



# JUIndoorLoc: A Ubiquitous Framework for Smartphone-Based Indoor Localization Subject to Context and Device Heterogeneity

Priya Roy<sup>1</sup> · Chandreyee Chowdhury<sup>1</sup> · Dip Ghosh<sup>2</sup> · Sanghamitra Bandyopadhyay<sup>2</sup>

Published online: 19 February 2019

© Springer Science+Business Media, LLC, part of Springer Nature 2019

## Abstract

A new era of ubiquitous indoor location awareness is on the horizon especially for context sensing, ambient assisted living and many other smart city applications. Although indoor localization plays a pivotal role in making the environment smarter, it is still very difficult to compare state-of-the-art localization algorithms due to the scarcity of standard databases. Publicly available databases are neither fine-grained nor contain data for different conditions. Received Signal Strength Indicator (*RSSI*) of *Wi-Fi* signals vary with indoor environment (open/closed room, presence/absence of user, temperature etc.) and scanning smart hand-held devices. Thus, localization accuracy varies with various environmental conditions and also granularity of location points (cell). Consequently, in this paper, our contribution is two-fold. First, we present a comprehensive indoor localization dataset, subject to different domains-spatial, temporal, context and device. *RSSI* data has been collected with cell sizes as small as 1 m × 1 m from three floors of a building of our University using an Android application built for this purpose. This multi-floor dataset is available online at [https://drive.google.com/open?id=1\\_z1qhoRIcpineP9AHkfVGCfB2Fd\\_e-fD](https://drive.google.com/open?id=1_z1qhoRIcpineP9AHkfVGCfB2Fd_e-fD). Our experimental results show that maximum of 71.78% classification accuracy can be achieved for state-of-the-art classifiers when training and testing samples are taken in different environmental conditions and from smartphones having different configurations. Single classifier cannot easily be modified to suit these variations without losing its generality. So, to overcome these conditional dependencies, our second contribution is to propose a framework for indoor localization, *JUIndoorLoc* and design an ensemble of condition specific classifiers as part of the framework to take care of context and device heterogeneity. Consequently, this ensemble of condition specific classifiers is implemented and found to predict a location with 91.74% accuracy (1.87 m) for our dataset.

**Keywords** Indoor localization · Machine learning · *RSSI* · *Wi-Fi* · Ensemble classifier · KNN

---

✉ Chandreyee Chowdhury  
chandreyee.chowdhury@gmail.com

Extended author information available on the last page of the article

## 1 Introduction

A spectacular growth of indoor localization studies can be witnessed today as many real-world applications including smart home, smart campus, disaster management need to localize user in order to provide services. The goal of indoor localization is to identify the exact location of a smart device within a multi-storied building. This domain is enhanced by many researcher's effort from past two decades. The *GPS* system does not work in indoor environment properly as signal strength fluctuates badly and location update may not be received by user's device at regular intervals [1, 2]. Nowadays, every location of most buildings (like universities, hospitals, offices, shopping malls etc.) are covered by Wireless LAN, thus indoor localization based on *Wi-Fi* (IEEE 802.11 *WLAN* standard) signal strength or *RSSI* has become a prominent approach as it does not require any additional hardware devices to be installed in a building.

Most of the *WLAN* localization systems need two phases: training and testing. In training phase, a radio map of an area is constructed based on *RSSI* values of available Access Points (*APs*) as scanned by smartphones. Later, in the testing phase, received signal strength of the available *APs* are recorded at a current instance and predict current location of the smartphone based on the data collected in training phase [3].

Though *RSSI* fingerprinting based localization technique is a popular approach, substantial challenges still exist. In an indoor environment, *RSSI* values frequently change with environmental conditions even when data is collected by the same device at the same location [4]. Environmental conditions include opening/closing of doors and/or windows, humidity and temperature variation, presence and absence of people and other interfering devices [5]. Moreover, the heterogeneity of devices is also an important factor. Not only signal strength varies from one device to the other, but devices may also detect different sets of *APs* [6] from the same location. Thus, indoor localization techniques should be validated against datasets that pertains to different environmental conditions. However, there is a serious lack of publicly available database in this field. Each approach presents its estimated results using its own database and describes how the experiments are carried out [7]. Under these circumstances, comparing different methods are not possible since the distinctiveness of each experiment is hardly reproducible. In the Pattern Recognition and Machine Learning research fields, the common practice is to test the results of the works by applying it on a widely accepted standardized dataset (like the datasets available in *UCI* Machine Learning Repository [8]) or providing the dataset for other researchers to compare. Thus different methodologies can be fairly compared on a common ground.

Datasets for different devices under the same context or that of multiple buildings can be found in [9–13]. As a room is considered to be a location point, accuracy with respect to a room was given in [9]. Moreover, the authors in [9] reported that the signal strengths vary with respect to a given device at different times and at a certain time with different devices. This motivates the importance of having a dataset that comprehensively presents *RSSI* data under different ambiance and device configurations. Additionally, in order to achieve better accuracy, location points need to be more fine-grained.

Generally, the ambiance and device configuration vary at the time of train and test data collection. These variations typically affect *RSSI* as discussed earlier. As a result, the classification accuracy of a base learner is also degraded. An ensemble model containing a number of base learners has a generalization ability, which is usually much stronger than individual base learners. Few ensemble based localization methods have

been proposed in [14–16] without considering indoor contextual heterogeneity. The location points considered in these works are mostly not fine-grained either.

Consequently, in this paper, our main contribution is twofold.

- First, we design and present a benchmark dataset for indoor localization subject to temporal, context (like open/closed room and presence/absence of user) and device heterogeneity.
- In this regard, the second contribution is to propose a ubiquitous indoor localization framework, JUIndoorLoc that incorporates an ensemble technique for condition-based classifiers. This is designed to mitigate the effect of these heterogeneities while predicting the location of a user/device. The framework is ubiquitous as it does not require any specialized hardware but uses the existing Wi-Fi infrastructure of the buildings and smartphones commonly used by people.

To efficiently deal with the problem, the main characteristics of the dataset are:

- Data are captured with  $1\text{ m} \times 1\text{ m}$  cell size from three floors of a building so that different *RSSI* patterns of rooms, laboratories, corridors and stairs can be investigated.
- The numbers of *APs* appearing in the dataset are 172.
- Total 1000 location points from three floors are covered.
- *RSSI* data has been collected by 4 Android devices with different configurations.

An Android application for *Wi-Fi* data collection is built to collect *Wi-Fi* footprints from different areas. The application takes a floor plan as input and partitions the whole area into a number of cells as specified by the user. Each cell acts as a location point that is assigned a unique identifier and *RSSI* values are collected for those location points. The application is generic and can be configured for any specific map or cell size.

Rest of the paper is organized as follows. Section 2 discusses main features and limitations of other available datasets. Design of proposed *JUIndoorLoc* is described in Sect. 3 while Sect. 4 describes the design of a condition-based ensemble classifier. Section 5 analyses the results of state-of-the-art classifiers and condition-based ensemble classifier. Finally, Sect. 6 concludes the paper.

## 2 Related Work

This section is organized as follows. First, we discuss some prior works related to indoor positioning. Then we briefly describe main features and limitations of some publicly available indoor localization datasets.

### 2.1 Prior Works of Indoor Positioning

Indoor localization and positioning techniques are roughly classified into two categories: (i) *Statistical approach* and (ii) *Machine Learning approach*.

### 2.1.1 Statistical Approach

*RADAR*, a user localization and tracking system for indoor environment, is proposed in [17]. It used the nearest neighbor technique in signal space to determine location of a device. In this work, authors showed 2–3 m accuracy in the proposed work. Another system *Horus* [18] played a major role in this field of research. It is a well known indoor localization system in statistical approach, which requires less computational resources. To achieve better accuracy, the authors proposed different modules to address the causes of wireless channel variations. The system used joint clustering techniques and probabilistic methods. The experimental results showed that this system could achieve 90% accuracy and maximum 2.1 m error. To reduce positioning error, a fuzzy control system, combining Dead Reckoning (DR) and *RSSI*, was proposed in [19] and reported an average error of 0.625 m. However, information about the experimental region and scanning devices were not provided. Using *Wi-Fi* and Bluetooth a novel fingerprint algorithm was proposed in [20]. It adopted log-distance path loss model for removing unwanted *RSSI* data. They had also used Hausdorff distance algorithm and median filter to minimize their database and eliminate unnecessary signals of *Wi-Fi APs*. They had found an improvement in positioning error by 0.695 m. Their experiment was conducted in a small indoor region mainly a corridor of  $2.3 \times 34.8$  m and a room of  $6.4 \times 5.2$  m without any obstacles. As fingerprinting is very time consuming, a quick radio fingerprint collection (QRFC) algorithm was proposed in [21] for collecting fingerprints rapidly using a smart-device and its' in-built motion sensor. The time needed for collecting samples was almost close to the time of walking slowly through a path. In their experiments, they found no notable difference in their accuracy. Although QRFC requires very less time to complete the whole task, the use of motion sensor incurs errors which cause incorrect step detection.

### 2.1.2 Machine Learning Approach

To apply machine learning techniques for indoor localization, a model is first generated based on collected fingerprints in offline phase. Then current location is predicted by the model based on the data collected in online phase. Many works can be found that uses single classifier as in [22–25] where the authors also apply heuristics about signal propagation. A graph-based indoor sub-area localization technique, *GraphLoc* is proposed in [26]. This work does not require any specialized device and the fingerprint map is generated by the crowd-sourcing approach. First, logical floor graph is constructed using inherent characteristics of *Wi-Fi* signals. Then, the problem of constructing fingerprint map is formulated as a graph mapping problem between logical floor graph and physical floor graph. Finally, in online phase, a Bayesian-based method is utilized to predict the unknown location. The reported average localization accuracy was 88.2%.

However, the changing signal characteristics for different conditions, such as device and context heterogeneity can be better captured by multi-classifier models such as ensemble. In [14], a novel technique to localize people in the indoor environment was proposed. A simple decision tree, J48G and a more advanced Soft Computing algorithm called Fuzzy Unordered Rule Induction Algorithm (*FURIA*) had considered as base classifiers in order to derive ensemble of classifiers. Their proposed work had implemented in two real-world indoor environments. An ensemble of some weak position estimators was proposed in [16] for developing a robust indoor positioning model. The area of a user was estimated first. Then the weight

of each weak estimator was updated according to the estimated area. Based on all of the weighted weak estimators, the user's location was computed. *Wi-Fi* fingerprints had collected from one floor (29.8 m × 16.3 m) of a building using one Android device. However, location points were not fine grained and selected in a random way. In *Loco* [15], authors use Bluetooth low energy (BLE) beacons along with *Wi-Fi* signals for better indoor positioning. The authors applied boosting for complexity reduction. Almost 56 BLE beacons were deployed on the ceilings of their experimental area. The signal strengths of 159 *Wi-Fi APs* were observed. Applying boosting method this system is found to achieve 96.6% accuracy at room level. Thus precise positioning within a room is not possible.

A condition-based ensemble classifier based on majority voting scheme is proposed in [27]. This scheme is implemented on a relatively small dataset for 2 m × 2 m cells. Only majority voting method is considered in the work and the dataset is not made public, it is neither sufficiently fine-grained, nor it considered several types of indoor regions like rooms and corridors etc.

## 2.2 Publicly Available Indoor Localization Datasets

Main features and limitations of some publicly available datasets are briefly described in Table 1. *UJIIndoorLoc* [9] is first publicly available database for indoor localization. Although this dataset covered a large area, a comprehensive benchmark was not provided. Only the results of simple *INN* algorithm were reported with 89.92% success rate. In *UJIIndoorLoc*, *RSSI* data was collected from the center of every room and from another position in front of every door. Hence, in this work, positioning error was huge, nearly 7.9 m. In [11], a magnetic database was published consisting of inertial sensor data. Unfortunately, these type of sensors generate errors which incur high noise in location prediction. Besides, the datasets published in [12, 13] contain *RSSI* specific data very small regions. In recent past, few crowd sourced indoor localization datasets [12, 30] are published. In [30] at the time of data collection the location of a user is taken as manual input from that user. As a result, incorrect location point labeling problem is occurred as the data had been captured by the crowd. Moreover, in the above mentioned datasets, *RSSI* data of various time instants and indoor environments were recorded. However, these datasets did not provide any information about the indoor environments in which the data samples were taken.

Thus, existing indoor localization techniques heavily depends on fingerprinting effort and more importantly, the fingerprints change due to environmental conditions. Though in many research works authors use their own experimental setups which are often impossible to reproduce and hence make the whole process hard to compare. In this context, the current paper proposes a fine grained multi-device dataset of *Wi-Fi* signals. Thus, the next section elaborates about the data collection mechanism and attributes of our dataset that we intend to publish for further enhancement. In order to show the effectiveness of the benchmark dataset, an ensemble of condition specific classifiers is also designed and tested on the dataset as described in subsequent sections.

## 3 Design of JUIndoorLoc

In this section, detailed design of *JUIndoorLoc* is presented. First, the framework of *JUIndoorLoc* is described followed by data collection process and description of pre-processing techniques applied on the data before storing it into the database. The dataset description along with the motivation of such design is also discussed.

**Table 1** Main features and limitation of other available Indoor localization datasets

Dataset	Main features	Limitations
UIJIndoorLoc [9]	The dataset consisted of 21,049 samples captured from 3 different buildings with 4 floors each by 21 devices. 933 reference points were covered and 520 APs were found in the dataset	Results of simple <i>INN</i> algorithm was provided. Comprehensive benchmark is missing
Movement prediction of user in indoor for AAL application [28]	RSS data was collected for Ambient Assisted Living (AAL) applications. A benchmark for user movement prediction in a real-life office environment using RSS data was introduced	RSS values of only 4 anchors were captured. Two labels were assigned according to the movement paths and binary classification was used to predict user movement
GEOTEC laboratory dataset [11]	This magnetic dataset was recorded by one device and consisted of 281 continuous samples; among them 270 were for training and 11 were for testing. It contained magnetometer, accelerometer and orientation sensor data. Covered area was approximately 200 m <sup>2</sup>	The sensor data were found to mostly depend on user velocity and the orientation of the device
IPIN 2016 Tutorial dataset [12]	This dataset consisted of 927 data sample for training and 702 for testings. The 177 attributes of this dataset contained the <i>Wi-Fi</i> fingerprint data from 168 APs, the <i>GPS</i> coordinates and other relevant information	Only a small corridor ( $\approx 120$ m <sup>2</sup> ) was covered as the aim of this dataset was to investigate a small scenario
ALCALA 2017 Tutorial dataset [13]	Data were collected in the same corridor as the IPIN 2016 Tutorial. The dataset consisted of 670 training samples and 405 testing samples. The 154 attributes of the dataset contained the <i>Wi-Fi</i> fingerprint data from 152 APs and the <i>GPS</i> coordinates where it was recorded	Some training fingerprints were incorrectly labeled for the purpose of testing. Thinner grid was used to capture data but the size of the grid is not mentioned
IPIN 2016 Off-Site Competition [29]	This dataset was used in an off-site competition of <i>IPIN 2016</i> conference. sensor and RSS samples were acquired from 4 buildings by multiple devices (approximately 6)	Data were collected from some trajectories and stored in the log files in a complex format. A parser, written in MATLAB, was provided to read the log files in a desirable format
The mannheim/compass dataset [13]	This <i>Wi-Fi</i> -based dataset was recorded from the second floor (221 m <sup>2</sup> area) of a building. Training data were captured from 130 reference points with a spacing of 1.5 m. Test data were captured from 46 reference points and selected in a random fashion. The test-bed was covered by 12 APs and additional 13 APs were installed	In the provided floor plan, data were collected only from corridor. Number of devices used to record data was not reported properly

**Table 1** (continued)

Dataset	Main features	Limitations
Wi-Fi Crowd-sourced dataset [30]	<p>This dataset was collected by 8 users and 21 different types of Android devices from different positions of a five storied building. Samples were taken from random position, i.e., not grid based. 991 APs were found in their dataset. The dataset is available as 4 'csv' files (Coordinate, RSS, Date and Device file)</p>	<p>It was mentioned that Wi-Fi hotspots were existed among 991 APs but these were not eliminated from their dataset. As these APs were not consistent they incur noise in the location prediction</p>

### 3.1 Framework of JUIndoorLoc

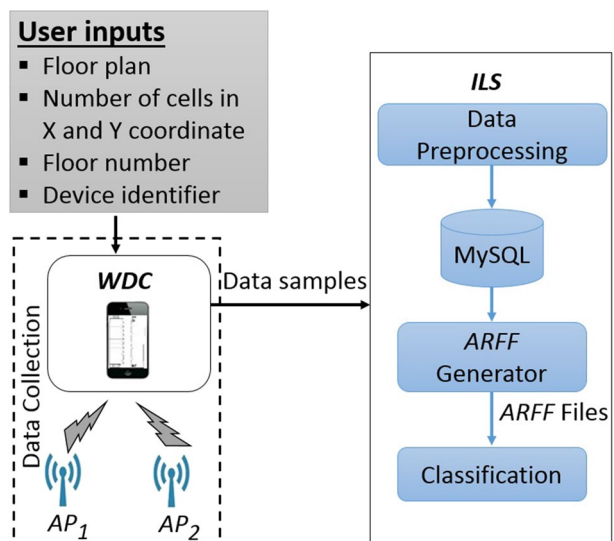
The major modules of this framework are *Wi-Fi Data Collector (WDC)* and *Indoor Localization Server (ILS)* as depicted in Fig. 1.

*WDC* application is used for collecting *RSSI* signals of every location points of an indoor environment and for transferring the collected data to the localization server, *ILS*, to create a knowledge base of the environment. *WDC* runs on Android devices and communicates with *ILS* using socket. Authorized users of this system can upload a floor plan of an area as an image as shown in Fig. 2. Administrator of an indoor environment can manage this user. The floor plan is divided into a number of cells (represented by boxes in the figure) according to the desired granularity. Thus, the area covered by a cell can be varied according to requirement. *RSSI* data is collected for each cell by tapping a cell on the grid shown in the *WDC* application. After tapping on a cell, data is captured for 120 seconds. Each cell is assigned a unique identifier by the application. The collected data is sent to *ILS* and stored in a *MySQL* database. This dataset is available from online in *.csv* file format. In server side, an application (*ARFF Generator*) is developed to generate *ARFF* (Attribute Relation File Format) file based on collected data. This generated file is taken as an input to different machine learning algorithms for classification.

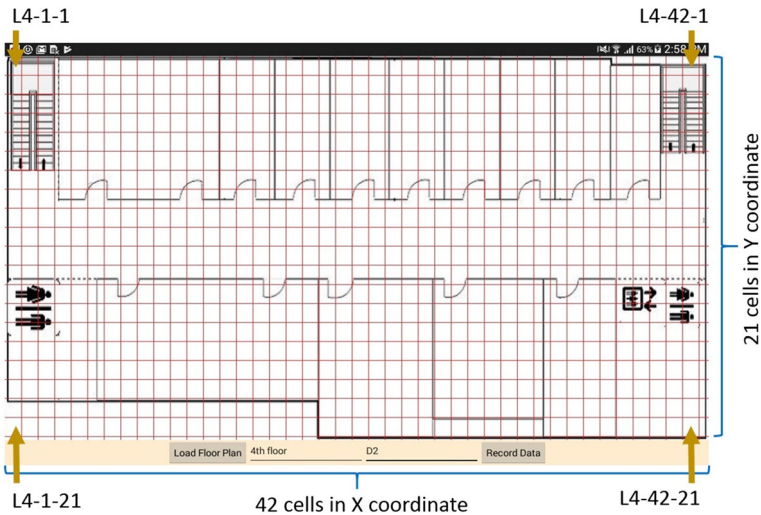
### 3.2 Data Collection

Training and test data have been collected for a duration of 31 days and 5 days respectively. As the indoor environment is not consistent, any existing *AP* can be removed or new *APs* can be installed. To deal with this kind of temporal and environmental changes and to identify stable *APs*, a gap of two month has been maintained between training and test data collection. Using *WDC* application, mobile device measures the signal strength of available *APs* from a location point and sends the data to a server to analyze. In this phase, the *RSSI* dataset is collected from all possible location point of the region and the recorded *RSSI*

**Fig. 1** Block diagram of the framework of JUIndoorLoc







**Fig. 2** Android based *Wi-Fi* Data Collector (*WDC*) application for collecting *RSSI* value in offline phase

values are stored along with collection time, location point identifier, device identifier and detailed features of *APs* like Basic Service Set Identifiers (*BSSIDs*) and Service Set Identifiers (*SSIDs*). At the time of data collection, it has been observed that at a specific location point and time instant, *RSSI* significantly varies with hardware. Signal strength also varies with time at same location and for a specific device. Moreover, different factors of the environment such as the presence of obstacles, the condition of weather as well as, the presence of human beings and other devices may also affect *Wi-Fi* signal strength. These observations motivate us to collect data for different perspectives so that robustness of localization algorithms can be analyzed with the dataset. To analyze the pattern of *RSSI* data, we have collected data in different perspectives as follows:

1. **Spatial:** Instead of collecting data from the center of a cell, data has been collected from different points around a cell (based on open space and obstacles).
2. **Temporal:** Data has been collected at different time of a day, to understand varying nature due to different environmental effects.
3. **Context:** Different contextual heterogeneities are considered while collecting data. These are given below:
  - (a) In open room and presence of human ( $C_1$ ),
  - (b) In open room and absence of human ( $C_2$ ),
  - (c) In closed room and presence of human ( $C_3$ ),
  - (d) In closed room and absence of human ( $C_4$ ).
4. **Device:** Four Android devices ( $D_1, D_2, D_3$  and  $D_4$ ) are used for data collection to understand the variation of the radio signal.

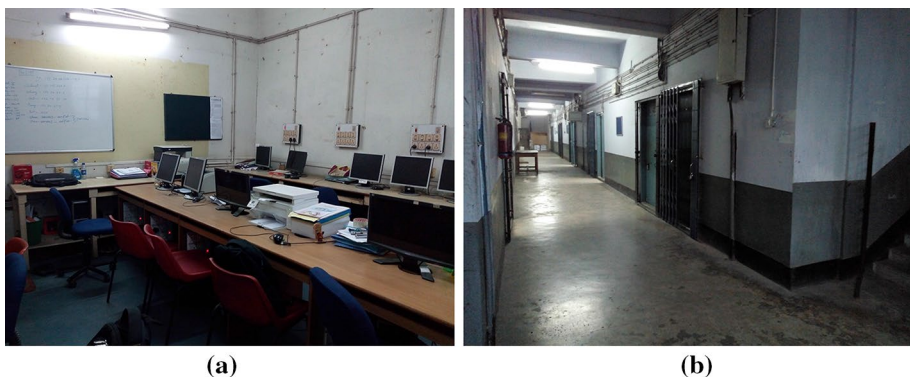
The experiment is carried out on the 3rd, 4th and 5th floor of a 5 storied-building of our University. In this building, each floor covers 882 m<sup>2</sup> area, with a length of 42 m and width of 21 m and consists of faculty rooms, classrooms, seminar rooms, research

labs and corridor as shown in Fig. 3. 172 APs are installed at specific positions covering the whole experimental region and every location point is covered by at least 3 APs. Firstly, data are collected dividing the whole region into  $2\text{ m} \times 2\text{ m}$  cells. However, for fine grained results, later on, the region is divided into  $1\text{ m} \times 1\text{ m}$  cells or location points. Thus,  $X$  and  $Y$  coordinates of each floor is divided into 42 and 21 cells respectively as shown in Fig. 2. Each cell is distinguished by a unique identifier by naming them as  $L$  <floor number>-<  $X$  coordinate>-<  $Y$  coordinate>. Suppose, identifier of a cell is  $L4 - 22 - 8$ . It represents that the cell is in the 4th floor and its  $X$  and  $Y$  coordinates are 22 and 8 respectively. Due to the presence of obstacles,  $RSSI$  values have not been collected from all cells. Note that, in this experimental region ground truth locations are cells of  $1\text{ m} \times 1\text{ m}$  dimension. Ground truth is the data that is known to be correct. However, there is a maximum error of  $\sqrt{2} / 2$  in the ground truth as the data have been collected from different points around a cell. In order to take care of device heterogeneity, data is collected from 4 different types of devices namely, Samsung Galaxy Tab 10 (*Android version 4.0*), Samsung Galaxy Tab E (*Android version 5.0*), Samsung Galaxy Tab 2 (*Android version 4.1.1*) and Motorola Moto E (*Android version 5.1*). Fingerprints are collected by each device covering the total area of three floors, which is divided into 1000 number of cells. The  $RSSI$  scan is repeated 2 to 3 times at a cell from each device to collect around 15 fingerprints and among them 7 fingerprints are selected randomly.

### 3.3 Data Preprocessing

Raw data needs preprocessing before analysis. Our *Wi-Fi* data are preprocessed in the following manner:

- *Elimination of duplicate entries*: In preprocessing, duplicate entries are removed as these may affect the accuracy of the system.
- *Filling entries of unobserved APs*: Due to the presence of obstacles, limited coverage range of *Wi-Fi* and nearby interfering devices along the path, the  $RSSI$  values are not received from all APs from all cells. So to prepare dataset that can be fed to machine learning algorithms, there are some missing features, which needs to be



**Fig. 3** Experimental regions; **a** research lab of 3rd floor; **b** corridor connecting different rooms on the 4th floor

filled. Note that, in our dataset for very poor signal strength, the *RSSI* value is nearly  $-100$  dBm. So, for *JUIndoorLoc*, these missing features are filled with  $-110$  dBm.

- *Removal of Wi-Fi hotspots*: *Wi-Fi* hotspots are eliminated as they are not stable. *RSSI* data has been recorded for many days and also different times in a day. Hence, the *APs* available only for few days or some specific time instant in a day are easily detected and removed. Thus, we have considered those *APs* that were alive for the entire duration of our data collection for further analysis.

### 3.4 Description of Dataset

A dataset has been created based on collected data from the experimental region. This *Wi-Fi* dataset contains 25,364 samples. Each sample has 177 different fields as mentioned below.

(1)	(2)	–	(173)	(174)	(175)	(176)	(177)
$C_{id}$	$AP_{001}$	–	$AP_{172}$	$R_s$	$H_{pr}$	$D_{id}$	$T_s$
$L4 - 22 - 8$	$- 54$	–	$- 88$	1	0	$D_1$	1468832345801

$C_{id}$  (column 1) corresponds to a value for uniquely identifying each cell. It is used to represent the location point where the *RSSI* values are recorded. This dataset contains 1000 unique cells with each of size  $1\text{ m} \times 1\text{ m}$ . Each *AP* is identified by a Service Set Identifier (*SSID*) and Basic Service Set Identification (*BSSID*). However, it is observed that many *APs* may have the same *SSID*. Thus, *APs* are identified by the *BSSIDs* or *MAC* addresses. Due to privacy reasons, these *BSSIDs* are renamed to  $AP_{001}$  to  $AP_{172}$ . *RSSI* values of 172 different *APs*, the most important data in fingerprinting based indoor localization, are represented in  $AP_{001}$  to  $AP_{172}$  (column 2—173). Context heterogeneity in terms of open and closed rooms (represented by 1 and 0 respectively) is shown in  $R_s$  (column 174) and presence and absence of human (represented by 1 and 0 respectively) are depicted in  $H_{pr}$  (column 175).  $D_{id}$  (Column 176) represents unique identifier assigned to each device.  $T_s$  (column 177) indicates the data collection time in milliseconds. Note that, *RSSI* values and  $C_{id}$  are used to train different machine learning algorithms. Other attributes are used to distinguish between training and testing conditions.

The whole *Wi-Fi* dataset is divided into two different sets: (i) *Training* (23,904 samples) and (ii) *Test* (1460 samples) datasets. Two datasets contain same attributes as mention above. Training dataset contains *RSSI* values of 1000 cells captured by 4 different devices, whereas test dataset contains *RSSI* values of some specific cells captured by 2 devices.

### 3.5 Discussions on JUIndoorLoc

Data from *JUIndoorLoc* are analyzed for different environmental factors in order to have a close look at the issues and challenges. Note that, in a cell. *RSSI* scan is performed for 120 s and in this time period 3 to 4 fingerprints are selected out of 5 to 6 captured fingerprints. The selected values are averaged and plotted in the Fig. 4. The scan is repeated at every 15 minutes from (11:00 am) to (07:00 pm). At a particular time instant, for every *APs* an average of statistical *RSSI* values received in this scan duration is considered. Variation of the

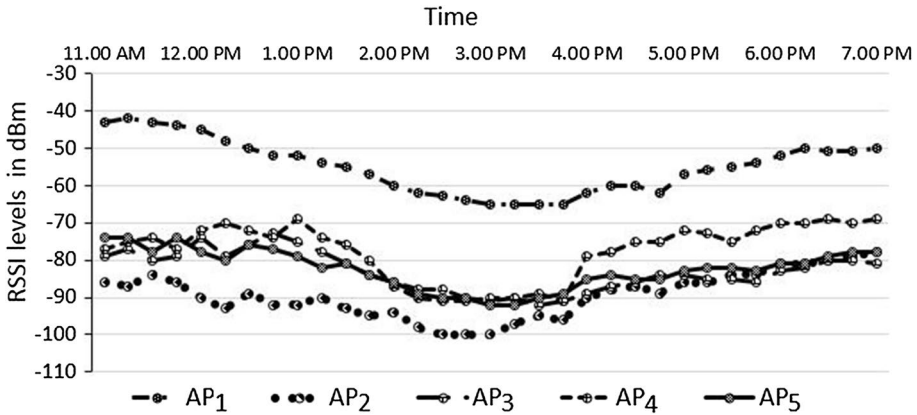


Fig. 4 Variation of *RSSI* values with different times at a specific location

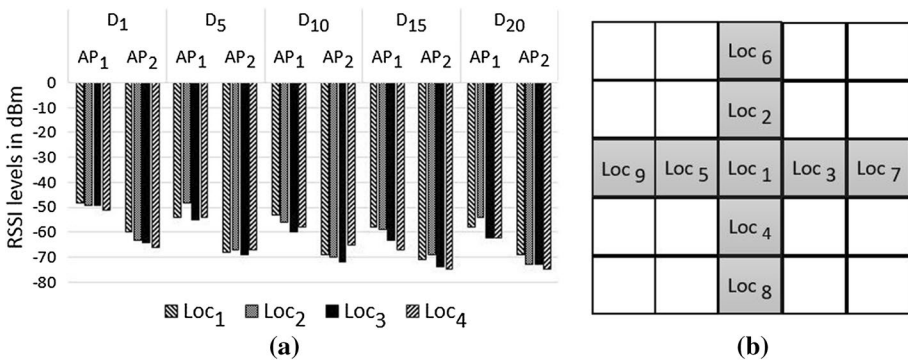


Fig. 5 **a** Effect of interference for varying number of devices;  $D_i$  indicates  $i$  number of devices used in the experimental setting for 4 locations; **b** map showing the position of 9 neighboring locations

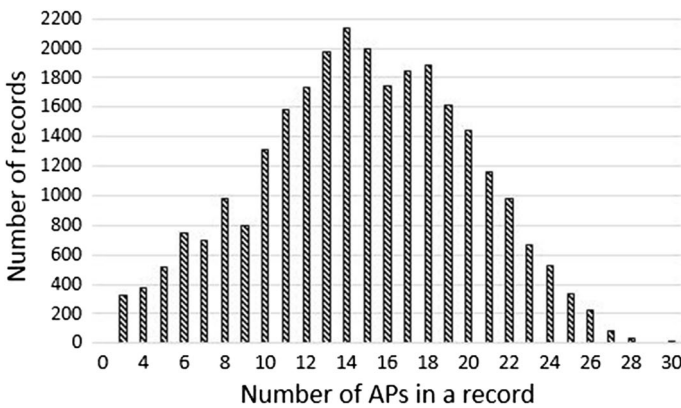
signal strengths of 5 APs are shown in Fig. 4 with respect to time, using same device and same location. It can be observed that the signal strengths in morning (11:00 am) and evening (07:00 pm) are almost same with a drop at around (02:00 pm) for all APs. At morning and evening, the location is less crowded. Thus, the number of nearby interfering devices are also less while at around (02:00 pm) the place is more crowded.

The *RSSI* signal is also affected by varying number of nearby interfering devices. *RSSI* variation of 4 neighboring locations due to the presence of 1 to 20 interfering devices is shown in Fig. 5a. This experiment is conducted in a research lab of the 3rd floor. In the presence of one interfering device, *RSSI* values of two APs,  $AP_1$  and  $AP_2$  are  $-48$  dBm and  $-60$  dBm respectively for Location 1. However, these values gradually decrease with the presence of increasing number of interfering devices and reach a steady state when the interfering devices are increased to 15 and 20.

Correlation of *RSSI* values of 9 neighboring locations ( $Loc_1$  to  $Loc_9$ ) is shown in Table 2 and the position of neighboring locations are shown in Fig. 5b. One user, carrying a smartphone, captures *RSSI* values from  $Loc_1$  and then moves to the next neighboring location. In this way, the user collects *RSSI* values of 9 neighboring locations. This experiment is

**Table 2** Correlation of *RSSI* gradient of 9 neighboring locations

	<i>Loc</i> <sub>1</sub>	<i>Loc</i> <sub>2</sub>	<i>Loc</i> <sub>3</sub>	<i>Loc</i> <sub>4</sub>	<i>Loc</i> <sub>5</sub>	<i>Loc</i> <sub>6</sub>	<i>Loc</i> <sub>7</sub>	<i>Loc</i> <sub>8</sub>	<i>Loc</i> <sub>9</sub>
<i>Loc</i> <sub>1</sub>	1	–	–	–	–	–	–	–	–
<i>Loc</i> <sub>2</sub>	0.8283	1	–	–	–	–	–	–	–
<i>Loc</i> <sub>3</sub>	0.8948	0.9426	1	–	–	–	–	–	–
<i>Loc</i> <sub>4</sub>	0.9619	0.8886	0.9303	1	–	–	–	–	–
<i>Loc</i> <sub>5</sub>	0.8888	0.8743	0.8792	0.9692	1	–	–	–	–
<i>Loc</i> <sub>6</sub>	0.7388	0.8463	0.5698	0.5899	0.7422	1	–	–	–
<i>Loc</i> <sub>7</sub>	0.7675	0.5543	0.8579	0.6274	0.6974	0.6498	1	–	–
<i>Loc</i> <sub>8</sub>	0.8054	0.5386	0.6742	0.8672	0.7169	0.6021	0.6825	1	–
<i>Loc</i> <sub>9</sub>	0.8148	0.4928	0.7263	0.7935	0.8563	0.7136	0.5906	0.6369	1



**Fig. 6** Total number of records with respect to number of *APs* detected in a single capture

repeated 8 times. It has been observed that *RSSI* values of adjacent locations have a strong positive linear correlation, as the correlation coefficient *r* is close to +1 (0.8 to 0.9). However, correlation coefficient varies between 0.5 and 0.7 for the locations that are not adjacent to each other.

While Figs. 4 and 5a show behavior of the dataset with time and environmental factors for some selected samples, Figs. 6, 7 and 8 present a general discussion of the dataset. Note that, each dataset record corresponds to an *RSSI* scan. At most 30 *APs* are detected in *RSSI* scans as depicted in Fig. 6. However, maximum dataset records (*RSSI* scans) contain 14 *APs*. All location points considered in the dataset are found to be covered by some *AP*. As the behavior of *Wi-Fi* signals vary with its signal strength, it is important to look at the proportion of dataset records that are found to provide appreciable *RSSI*.

Figure 7 depicts the number of *APs* with different *RSSI* levels that are detected in different time instants. Dataset records contain different *RSSI* levels ranging from –11 to –100 dBm. *RSSI* level of the maximum number of *APs* are found to lie between –81 and –90 dBm irrespective of time. Total records (number of appearance in the dataset) with this *RSSI* level are 147,216 that are spread over 3 floors as shown in Fig. 8. Very strong *RSSI* signals (–11 dBm to –30 dBm) and weak *RSSI* signals (–91 dBm to –100 dBm) are detected from only few *APs* (Fig. 7) and captured by few dataset records (Fig. 8). Hence,

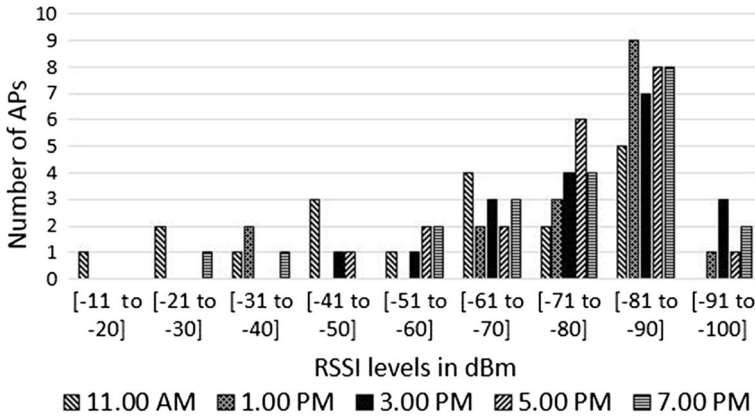


Fig. 7 The number of APs detected with different RSSI levels between 11:00 am to 07:00 pm

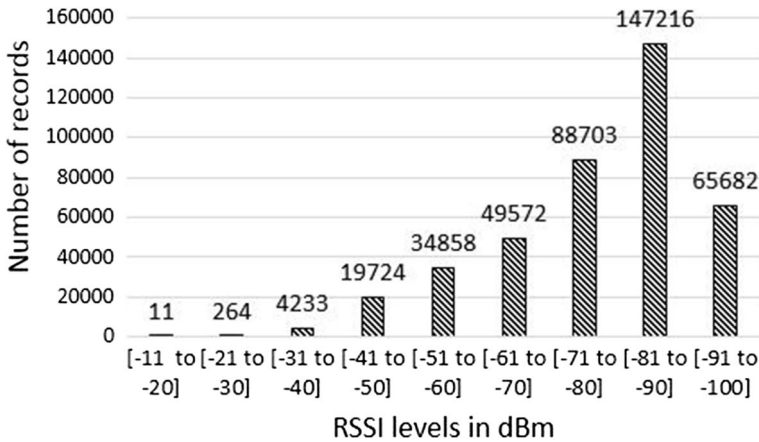


Fig. 8 Number of records (number of appearance in dataset) with different RSSI levels

learning based methods can be applied to predict as more records can be found with moderate but comparable RSSI levels.

In this regard, our proposed conditional ensemble classifier is described in the next section that deals with context and device heterogeneity.

### 4 Design of Conditional Ensemble Classifier

RSSI values are vulnerable to time (as shown in Fig. 4), contexts (as shown in Fig. 5a) and devices. So, localization performance may degrade when training and test conditions are different. For instance, if the training data are collected from one device, then it would be difficult to predict the location of a user carrying a different device with state-of-the-art classifiers. In this scenario, classifiers are not capable to generalize all the conditions

individually. Hence, the combination of multiple classifiers can improve accuracy, robustness and efficiency over individual classifier.

In our indoor localization system, a set of all location points is represented by  $LP = \{lp_1, lp_2, lp_3, \dots, lp_n\}$ . Here, the dataset is represented as  $DS = \{ds_{c_1d_1}, ds_{c_2d_1}, \dots, ds_{c_md_1}, ds_{c_1d_2}, ds_{c_2d_2}, \dots, ds_{c_md_2}, \dots, ds_{c_1d_p}, ds_{c_2d_p}, \dots, ds_{c_md_p}\}$  where  $m$  and  $p$  denote number of contexts and devices respectively. The dataset ( $DS$ ) is a function of context and device which is represented as  $DS = f(Context, Device)$ .  $DS'$  is generated after preprocessing the dataset  $DS$ .  $DS'$  contains the feature set. In our case, features are the different  $RSSI$ s received from  $AP$ s,  $AP = \{ap_1, ap_2, ap_3, \dots, ap_q\}$  available while collecting data. An instance  $X_j$  of dataset  $DS'$  is represented by  $\{x_1, x_2, x_3, \dots, x_q, lp'_j\}$ , where  $x_i$  represents the received  $RSSI$  value of  $i$ th  $AP$  and  $lp'_j$  represents the location label where  $lp'_j \in LP$ .

Using a learning algorithm, the objective of indoor localization problem is to identify the location set  $LP$  from dataset  $DS'$  using feature space  $AP$  with a function  $g : DS' \rightarrow LP$ . Here,  $g$  is a member of hypothesis space and it best fits the dataset  $DS'$  using a loss function  $l : LP \times LP \rightarrow R$  such that if for an instance  $j$  of the training model, the location label is  $lp_j$  and the predicted label is  $lp'$  then the loss is computed as  $l(lp_j, lp')$ . In order to solve the problem of indoor localization more effectively, more than one learning algorithms are used to predict the location from dataset  $DS'$ . Based on different conditions of the training dataset ( $DS'$ ) the learning algorithms are trained using cross-validation. Depending upon the localization accuracy, a base classifier is chosen for the conditional datasets.

The training datasets, say  $k$  number of datasets, are selected based on various context and device as depicted in Fig. 9. These  $k$  training datasets are tuned with a base classifier. The test dataset, containing data of all contexts and devices is classified with each of the  $k$  condition based classifiers,  $CF = \{cf_1, cf_2, cf_3, \dots, cf_k\}$ . Finally, the results of  $k$  condition based classifiers are taken as input to the ensemble methods to get final prediction result. Indoor localization problem can be solved effectively using  $k$ -condition based Ensemble Classifier  $E : DS'' \times CF \rightarrow LP$ , where  $E$  performs either majority voting or the average of probabilities using  $CF$  on test dataset  $DS''$  and estimated location point is returned as output.

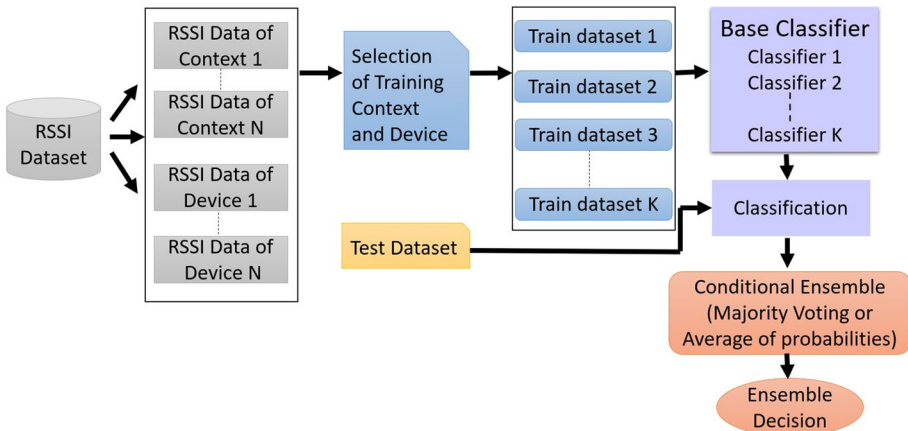


Fig. 9 Block diagram of conditional ensemble classifier

#### 4.1 Majority Voting

Here, all test instances are classified by every individual condition specific classifier, then the final prediction result for the ensemble is obtained by majority voting of these results of the individual classifiers. In case of a tie,  $cf'_i$ 's may be assigned relative weights based on their performance accuracy. The accuracy ( $A^{mv}$ ) of ensemble classifier is obtained by,

$$A^{mv} = \frac{T_c}{T_i} \times 100\%$$

Here,  $T_c$  and  $T_i$  represent the total number of correctly predicted instance and the total number of instances of test dataset respectively.

#### 4.2 Average of Probabilities

In our proposed conditional ensemble classifier, we have used  $k$  condition based classifiers ( $cf_1, cf_2, \dots, cf_k$ ). Suppose, there are  $n$  location points and each location point corresponds to a class. Thus, we have  $n$  classes. Using a classifier, a test dataset is classified and probability of a class  $P(CL_i)$  in test dataset is obtained by,

$$P_{cf_j}(CL_i) = \frac{N_c}{N'_i}$$

Here,  $N_c$  and  $N'_i$  represent the number of correctly predicted instances of that class and the total number of instances of that class respectively. Similarly,  $k$  classifiers are used to classify test data and probability of each class is calculated. Thus, mean of the probability  $P_m(CL_i)$  of the  $i$ th class is obtained by,

$$P_m(CL_i) = \frac{\sum_{j=1}^k P_{cf_j}(CL_i)}{k} \quad \text{where } i = 1 \dots n$$

Hence, the accuracy ( $A^{ap}$ ) of ensemble of  $k$  classifiers is calculated by,

$$A^{ap} = \frac{\sum_{i=1}^n P_m(CL_i)}{n} \times 100\%$$

In this context, experimental results of indoor localization for the collected dataset using state-of-the-art classifiers and the proposed ensemble of condition-based classifiers are shown in the next section.

### 5 Results and Discussions

The performance of dataset collected for *JUIndoorLoc* has been analyzed with two different perspectives. First, state-of-the-art classifiers of *Weka* 3.9 tool and a statistical method, *Horus* [18] are used to evaluate location prediction performance of our proposed dataset. This is followed by performance analysis of our proposed ensemble of condition-based classifiers in Sect. 5.2. All the experiments are performed on Intel Pentium quad core machine with 1.60 GHz processor and 4 GB RAM.



**Table 3** *JUIndoorLoc* results with 1NN in conjunction with Euclidean Distance

Error in positioning	Success rate	Time
1.24 m	91.67%	1.44 ms

**Table 4** Accuracies of used classifiers in different locations: Accuracy (in %) ( $A_p$ ), Average error in meter ( $E_m$ )

Dataset	Metric	KNN	K*	Bayes Net.	J48	SVM	Horus
Room	$A_p$	87.33	85.45	85.58	84.72	86.25	85.45
	$E_m$	1.21	1.42	1.89	1.24	1.82	1.16
Corridor	$A_p$	89.37	88.41	82.50	83.37	85.76	85.31
	$E_m$	1.07	1.35	1.79	1.31	1.28	1.36
Stairs	$A_p$	90.12	88.46	87.57	85.22	86.65	87.91
	$E_m$	1.57	1.41	1.41	1.22	1.32	1.25
Entire floor	$A_p$	88.52	86.50	85.32	84.71	85.38	87.24
	$E_m$	1.24	1.77	1.68	1.87	1.54	1.46

All accuracies are obtained by performing fivefold cross-validation

## 5.1 Performance of Dataset Using State-of-the-Art Classifiers

A baseline is a method that uses randomness, simple summary statistics, heuristics or machine learning to create predictions for a dataset. In [9] the distance based technique  $K$  Nearest Neighbor ( $KNN$ ) is considered as a baseline for comparison purposes. They have established 1NN technique ( $K = 1$ ) in conjunction with the Euclidean Distance. This method when applied to our dataset yields the results as reported in Table 3.

The success rate corresponds to the percentage of samples that are correctly located inside the corresponding cell. Error in positioning represents the average error in meters. The average time in milliseconds required to obtain the precise location per sample is mentioned in the time field. The corresponding error and success rate as reported in [9] for the same baseline method are 7.9 m and 89.92% respectively.

The  $RSSI$  data from various regions such as room, corridor, stairs and entire floor are analyzed as shown in Table 4. All these 4 datasets contain data of various time instants taken at a specific context and device. Different machine learning algorithms such as **KNN**, **K\***, **Bayesian Network**, **J48** and **SVM** are used to classify data apart from the basic algorithm of **Horus** [18]. The default parameters provided by *Weka* toolkit for the above-mentioned algorithms are used in the experiments. Classification accuracy obtained by these algorithms are measured in percentage ( $A_p$ ) and in the form of distance ( $E_m$ ) from actual cell (in case of misclassification). These results show that for various regions, the accuracy of location estimation varies significantly in different approaches. However, it has been observed that in different area accuracy of any algorithm does not change significantly, though population density changes with time. The basic algorithm of *Horus* [18] is applied to our dataset for validation. In different area, the performance of *Horus* [18] system ranges between 85.31 and 87.91% and error in positioning (meter) lies between 1.16 and 1.46 m. In our datasets,  $KNN$  algorithm performs better than other classifiers in different locations. The average case accuracy of  $KNN$  algorithm ranges between 87.33 and 90.12%. In  $KNN$ , worst case error of location prediction lies between 1.07 and 1.57 m. However, accuracies

of  $K^*$ , *Bayesian Network*, *J48* and *SVM* vary between 82 and 88% and the overall meter level accuracy ranges from 1.22 to 1.89 m.

## 5.2 Performance of Proposed Condition-Based Ensemble Classifiers

### 5.2.1 Estimating Location Accuracies with Experimental Dataset Using Cross-Validation Method

Some experiments are performed with our dataset to estimate unknown location points. In Table 5, 7 subsets of data are taken from our dataset. Each subset of data contains *RSSI* data of a specific context (such as  $C_1$  to  $C_4$ ) taken from a specific device (such as  $D_1$  to  $D_4$ ) in different times in a day. Classification accuracies are obtained by performing five-fold cross-validation of some well known classifiers like *KNN*, *Bayesian Network*, *J48* and *SVM*. In most of the subsets of data, *KNN* provide reasonably better prediction accuracy than other three classifiers. The default values of the parameters of all classifiers are kept unchanged.

### 5.2.2 Estimating Location Accuracies Using Separate Training and Testing Dataset

Rather than using the same subset of data with cross-validation method, in our next phase a different condition (context and device) specific test dataset is used to evaluate classification accuracies. This step is essential to verify whether the classifiers can accurately estimate the location points, when the user is in different environment or with different types of devices than the state in which the classifiers are trained. In Table 6 each conditional subset of data is used to train classifiers and a different condition specific data is used as a test set to predict location accuracy. The default values of the parameters of all classifiers are kept unchanged. The performance of every classifier decreases in Table 6. However, the performance of *KNN* classifier is well enough in some cases despite of different condition specific datasets. Hence, *KNN* is chosen as the base classifier for ensemble method in the next section.

### 5.2.3 Evaluating the Performance of Ensemble Method

The performance of proposed ensemble method is evaluated by taking 7 dissimilar condition specific dataset as train datasets. The test sets are taken for two cases:

**Table 5** Classification accuracies (in %) obtained on experimental datasets with different machine learning algorithms

Dataset	KNN	Bayes Net.	J48	SVM
$C_4-D_2$	86.79	87.97	85.97	83.24
$C_3-D_3$	88.14	80.47	82.77	84.43
$C_2-D_1$	73.47	62.60	70.62	69.18
$C_2-D_3$	89.13	72.70	84.80	80.73
$C_4-D_4$	86.07	74.18	87.54	85.47
$C_3-D_1$	87.32	73.40	80.36	88.62
$C_1-D_2$	88.37	80.55	86.23	81.48

All accuracies are obtained by performing fivefold cross-validation

**Table 6** Classification accuracies (in %) obtained on experimental datasets with different classifiers

Train Set	Test Set	KNN	Bayes Net.	J48	SVM
$C_4-D_2$	$C_1-D_2$	<b>71.78</b>	66.48	68.36	62.86
$C_3-D_3$	$C_3-D_1$	67.56	58.61	60.58	63.15
$C_2-D_1$	$C_4-D_4$	45.73	41.83	47.62	40.68
$C_2-D_3$	$C_4-D_2$	70.36	64.26	68.74	65.87
$C_4-D_4$	$C_3-D_3$	69.82	61.15	67.36	60.74
$C_3-D_1$	$C_2-D_1$	57.64	59.52	62.85	58.60
$C_1-D_2$	$C_2-D_3$	66.26	60.22	63.27	64.28

Maximum accuracy that is obtained using different condition-specific datasets

**Table 7** Classification accuracies (in %) of two cases are obtained by each condition specific classifier with two test datasets and Ensemble method

Train Set	K	Classifier	Case-I	Case-II
$C_4-D_2$	3	Classifier 1	62.28	58.53
$C_3-D_3$	5	Classifier 2	59.32	57.16
$C_2-D_1$	3	Classifier 3	54.68	51.64
$C_2-D_3$	3	Classifier 4	77.96	71.36
$C_4-D_4$	7	Classifier 5	73.30	69.27
$C_3-D_1$	5	Classifier 6	79.66	76.43
$C_1-D_2$	3	Classifier 7	80.50	78.84
–	–	Ensemble (Majority Voting)	<b>91.74</b>	<b>88.61</b>
–	–	Ensemble (Avg. of probability)	<b>89.53</b>	<b>89.13</b>

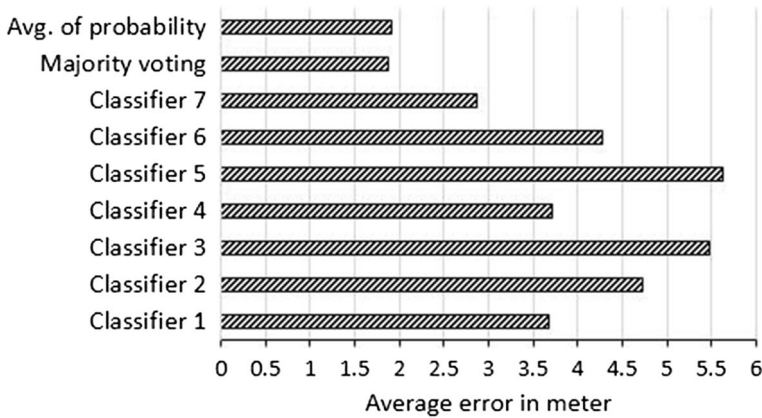
$K$  represents the parameter of  $KNN$ . In Case-I test set contains instances for  $C_i-D_i$  where  $i = 1$  to 4 and each condition,  $C_i-D_i$ , matches with the 7 training conditions. In Case-II, test set  $C_4-D_3$  is used

Accuracy obtained for the ensemble of classifiers

Case-I: The test dataset is collected in such a way that it contains instances for 7 different conditions of train datasets. The results are summarized in Table 7.

Case-II: In this case, the test dataset is prepared to contain the instances for a specific condition that is not included in train datasets. Here, test instances are collected using device  $D_3$  in a closed room with no other users in the vicinity (denoted as context  $C_4$  as mentioned in Sect. 3.2). The 7 condition specific classifiers are not trained for this specific combination of condition,  $C_4-D_3$  as shown in Table 7.

In both cases, the instances of training and test datasets are distinct from each other. Here,  $KNN$  is considered as a base classifier for ensemble method since in the previous sections the performance of  $KNN$  classifier is found to be stable. Each of the 7 training datasets is cross-validated and the parameter  $K$  is tuned for each of the conditions with different values. After tuning the parameter  $K$  for each of the individual training datasets, the classification with test dataset is performed. The results of each condition specific classifier varies between 54.68 and 80.50% as shown in Table 7. However, combining the prediction result of 7 condition specific classifiers with majority voting technique, the classification accuracy is increased to 89.43%. While combining classifier results through average probabilities, 87.78% accuracy can be obtained. Moreover, in Table 7 prediction accuracies of both ensemble method majority voting and the average of probabilities are 88.61% and



**Fig. 10** Average error of individual classifiers and ensemble methods in meter for Case-I

89.13% respectively. Although individual classifiers can not properly predict various conditional data but together they cover all the conditions and their united decision can achieve higher prediction accuracy than individual classifiers.

The average error in meter for Case-I are obtained from our floor map and depicted in Fig. 10. The average error of ensemble methods (1.87 m and 1.92 m) are subsequently lower than that of individual classifiers. Hence, our proposed ensemble methods are found to better cope with device and context heterogeneity than the condition based individual classifiers in terms of classification accuracy and average error in meter.

## 6 Conclusion and Future Directions

This paper introduces a new indoor localization dataset, based on *Wi-Fi* signal strength subject to spatial, temporal, contextual and device heterogeneity. The dataset description has been detailed, including the attributes used in the dataset and their significance. An indoor localization framework, *JUIndoorLoc* is designed that uses an ensemble of condition-based classifiers designed to mitigate the effects of device and context heterogeneity in indoor positioning. Samples are collected in an anonymized fashion for three floors of our departmental building of our University. Total 25,364 samples are collected from 1000 cells each of size 1 m × 1 m. Our dataset is validated against state-of-the-art classification algorithms and the basic algorithm of *Horus* [18] to justify its applicability for comparing both machine learning and statistical approaches of indoor localization. Data are also analyzed for signal variations due to factors like differing indoor environments and devices. Significant localization accuracy (62.60% to 88.37%) has been obtained when training and test samples belong to same contexts and devices. However, location prediction accuracy is found to drop (41.83% to 71.78%) when training and test conditions are different. Our proposed condition specific ensemble classifier is found to efficiently overcome these difficulties achieving 91.74% accuracy. These results emphasize that our dataset provides a comprehensive dataset to make comparisons among different methods in the field.

In future, we are planning to work on reducing the effort of fingerprinting. One of the main issues with indoor localization is the effort needed for precise fingerprinting.

Localization accuracy depends greatly on it. Crowdsourcing can be an alternative to reduce the fingerprinting effort but precision of fingerprinting may be degraded. Thus, in future we plan to investigate novel techniques that would provide considerable localization accuracy even for minimal and/or imprecise fingerprint data. Another future dimension is to provide real time indoor navigation technique based on semi-supervised learning mechanisms.

## References

1. Liu, H., Darabi, H., Banerjee, P., & Liu, J. (2007). Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(6), 1067–1080.
2. Gu, Y., Lo, A., & Niemegeers, I. (2009). A survey of indoor positioning systems for wireless personal networks. *IEEE Communications Surveys Tutorials*, 11(1), 13–32.
3. He, S., & Chan, S. H. G. (2016). Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys Tutorials*, 18(1), 466–490.
4. Shu, Y., Huang, Y., Zhang, J., Coué, P., Cheng, P., Chen, J., et al. (2016). Gradient-based fingerprinting for indoor localization and tracking. *IEEE Transactions on Industrial Electronics*, 63(4), 2424–2433.
5. Shih, C. Y., Chen, L. H., Chen, G. H., Wu, E. H. K., & Jin, M. H. (2012). Intelligent radio map management for future WLAN indoor location fingerprinting. In *2012 IEEE wireless communications and networking conference (WCNC)* (pp. 2769–2773).
6. Marques, N., Meneses, F., & Moreira, A. (2012). Combining similarity functions and majority rules for multi-building, multi-floor, WiFi positioning. In *2012 International conference on indoor positioning and indoor navigation (IPIN)* (pp. 1–9).
7. Xiao, J., Zhou, Z., Yi, Y., & Ni, L. M. (2016). A survey on wireless indoor localization from the device perspective. *ACM Computing Surveys*, 49(2), 25:1–25:31. <https://doi.org/10.1145/2933232>.
8. Asuncion, A., & Newman, D. (2007). UCI machine learning repository. <https://archive.ics.uci.edu/ml/index.php>.
9. Torres-Sospedra, J., Montoliu, R., Martínez-Usó, A., Avariento, J. P., Arnau, T. J., Benedito-Bordonau, M., et al. (2014). Ujiiindoorloc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems. In *2014 International conference on indoor positioning and indoor navigation (IPIN)* (pp. 261–270).
10. Torres-Sospedra, J., Rambla, D., Montoliu, R., Belmonte, O., & Huerta, J. (2015). UJIIIndoorLoc-Mag: A new database for magnetic field-based localization problems. In *2015 International conference on indoor positioning and indoor navigation (IPIN)* (pp. 1–10).
11. Torres-Sospedra, J., Montoliu, R., Mendoza-Silva, G. M., Belmonte, O., Rambla, D., & Huerta, J. (2016). Providing databases for different indoor positioning technologies: Pros and cons of magnetic field and Wi-Fi based positioning. *Mobile Information Systems*. <http://doi.org/10.1155/2016/6092618>.
12. Montoliu, R., Sansano, E., Torres-Sospedra, J., & Belmonte, O. (2017). IndoorLoc platform: A public repository for comparing and evaluating indoor positioning systems. In *2017 International conference on indoor positioning and indoor navigation (IPIN)* (pp. 1–8).
13. King, T., Kopf, S., Haenselmann, T., Lubberger, C., & Effelsberg, W. (2008). CRAWDAD dataset mannheim/compass (v. 2008-04-11). Downloaded from <https://crawdad.org/mannheim/compass/20080411/fingerprint>, traceset: fingerprint.
14. Trawiński, K., Alonso, J. M., & Hernández, N. (2013). A multiclassifier approach for topology-based wifi indoor localization. *Soft Computing*, 17(10), 1817–1831. <https://doi.org/10.1007/s00500-013-1019-5>.
15. Cooper, M., Biehl, J., Filby, G., & Kratz, S. (2016). LoCo: Boosting for indoor location classification combining Wi-Fi and BLE. *Personal and Ubiquitous Computing*, 20(1), 83–96. <https://doi.org/10.1007/s00779-015-0899-z>.
16. Taniuchi, D., & Maekawa, T. (2014). Robust Wi-Fi based indoor positioning with ensemble learning. In *2014 IEEE 10th International conference on wireless and mobile computing, networking and communications (WiMob)* (pp. 592–597).
17. Bahl, P., & Padmanabhan, V. N. (2000). Radar: An in-building RF-based user location and tracking system. In *Proceedings IEEE INFOCOM 2000. Conference on computer communications. Nineteenth annual joint conference of the IEEE computer and communications societies (Cat. No.00CH37064)* (vol. 2, pp. 775–784).

18. Youssef, M., & Agrawala, A. (2005). The horus WLAN location determination system. In *Proceedings of the 3rd international conference on mobile systems, applications, and services, ser. MobiSys '05* (pp. 205–218). New York, NY: ACM. <http://doi.org/10.1145/1067170.1067193>.
19. Yen, C.-T., & Ke, C.-H. (2017). Improving tracking error by dead reckoning and RSSI technologies with a fuzzy fusion scheme in indoor location. *Microsystem Technologies*. <https://doi.org/10.1007/s00542-017-3614-3>.
20. Seong, J.-H., & Seo, D.-H. (2018). Environment adaptive localization method using Wi-Fi and bluetooth low energy. *Wireless Personal Communications*, 99(2), 765–778. <https://doi.org/10.1007/s11277-017-5151-x>.
21. Liu, H.-H. (2017). The quick radio fingerprint collection method for a WiFi-based indoor positioning system. *Mobile Networks and Applications*, 22(1), 61–71. <https://doi.org/10.1007/s11036-015-0666-4>.
22. Gu, Y., Liu, J., Chen, Y., & Jiang, X. (2014). Constraint online sequential extreme learning machine for lifelong indoor localization system. In *2014 International joint conference on neural networks (IJCNN)* (pp. 732–738).
23. Ahriz, I., Oussar, Y., Denby, B., & Dreyfus, G. (2010). Full-band GSM fingerprints for indoor localization using a machine learning approach. *International Journal of Navigation and Observation*. <http://doi.org/10.1155/2010/497829>.
24. Niu, J., Wang, B., Shu, L., Duong, T. Q., & Chen, Y. (2015). ZIL: An energy-efficient indoor localization system using ZigBee radio to detect WiFi fingerprints. *IEEE Journal on Selected Areas in Communications*, 33(7), 1431–1442.
25. Moreno, V., Zamora, M. A., & Skarmeta, A. F. (2016). A low-cost indoor localization system for energy sustainability in smart buildings. *IEEE Sensors Journal*, 16(9), 3246–3262.
26. Chen, Y., Guo, M., Shen, J., & Cao, J. (2017). Graphloc: A graph-based method for indoor subarea localization with zero-configuration. *Personal and Ubiquitous Computing*, 21(3), 489–505. <https://doi.org/10.1007/s00779-017-1011-7>.
27. Ghosh, D., Roy, P., Chowdhury, C., & Bandyopadhyay, S. (2016). An ensemble of condition based classifiers for indoor localization. In *2016 IEEE International conference on advanced networks and telecommunications systems (ANTS)* (pp. 1–6).
28. Bacciu, D., Barsocchi, P., Chessa, S., Gallicchio, C., & Micheli, A. (2014). An experimental characterization of reservoir computing in ambient assisted living applications. *Neural Computing and Applications*, 24(6), 1451–1464.
29. Torres-Sospedra, J., Jiménez, A. R., Knauth, S., Moreira, A., Beer, Y., Fetzer, T., et al. (2017). The smartphone-based offline indoor location competition at IPIN 2016: Analysis and future work. *Sensors*, 17(3). <http://www.mdpi.com/1424-8220/17/3/557>.
30. Lohan, E. S., Torres-Sospedra, J., Leppäkoski, H., Richter, P., Peng, Z., & Huerta, J. (2017). Wi-Fi crowdsourced fingerprinting dataset for indoor positioning. *Data*, 2(4). <http://www.mdpi.com/2306-5729/2/4/32>.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Priya Roy** is currently pursuing the Ph.D. degree in Computer Science and Engineering at Jadavpur University, India under the supervision of Dr. Chandreyee Chowdhury. Her research interests include indoor localization and machine learning. She received her B.Tech. and M.Tech. degrees in Computer Science and Engineering from Dream Institute of Technology and Jadavpur University, India, in 2012 and 2016, respectively.



**Chandreyee Chowdhury** is a faculty in the department of Computer Science and Engineering at Jadavpur University, India. She received Ph.D. in Engineering from Jadavpur University in 2013 and M.E. in Computer Science and Engineering from Jadavpur University in 2005. She has published a number of research articles in International journals and conferences. Her research interests include routing issues of Wireless Body Area Networks, mobile crowd-sensing, and applications of smartphone sensing. She is a member of IEEE and IEEE Computer Society.



**Dip Ghosh** is a Doctoral student in Indian Statistical Institute, Kolkata under the supervision of Prof. Sanghamitra Bandyopadhyay. He did his masters and bachelor degree from the University of Southampton, UK and University of Kalyani, India respectively. His primary research interests are Bioinformatics, Machine Learning and Pattern Recognition.



**Dr. Sanghamitra Bandyopadhyay** did her B.Tech., M.Tech. and Ph.D. in Computer Science from Calcutta University, IIT Kharagpur and ISI respectively. She is currently a Professor (Higher Administrative Grade) and director at the Indian Statistical Institute, Kolkata, India. She has worked in various Universities and Institutes world-wide including in USA, Australia, Germany, China, France, Italy, Slovenia and Mexico. She has delivered invited lectures in many more countries. She has authored/co-authored more than 300 research article in international journals, conferences and book chapters, and published six authored and edited books from publishers like Springer, World Scientific and Wiley. She has also edited journals special issues in the area of soft computing, data mining, and bioinformatics. Her research interests include computational biology and bioinformatics, soft and evolutionary computation, pattern recognition and data mining.

## Affiliations

Priya Roy<sup>1</sup> · Chandreyee Chowdhury<sup>1</sup>  · Dip Ghosh<sup>2</sup> · Sanghamitra Bandyopadhyay<sup>2</sup>

Priya Roy  
priyaroy.rs@jadavpuruniversity.in

Dip Ghosh  
dipgi2005@gmail.com

Sanghamitra Bandyopadhyay  
sanghami@isical.ac.in

<sup>1</sup> Computer Science and Engineering, Jadavpur University, Kolkata, India

<sup>2</sup> Indian Statistical Institute, Kolkata, India