IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 10, 2021, accepted January 15, 2021, date of publication January 20, 2021, date of current version January 28, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3053045

A Machine Learning Approach to Enhance the Performance of D2D-Enabled Clustered Networks

SAAD ASLAM[1,2], FAKHRUL ALAM[1], (Senior Member, IEEE), SYED FARAZ HASAN[1], (Senior Member, IEEE), AND MOHAMMAD A. RASHID[1], (Senior Member, IEEE)

[1]Department of Mechanical & Electrical Engineering, School of Food & Advanced Technology, Massey University, Auckland 0632, New Zealand
[2]School of Professional Engineering, Manukau Institute of Technology, Auckland 2104, New Zealand

Corresponding author: Saad Aslam (s.aslam@massey.ac.nz)

The work of Saad Aslam was supported in part by the Massey University Doctoral Scholarship.

ABSTRACT Clustering has been suggested as an effective technique to enhance the performance of multicasting networks. Typically, a cluster head is selected to broadcast the cached content to its cluster members utilizing Device-to-Device (D2D) communication. However, some users can attain better performance by being connected with the Evolved Node B (eNB) rather than being in the clusters. In this article, we apply machine learning algorithms, namely Support Vector Machine, Random Forest, and Deep Neural Network to identify the users that should be serviced by the eNB. We therefore propose a mixed-mode content distribution scheme where the cluster heads and eNB service the two segregated groups of users to improve the performance of existing clustering schemes. A D2D-enabled multicasting scenario has been set up to perform a comprehensive simulation study that demonstrates that by utilizing the mixed-mode scheme, the performance of individual users, as well as the whole network, improve significantly in terms of throughput, energy consumption, and fairness. This study also demonstrates the trade-off between eNB loading and performance improvement for various parameters.

INDEX TERMS Clustering algorithm, content multicasting, D2D enabled networks, deep neural networks, eNB loading, machine learning, random forest, support vector machine, user segregation.

I. INTRODUCTION

The proliferation of cellular devices and rapid rise of Internet usage has produced unprecedented growth in wireless traffic [1]. To accommodate such growth, 5G, and beyond networks are expected to achieve capacity improvement in the order of 1000 [2]. Various techniques have been proposed in the literature to improve the capacity of the network such as transmission in mmWave, cell densification, and massive MIMO [2]. However, apart from capacity improvement techniques, researchers have also considered studying the type of traffic most demanded by the users [3], [4]. Repetitive patterns of content requests were noticed which led researchers to several efficient ways of content dissemination. It encouraged the methodologies for caching the contents [5]–[7] for social-aware networks using D2D communication [8] that entails direct transmission between two user equipment [9]. It reduces the transmission from the core network thereby reducing delays and improving the energy consumption of devices.

D2D-enabled social-aware networks have been widely studied with the clustering concept. Various clustering algorithms have found their application for D2D-enabled networks. These clustering algorithms can be classified into four main categories; distance and similarity-based, hierarchical, squared error-based and graph theory-based algorithms [10]. In addition, there is a class of density-based clustering algorithms [11]–[14].

One of the main advantages of considering clustering for social-aware networks is that social awareness among the devices optimizes the network performance and brings cluster stability [15], [16]. Clustering is also accepted as one of the effective ways to exploit traffic characteristics to improve spectral efficiency [17]. Clustering in a multicasting scenario intends to improve performance by reducing energy consumption [18]–[24] and increasing throughput [25]–[28]. These works show that the importance of clustering is not specific to content-centric broadcast networks and can be applied

The associate editor coordinating the review of this manuscript and approving it for publication was Tony Thomas.

TABLE 1. Summary of clustering works and their performance metrics.

Research	User Segregation	Performance Parameters
Zhou et al (2013) [22]	✖	Energy Consumption
Xu et al (2014) [23]	✖	Energy Consumption, Spectral Efficiency
Militanoet al (2015) [24]	✖	Energy Consumption
Asadi et al (2016) [25]	✖	Throughput
Bassoy et al (2016) [26]	✖	End-to-End Delay, Cluster Stability
Xia et al (2017) [27]	✖	Energy Consumption
Niu et al (2017) [28]	✖	Energy Consumption
Kitagawa et al (2017) [29]	✖	Throughput
Ren et al (2017) [30]	✖	Throughput
Tulu et al (2017) [31]	✖	End-to-End Delay, Cluster Stability
Xu et al (2017) [32]	✖	Energy Consumption, Spectral Efficiency
Huang et al (2017) [33]	✖	Energy Consumption, Spectral Efficiency
Li et al (2017) [34]	✖	Energy Consumption, Spectral Efficiency
Duan et al (2017) [35]	✖	Energy Consumption, Spectral Efficiency
Yaacoub et al (2018) [36]	✖	Energy Consumption, Cluster Stability
Zhao et al (2018) [37]	✖	Energy Consumption
Khan et al (2018) [38]	✖	Energy Consumption
Sharafesddine et al (2018) [39]	✖	Throughput
Weijing et al (2018) [40]	✖	End-to-End Delay
Yin et al (2019) [41]	✖	Energy Consumption
Yin et al (2020) [42]	✖	Throughput
Aslam et al (2020) [43]	✖	Throughput, Energy Consumption, Spectral Efficiency, Fairness

to various applications as shown in the literature [29]–[43]. Table 1 summarizes various studies focusing on clustering.

Since clustering is meant for performance enhancement of various parameters, therefore a lot of research works have been dedicated to designing new clustering algorithms, mentioned in Table 1. However, a fundamental question would be, is there any scope of improvement within the clustering process itself? To answer this question, let us consider a few users interested in the same content. Once the users sharing a common interest are identified (several processes exist in the literature for content/interest identification), they are either served directly by the eNB or by the Cluster Heads (CHs) in clusters. Clusters are formed only for the users that are interested in the same content. Different research works that consider clustering, place all the users/nodes in clusters whereas on the other hand if clustering is not performed then all the users are associated with eNB. However, we believe there is a better approach to clustering through user segregation; some users communicate with CHs while being in clusters, whereas the remaining communicate directly with the eNB. Nevertheless, as a result of this segregation, only a certain percentage of users communicate directly with eNB and the majority of the users still stay in clusters and exploit D2D communication.

To further elaborate on the proposed concept, let us consider a social gathering, such as a football match. Users (socially aware nodes) are interested in the same videos of their favorite players. In a conventional clustering scheme, all these users sharing a common interest will be considered for clustering. However, even within a socially connected group, user segregation needs to take place to enhance the performance. Hence, in the proposed scheme, there are three types of users; CHs (responsible for fetching the content from eNB), cluster members (part of the cluster and receiving the requested content from CH via D2D), and segregated users (downloading the content directly from the eNB). Various studies presented in Table 1 suggest that this user segregation has not been reported in the literature. In this article, we show that segregation has a significant impact on the performance of both individual users as well as the network.

It should be noted that the concept of user segregation is not specific to any particular application of the D2D clustered network. It can be applied to any clustering scenario and/or application. However, we have evaluated the proposed scheme for a content-sharing D2D enabled network. It is also understood that D2D communication is meant to offload the central controller and socially connected users are meant to be in the same group. However, we focus on optimizing the clustering scenario by introducing the segregation concept. Moreover, it has been extensively discussed in the literature that eNBs often have spare capacity [16], [18], [19] and we are proposing to exploit this capacity leading to better performance and more efficient utilization of resources. Several research works have exploited the spare capacity to improve the system's performance [17], [20], [21], but this concept has not been considered for clustering.

For the proposed scheme to work, the fundamental task would be to categorize the users into two groups; one group contains all the users better served in clusters while the other group consists of users better off with the eNB. To perform this segregation, we opt for the Machine Learning (ML) approach. ML is an ideal tool for the proposed problem since we take advantage of offline training, without involving the eNB, making the training, and segregation process distributive. The machine learning algorithm employed are: Support Vector Machine (SVM) [44], [45], Random Forest (RF)[46], and Deep Neural Network (DNN)[47]. Literature suggests that SVM and DNN have been widely used for classification problems [48]. Moreover, the problem discussed in this manuscript is a binary classification problem which has found even more applications for SVM and DNN [49]. Specifically, for different applications of wireless networks from the physical layer to the application layer, SVM and DNN have been used [50]. Similarly, RF has found many use-cases for classification and regression problems targeting wireless networks [48], [51]. The concept of the proposed scheme; user segregation, utilization of ML algorithms is shown in Fig. 1.

A. MOTIVATION AND CHALLENGES

It has been well established that clustering improves the performance of the network. However, the effect of clustering on the performance of the individual users have not been investigated. Some users are disadvantaged by being in the clusters since they can be better served by the eNB. On the contrary, clustering scenarios considered in the literature place all the users/nodes in the clusters. Therefore, this study aims to identify the users better served in clusters and the rest communicate directly with the eNB. Moreover, the user segregation should not impact the network latency. Therefore, ML algorithms that take advantage of offline training are employed. We demonstrate that such user segregation considerably improves various performance parameters.

To perform the user segregation, training the machine learning algorithms is the first task. We need a significant amount of data to train the algorithms. Unlike, other domains such as image processing, data sets are not publicly available for wireless networks. Therefore, extensive simulations are required to generate the training corpus. For a practical solution, data collection opportunities in a live network to construct real-world training corpus need to be identified as well. Later in the subsequent sections, we provide details on each of these accomplished tasks.

B. CONTRIBUTIONS

The major contributions of our work are as follows:

- A user segregation scheme targeting D2D clustering has not been reported in the literature. Our work clearly shows that substantial improvement in terms of throughput, energy consumption, and fairness can be achieved as a result of applying user segregation. It should be noted that we applied this concept to four clustering algorithms, and it improves the performance of every algorithm.
- A binary classification model has been designed and trained to identify the users that should communicate directly with eNB as opposed to being in clusters. This model is trained completely offline and therefore does not increase the workload of the central controller. Moreover, owing to the offline training, explicit network measurements of the live network are not required. Therefore, network latency is not substantially affected.
- Multiple machine learning algorithms are investigated to ascertain their suitability as classifiers for user segregation. It is found that SVM performs the best, followed by RF and DNN.
- We have explored and identified various data collection opportunities in a cellular network for constructing the machine learning training corpus. These opportunities are outlined with respect to the segregation problem as well.
- This work clearly demonstrates the trade-off between eNB loading and performance improvement. It provides an opportunity for the cellular networks to select an improvement factor based on the serving capacity of the eNB. Moreover, our results also show that binary classification is adequate, and training a multiclass classifier is not warranted.

The rest of the manuscript is organized into five sections. Section 2 presents the rationale for using ML and data collection opportunities in a cellular network. The system model is presented in Section 3. The proposed DNN architecture, utilization of SVM and RF, and all the relevant details of ML implementation are detailed in Section 4. Results are provided in Section 5, while the conclusion is provided in the last section, also discussing future research directions.

II. MACHINE LEARNING FOR CLUSTERING APPLICATIONS OF NEXT-GENERATION CELLULAR NETWORKS

As we are moving towards the standardization of 5G, the focus has shifted towards developing the technologies necessary to realize the next-generation cellular networks such as 6G. Besides the improvement in the current techniques, ML approaches have been recognized by many researchers as a potential tool to provide optimal solutions to the complexities of the 6G network [52]. It should be noted that ML algorithms not only deliver network optimization solutions, they profoundly change the architecture of the 6G network. Therefore, data storage and learning servers are now an essential part of the cellular network architecture [53]–[58].

Recent works have focused on ML applications of clustered cellular networks [59], [60]. 6G is envisioned to have various D2D use-cases and research suggests that many of these applications are better realized by applying clustering algorithms [61]. A few preliminary studies have shown that clustering is going to be an integral part of 6G [62]. We have summarized various ML applications specific to cellular networks and clustering in Fig. 2.

Machine Learning has been explored in various wireless network applications. Wireless Sensor Networks (WSN) employ machine learning techniques to adapt to dynamic environments. WSN utilizes machine learning to optimize

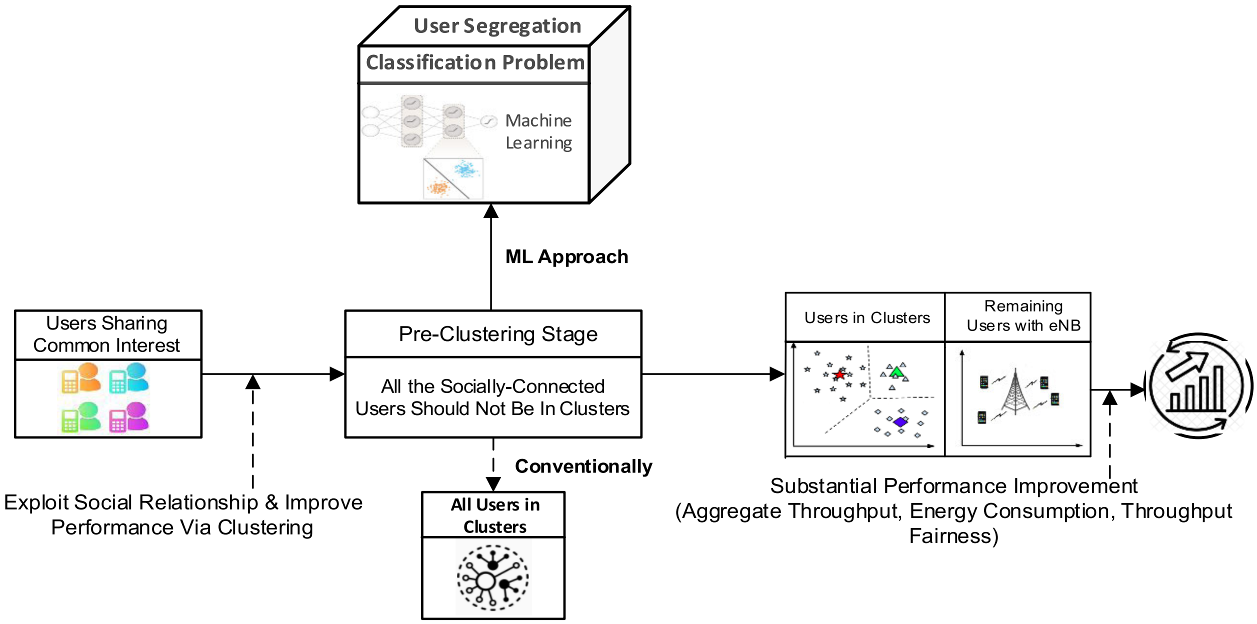


FIGURE 1. Proposed user segregation before clustering. Segregation is carried out using machine learning and as a result, few users are shown communicating in clusters whereas the rest communicate directly with the eNB.

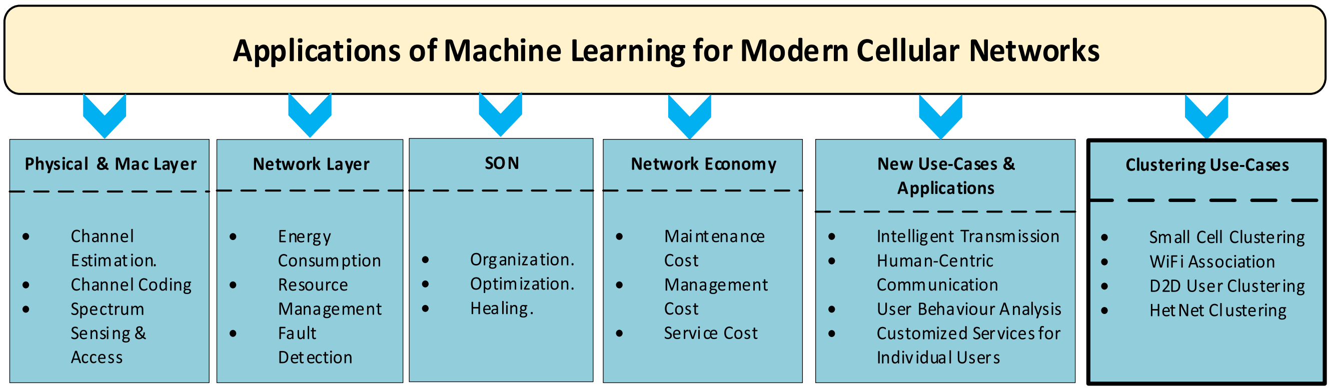


FIGURE 2. A range of machine learning applications for cellular networks.

energy consumption, scheduling, routing, and security [49], etc. Authors in [63] jointly consider cognitive radio and machine learning to investigate the complex spectrum requirements of the communication system. The authors propose a new model of anti-jamming, based on machine learning. On the other hand, [64] investigates the spectrum sharing problem. A multi-agent learning framework has been introduced to optimize the spectrum sharing process.

It is important to note that all these applications of ML can be realized if we have the data for learning. Data is being generated by thousands of cellular users each day. Therefore, data is available and cellular operators (COs) need to make use of this data.

Owing to the benefits of ML and offline learning, we utilize ML for the proposed work. It is synergistic to the other ML proposals as well. However, we are targeting a clustering application and proposing user segregation for performance improvement. Moreover, if we consider an online network, then solving this problem conventionally, without utilizing the ML, will present an impractical scenario, where the signalling required to set up an eNB assisted solution would be unrealistic. Therefore, offline training makes the scheme distributive and hence, realizable in real-world scenario.

A. DATA COLLECTION FOR MACHINE LEARNING

The data required for ML can be collected by UEs, Core Network (CN), and the Radio Access Network (RAN) [65]. Fig. 3 presents the details of different data collection sources within the cellular network [65]. Once the data from different sources come in, it needs to be processed so that machine learning algorithms can be designed to improve the performance of the network. The data required by the proposed study to perform the user segregation can be obtained from

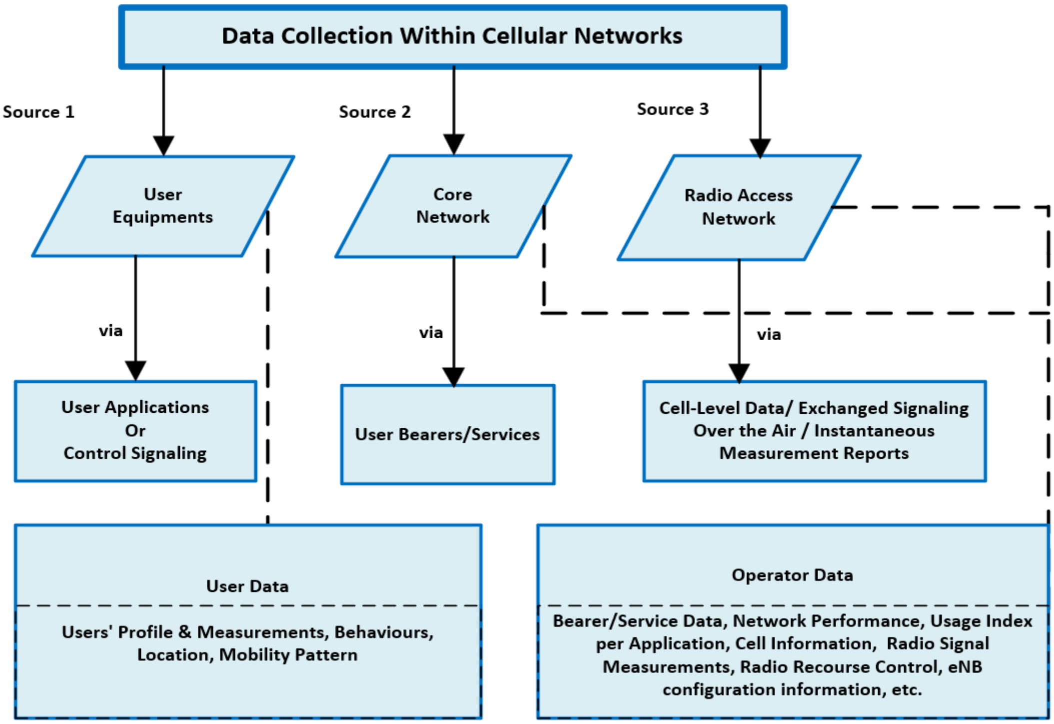


FIGURE 3. Data collection opportunities to construct training corpus in cellular networks.

RAN, mentioned as Source 3 in Fig. 3. The specific details of the data (input features) are detailed in Section IV.

III. SYSTEM MODEL

We consider a geographical area of 1 sq.km where users follow a uniform random distribution. The eNB is placed at the center of the cell. It should be noted that the placement of the eNB does not affect machine learning since users are randomly distributed and therefore, the distance of users to the eNB and/or distance of users among themselves is random as well. Moreover, the geographical location of users for which the clustering is taking place does not specifically represent a macro cell or small cell. The concept can be validated for any scenario since the learning was not subjected to any of these factors (geographical location of eNB and/or type of cell).

A typical network model is presented in Fig. 4. We assume that users are interested in certain content. These users are going to be served in clusters via a CH (UEs shown with red tops). However, there are certain nodes better served with the eNB and are communicating directly in Fig. 4 (UEs shown with blue tops). Moreover, the machine learning aspect is shown as well, complementing the network model.

The users are represented by the set U, which comprises of both Cellular User Equipment (CUE) and D2D User Equipment (DUE), such that, $U = 2U_D + U_C$, where $2U_D$ represents U_D D2D pairs and U_C are cellular users. All the list of symbols is summarized in Table 2. We may write the

TABLE 2. List of symbols.

Symbol	Representation
$\boldsymbol{i}$	Index of CUEs
$\boldsymbol{j}$	Index of DUEs
B	Bandwidth of the Transmission Channel
$\boldsymbol{N_o}$	Noise Spectral Density
$\boldsymbol{SNR_{U_{C_i}}}$	SNR of the ith CUE
$\boldsymbol{SNR_{U_{D_j}}}$	SNR of the jth DUE
$\boldsymbol{P_{eNB}}$	Transmit power of the eNB
$\boldsymbol{h_{eNB,U_{C_i}}}$	Channel between the eNB and the ith CUE
$\boldsymbol{P_m^T}$	D2D Transmitter indexed as m
$\boldsymbol{h_{U_{D_j},m}}$	Channel between the mth DUE Transmitter and D2D receiver denoted by U_{D_j}

capacity of the system represented by both types of users as;

$$C_U = W\left(\sum_{i=1}^{U_C} \log_2\left(1 + SNR_{U_{C_i}}\right) + \sum_{j=1}^{U_D} \log_2\left(1 + SNR_{U_{D_j}}\right)\right) \quad (1)$$

where $i\&j$ donates indexes of CUEs and DUEs respectively and W represents the bandwidth of the cellular network.

$SNR_{U_{C_i}}$ and $SNR_{U_{D_j}}$ represents SNR of the cellular and D2D users respectively, given by (2) & (3) ;

$$SNR_{U_{C_i}} = \frac{P_{eNB} h_{eNB,U_{C_i}}}{N_o B} \quad (2)$$

$$SNR_{U_{D_j}} = \frac{P_m^T h_{U_{D_j},m}}{N_o B} \quad (3)$$

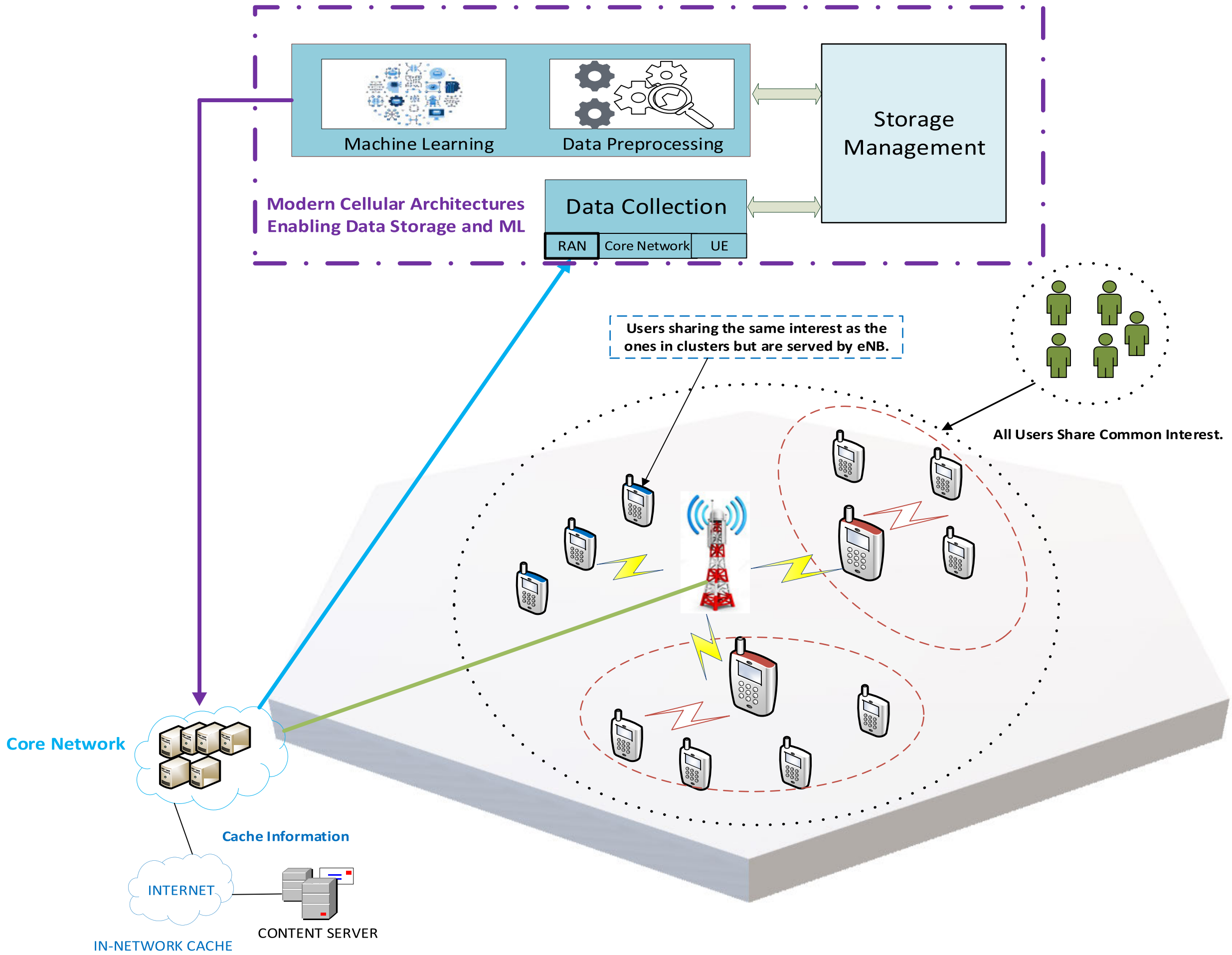


FIGURE 4. The network model supported by machine learning. A separate learning server, linked with core networks, have been considered an important part of modern cellular networks.

The channel gain h is modelled by a Rayleigh fading channel. It is important to note that in this study, we encapsulate both distance-dependent attenuation (i.e. path loss) and shadowing effect into the channel model. Shadowing is defined by a Gaussian random variable with a zero mean and standard deviation $\sigma = 8$dB.

As can be observed from (1-3), the sum capacity of a network depends on the accumulative SNRs of both CUEs and DUEs. Moreover, if we assume that the transmission power (of both eNB and DUE) and channel bandwidth are constant, then the SNR will significantly depend on the channel between the transmitter and the receiver, which is valid for both types of users. Nevertheless, it is worth mentioning that even with the varying transmission power, SNR will be influenced by the channel conditions. Therefore, the physical link between the transmitter and the receiver plays a vital role in determining the achievable data rates. This is important in selecting the input features for training the classifier.

Our goal is to maximize (1), utilizing the concept of user segregation, executed by ML algorithms. From a broader perspective, it is a binary classification problem where a user needs to be added to either of the two groups i.e. with eNB or with clusters. To perform this segregation, we utilize ML classifiers. The target for ML is based on maximizing the throughput of a particular node. Since we are considering a multicasting application, it should be noted that within a cluster the maximum achievable rate of cluster member depends on the worst physical link, to ensure that all the members receive the required data. In a given cluster k, the achievable rate of the cluster member m_k is be given by (4);

$$R_{m_k} = Blog_2\left(1 + \frac{P_{CH_k} h_{m_k,CH_k}}{N_o B}\right) \tag{4}$$

CH_k is the cluster head of kth cluster and P_{CH_k} represents its transmission power where h_{m_k,CH_k} is the worst physical link of the cluster.

A user is selected for either of the two groups based on the achievable data rate being higher in that group (i.e. either with the eNB or the CH) as opposed to being part of the other group. This fact gives the user opportunity to not only reduce its energy consumptions but also increase the throughput fairness of the system. It should be mentioned here that the ML algorithm is trained offline before it is tested on a mobile network (or, as in our case tested with the simulation

TABLE 3. Input features and true labels.

Feature (Mean Value)	True Labels	Labels Representation
1. Distance of a user with eNB 2. Distance of a user with other users 3. Pathloss of a user with eNB 4. Pathloss of a user with other users	0	Users with eNB
	1	Users in Clusters

environment). For the classification, we propose to utilize SVM, RF, and DNN. These algorithms were benchmarked against four widely used classification techniques and are found to be superior (please see section V).

IV. PROPOSED MACHINE LEARNING APPROACH

The data generated for the training purpose is obtained via the simulations. It conforms with different research works conducted in the field of wireless networks [66]–[68]. It is explained in [69], that the comparison of different ML algorithms can be made if we have common datasets. However, while this is true for computer vision, voice, and image processing, the wireless communication domain is unfortunately not having common datasets because it inherently deals with the data that can be accurately generated by simulations. However, as discussed in Section II, data is available to the COs, that could be potentially used to set up common benchmarks so the effectiveness of the simulation-based training can be evaluated. The overall learning algorithm is presented in Fig. 5. The algorithm is trained completely offline. However, data required (distance and channel conditions) for an online implementation, are the typical data required for forming clusters and can be obtained via the D2D discovery process which takes place before the communication starts. The details of such relevant process can be found in the clustering literature (e.g. see [25], [43] that shows network latency is not impacted significantly.

The selection of input features took place keeping in view the target of throughput maximization. The details of the explored ML algorithms including the tuning of hyperparameters and complexity have been provided in subsequent subsections and Section V.

A. THE PROPOSED DNN ARCHITECTURE

The proposed architecture is shown in Fig. 6 (a). Utilizing this architecture, we performed the user segregation for the proposed scheme. The output of the DNN gives a probability value, which classifies a node that should be either serviced by the eNB or the CH. The DNN consist of three layers. The input layer is termed as L_{IN}, whereas the hidden layers are termed as L_{H1}, L_{H2} respectively. The selected features and dependent vector/label are presented in Table 3.

Before the actual values of the weights can be found, the weights are randomly initialized from values between 0 and 1. Once the weights are initialized, we move to the first hidden layer where a dot product of the initialized weights and bias is performed. It can be represented as the following equation;

$$X_{L_{H1}} = V_{L_{IN}} . W_{L_{IN}} + b_{L_{IN}} \tag{5}$$

$V_{L_{IN}}$ trepresents the input vector (shown in fig. 6 (a)), $b_{L_{IN}}$ is the bias and weights are represented by $W_{L_{IN}}$. Once the input passes through the first layer, it becomes a neuron to be processed by the other layers. Now we use $X_{L_{H1}}$ to pass through our first activation function i.e. Rectified Linear Unit (ReLU) [70], which creates the first hidden layer and its output becomes the input of the second hidden layer.

The output layer of the proposed DNN architecture is composed of one neuron with the output given by (6).

$$Out_{L3} = sig(Out_{L2}.W_{L_{H2}} + b_{L_{H2}}) \tag{6}$$

$(Out_{L2}.W_{L_{H2}})$ represents the dot product between the input vector $V_{L_{H2}}$(i.e. the data vector from second hidden layer) and the corresponding weights $W_{L_{H2}}$ The bias of the second hidden layer is represented by $b_{L_{H2}}$.

Out_{L3} is the output from the last layer. Sig is the sigmoid function given by (7).

$$sig\,(x) = \frac{1}{1 - e^{-x}} \tag{7}$$

It computes the probabilities for the two classes. It can be mathematically represented as the following:

$$p_{out} = \begin{cases} p_{cluster} & if\ Out_{L3} > 0.5; \\ p_{eNB} & otherwise \end{cases} \tag{8}$$

In (8), $p_{cluster}$ represent the nodes that should be in clusters whereas the nodes that should be served by the eNB are represented by p_{eNB}. Ideally, all the nodes with clusters should have a probability of 1, and others should have a probability of zero.

The output that we obtained from DNN is then evaluated for error. It is calculated using binary cross-entropy. The loss determines the misclassification between the target and the predicted one. It should be noted that the simulation enabled us to possess the target information. The loss function is evaluated after every training iteration. It is given by (9):

$$LF\,(p_{cluster}, p_{eNB}) = -\frac{1}{N}\sum_{i=1}^{N}([TrueLabel_{cluster}]\,log\,(p_{outi}) + [[TrueLabel_{eNB}]]\,log\,(1 - p_{outi})) \tag{9}$$

The loss function is averaged over all the training samples N, at the end of each iteration. Once, the first iteration is complete, we backward propagate our gradient descent to update our weight parameters.

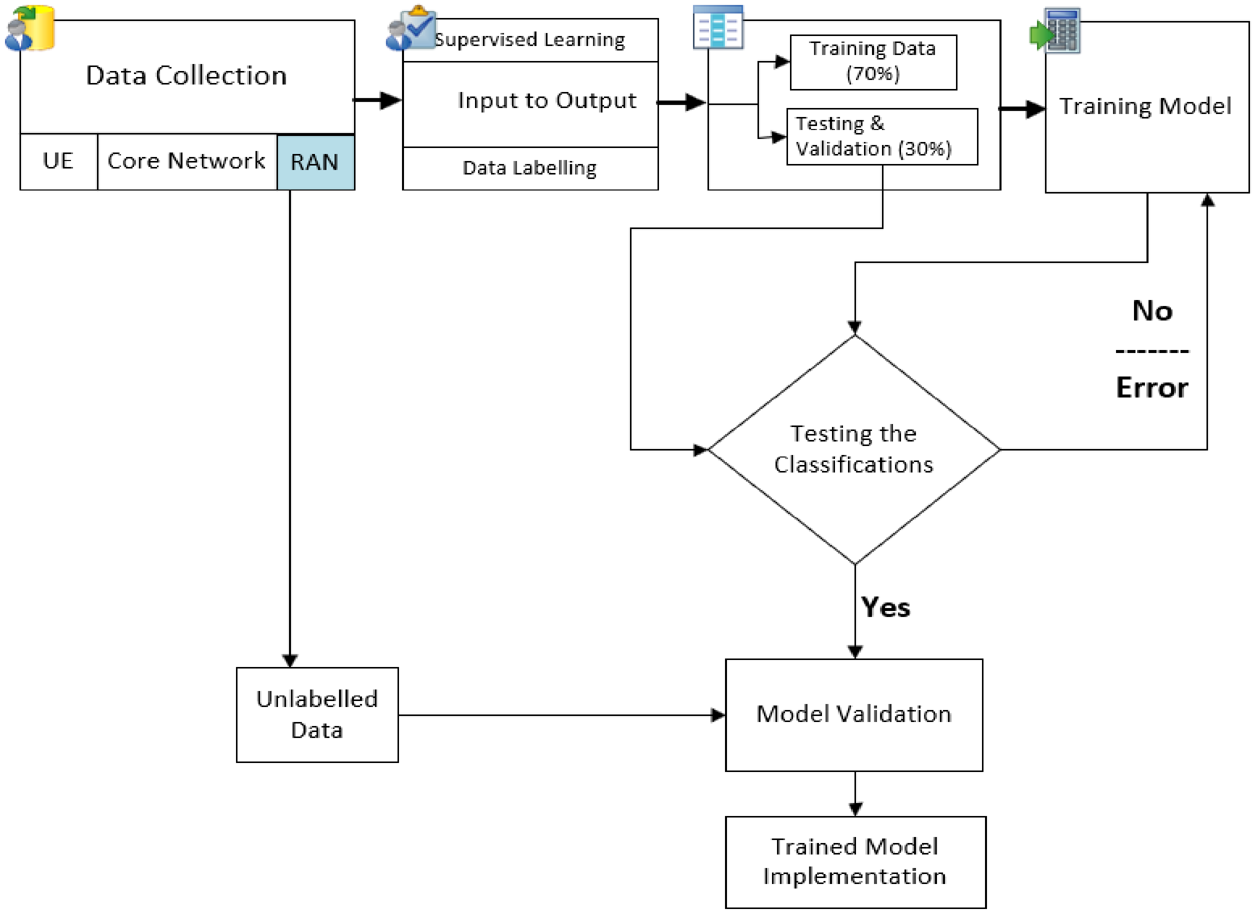


FIGURE 5. The learning algorithm flow chart. It should be noted that all the tasks are performed offline except trained model implementation.

B. SUPPORT VECTOR MACHINE

The objective of the SVM is to find an optimal hyperplane that segregates the input data into two categories [55] (e.g. whether a particular user should be in the cluster as opposed to being with eNB). The overall schematic of SVM applied to the proposed study is shown in Fig. 6 (b).

Each input instance of an SVM, denoted by I_{SVM}, represents a pair (a_i, b_i), where $a_i \in \mathbb{R}^n$ is the data instance (shown in fig. 6 (b)), and b_i represents the binary class label (given in (11)). A class can be characterized as either positive or negative. Therefore, in this setting, "nodes with clusters" belong to the positive class while "nodes with eNB" belong to the negative class. Given this information, we may write the hyperplane as;

$$w.I_{SVM} + C = 0 \tag{10}$$

where the classifier can be defined as;

$$f\left(I_{SVM}\right) = \begin{cases} (w.I_{SVM} + C), & b_i < -1 \\ (w.I_{SVM} + C), & b_i \geq +1 \end{cases} \tag{11}$$

In the above-given equation, w represents attached weights and C is a constant.

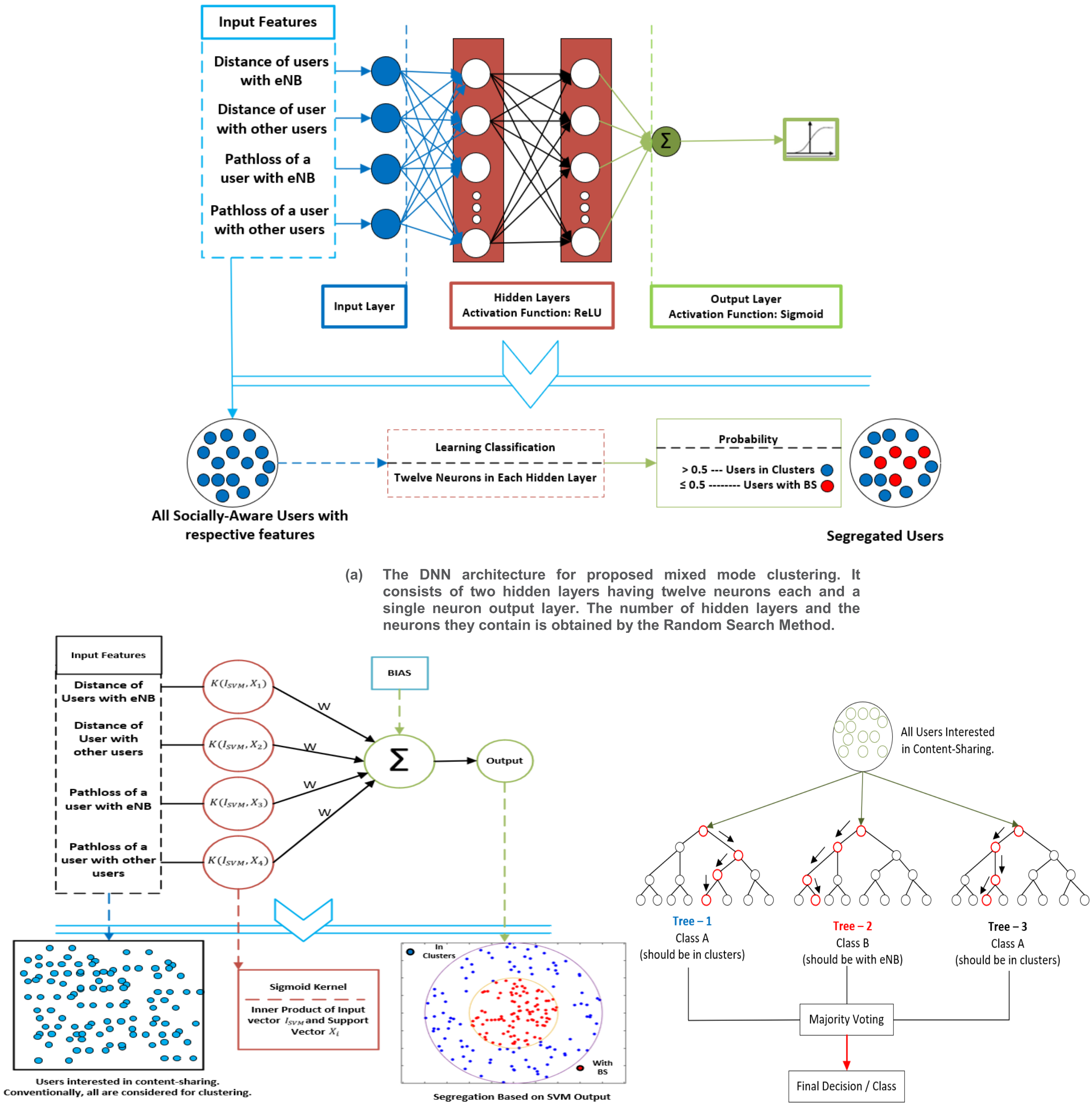
TABLE 4. Comparison of accuracy for different SVM kernel functions.

Kernel Function Explored	Accuracy (%)
Linear	82.33
Gaussian	88
Polynomial	93.33
Radial Basis Function	94.5
Sigmoid	96

The performance of the SVM is dependent on the Kernel function. We empirically selected Sigmoid Kernel as it provided higher accuracy in classifying the users compared to other functions such as Polynomial, Gaussian, and Radial basis function, etc. (please see Table 4).

C. RANDOM FOREST

Random Forest belongs to the class of ensemble algorithms (e.g. bagging and boosting) that utilizes the combination of trees to increase the accuracy and make stable predictions. RF can be used for both classification and regression problems. In this study, we use RF for the proposed binary classification problem [71]. The tree ensemble created by RF is trained on bootstrapped training data. Majority vote of trees decides the



(a) The DNN architecture for proposed mixed mode clustering. It consists of two hidden layers having twelve neurons each and a single neuron output layer. The number of hidden layers and the neurons they contain is obtained by the Random Search Method.

(b) SVM architecture implemented with Sigmoid Kernel, for the proposed mixed mode clustering.

(c) The Random Forest Concept as applied to the User Segregation problem.

FIGURE 6. **The application of ML algorithms for the proposed user segregation.**

classifier output [72]. The classification decision is dependent on the attribute/feature selection approach. In this work, Gini Index (GI) has been utilized, which evaluates the impurity of a feature with respect to the class. GI can be written as given in (12) [73];

$$\sum \sum_{a \neq b} ((f(NS_a, TS)\ /\ |TS|)\ ((f(NS_b, TS)\ /\ |TS|) \quad (12)$$

where NS_a is the node selection class 'a' (node belonging to cluster in this case), TS is the training set. The probability of the selected node is defined by $(f(NS_a, TS)\ /\ |TS|$.

Once the RF is trained on $'N'$ trees, each new dataset is passed down the trees, and the forest chooses a class based on the majority vote of trees. As described earlier, we utilize RF to classify the nodes that should be with eNB as opposed to being in the clusters. The concept of RF as applied to the user segregation problem is illustrated in Fig. 6 (c).

TABLE 5. Simulation parameters.

Parameters	Value
Simulation Platform	MATLAB
Channel Model	Rayleigh Distributed
User Placement	Uniformly Distributed
Node Density – Interested in Same Content	100 to 1000
Cluster Size	Variable
Number of Clusters	Variable
Transmit Power of CH	1.425 Joules/s
Power required to receive data from eNB	1.8 Joules/s
Power required to receive data from CH	0.925 Joules/s
Content Considered	A file of size 100kBits

V. RESULTS & DISCUSSION

We generated 340,000 data sets (with input features mentioned in Table. 3), 70% were used to train the model (i.e. training set) while the remainder 30% is equally partitioned for testing & validation.

The dataset used for training can be accessed at GitHub (https://github.com/Saad7861004/Machine-Learning-For-Wireless-Cellular-Networks.git). A standard technique used in the literature to define training and testing sets for validating the learning model is cross-validation [74]. Therefore, k-fold cross-validation was applied in this study with k = 10, a commonly used value [75]. Other common values of k such as k = 3, 5 were explored showing similar trends.

The ML related results such as accuracy, loss, and Receiver Operating Characteristic curves (ROC) are obtained from the validation set whereas the test data is used to evaluate the system's performance. It should be noted that we are testing the proposed scheme for a multimedia application in a multicasting scenario. Users are interested in the content of size 100kB, following the standard literature [76]. Various user densities have been considered to demonstrate the performance and effect of increasing density on different parameters. Moreover, these users are interested in sharing a common interest. Hence, it does not represent the total users of a cell. As discussed earlier in this document, as a result of classification, some users are in clusters whereas the remaining communicate directly with the eNB. Users in clusters communicate in D2D mode since the CH fetches the requested content and delivers it to its cluster members whereas the rest download the content directly from the eNB. All the simulation parameters are taken from the literature [36], [77]–[79]. The parameters of interest are summarized in Table 5.

A. HYPERPARAMETERS OPTIMIZATION

The objective is to find the optimal solution for each of the employed ML algorithms. To achieve this goal, we explored various values of the hyperparameters, mentioned in Table 6, for each of the classification technique. Literature suggested that we can exploit the following search strategies to find the optimal solution. These strategies include; Random Search, Grid Search, Heuristic, and Exhaustive Search [80].

An extensive study on hyperparameter optimization [81] suggests that Random Search is a significantly better technique for various types of data sets as compared to Search Space, Grid Search, etc. We, therefore, applied Random Search to optimize the hyperparameters (for all the ML algorithms). We train the learning model using the training data and use Random Search on k-fold cross-validation to tune the hyperparameters. It should be noted that many other related articles targeting wireless network applications are using the Random Search method for hyperparameters optimization as well [82]. All the values of hyperparameters are detailed in Table 6.

B. COMPLEXITY FOR TRAINING THE ALGORITHMS

It is mentioned earlier in Section I that the proposed study takes advantage of offline training. Only the trained classifier/algorithm will be implemented in a live network. Therefore, the proposed algorithm training does not affect network latency significantly. The total execution time for training the algorithms are detailed in Table 7.

The experiments were performed on a 64-bit Intel 4600 GPU, Core i7-4790 CPU @ 3.60 - 4 GHz processor. The system had 12 GB of RAM. MATLAB was used for the training purpose. SVM, RF, and DNN are standard classifiers. Therefore, the complexity of these algorithms in terms of hardware requirements and Big O notations can be found in the literature [83]–[85] and are quoted in Table 7.

C. COMPARISON OF CLASSIFICATION TECHNIQUES

It has been reported in the literature that SVM, RF, and DNN are useful for classification purposes especially for applications targeting wireless networks [86]. However, other classification techniques are available and investigated in this study as well. The accuracy of each technique is detailed in Table 8. SVM, RF, and DNN give the best accuracy among all the classification techniques. In this context, accuracy is defined as the percentage of correctly classified users. As discussed in Section III, the target was to select a user for one of the two groups based on the throughput maximization, and hence the output of the three algorithms was compared with true labels to determine the accuracy of the classification. The results given in Table 8 show that we achieved better accuracy for SVM as compared to RF and DNN. Moreover, it is suggested in the literature [85], [87] that SVM and RF have lower execution time as compared to DNN (supported by our findings shown in Table 7 as well), therefore they present lower computational burden on the learning servers as well.

D. COMPARISON OF ACCURACY AND LOSS OF THE TRAINED ALGORITHMS

Fig. 7 (a) presents the accuracy of the three algorithms for testing data against different epochs. It can be observed that the accuracy of all the algorithms keeps on increasing with the increasing number of epochs. On the other hand, it can be seen in Fig. 7 (b) that the loss values keep

TABLE 6. Hyperparameters explored for various machine learning algorithms.

Algorithm	Hyperparameters Explored		Optimized Solution/Hyperparameters
DNN	η (Learning Rate for Stochastic Gradient Descent)	$10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1$	Hidden Layer: 2 Number of Neurons:12, Gradient Descent: 10^{-3}
	Epochs	150	
	Batch Size	256	
	Hidden Layers	1,2,3,4,5	
	Activation Function	ReLU, Sigmoid	
	Weight and Bias Initiation	Random	
	Number of Neurons/Nodes in First & Second Hidden Layer	First Hidden Layer: 5 to 40 (step of 2) Second Hidden Layer: 0 to 40 (step of 2)	
SVM	Kernel	Linear, Gaussian, Polynomial, RBF, Sigmoid.	Kernel: Sigmoid C = 1 $\Upsilon = 10^{-1}$
	C	0.03, 0.1, 0.2, 0.3, 0.4. 1,2,3,4,5,10,100.	
	ϒ	$10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$	
Random Forest	Number of Trees/Estimators	10, 50, 100, 200, 300	Number of Tress/Estimators: 200 Criterion: Gini Max_Depth: 5
	Max Depth	2, 5, 10, 20	
	Function/Criterion to evaluate each split	Gini, Entropy	
	Bootstrap	True, False	
	Min, samples leaf	[1,20]	
	Min. samples split	[2,20]	
Decision Trees	Learning Rates	0.0001, 0.001, 0.01, 0.1, 0.2, 0.3	Learning Rate = 0.01, Decision Tress = 300, Splitter = Random.
	Number of Decision Trees/ Estimators	100, 200, 300, 400, 500	
	Splitter	Best, Random	

TABLE 7. Comparison of complexity for different machine learning algorithms.

Algorithm	Multipliers	Adders	Complexity in General (Worst Case Running Time)	Total Execution Time (sec)
DNN	$\sum_{k=2}^{K} N_{k-1} N_k$	$\sum_{k=2}^{K} (N_{k-1} N_k) N_k$	$O(L^4)$	3401.5
SVM	$N_{SV} * M$	$2N_{SV} * M$	$O(N_{SV} D)$	2307.1
RF	0 *(since it is a classification problem)*	$N_{trees} - 1$	$O(M N_{trees})$	1874

$N_{SV} = number\ of\ support\ vectors, M = total\ number\ of\ features, L = total\ number\ of\ layers\ of\ DNN,$
$N_k = number\ of\ neurons\ in\ the\ kth\ layer, D = data\ points, N_{trees} = number\ of\ trees.$

TABLE 8. Comparison of accuracy for different classification techniques.

Classification Techniques Accuracy (%)							
Decision Trees	Discriminant Analysis	k-Nearest Neighbor (k = 5)	Ensemble Classifier		DNN	RF	SVM
Medium Tree	Linear Quadratic	Weighted KNN	Boosted Trees	Bagged Trees			
73.3	76.3	61.4	74.3	73.6	85.8	93.33	96

on decreasing with the increase in the number of epochs suggesting no overfitting. It is further elaborated in the next section. The number of epochs were selected to be 150 based on the satisfactory error rate and flattening of the curves.

E. ROC CURVES

ROC curves, as shown in Fig. 8, demonstrate that the model is not overfitting as the curves are not all into the left corner. Since the curves are closer to the true positive axis, therefore, it shows the false positive rate is well within limits. Moreover,

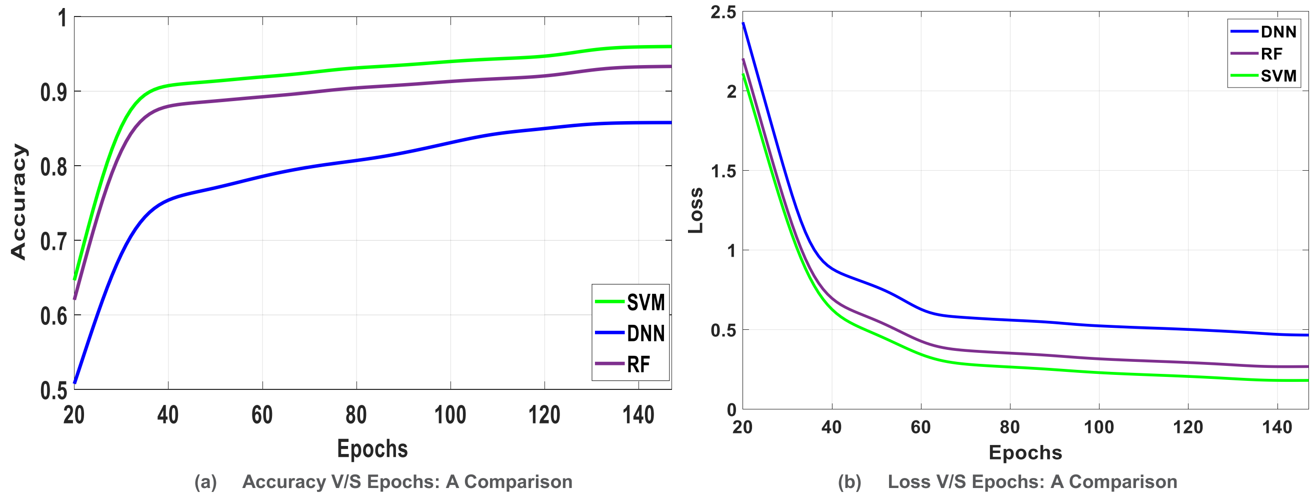


FIGURE 7. Comparison of the classification accuracy and loss based on the validation data set.

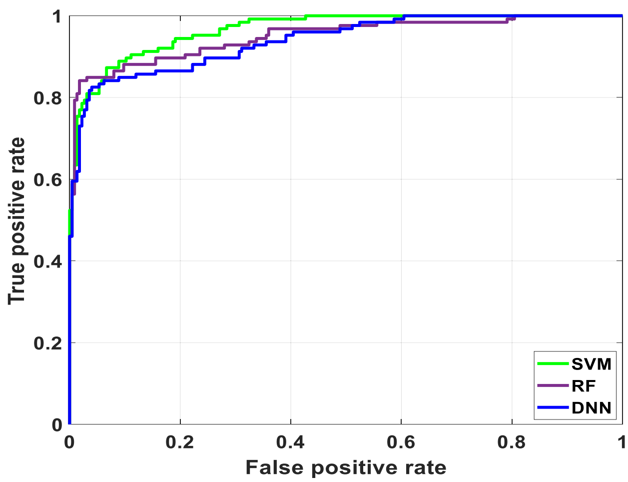


FIGURE 8. The ROC curves for the trained model.

it can be seen from Fig. 8 that area under the SVM curve is greater than that of RF and DNN which is another indication of SVM being more accurate subject to the user segregation problem.

F. ANALYSIS OF THE PERFORMANCE PARAMETERS

In this section, we explore the impact of the proposed mixed-mode scheme on the performance of the system. All the results shown in this section are based on the SVM classification since it is shown in the previous section that it produces the best classification results.

To demonstrate the performance, we have applied the proposed scheme and segregated the users for four different clustering algorithms. It should be noted that since we are considering different densities of socially aware nodes, so the number of clusters formed are not fixed. Moreover, as opposed to existing literature, we are not making an arbitrary number of clusters rather we are using Calinski-Harabasz criterion to select the number of clusters for a given node density [88].

The performance of the proposed scheme is benchmarked against that of the corresponding standard clustering scheme that keeps all the users within the clusters. Two of the benchmarked clustering algorithms represent state-of-the-art whereas the other two represent the classical clustering techniques. The classical algorithms, K-Medoids and Density-Based Spatial Clustering of Applications With Noise (DBSCAN) [12] can be widely found in the literature whereas the work presented in [31] and [43] are recently proposed algorithms, in this document, we termed them as "EBC" and "Multi-Factor" respectively. EBC algorithm selects CHs based on the entropy of betweenness centrality, which considers social-interest and shortest path between the users. The social factor is also considered in Multi-Factor scheme, which takes channel conditions as well as the distance between the users into consideration for selecting CHs and forming clusters. We deliberately selected these algorithms since they consider the social interest and it has been shown in the literature that it improves the performance of the network. Therefore, the proposed segregation concept is compared with the best performing algorithms. Our results also highlight that irrespective of the clustering algorithm, the proposed scheme significantly enhances the system's performance.

1) THROUGHPUT

The throughput performance is demonstrated in Fig. 9. The solid line represents 'all in clusters' scenario whereas the dotted line in each case demonstrates the impact of the proposed scheme. It should be noted that dotted lines represent the same clustering approach as solid lines, the only difference is that some users (based on classification) are serviced by eNB. In the case of classical schemes, at the user density of one thousand, the percentage increase in the throughput is approximately 42% and 34% for K-Medoids and DBSCAN respectively whereas for both other schemes the increase is

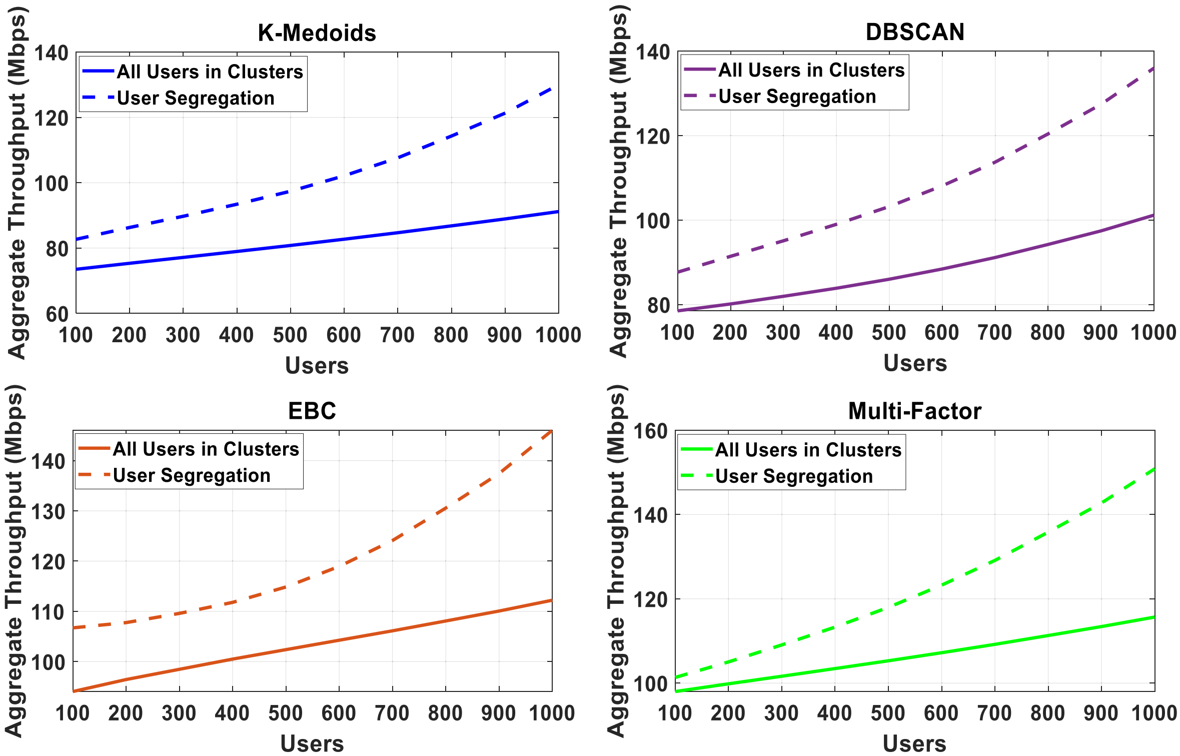


FIGURE 9. Throughput performance: A comparison.

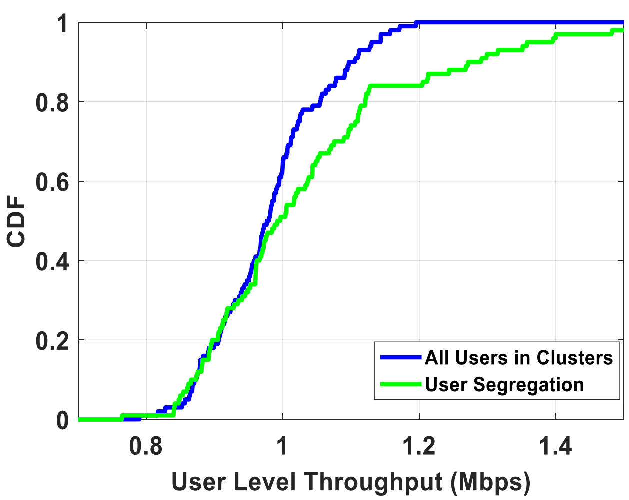


FIGURE 10. CDF of the throughput (100 Users). Similar trends were observed at other user densities with performance gap more pronounced.

approximately 30%. It is a significant improvement in all cases. The proposed scheme was able to achieve the best improvement for K-Medoids since it only considers distance for clustering users and distance only may not be the best metric for clustering since users in proximity may not have the best channel conditions due to various factors such as shadowing. However, in the proposed user segregation scheme, shadowing was considered for training the algorithms.

The throughput result is further elaborated in Fig. 10. It represents the Cumulative Distribution Function (CDF) of throughput benchmarked against MF scheme. We are presenting the result for the MF scheme only since it produces the best throughput as compared to the other two cases. It can be seen that the proposed scheme is performing better. A clear difference in the performance of the users can be seen at the 90th percentile. Similar trends were observed for all clustering schemes and for other user densities.

2) ENERGY CONSUMPTION

The result for the energy consumption of the users is demonstrated by Fig. 11. Downlink energy consumption is considered in this study. We have utilized the energy consumption model presented in [79]. We assume that the content demanded by the users is a file of size "F_S" *bits*. Suppose this file needs to be transmitted from CH_k to cluster member m_k having an achievable data rate of R_{m_k}. The time required to transmit this file is $\left(\frac{F_S}{R_{m_k}}\right)$ *seconds*. Therefore, energy consumption E_k in one of the clusters $'k'$ can be written as;

$$E_k = \frac{F_s P_{chrx}}{R_{CH_k}} + \frac{F_s P_{CH_k}}{R_{m_k}} + \sum\nolimits_{\substack{k \neq m \\ \forall m}} \frac{F_s P_{mrx}}{R_{m_k}} \tag{13}$$

Equation (13) represents the sum of three independent terms where the first term indicated the energy consumed by the CH to receive data (P_{chrx} represents the power consumed by the CH to receive the content from eNB and R_{CH_k} is the achievable rate at the cluster head). The second term

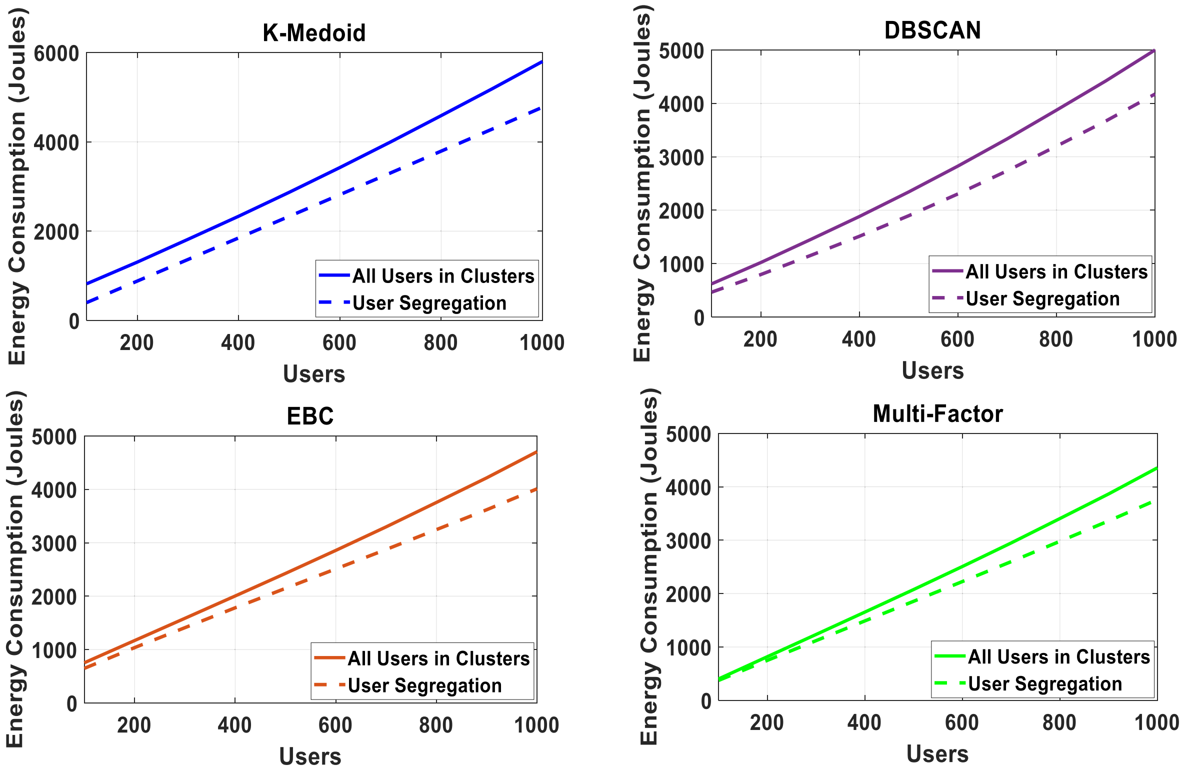


FIGURE 11. Energy consumption: A comparison.

represents the energy consumed by the CH to transmit the data to its cluster members (P_{CH_k} is the transmit power of cluster head and R_{m_k} is the achievable rate of cluster member given in (4)). The sum of the energy consumed by the cluster members to receive the demanded content is shown by the last term (where P_{mrx} represents the power consumed by the cluster member to receive content from the CH). The energy model clearly shows that if the file size is constant (as mentioned earlier it is 100kB), the energy consumption of the nodes significantly depends on the achievable data rates. Since the previous result demonstrates that user segregation improves the individual rates for the users, therefore energy consumption will be reduced as well. The reduction in energy consumptions at a user density of one thousand is approximately 17.67% for K-Medoids, 16.5% for DBSCAN, 14.77% for EBC, and 13.66% for the Multi-Factor scheme. The CDF of energy consumption is shown in Fig. 12, for the Multi-Factor scheme. The performance of the proposed scheme is better in most of the quartiles.

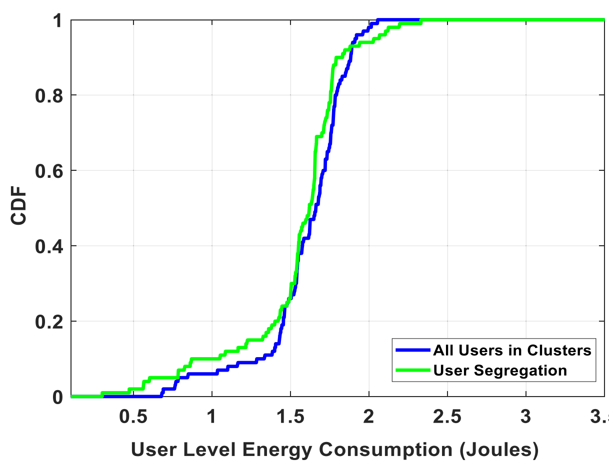


FIGURE 12. Energy consumption CDF (100 Users). Similar trends were observed at other user densities with performance gap more pronounced.

3) THROUGHPUT FAIRNESS

Jain's fairness model [89] as defined by (14) has been used to evaluate the fairness of the scheme.

$$J(x) = \frac{\left(\sum_{i=1}^{M} x_i\right)^2}{M \sum_{i=1}^{M} x_i^2} \tag{14}$$

x_i represents the throughput of the i^{th} user, given that there are total M users.

The result for throughput fairness is shown in Fig. 13. At the user density of one thousand, the improvement in fairness is 26.3% for K-Medoids, 26.5% for DBSCAN, 13.5% for EBC, and 20% for the fourth scheme. Since we have already shown earlier that as compared to 'all in clusters' scenario, user segregation improves achievable rates for a considerable percentage of users (precisely 20-30% users on

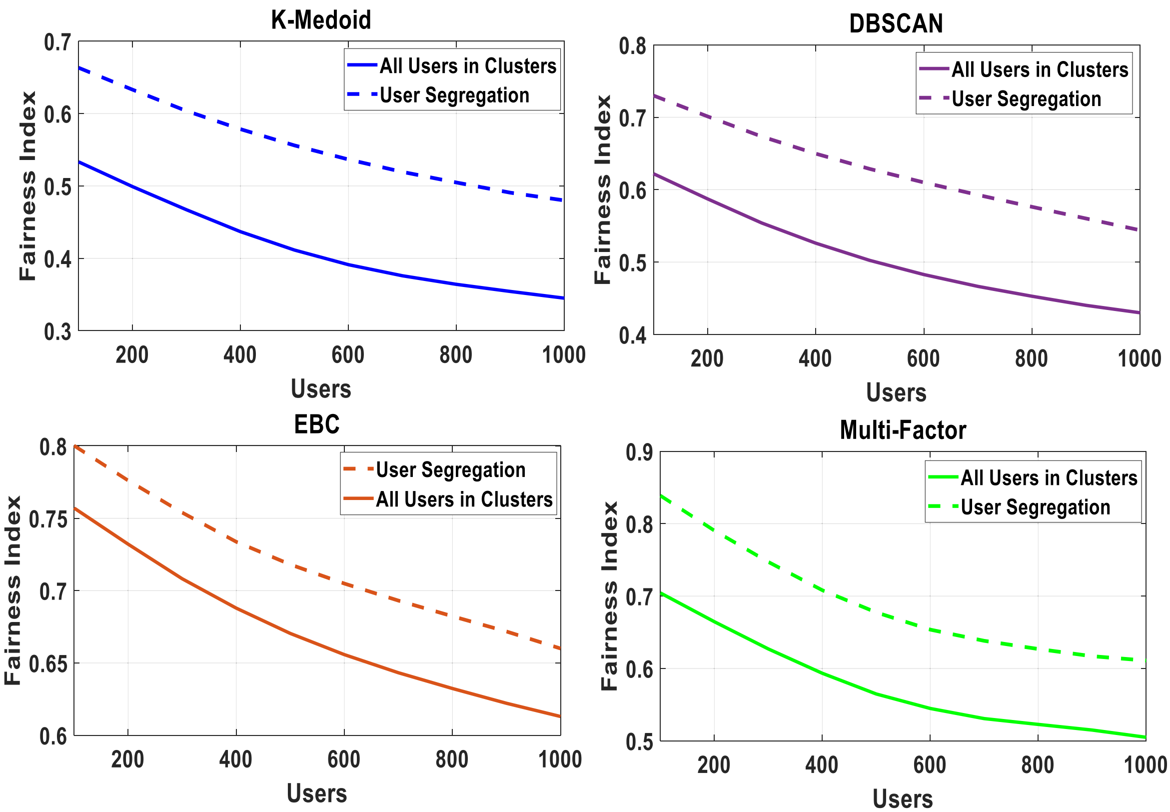


FIGURE 13. Throughput fairness: A comparison.

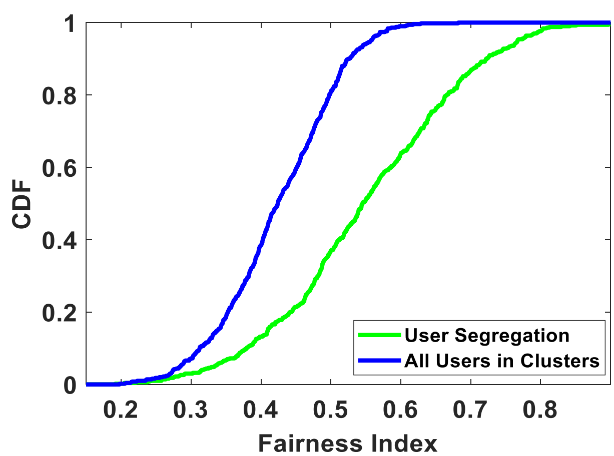


FIGURE 14. Throughput fairness CDF (100 Users). Similar trends were observed at other user densities with performance gap more pronounced.

average), as a result, distribution of throughput among the users is significantly improved therefore the system is fairer to the users.

The result presented in Fig. 14 shows the CDF of throughput fairness for the proposed scheme compared to conventional clustering. It is clear that after the 10^{th} percentile, fairness for the user segregation scenario is better in all percentiles. Therefore, better fairness is not achieved by only favoring a few users while neglecting a large number of users.

G. TRADE-OFF BETWEEN PERFORMANCE PARAMETERS AND eNB LOADING

Our simulations suggest that the performance parameters are getting significantly improved at the expense of eNB loading. Hence, it becomes necessary to select an appropriate loading factor depending on the capacity of the eNB. To present this trade-off, we randomly selected different percentages of users from the total number of users for which the performance was improved as a result of applying user segregation. We considered one hundred different random combinations, calculated the performance parameters for each combination, and then averaged the results. Fig. 15 shows two performance bounds; an upper bound of performance improvement (topmost curve, user segregation with 100% loading) meaning all users who have been identified to be better off with the eNB are being serviced by the eNB, and the lower bound (not segregated, all users in clusters). However, we may select a certain percentage of users according to the spare capacity of the eNB. The three middle curves represent three different loading factors, 10%, 50%, and 80% (bottom to top: low loading, average loading, and high loading) of the total users for which the performance was improved. Greater the loading factor, greater is the performance improvement. Therefore, it presents an opportunity for the cellular network to select a particular loading factor and trade it off with an improvement in the performance.

It should be noted that the result shown in Fig. 15 considers randomly selected users so it might disadvantage some users

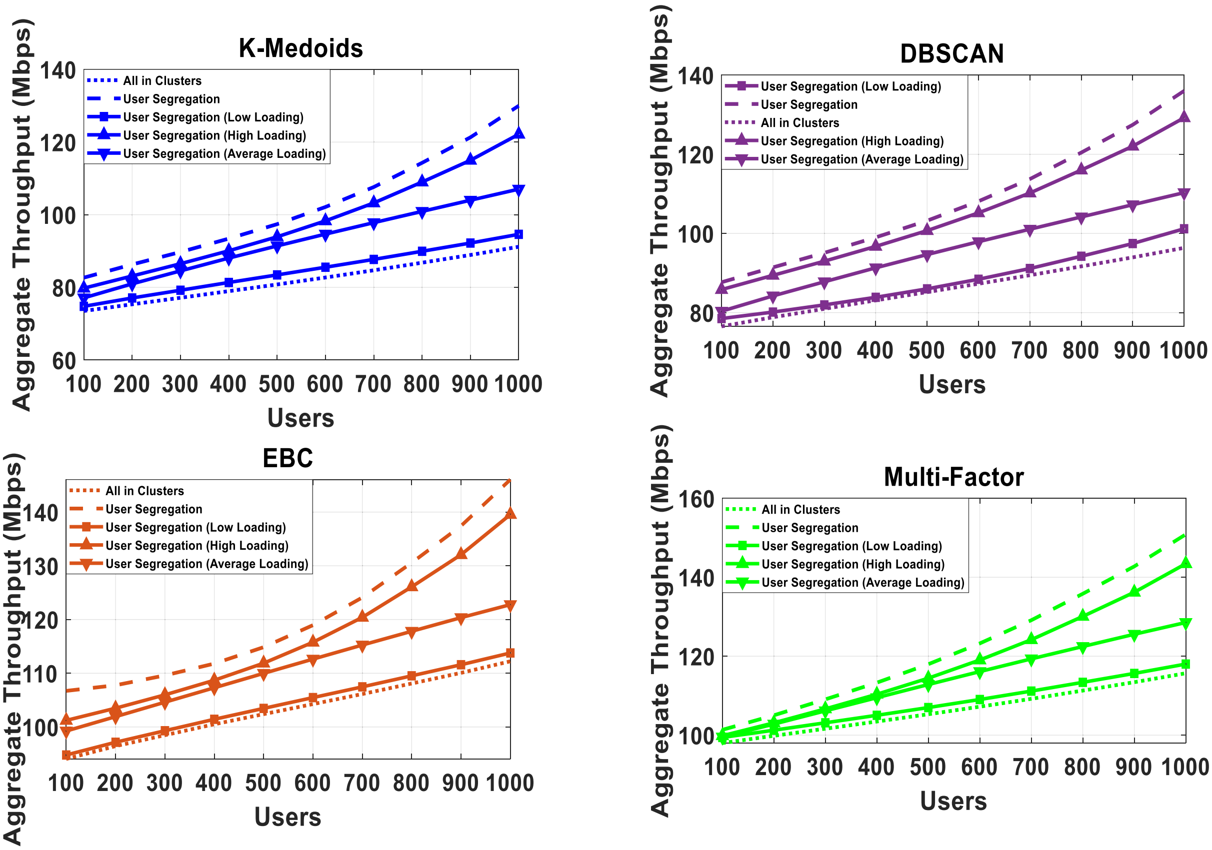


FIGURE 15. Improvement in aggregate throughput for different loading factors.

TABLE 9. Random to best selection: Performance improvement for different parameters.

Clustering Scheme	Aggregate Throughput (% Increase)			Energy Consumption (% Decrease)			Throughput Fairness (% Increase)		
	Low Loading	Average Loading	High Loading	Low Loading	Average Loading	High Loading	Low Loading	Average Loading	High Loading
K-Medoid	2.74	3.03	3.59	1.11	1.66	2.18	1.88	2.11	2.67
DBSCAN	1.75	3.13	3.46	1.29	1.78	2.28	1.76	2.07	2.49
EBC	1.81	3.31	3.22	0.96	1.21	1.77	1.01	1.67	2.11
MF	0.83	2.79	2.86	0.67	0.98	1.27	1.56	1.97	2.32

that are having the best performance. Therefore, to investigate this, we selected the same percentages for best users (i.e. the users that have the best improvement of all the users) and compared the performance with the random selection. Table 9 shows performance improvement is not significant, as we change the selection criteria from random to best users. Moreover, this percentage increase in performance is reported at a user density of one thousand. Therefore, this improvement will be even less at other user densities considered in this study. Similar marginal benefits (can be seen in Table 9) were observed for the other performance parameters i.e. energy consumption and throughput fairness. Therefore, we can conclude that a binary classifier is adequate, and training a multiclass classifier is not warranted.

VI. CONCLUSION

In this paper, we have proposed a mixed-mode clustering scheme based on user segregation. The concept relies on the fact that all users should not be part of a cluster as there are always some users that are better served by the eNB. We applied ML algorithms to perform this classification and compared the accuracy of different classification techniques. SVM, RF, and DNN were found to be the most promising classifiers. The results shown for accuracy and ROC demonstrate the effectiveness of the trained algorithm for the proposed scheme. The trained model was tested on a D2D-enabled content-sharing multicasting scenario. As per the classification outcome, a portion of the users were directly fetching the required content from the eNB. The results presented the impact of the proposed scheme on performance parameters such as throughput, energy consumption, and fairness. A comprehensive comparative study was conducted to demonstrate that irrespective of the clustering technique, performance gets improved by applying the proposed segregation scheme. Specifically, at a node density of one thousand, throughput gets significantly improved by 42% for K-Medoids, 34% for DBSCAN, and approximately 30% for the other two clustering schemes. Energy consumption was reduced by; 17.67% for K-Medoids, 16.5% for DBSCAN, 14.77% for EBC, and 13.66% for the Multi-Factor scheme. Throughput fairness showed improvement by 26.3% for K-Medoids, 26.5% for DBSCAN, 13.5% for EBC, and 20% for the fourth scheme. As a result of segregation some users communicate directly to the eNB, therefore we presented a tradeoff in performance improvement for various loading factors. The margin of improvement can be selected based on eNB's loading capability and spare capacity. This study also demonstrated that as compared to DNN, Random Forest and SVM perform better with relatively smaller training samples subject to classification scenarios.

Going further, it would be interesting to compare the results with the ones obtained from ML algorithms trained on live network data. Moreover, the proposed method is evaluated for multimedia application and it should be explored for other applications as well.

REFERENCES

[1] *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update 2015–2020 White Paper*, I. Cisco, San Jose, CA, USA, 2015.
[2] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.
[3] B. A. Ramanan, L. M. Drabeck, M. Haner, N. Nithi, T. E. Klein, and C. Sawkar, "Cacheability analysis of HTTP traffic in an operational LTE network," in *Proc. Wireless Telecommun. Symp. (WTS)*, Apr. 2013, pp. 1–8.
[4] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 142–149, Apr. 2013.
[5] R. Ullah, M. A. U. Rehman, M. A. Naeem, B.-S. Kim, and S. Mastorakis, "ICN with edge for 5G: Exploiting in-network caching in ICN-based edge computing for 5G networks," *Future Gener. Comput. Syst.*, vol. 111, pp. 159–174, Oct. 2020.
[6] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8402–8413, Dec. 2013.
[7] N. Golrezaei, A. G. Dimakis, and A. F. Molisch, "Scaling behavior for device-to-device communications with distributed caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4286–4298, Jul. 2014.
[8] N. Giatsoglou, K. Ntontin, E. Kartsakli, A. Antonopoulos, and C. Verikoukis, "D2D-aware device caching in mmWave-cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 9, pp. 2025–2037, Sep. 2017.
[9] M. N. Tehrani, M. Uysal, and H. Yanikomeroglu, "Device-to-device communication in 5G cellular networks: Challenges, solutions, and future directions," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 86–92, May 2014.
[10] R. Xu and D. Wunsch, "Survey of clustering algorithms," *IEEE. Trans. Neural Netw.*, vol. 16, no. 3, pp. 645–678, May 2005.
[11] D. Cheng, Q. Zhu, J. Huang, Q. Wu, and Y. Lijun, "Clustering with local density peaks-based minimum spanning tree," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 2, pp. 374–387, Feb. 2021.
[12] M. Du, S. Ding, Y. Xue, and Z. Shi, "A novel density peaks clustering with sensitivity of local density and density-adaptive metric," *Knowl. Inf. Syst.*, vol. 59, no. 2, pp. 285–309, May 2019.
[13] M. Du, S. Ding, and Y. Xue, "A robust density peaks clustering algorithm using fuzzy neighborhood," *Int. J. Mach. Learn. Cybern.*, vol. 9, no. 7, pp. 1131–1140, Jul. 2018.
[14] M. Du, S. Ding, X. Xu, and Y. Xue, "Density peaks clustering using geodesic distances," *Int. J. Mach. Learn. Cybern.*, vol. 9, no. 8, pp. 1335–1349, Aug. 2018.
[15] M. F. Khan, K.-L.-A. Yau, R. M. Noor, and M. A. Imran, "Survey and taxonomy of clustering algorithms in 5G," *J. Netw. Comput. Appl.*, vol. 154, Mar. 2020, Art. no. 102539.
[16] N. Ul Hasan, W. Ejaz, N. Ejaz, H. S. Kim, A. Anpalagan, and M. Jo, "Network selection and channel allocation for spectrum sharing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 980–992, 2016.
[17] C. Xu, L. Song, Z. Han, Q. Zhao, X. Wang, X. Cheng, and B. Jiao, "Efficiency resource allocation for device-to-device underlay communication systems: A reverse iterative combinatorial auction based approach," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 348–358, Sep. 2013.
[18] I. Filippini, A. E. C. Redondi, and A. Capone, "Beyond cellular green generation: Potential and challenges of the network separation," *Mobile Inf. Syst.*, vol. 2017, pp. 1–11, Jan. 2017.
[19] A. Tanveer, Z. U. Khan, A. N. Malik, and I. M. Qureshi, "Flexible queuing model for number of active users in cognitive radio network environment," *Wireless Commun. Mobile Comput.*, vol. 2018, pp. 1–6, Dec. 2018.
[20] Y. Li, D. Jin, J. Yuan, and Z. Han, "Coalitional games for resource allocation in the device-to-device uplink underlaying cellular networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3965–3977, Jul. 2014.
[21] Y. Zhao, Y. Li, X. Chen, and N. Ge, "Joint optimization of resource allocation and relay selection for network coding aided device-to-device communications," *IEEE Commun. Lett.*, vol. 19, no. 5, pp. 807–810, May 2015.
[22] Y. Zhou, H. Liu, Z. Pan, L. Tian, J. Shi, and G. Yang, "Two-stage cooperative multicast transmission with optimized power consumption and guaranteed coverage," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 2, pp. 274–284, Feb. 2014.
[23] L. Xu, G. M. P. O'Hare, and R. Collier, "A balanced energy-efficient multihop clustering scheme for wireless sensor networks," in *Proc. 7th IFIP Wireless Mobile Netw. Conf. (WMNC)*, May 2014, pp. 1–8.
[24] L. Militano, M. Condoluci, G. Araniti, A. Molinaro, A. Iera, and G.-M. Muntean, "Single frequency-based device-to-device-enhanced video delivery for evolved multimedia broadcast and multicast services," *IEEE Trans. Broadcast.*, vol. 61, no. 2, pp. 263–278, Jun. 2015.
[25] A. Asadi and V. Mancuso, "Network-assisted outband D2D-clustering in 5G cellular networks: Theory and practice," *IEEE Trans. Mobile Comput.*, vol. 16, no. 8, pp. 2246–2259, Aug. 2017.
[26] S. Bassoy, M. Jaber, M. A. Imran, and P. Xiao, "Load aware self-organising user-centric dynamic CoMP clustering for 5G networks," *IEEE Access*, vol. 4, pp. 2895–2906, 2016.
[27] Z. Xia, J. Yan, and Y. Liu, "Cooperative content delivery in multicast multihop device-to-device networks," *IEEE Access*, vol. 5, pp. 6314–6324, 2017.
[28] Y. Niu, Y. Liu, Y. Li, X. Chen, Z. Zhong, and Z. Han, "Device-to-device communications enabled energy efficient multicast scheduling in mmWave small cells," *IEEE Trans. Commun.*, vol. 66, no. 3, pp. 1093–1109, Mar. 2018.
[29] K. Kitagawa, H. Homma, Y. Suegara, and Y. Kishi, "A user selection algorithm for D2D multicast communication underlaying cellular systems," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2017, pp. 1–6.
[30] H. Ren, F. Jiang, and H. Wang, "Resource allocation based on clustering algorithm for hybrid device-to-device networks," in *Proc. 9th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2017, pp. 1–6.
[31] M. M. Tulu, R. Hou, C. Li, and M. D. Amentie, "Cluster head selection method for content-centric mobile social network in 5G," *IET Commun.*, vol. 12, no. 4, pp. 402–408, Mar. 2018.

[32] L. Xu, G. O'Hare, and R. Collier, ''A smart and balanced energy-efficient multihop clustering algorithm (smart-BEEM) for MIMO IoT systems in future networks,'' *Sensors*, vol. 17, no. 7, p. 1574, Jul. 2017.
[33] J. Huang, P. Zhou, K. Luo, Z. Yang, and G. He, ''Two-stage resource allocation scheme for three-tier ultra-dense network,'' *China Commun.*, vol. 14, no. 10, pp. 118–129, Oct. 2017.
[34] Y.-Y. Li, Z.-G. Ma, W.-B. Yan, and L.-Y. Liu, ''Weighted graph based clustering and local mobility management for dense small cell network with X2 interface,'' *Wireless Pers. Commun.*, vol. 95, no. 4, pp. 3771–3783, Aug. 2017.
[35] X. Duan, Y. Liu, and X. Wang, ''SDN enabled 5G-VANET: Adaptive vehicle clustering and beamformed transmission for aggregated traffic,'' *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 120–127, Jul. 2017.
[36] E. Yaacoub, ''Cooperative energy efficient D2D clustering in LTE–A with enhanced QoS,'' *Telecommun. Syst.*, vol. 67, no. 3, pp. 401–414, Mar. 2018.
[37] Y. Zhao and W. Song, ''Energy-aware incentivized data dissemination via wireless D2D communications with weighted social communities,'' *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 4, pp. 945–957, Dec. 2018.
[38] Z. Khan and P. Fan, ''A multi-hop moving zone (MMZ) clustering scheme based on cellular-V2X,'' *China Commun.*, vol. 15, no. 7, pp. 55–66, Jul. 2018.
[39] S. Sharafeddine and O. Farhat, ''A proactive scalable approach for reliable cluster formation in wireless networks with D2D offloading,'' *Ad Hoc Netw.*, vol. 77, pp. 42–53, Aug. 2018.
[40] W. Qi, Q. Song, X. Wang, L. Guo, and Z. Ning, ''SDN-enabled social-aware clustering in 5G-VANET systems,'' *IEEE Access*, vol. 6, pp. 28213–28224, 2018.
[41] L. Yin, J. Gui, and Z. Zeng, ''Improving energy efficiency of multimedia content dissemination by adaptive clustering and D2D multicast,'' *Mobile Inf. Syst.*, vol. 2019, pp. 1–16, Mar. 2019.
[42] M. Yin, W. Li, L. Feng, P. Yu, and X. Qiu, ''Emergency communications based on throughput-aware D2D multicasting in 5G public safety networks,'' *Sensors*, vol. 20, no. 7, p. 1901, Mar. 2020.
[43] S. Aslam, F. Alam, S. F. Hasan, and M. Rashid, ''A novel weighted clustering algorithm supported by a distributed architecture for D2D enabled content-centric networks,'' *Sensors*, vol. 20, no. 19, p. 5509, Sep. 2020.
[44] H. Kaneko and K. Funatsu, ''Application of online support vector regression for soft sensors,'' *AIChE J.*, vol. 60, no. 2, pp. 600–612, Feb. 2014.
[45] M. A. Hearst, S. T. Dumais, E. Osman, J. Platt, and B. Scholkopf, ''Support vector machines,'' *IEEE Intell. Syst. Appl.*, vol. 13, no. 4, pp. 18–28, Jul./Aug. 2008.
[46] L. Breiman, ''Random forests,'' *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[47] X. Yuan, B. Huang, Y. Wang, C. Yang, and W. Gui, ''Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE,'' *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 3235–3243, Jul. 2018.
[48] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, ''Application of machine learning in wireless networks: Key techniques and open issues,'' *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3072–3108, 4th Quart., 2019.
[49] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, ''Machine learning in wireless sensor networks: Algorithms, strategies, and applications,'' *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 1996–2018, 4th Quart., 2014.
[50] J. Wang, C. Jiang, H. Zhang, Y. Ren, K.-C. Chen, and L. Hanzo, ''Thirty years of machine learning: The road to Pareto-optimal wireless networks,'' *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 1472–1514, 3rd Quart., 2020.
[51] Y. Wang, K. Wu, and L. M. Ni, ''WiFall: Device-free fall detection by wireless networks,'' *IEEE Trans. Mobile Comput.*, vol. 16, no. 2, pp. 581–594, Feb. 2017.
[52] R. Shafin, L. Liu, V. Chandrasekhar, H. Chen, J. Reed, and J. C. Zhang, ''Artificial intelligence-enabled cellular networks: A critical path to beyond-5G and 6G,'' *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 212–217, Apr. 2020.
[53] S. Y. Han, N. B. Abu-Ghazaleh, and D. Lee, ''Efficient and consistent path loss model for mobile network simulation,'' *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1774–1786, Jun. 2016.
[54] J. Liu, R. Deng, S. Zhou, and Z. Niu, ''Seeing the unobservable: Channel learning for wireless communication networks,'' in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2015, pp. 1–6.
[55] S. Hu, Y.-D. Yao, and Z. Yang, ''MAC protocol identification using support vector machines for cognitive radio networks,'' *IEEE Wireless Commun.*, vol. 21, no. 1, pp. 52–60, Feb. 2014.
[56] M. G. Kibria, K. Nguyen, G. P. Villardi, O. Zhao, K. Ishizu, and F. Kojima, ''Big data analytics, machine learning, and artificial intelligence in next-generation wireless networks,'' *IEEE Access*, vol. 6, pp. 32328–32338, 2018.
[57] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, ''Artificial neural networks-based machine learning for wireless networks: A tutorial,'' *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3039–3071, 4th Quart., 2019.
[58] R. D. Timoteo, D. C. Cunha, and G. D. Cavalcanti, ''A proposal for path loss prediction in urban environments using support vector regression,'' in *Proc. Adv. Int. Conf. Telecommun*, 2014, pp. 1–5.
[59] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, ''Machine learning paradigms for next-generation wireless networks,'' *IEEE Wireless Commun.*, vol. 24, no. 2, pp. 98–105, Apr. 2017.
[60] M. Polese, R. Jana, V. Kounev, K. Zhang, S. Deb, and M. Zorzi, ''Machine learning at the edge: A data-driven architecture with applications to 5G cellular networks,'' *IEEE Trans. Mobile Comput.*, early access, Jun. 3, 2020, doi: 10.1109/TMC.2020.2999852.
[61] S. Zhang, J. Liu, H. Guo, M. Qi, and N. Kato, ''Envisioning device-to-device communications in 6G,'' *IEEE Netw.*, vol. 34, no. 3, pp. 86–91, May 2020.
[62] Z. Na, Y. Liu, J. Shi, C. Liu, and Z. Gao, ''UAV-supported clustered NOMA for 6G-enabled Internet of Things: Trajectory planning and resource allocation,'' *IEEE Internet Things J.*, early access, Jun. 23, 2020, doi: 10.1109/JIOT.2020.3004432.
[63] X. Wang, J. Wang, Y. Xu, J. Chen, L. Jia, X. Liu, and Y. Yang, ''Dynamic spectrum anti-jamming communications: Challenges and opportunities,'' *IEEE Commun. Mag.*, vol. 58, no. 2, pp. 79–85, Feb. 2020.
[64] X. Wang, Y. Xu, C. Chen, X. Yang, J. Chen, L. Ruan, Y. Xu, and R. Chen, ''Machine learning empowered spectrum sharing in intelligent unmanned swarm communication systems: Challenges, requirements and solutions,'' *IEEE Access*, vol. 8, pp. 89839–89849, 2020.
[65] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, and W. Xiang, ''Big data-driven optimization for mobile networks toward 5G,'' *IEEE Netw.*, vol. 30, no. 1, pp. 44–51, Jan. 2016.
[66] M. Najla, Z. Becvar, P. Mach, and D. Gesbert, ''Predicting device-to-device channels from cellular channel measurements: A learning approach,'' 2019, *arXiv:1911.07191*. [Online]. Available: http://arxiv.org/abs/1911.07191
[67] T. O'Shea and J. Hoydis, ''An introduction to deep learning for the physical layer,'' *IEEE Trans. Cognit. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.
[68] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, ''Deep learning for wireless physical layer: Opportunities and challenges,'' *China Commun.*, vol. 14, no. 11, pp. 92–111, Nov. 2017.
[69] X. Cheng, L. Fang, L. Yang, and S. Cui, ''Mobile big data: The fuel for data-driven wireless,'' *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1489–1516, Oct. 2017.
[70] S. Qian, H. Liu, C. Liu, S. Wu, and H. S. Wong, ''Adaptive activation functions in convolutional neural networks,'' *Neurocomputing*, vol. 272, pp. 204–212, Jan. 2018.
[71] A. Liaw and M. Wiener, ''Classification and regression by randomforest,'' *R News*, vol. 2, no. 3, pp. 18–22, 2002.
[72] P. O. Gislason, J. A. Benediktsson, and J. R. Sveinsson, ''Random forest classification of multisource remote sensing and geographic data,'' in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, vol. 2, Sep. 2004, pp. 1049–1052.
[73] M. Pal, ''Random forest classifier for remote sensing classification,'' *Int. J. Remote Sens.*, vol. 26, no. 1, pp. 217–222, Jan. 2005.
[74] K. K. Dobbin and R. M. Simon, ''Optimally splitting cases for training and testing high dimensional classifiers,'' *BMC Med. Genomics*, vol. 4, no. 1, p. 31, Dec. 2011.
[75] G. James, D. Witten, and R. T. T. Hastie, *An Introduction to Statistical Learning*. New York, NY, USA: Springer, 2013.
[76] A. Ometov, A. Orsino, L. Militano, D. Moltchanov, G. Araniti, E. Olshannikova, G. Fodor, S. Andreev, T. Olsson, A. Iera, J. Torsner, Y. Koucheryavy, and T. Mikkonen, ''Toward trusted, social-aware D2D connectivity: Bridging across the technology and sociality realms,'' *IEEE Wireless Commun.*, vol. 23, no. 4, pp. 103–111, Aug. 2016.
[77] *3GPP TSG RAN Physical Layer Aspects For Evolved UTRA, V7. 1.0*, document TR 25.814, 3GPP, 2006.
[78] K. Mahmud, M. Inoue, H. Murakami, M. Hasegawa, and H. Morikawa, ''Energy consumption measurement of wireless interfaces in multi-service user terminals for heterogeneous wireless networks,'' *IEICE Trans. Commun.*, vol. E88-B, no. 3, pp. 1097–1110, Mar. 2005.

[79] E. Yaacoub and O. Kubbar, "Energy-efficient device-to-device communications in LTE public safety networks," in *Proc. IEEE Globecom Workshops*, Dec. 2012, pp. 391–395.
[80] D. Stathakis, "How many hidden layers and nodes?" *Int. J. Remote Sens.*, vol. 30, no. 8, pp. 2133–2147, Apr. 2009.
[81] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 281–305, Feb. 2012.
[82] H. Yang, X. Xie, and M. Kadoch, "Machine learning techniques and a case study for intelligent wireless networks," *IEEE Netw.*, vol. 34, no. 3, pp. 208–215, May 2020.
[83] M. J. Kearns, *The Computational Complexity of Machine Learning*. Cambridge, MA, USA: MIT Press, 1990.
[84] S. Sra, S. Nowozin, and S. J. Wright, *Optimization for Machine Learning*. Cambridge, MA, USA: MIT Press, 2012.
[85] Y. Zhao, C. Shi, T. Yang, J. Du, Y. Zang, D. Wang, X. Chen, L. Wang, and Z. Zhang, "Low-complexity and joint modulation format identification and OSNR estimation using random forest for flexible coherent receivers," *Opt. Commun.*, vol. 457, Feb. 2020, Art. no. 124698.
[86] N. Ahad, J. Qadir, and N. Ahsan, "Neural networks in wireless networks: Techniques, applications and guidelines," *J. Netw. Comput. Appl.*, vol. 68, pp. 1–27, Jun. 2016.
[87] K. Li, Y. Wu, Y. Nan, P. Li, and Y. Li, "Hierarchical multi-class classification in multimodal spacecraft data using DNN and weighted support vector machine," *Neurocomputing*, vol. 259, pp. 55–65, Oct. 2017.
[88] S. Lukasik, P. A. Kowalski, M. Charytanowicz, and P. Kulczycki, "Clustering using flower pollination algorithm and Calinski-Harabasz index," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 2724–2728.
[89] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination," Eastern Res. Lab., Digit. Equip. Corp., Hudson, MA, USA, Tech. Rep. DEC-TR-301, 1984.

SAAD ASLAM received the B.S. (Hons.) and M.S. degrees in electrical engineering from CIIT, Lahore, Pakistan, in 2010 and 2014, respectively. He is currently pursuing the Ph.D. degree with Massey University, Auckland, New Zealand. He is also working as a Lecturer with the School of Professional Engineering, Manukau Institute of Technology, Auckland. His research interests include D2D communication, clustering algorithms, 5G and machine learning. He received the Gold Medal for his M.S. degree.

FAKHRUL ALAM (Senior Member, IEEE) received the B.Sc. degree (Hons.) in electrical and electronic engineering from BUET, Dhaka, Bangladesh, in 1996, and the M.S. and Ph.D. degrees in electrical engineering from Virginia Tech, Blacksburg, VA, USA, in 1999 and 2002, respectively. He is currently an Associate Professor with the Department of Mechanical & Electrical Engineering, School of Food & Advanced Technology, Massey University, Auckland, New Zealand. His research interest includes indoor localization, 5G and visible light communication, IoT and wireless sensor networks.

SYED FARAZ HASAN (Senior Member, IEEE) received the bachelor's degree (Hons.) in electrical engineering from the NED University of Engineering and Technology, Pakistan, in 2008, and the Ph.D. degree from Ulster University, U.K., in 2011, on Vice Chancellor's Research Scholarship. He was a Postdoctoral Researcher with the Korea Advanced Institute of Science and Technology (KAIST), South Korea, from 2011 to 2012, and an Assistant Professor with Sungkyunkwan University, South Korea, from 2012 to 2014. He is currently a Senior Lecturer with the Department of Electrical and Mechanical Engineering, Massey University, New Zealand, where he leads the Telecommunication and Network Engineering research group. His research interests include device-to-device communication, mm-wave communication, and energy harvesting and localization. He is also the Distinguished Speaker of ACM, and the Impact Creator of IEEE.

MOHAMMAD A. RASHID (Senior Member, IEEE) received the M.Sc. (engineering) degree in electronics engineering with a specializing in engineering cybernetics systems from the Wroclaw University of Science and Technology, in 1978, and the Ph.D. degree from the University of Strathclyde, U.K., in 1986. He is currently a Senior Lecturer with the Department of Mechanical & Electrical Engineering, Massey University, New Zealand. His research interests include embedded systems and the Internet of Things, network access protocols, the IoT security and blockchain, big data analytics, and ERP/enterprise systems.