Supervised Learning

Daniela Alexandra Embus [1], Andres Julián Castillo [1], Fulvio Yesid Vivas [1], Oscar Mauricio Caicedo [1,*,†] and Armando Ordóñez [1,2,†]

[1]   Departamento de Telemática, Universidad del Cauca, Popayán, Cauca 190002, Colombia;
      dembus@unicauca.edu.co (D.A.E.); ajcastillo@unicauca.edu.co (A.J.C.); fyvivas@unicauca.edu.co (F.Y.V.);
      jaordonez@unicauca.edu.co (A.O.)
[2]   Intelligent Management Systems Group, Fundación Universitaria de Popayán, Popayàn,
      Cauca 190002, Colombia
*   Correspondence: omcaicedo@unicauca.edu.co; Tel.: +57-(2)-8209900 (ext. 2127)
†   These authors contributed equally to this work.

check for updates

Abstract: Network selection plays a pivotal role in ensuring efficient handover management. Some existing approaches for network selection may use one criterion, such as RSSI (Received Signal Strength Indicator) or SINR (Signal to Interference Noise Ratio). However, these approaches are reactive and may lead to incorrect decisions due to the limited information. Other multi-criteria-based approaches use techniques, such as statistical mathematics, heuristics methods, and neural networks, to optimize the network selection. However, these approaches have shortcomings related to their computational complexity and the unnecessary and frequent handovers. This paper introduces NetSel-RF, a multi-criteria model, based on supervised learning, for network selection in WiFi networks. Here, we describe the created dataset, the data preparation and the evaluation of diverse supervised learning techniques (Random Forest, Support Vector Machine, Adaptive Random Forest, Hoeffding Adaptive Tree, and Hoedding Tree techniques). Our evaluation results show that Random Forest outperforms other algorithms in terms of its accuracy and Matthews correlation coefficient. Additionally, NetSel-RF performs better than the Signal Strong First approach and behaves similarly to the Analytic Hierarchy Process–Technique for Order Preferences by Similarity to the Ideal Solution (AHP-TOPSIS) approach in terms of the number of handovers and throughput drops. Unlike the latter, NetSel-RF is proactive and therefore is more efficient regarding Quality of Services (QoS) and Quality of Experience (QoE) since the end-devices perform the handover before the network link quality degrades.

Keywords: network selection; multi-criteria; random forest; AHP-TOPSIS; strongest signal first

1. Introduction

Handover Management (HM) plays a fundamental role in wireless communications. HM is the process by which a mobile device maintains an active connection when the user roams from the coverage area of one network to another [1]. HM comprises three phases [2,3]: initiation, selection, and execution. Handover Initiation collects all information required to identify and determine the neighbor networks, their parameters, and their available services. Network Selection chooses the best available network by taking into account diverse parameters and evaluation metrics. Handover Execution establishes the connection and releases resources. This paper focuses on the selection phase, which is crucial to ensuring service continuity, providing Quality of Services (QoS), and satisfying the Quality of Experience (QoE) [4,5].

Traditional approaches for Network Selection use a single criterion to decide the next network (e.g., Access Point (AP) or Base Station) to which devices must connect as they move [6]. The Signal Strong First (SSF) is the most widely used traditional approach, which performs the selection by comparing the RSSI of the current network with available adjacent networks [1]. Other approaches based on a single criterion use the available bandwidth [7], load in the network [8], or Signal to Interference Noise Ratio (SINR) [9] to select the network. As these approaches only use one criterion, they ignore important information, leading to throughput drops, unnecessary handovers, and even service disruption [6,10,11].

Other approaches use multiple criteria to address the shortcomings mentioned above. The selection criteria can be classified into network, user, device, and application criteria [12]. Network criteria include parameters that describe the wireless network characteristics (e.g., coverage area and Received Signal Strength Indicator (RSSI)). User criteria comprise parameters related to user preferences. Device criteria encompass parameters that provide information about the mobile device, such as battery consumption, and location. Application criteria cover parameters related to the QoS requirements of real or non-real time applications, such as the bandwidth consumed by such applications. Some multi-criteria approaches (also known as Multi-Attribute Decision-Making (MADM)) include the Simple Additive Weighting (SAW), Gray Relational Model, and the Technique for Order Preferences by Similarity to the Ideal Solution (TOPSIS) [4,13]. These approaches have some shortcomings related to skipping diverse selection criteria, which results in incorrect network selection [6]. Other relevant multi-criteria approaches use techniques such as mathematical models [14], Neural Networks (NN) [12,15], Fuzzy Logic (FL) [16,17], and Game Theory [6] to perform network selection. However, these techniques can decrease the throughput during the handover [18]. Hybrid solutions combine multi-criteria approaches [19]; however, their major problem is their high computational complexity.

In this paper, we introduce NetSel-RF: a multi-criteria model based on supervised learning which is intended to overcome the shortcomings mentioned above and achieve efficient network selection in WiFi networks. To create this model, we constructed a dataset, performed data preparation, carried out feature selection, and applied different supervised Machine Learning (ML) techniques. Our evaluation results reveal several facts. First, to build up NetSel-RF, the Random Forest (RF) technique [20] performs better than the Hoedding Tree (HT) [20], Adaptive Random Forest (ARF) [21], Hoeffding Adaptive Tree (HAT) [21], Support Vector Machine (SVM) [22] techniques regarding accuracy [23] and the Matthews correlation coefficient (MCC) [24]. Second, NetSel-RF outperforms the SSF approach in terms of the number of handovers, ping-pong, and throughput drops. Third, the proposed method behaves similarly to the AHP-TOPSIS approach regarding the number of handovers and throughput drops. These are as a result of the fact that our model uses criteria coming from the network, user preferences, end-devices, and applications simultaneously. Furthermore, our model is proactive; that is, the mobile selects a new AP while it is still connected to another one.

The remainder of this paper is organized as follows. Section 2 reviews the related work. Section 3 introduces the network selection approach based on multiple criteria and ML. Section 4 presents the efficiency of NetSel-RF in comparison with AHP-TOPSIS and SSF. Finally, Section 5 contains some concluding remarks and plans for future work.

2. Related Work

In SSF, the mobile devices switch to another AP or Base Station (BS) when the RSSI level of the currently associated network is lower than a threshold (e.g., −90 dBm). During this threshold violation, mobile devices select the adjacent AP/BS with a stronger RSSI [25]. Network selection based only on RSSI has some shortcomings [13]—discontinuity of service, unnecessary handovers, and noticeable delays during the handover—which are more evident in dense networks.

Regarding multi-criteria approaches, Khan et al. [16] presented a generic handover management approach that operates in three phases. First, it identifies the appropriate place for the handover

initiation based on the estimated coverage area. Second, it uses FL to eliminate inappropriate networks. Third, it selects an optimal network using the TOPSIS algorithm by considering several parameters such as delay, jitter, bit error rate (BER), packet loss, communication cost, response time, and network load. Sadik et al. [19] proposed a solution to ensure QoS during HM which operates as follows. First, it uses network-related parameters such as SINR in the HM initiation phase; second, it uses AHP-TOPSIS to select the network with the best predicted QoE based on user-related parameters, such as movement direction and the Mean Opinion Score (MOS), as well as device-related parameters such as the battery level.

Yu et al. [13] introduced an approach that combines network (i.e., available bandwidth, latency, delay jitter, packet loss ratio, and BER) and user parameters (i.e., service price and user preferences). This approach uses FAHP, entropy, and TOPSIS to calculate a utility value for all candidate networks using the parameters mentioned above. A network with a utility value higher than the current network is selected. Goyal et al. [17] proposed a network selection approach considering several parameters such as RSSI, velocity, bandwidth, load on evolved node B (eNB), power transmission, dwell time, and cell radius. The handover procedure is initiated when the RSSI of the current network drops below a predefined threshold, and the dwell time is higher than the handover time. This approach selects the new best AP/BS by an objective function computed by the Fuzzy Analytical Hierarchy Process.

Goyal et al. [14] introduced an approach that, through simple mathematical formulation, gives dynamic and realistic weights to parameters, such as bandwidth, distance, velocity, power consumption, and network usage charges. This approach uses the dynamic weights to determine the best AP/BS to connect the mobile device. Alotaibi et al. [12] present an NN-based approach for HM, which includes three elements: the Vertical Handover Manager handles the handoff requests carried out by mobile devices; the Attributes Collector recovers network, device, and user criteria to feed an NN; and the Artificial Network Selector uses this NN to choose the best available wireless network. Ali et al. [15] proposed an approach that uses a Feed-Forward NN (FNN) with two levels to select the best AP/BS. This approach gathers parameters from the network parameter and QoE from eNB and trains its FNN by using them and considering the experience gained from past handover decisions.

Table 1 lists some shortcomings of the multi-criteria approaches. First, none of the methods mentioned above use network, device, applications, and user criteria jointly. Skipping parameters may affect the network selection, which can lead to unstable connection, unnecessary device resource consumption, and unneeded handovers. Second, approaches based on NN and FNN have a high computational complexity which can increase the time of network selection, thus decreasing throughput during the handover. Third, MADM approaches suffer when determining the most suitable weights for the criteria, which can affect network selection. In summary, the network selection in HM is still an open issue since multi-criteria approaches need to be optimized to avoid unnecessary handovers and increase the throughput.

Table 1. Multi-criteria based related work.

Proposal	Criterion				Metric			Complexity
	Network	Devices	User	Application	# of Handovers	Throughput	Type Handover	
[16]	√		√		√	√		High
[19]	√	√	√		√	√		Medium
[13]	√		√		√	√		High
[17]	√	√						Medium
[14]	√	√	√		√			Medium
[12]	√	√	√			√		High
[15]	√		√			√		High
NetSel-RF	√	√	√	√	√	√	√	Medium

3. NetSel-RF Model

This section presents our classification model for optimizing the network selection in a WiFi-based environment. In particular, this section presents the motivation, methodology, interpretation of data, data preparation, and model construction.

3.1. Motivation

Figure 1 shows a user with a mobile device (*dev*1) moving through the hall. This *dev*1 needs to be always connected to the best network. As mentioned above, traditional handover approaches consider only RSSI for making decisions [26]; however, his approach produces frequent disconnection and inefficient seamless handovers, leading to handover operation failures [6].

Several approaches have addressed these problems by using multiple criteria considering the handover decision parameters related to QoS and QoE [18]. Nevertheless, current approaches based on multi-criteria can perform incorrect network selection by skipping one or more criteria related to network, devices, users, or applications parameters. In this paper, we argue that a supervised learning model based on multiple criteria allows the network selection to be optimized by considering the parameters of networks, devices, users, and applications. In this way, such a model can learn the best network to satisfy QoS and QoE. Additionally, supervised learning allows the new network to be selected proactively; the devices perform a handover before the network link quality degrades.
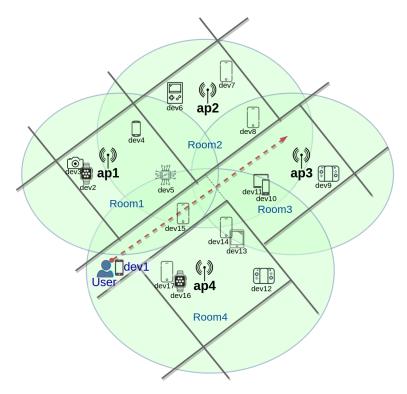

Figure 1. Motivation Scenario.

3.2. Methodology

We followed the Cross Industry Standard Process for Data Mining methodology (CRISP-DM) [27] to build our classification model. The aim was to optimize the network selection by choosing the best AP available in Wi-Fi networks. We carried out three CRISP-DM steps, namely data interpretation, data preparation, and modeling. Data interpretation generates, gathers, and defines the initial dataset that includes the set of data and the set of features related to the selection of the best AP to perform a hand-off. Data preparation covers all activities necessary to construct the final dataset from the

initial dataset; data preparation tasks include attribute selection as well as the transformation and cleaning of data. Modeling refers to the assessment of several supervised ML algorithms to choose the one that best foresees the AP to make the handover decision. The selection metrics are the true positive rate (TPR) [28], false positive rate (FPR) [29], Matthews correlation coefficient (MCC) [24], and time of classification (T_C) [28].

3.3. Data Interpretation

This phase builds an initial dataset with the appropriate features (i.e., parameters) to select the best available AP in a WiFi network. This phase includes the following tasks: first, an experimental scenario is designed and implemented; second, the relevant parameters/attributes to optimize the AP selection are chosen; third, the target variable is defined; fourth, the dataset is filled by capturing the attributes when the devices are moving and have at least two APs in range.

Figure 2 shows the scenario built to obtain the initial dataset. This scenario includes four APs (described in Table 2), together with 24 mobile devices (*devs*). In particular, *dev*1, *dev*2, *dev*3, and *dev*4 perform a circular movement, and *dev*5, *dev*6, *dev*7, and *dev*8, carry out a linear movement. In turn, *dev*9 to *dev*24 perform a random movement. All devices perform the movement in 500 steps at a speed of 1 m/s and communicate at 2462 GHz. The scenario described was deployed by using the Mininet-WiFi [30] emulator in a virtual machine with Ubuntu Server 16.04.

Table 2. Access Point (AP) setup.

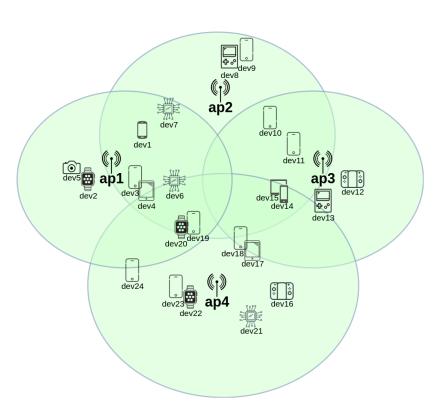| AP	SSID	Position [x,y,z] (m)	Range (m)	Channel	Total_Users_Support
ap1	ap1-ssid	[50.0,150.0,0]	45	1	15
ap2	ap2-ssid	[90.0,180.0,0]	50	7	18
ap3	ap3-ssid	[130.0,150.0,0]	45	11	15
ap4	ap4-ssid	[90.0,90.0,0]	57	11	20

Figure 2. Dataset scenario set-up.

In the second step, we investigated the most used parameters in the literature related to the network selection phase. Table 3 briefly describes the parameters chosen for our initial dataset and classifies them according to their type: network, device, application, and user. In the third task, we defined AP_target as the dependent parameter/attribute (also known as output variable) to predict; this variable is categorical because it allows the prediction of the AP to which the mobile device should connect taking into account the user preference. In the fourth task, we proceeded to fill up the dataset. It is relevant to mention that we gathered the data when mobile devices (*devs*) moved and had at least two candidate APs in-range.

Table 3. Parameter description.

Proposal	Criterion	Parameter	Description
[14,16,17]	Network	AP	Indicates which are the candidate APs that are in range of the mobile device.
		RSSI	Reference scale for measuring the power level of the signals received by a device and determining if the signal is sufficient to get a good wireless connection.
		AP Ocupation	Percentage of users connected to the AP concerning the entire capacity of users support the AP.
[14]	Devices	Distance	Distance between mobile device and candidate AP.
[31]	Device and Application	Battery consumption	Estimated discharge percentage of the device when it is connected to the candidate AP taking into account the applications used by the user and the distance to the AP.
[32]	Application	Power consumption	Battery consumption in the mobile device due to the type of application that users are using in the handover moment.
[13]	User	User Preference	User preference between good signal quality, lower battery consumption or good QoS.

The result of this phase is the initial dataset with five input parameters/attributes and one output parameter. The input parameters are as follows: AP available, RSSI, AP occupation, battery consumption of devices, distance from the device to the AP, and power consumption of applications. The output parameter (dependent variable) is the selected AP. To create this dataset, we leveraged the global view of the network offered by Mininet-WiFi. This emulator provides a global view as it follows the Software-Defined Network concept [33]. With Mininet-WiFi, it is possible to collect parameters from end-devices and APs [34]. Listing 1 shows excerpts of code to illustrate how the aforementioned parameters were collected.

Listing 1: Parameter collection.

```
//AP available
dev.params["apsInRange"]
dev.params["associatedTo"]

//RSSI
ap_dis = dev.get_distance_to(ap_name)
dev_rssi = dev.get_rssi(ap_name,0,ap_dis)

//AP occupation
ap_num_dev = ap_name.params["associatedStations"]
ap_max_dis.params['maxDis']
ap_ocu = ((ap_num_dev*100.)/ap_max_dis)

//battery consumption of devices
dev_app = dev.params.get("app")
ap_dis = dev.get_distance_to(ap_name)
bat_temp = constant_battery_discharge * ap_dis * dev_app

//distance from the device to the AP
dev.get_distance_to(ap_temp)

//power consumption of applications
dev.params["pow"]
```
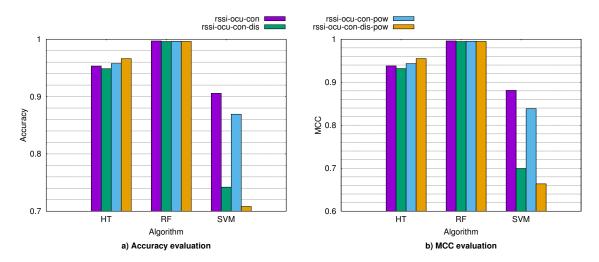
3.4. Data Preparation

This phase analyzes and processes the initial dataset in order to build the final dataset. The data preparation involves the assessment of the quality of the dataset and the reduction the dataset's dimension. Firstly, we captured the data from the Mininet Wi-Fi emulator. We noted that the data for APs in the range of mobile devices were correct in the dataset. For APs which were out of range, there were missing values in the dataset. Therefore, we decided to substitute the missing values with the maximum value of each parameter. In particular, we assigned −100 dBm for RSSI, 100% for AP occupation, and 1% for battery consumption. These maximum values in the dataset represent the non-selection of an AP outside the range of the mobile device.

Secondly, we selected the most relevant features with the aim of reducing the dataset's dimension and the computational complexity. To determine the final dataset features, we applied three ML classification algorithms to all datasets resulting from the combination of all features of the initial dataset. In particular, we used HT, RF, and SVM because the output variable of these methods is categorical. HT is an incremental decision tree learner for large data streams, which assumes that the data distribution is not changing over time [20]. Additionally, HT uses the simple idea that a small sample can often be sufficient to choose an optimal splitting attribute, which is supported mathematically by the Hoeffding bound that quantifies the number of observations needed to estimate some statistics within a prescribed precision [35]. RF consists of a large number of individual decision trees that operate as an ensemble. Each tree predicts a class, and the class with the most votes becomes the prediction [36]. SVM outputs an optimal N-dimensional hyperplane (N—the number of labels) that categorizes the output variable given labeled training data [22].

In this feature selection task, we used accuracy and MCC as metrics because they are typical in classification problems. Accuracy is the ratio between the number of correct predictions made and all predictions made [29]. MCC is a measure of the quality of classifications; it takes into account true and false positives and negatives. MCC is, in essence, a correlation coefficient value between −1 and 1. A coefficient of 1 represents a perfect prediction, 0 an average random prediction, and −1 an inverse prediction [24].

For the sake of readability, Figure 3 depicts only the evaluation results of four tested datasets regarding accuracy and MCC. Dataset 1, in purple, includes RSSI, AP occupation, and battery consumption. Dataset 2, in green, considers RSSI, AP occupation, battery consumption, and distance. Dataset 3, in blue, comprises RSSI, AP occupation, battery consumption, and power consumption. Dataset 4, in orange, includes RSSI, AP occupation, battery consumption, distance, and power consumption. Dataset 1 obtained the best performance regarding accuracy and MCC for all evaluated ML algorithms. Specifically, this dataset obtained the highest accuracy, which was near to 99.7%, and the highest MCC score, of around 0.996; thus, Dataset 1 was selected as the final dataset.

Figure 3. Feature selection; (**a**) Accuracy evaluation, (**b**) MCC evaluation.

Considering the accuracy and MCC results mentioned above, we reduced the dataset's dimension (i.e., resulting in lower computational complexity) from five to four input features; the final dataset (Dataset 1) includes the output variable AP_target and the input features: AP available, RSSI, AP occupation, and battery consumption. Note that the final dataset does not include the distance feature (no relevant); this may be due to its relationship with battery consumption. We filled the final dataset by capturing the features/parameters when the device is in motion and has at least two APs in range; note that we assigned the maximum values to the missing data. Our final dataset comprises a total of 10, 500 instances (see Table 4) [37].

Table 4. Excerpt of final dataset.

	station	ap1	rssi1	ocu1	con1	ap2	rssi2	ocu2	con2	ap3	rssi3	ocu3	con3	ap4	rssi4	ocu4	con4	AP_Target
1	sta1	0	−100	100	1.00	1	−69.0	66.67	0.031	1	−72.0	46.67	0.031	0	−100	100	1.00	ap3
2	sta2	0	−100	100	1.00	1	−55.0	55.56	0.010	1	−76.0	60.00	0.041	0	−100	100	1.00	ap2
3	sta3	0	−100	100	1.00	0	−100	100	1.00	1	−69.0	53.33	0.024	1	−76.00	45.00	0.052	ap3
4	sta4	0	−100	100	1.00	1	−74.0	66.67	0.046	1	−66.0	60.00	0.018	1	−77.00	40.00	0.056	ap3
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
10500	sta24	0	−100.0	100.0	1.0	1	−72.0	66.67	0.0376	1	−64.0	46.67	0.0169	0	−100.0	100.0	1.0	ap3

3.5. Modeling

We evaluated five classification algorithms, namely the three used in the features selection plus ARF and HAT. ARF is an adaptation of RF, which includes mechanisms to adapt to different kinds of concept drifts, given the same hyper-parameters. HAT adaptively learns from data streams that change over time without needing a fixed size of sliding window. The optimal size of the sliding window is a complicated parameter to guess for users since it depends on the rate of change of the distribution of the dataset [21]. We used these five algorithms because the target variable is categorical.

As evaluation metrics, we used the two used in the feature selection plus TPR, FPR, precision and time of classification (T_C). TPR is the number of class members classified correctly over the total number of class members (i.e., the label AP1 is used when AP1 is the correct selection for handover). In contrast to TPR, FPR is the number of class members classified incorrectly over the total number of class members (i.e., label the AP1 is used when AP3 is the correct selection for handover) [28]. The precision is the number of class members classified correctly over the total number of instances classified as class members [29]. T_C is the time spent by an algorithm to classify the selected AP [28]. Furthermore, we used cross-validation, which consisted of randomly dividing the final dataset into two parts: 80% for training and 20% for validation. As a result of the modeling phase, we obtained the classification model, called NetSel-RF.

Figure 4 shows the evaluation results regarding accuracy. These results reveal that RF and ARF obtain higher accuracy than the other evaluated algorithms. In particular, RF and ARF reach an accuracy of approximately 99.7% and 99.57%, respectively. These results are due to RF and ARF being able to handle large amounts of data with higher dimensionality; additionally, RF and ARF require less cleaning and pre-processing of data compared to other learning methods. Figure 4 depicts the evaluation results regarding MCC. These results reveal that RF and ARF also obtain a higher MCC than SVM, HT, and HAT. In particular, RF and ARF reach an MCC of approximately 0.996 and 0.994, respectively. These results are because tree-based algorithms are efficient at finding complex correlations. Therefore, RF and ARF provide a high correlation between real and predicted values for HM.

We also evaluated HT, HAT, RF, ARF, and SVM by using the confusion matrix, which is a fundamental tool to evaluate the performance of classification algorithms; this matrix allows us to determine quickly if the model is confusing different classes. In this matrix, each column represents the number of predictions per class, while each row represents the instances in the real class. Here, TPR and FRP represent the proportion of APs selected correctly and incorrectly, respectively.

Table 5 presents the performance evaluation results of the five algorithms mentioned above according to metrics derived from the confusion matrix and T_C. These results show that all algorithms have a higher TPR than than 90%. Again, RF and ARF behave better than the other evaluated algorithms, reaching percentages of around 99.7% and 99.57%, respectively. Furthermore, RF and ARF have a low FPR of about 0.29% and 0.43%, respectively. In addition, RF and ARF reach a high precision of around 99.68% and 99.69%, respectively. T_C is low for all tested algorithms, ranging between 0.66 and 5.146 ms. HT and SVM obtain the shortest T_C.
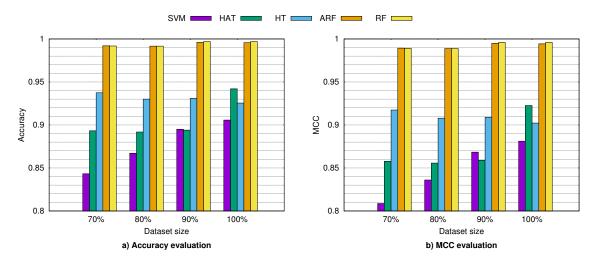

Figure 4. Modeling evaluation; (**a**) Accuracy evaluation, (**b**) MCC evaluation.

Table 5. T_C and metrics derived from the confusion matrix. TPR: true positive rate; FPR: false positive rate; RF: Random Forest; ARF: Adaptive Random Forest; SVM: Support Vector Machine; HAT: Hoeffding Adaptive Tree; HT: Hoedding Tree.

Algorithms	TPR%	FPR%	Precision%	T_C (ms)
RF	99.70	0.29	99.68	3.865
ARF	99.57	0.42	99.65	5.146
SVM	90.57	9.42	92.99	1.445
HAT	94.21	5.78	75.65	1.471
HT	92.55	7.44	74.44	0.666

From the cross-validation and confusion matrix analysis, we concluded the following: (a) none of the algorithms achieve the best results for all the metrics presented, (b) RF and ARF algorithms outperform the other algorithms regarding accuracy and MCC; and (c) the algorithms that obtained better performance in the TPR metric result in a high T_C and vice-versa. Considering these conclusions, we decided to use RF for network selection due to it being the model with the best balance between accuracy, MCC, and T_C. Thus, we built NetSel-RF by using RF. In the next section, we compare NetSel-RF with SSF and AHP-TOPSIS.

4. Network Selection

This section aims to evaluate the behavior of NetSel-RF regarding two metrics: the number of handovers and throughput drops. Furthermore, this section compares NetSel-RF to SSF and AHP-TOPSIS qualitative and quantitatively. The following subsections present the metrics, evaluation scenario, and results obtained.

4.1. Metrics and Evaluation Scenario

We compared NetSel-RF to SSF and AHP-TOPSIS regarding the number of handovers and throughput drops. The quantity of handovers is the number of transfers an end-device makes when it moves from a place to another one [13]; this number is affected by the ping-pong effect that occurs when the mobile device does not stay within the coverage of the selected AP and returns to the associated AP. The throughput drops represent the times that the number of bytes transmitted falls to zero due to handover.

With the aim of evaluating NetSel-RF, we used Mininet-WiFi to emulate a network formed by four APs with 802.11 g and 17 end-devices distributed in four rooms (see Figure 1). We analyzed the performance of NetSel-RF, SSF, and AHP-TOPSIS regarding the metrics mentioned above when *dev*1 carried out a linear movement at a speed of 1 m/s.

4.2. Results and Analysis

Figure 5 depicts how NetSel-RF, SSF, and AHP-TOPSIS behave regarding the number of handovers, revealing NetSel-RF has a higher number of unnecessary handovers than SSF and AHP-TOPSIS because it suffers from the ping-pong effect in the movements 100, 215, and 415. This ping-pong is caused by *AP*3 and *AP*4 offering identical conditions (i.e., the same RSSI level and percentage of users connected to AP) to *dev*1.
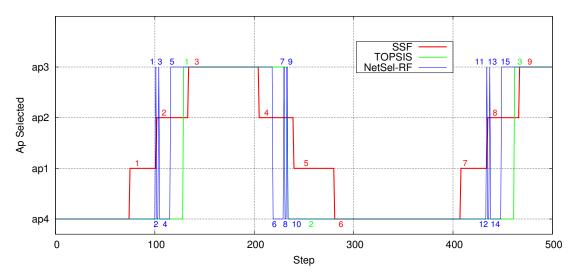

Figure 5. Number of handovers: NetSel-RF without movements module.

To address the ping-pong effect, we enhanced NetSel-RF with a movements module. This module uses a row vector to store the variable *AP_target* as well as the last seven locations (latitude and longitude) of end-devices. A movement is given by a location change. Thus, the handover is performed only when *AP_target* has the same value for all seven movements; otherwise, the end-devices remain connected to their current AP. It is important to highlight that we performed tests by considering 3, 5, 7, and 9 movements to choose the proper number to eliminate the ping-pong effect.

Table 6 exemplifies two vectors containing *AP_target*. NetSel-RF does not perform any handover for the *AP_target* presented in the first row since this row jumps between *AP*2 and *AP*3. Conversely, since the *AP_target* is the same for the seven location changes (movements) in the second row, NetSel-RF carries out the handover. In practice, we filled this vector for all devices in movement by using Mininet features: *dev.params*["*associatedTo*"] and *dev.params*["*position*"].

Table 6. AP_target example.

Variable / Movement	Mov1	Mov2	Mov3	Mov4	Mov5	Mov6	Mov7	
AP_target	AP2,loc1	AP2,loc2	AP2,loc3	AP3,loc4	AP2,loc5	AP3,loc6	AP2,loc7	X
AP_target	AP2,loc1	AP2,loc2	AP2,loc3	AP2,loc4	AP2,loc5	AP2,loc6	AP2,loc7	√

Figure 6 depicts the evaluation results regardubg the number of handovers carried out by the enhanced NetSel-RF, AHP-TOPSIS, and SSF. These results reveal that NetSet-RF and AHP-TOPSIS outperform SSF regarding this evaluation metric. In particular, the number of handovers for RSSI, AHP-TOPSIS, and our model are 9, 3, and 3, respectively.
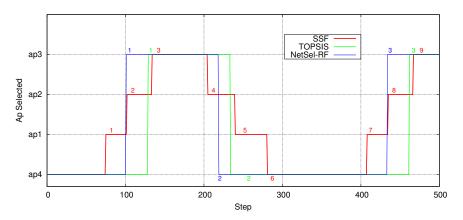

Figure 6. Number of handovers: NetSel-RF with movements module.

Figure 7 shows the evaluation results regarding the throughput drops of the enhanced NetSel-RF, AHP-TOPSIS, and SSF. These results reveal our model and AHP-TOPSIS outperform SSF regarding this evaluation metric. In particular, NetSel-RF and AHP-TOPSIS suffer 3 throughput drops (0*Mbps*). In turn, SSF has 7 drops. In summary, according to Figures 6 and 7, the multi-criteria based approaches behave better than SSF regarding the number of handovers and throughput drops.
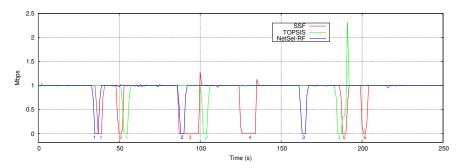
Figure 7. Throughput drops.

Figures 6 and 7 reveal that NetSel-RF behaves similarly to AHP-TOPSIS regarding the number of handovers and throughput drops. However, there is an essential difference between our model and AHP-TOPSIS. NetSel-RF performs transfers earlier than AHP-TOPSIS (as shown in Figure 8). This is due to the fact that our model is proactive and AHP-TOPSIS is reactive. Thus, NetSel-RF selects a new AP before suffering a disconnection. The proactive approaches are more effective than reactive ones regarding QoS and QoE since the end-devices perform the handover before the network link quality degrades, meaning that end-devices are always connected [38].
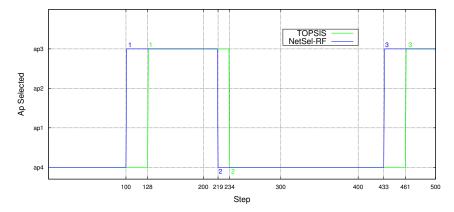

Figure 8. AP Selection.

4.3. Practicability

Since the NetSel-RF model needs information from devices (e.g., battery consumption) and APs (e.g., occupation), it requires a network global view for a real deployment. In this sense, NetSel-RF is envisioned to be deployed on WiFi networks following the SDN paradigm. SDN offers programmability, a global view, logically centralized control, and the decoupling of network control and packet forwarding. It is noteworthy that our training and validation datasets were collected by using the Mininet-WiFi emulator, which is SDN-oriented.

We analyzed the most common parameters used in the literature to build the final dataset used to train and validate NetSel-RF. The use of parameters other than those used in the data interpretation and data preparation phases is out of the scope of this paper. It is also worthy of note that we use the parameters available in the device and AP because, in practice, NetSel-RF is geared towards Software-Defined Wireless Networks [39–41], which is implicit from the data interpretation phase to the evaluation phase. For actual deployment, we suggest installing the NetSel-RF learning agent in the management plane [42] or in the cognitive plane [43].

5. Conclusions and Future Work

In this paper, we introduced a multi-criteria model based on supervised learning, called NetSel-RF, to cope with incorrect network selection decisions that lead to unnecessary handovers,

throughput drops, and complexity in the current HM solutions. We constructed a dataset, performed data preparation, carried out feature selection, and developed our model by using RF. Our model, based on multi-criteria (coming from the network, user preferences, end-devices, and applications) and supervised learning, outperformed the SSF approach regarding several metrics, such as the number of handovers (67%) and throughput drops (50%). Moreover, our model behaved similarly to AHP-TOPSIS regarding the metrics mentioned, but selecting a new AP without waiting for the current AP is beyond the capability of the model. Our approach is proactive, and AHP-TOPSIS is reactive; therefore, NetSelf-RF makes a hand-off to an AP with better conditions before the mobile device loses connection, in contrast to AHP-TOPSIS.

As future work, we intend to consider various devices carrying out the handover process at the same time. In this way, we plan to carry out the handover as a group, reducing the control messages that arrive at the AP. We also plan to evaluate the impact of proactive migrations in the throughput.

References

1. Jain, A.; Tokekar, S. Application based vertical handoff decision in heterogeneous network. *Procedia Comput. Sci.* **2015**, *57*, 782–788. [CrossRef]

2. Salih, Y.K.; See, O.H.; Ibrahim, R.W.; Yussof, S.; Iqbal, A. An overview of intelligent selection and prediction method in heterogeneous wireless networks. *J. Cent. South Univ.* **2014**, *21*, 3138–3154. [CrossRef]

3. Han, Z.; Lei, T.; Lu, Z.; Wen, X.; Zheng, W.; Guo, L. Artificial Intelligence-Based Handoff Management for Dense WLANs: A Deep Reinforcement Learning Approach. *IEEE Access* **2019**, *7*, 31688–31701. [CrossRef]

4. Drissi, M.; Oumsis, M.; Aboutajdine, D. A multi-criteria decision framework for network selection over LTE and WLAN. *Eng. Appl. Artif. Intell.* **2017**, *66*, 113–127. [CrossRef]

5. Lahby, M. Enhancing Modeling for Network Selection Using Graph Theory in Beyond 4G Networks. *Int. J. Bus. Data Commun. Netw.* **2020**, *16*, 48–69. [CrossRef]

6. Malathy, E.; Muthuswamy, V. State of art: vertical handover decision schemes in next-generation wireless network. *J. Commun. Inf. Netw.* **2018**, *3*, 43–52. [CrossRef]

7. Chi, C.; Cai, X.; Hao, R.; Liu, F. Modeling and analysis of handover algorithms. In Proceedings of the IEEE GLOBECOM 2007-IEEE Global Telecommunications Conference, Washington, DC, USA, 26–30 November 2007; pp. 4473–4477.

8. Nahida, K.; Yin, C.; Hu, Y.; Arain, Z.A.; Pan, C.; Khan, I.; Zhang, Y.; Rahman, G.S. Handover based on AP load in software defined Wi-Fi systems. *J. Commun. Netw.* **2017**, *19*, 596–604. [CrossRef]

9. Khalaf, G.A.F.M.; Badr, H.Z. A comprehensive approach to vertical handoff in heterogeneous wireless networks. *J. King Saud Univ. Comput. Inf. Sci.* **2013**, *25*, 197–205.

10. Khan, M.; Din, S.; Gohar, M.; Ahmad, A.; Cuomo, S.; Piccialli, F.; Jeon, G. Enabling multimedia aware vertical handover Management in Internet of Things based heterogeneous wireless networks. *Multimed. Tools Appl.* **2017**, *76*, 25919–25941. [CrossRef]

11. Din, S.; Ahmad, A.; Paul, A.; Rho, S. MGR: Multi-parameter Green Reliable communication for Internet of Things in 5G network. *J. Parallel Distrib. Comput.* **2018**, *118*, 34–45. [CrossRef]

12. Alotaibi, N.M.; Alwakeel, S.S. A neural network based handover management strategy for heterogeneous networks. In Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 9–11 December 2015; pp. 1210–1214.

13. Yu, H.W.; Zhang, B. A heterogeneous network selection algorithm based on network attribute and user preference. *Ad Hoc Netw.* **2018**, *72*, 68–80. [CrossRef]

14. Goyal, P.; Lobiyal, D.; Katti, C. Dynamic user preference based network selection for vertical handoff in heterogeneous wireless networks. *Wirel. Pers. Commun.* **2018**, *98*, 725–742. [CrossRef]

15. Ali, Z.; Baldo, N.; Mangues-Bafalluy, J.; Giupponi, L. Machine learning based handover management for improved QoE in LTE. In Proceedings of the NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium, Istanbul, Turkey, 25–29 April 2016; pp. 794–798.

16. Khan, M.; Ahmad, A.; Khalid, S.; Ahmed, S.H.; Jabbar, S.; Ahmad, J. Fuzzy based multi-criteria vertical handover decision modeling in heterogeneous wireless networks. *Multimed. Tools Appl.* **2017**, *76*, 24649–24674. [CrossRef]

17. Goyal, R.; Goyal, T.; Kaushal, S.; Kumar, H. Fuzzy AHP Based Technique for Handover Optimization in Heterogeneous Network. In Proceedings of the 2nd International Conference on Communication, Computing and Networking, Chandigarh, India, 29–30 March 2018; Springer: Singapore, 2019; pp. 293–301.

18. Khan, M.; Han, K. A vertical handover management scheme based on decision modelling in heterogeneous wireless networks. *IETE Tech. Rev.* **2015**, *32*, 402–412. [CrossRef]

19. Sadik, M.; Akkari, N.; Aldabbagh, G. SDN-based handover scheme for multi-tier LTE/Femto and D2D networks. *Comput. Netw.* **2018**, *142*, 142–153. [CrossRef]

20. Kumar, A.; Kaur, P.; Sharma, P. A Survey on Hoeffding Tree Stream Data Classification Algorithms. *CPUH Res. J.* **2015**, *1*, 28–32.

21. Bifet, A.; Gavaldà, R. Adaptive learning from evolving data streams. In Proceedings of the International Symposium on Intelligent Data Analysis, Lyon, France, 31 August–2 September 2009; Springer: Berlin/Heidelberg, Germany 2009; pp. 249–260.

22. Maulik, U.; Chakraborty, D. Remote Sensing Image Classification: A survey of support-vector-machine-based advanced techniques. *IEEE Geosci. Remote. Sens. Mag.* **2017**, *5*, 33–52. [CrossRef]

23. Boutaba, R.; Salahuddin, M.A.; Limam, N.; Ayoubi, S.; Shahriar, N.; Estrada-Solano, F.; Caicedo, O.M. A comprehensive survey on machine learning for networking: Evolution, applications and research opportunities. *J. Internet Serv. Appl.* **2018**, *9*, 16. [CrossRef]

24. Chicco, D.; Jurman, G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genom.* **2020**, *21*, 6. [CrossRef]

25. Ng, B.; Deng, A.; Qu, Y.; Seah, W.K. Changeover prediction model for improving handover support in campus area wlan. In Proceedings of the NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium, Istanbul, Turkey, 25–29 April 2016; pp. 265–272.

26. Goudarzi, S.; Hassan, W.H.; Anisi, M.H.; Soleymani, A.; Sookhak, M.; Khan, M.K.; Hashim, A.H.A.; Zareei, M. ABC-PSO for vertical handover in heterogeneous wireless networks. *Neurocomputing* **2017**, *256*, 63–81. [CrossRef]

27. Chapman, P.; Clinton, J.; Kerber, R.; Khabaza, T.; Reinartz, T.; Shearer, C.; Wirth, R. *CRISP-DM 1.0: Step-by-Step Data Mining Guide*; SPSS Inc.: Chicago, IL, USA, 2000; Volume 16.

28. Williams, N.; Zander, S.; Armitage, G. A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification. *ACM SIGCOMM Comput. Commun. Rev.* **2006**, *36*, 5–16. [CrossRef]

29. Raschka, S. An overview of general performance metrics of binary classifier systems. *arXiv* **2014**, arXiv:1410.5330.

30. Larasati, H.T.; Ilma, F.H.; Nuhamara, B.; Mustafa, A.; Hakimi, R.; Mulyana, E. Performance evaluation of handover association mechanisms in sdn-based wireless network. In Proceedings of the 2017 3rd International Conference on Wireless and Telematics (ICWT), Palembang, Indonesia, 27–28 July 2017; pp. 103–108.

31. Tuysuz, M.F.; Trestian, R. Energy-efficient vertical handover parameters, classification and solutions over wireless heterogeneous networks: a comprehensive survey. *Wirel. Pers. Commun.* **2017**, *97*, 1155–1184. [CrossRef]

32. Chamodrakas, I.; Martakos, D. A utility-based fuzzy TOPSIS method for energy efficient network selection in heterogeneous wireless networks. *Appl. Soft Comput.* **2011**, *11*, 3734–3743. [CrossRef]

33. Kim, H.; Feamster, N. Improving network management with software defined networking. *IEEE Commun. Mag.* **2013**, *51*, 114–119. [CrossRef]

34. Fontes, R.R.; Afzal, S.; Brito, S.H.; Santos, M.A.; Rothenberg, C.E. Mininet-WiFi: Emulating software-defined wireless networks. In Proceedings of the 2015 11th International Conference on Network and Service Management (CNSM), Barcelona, Spain, 9–13 November 2015; pp. 384–389.

35. Hoeglinger, S.; Pears, R. Use of hoeffding trees in concept based data stream mining. In Proceedings of the 2007 Third International Conference on Information and Automation for Sustainability, Melbourne, Australia, 4–6 December 2007; pp. 57–62.

36. Resende, P.A.A.; Drummond, A.C. A survey of random forest based methods for intrusion detection systems. *ACM Comput. Surv. (CSUR)* **2018**, *51*, 1–36. [CrossRef]

37. Castillo Morales, A.; Embus Gaviria, D. Final Dataset AP Selection and Algorithm Selection. Avaliable online: https://github.com/juliancastillom/DataSet-AP_selection (accessed on 6 May 2020).

38. Zeljković, E.; Marquez-Barja, J.M.; Kassler, A.; Riggio, R.; Latré, S. Proactive access point driven handovers in IEEE 802.11 networks. In Proceedings of the 2018 14th International Conference on Network and Service Management (CNSM), Rome, Italy, 5–9 November 2018; pp. 261–267.

39. Costanzo, S.; Galluccio, L.; Morabito, G.; Palazzo, S. Software defined wireless networks: Unbridling SDNs. In Proceedings of the IEEE 2012 European Workshop on Software Defined Networking, Darmstadt, Germany, 25–26 October 2012; pp. 1–6.

40. Tartarini, L.; Marotta, M.A.; Cerqueira, E.; Rochol, J.; Both, C.B.; Gerla, M.; Bellavista, P. Software-defined handover decision engine for heterogeneous cloud radio access networks. *Comput. Commun.* **2018**, *115*, 21–34. [CrossRef]

41. Moura, H.; Alves, A.R.; Borges, J.R.; Macedo, D.F.; Vieira, M.A. Ethanol: A Software-Defined Wireless Networking architecture for IEEE 802.11 networks. *Comput. Commun.* **2020**, *149*, 176–188. [CrossRef]

42. Estrada-Solano, F.; Ordonez, A.; Granville, L.Z.; Rendon, O.M.C. A framework for SDN integrated management based on a CIM model and a vertical management plane. *Comput. Commun.* **2017**, *102*, 150–164. [CrossRef]

43. Ayoubi, S.; Limam, N.; Salahuddin, M.A.; Shahriar, N.; Boutaba, R.; Estrada-Solano, F.; Caicedo, O.M. Machine Learning for Cognitive Network Management. *IEEE Commun. Mag.* **2018**, *56*, 158–165. [CrossRef]