

Advance Computer Network

ML based classification and prediction of context using WiFi RSSI

Name	ID
Shrut Shah	202311014
Dixit Lukhi	202311017
Abhay Manavadariya	202311032

Abstract

Indoor localization is becoming an important area of research due to the increasing need for location-based services in places like shopping malls, universities, and hospitals. Unlike outdoor systems like GPS, which use satellite signals, indoor localization needs different methods because walls and other obstacles block or weaken signals.

Wi-Fi-based indoor localization is popular because it's **cost-effective** and most indoor spaces already have Wi-Fi networks. By analyzing Wi-Fi signal strength (RSSI) from access points, machine learning models can accurately predict a user's location. However, this process faces challenges like **signal fluctuations**, different types of devices, and changes in the environment.

This project uses machine learning methods, including advanced models like **XGBoost** and **Voting Classifiers**, to predict specific indoor locations, such as "**lobbies**," using Wi-Fi signal data. It focuses on preparing the data, training accurate models, optimizing settings, and evaluating results to create a reliable system.

The results show that this approach effectively overcomes the challenges of Wi-Fi-based indoor localization, providing a practical and scalable solution for real-world needs.

I. PROBLEM STATEMENT

The problem is that accurately predicting the location of a person inside a building, specifically in areas like "lobbies," is challenging due to the variability of Wi-Fi signals. These signals are affected by factors such as obstacles (walls, furniture), time of day, and different devices. Since traditional outdoor systems like GPS don't work well indoors, alternative methods are needed for indoor localization.

Wi-Fi-based systems are commonly used for this purpose because most buildings already have Wi-Fi infrastructure. However, predicting locations like lobbies using Wi-Fi signal strength (RSSI) faces issues such as:

- **Signal variability:** Wi-Fi signals fluctuate due to environmental changes like movement or interference from other devices.
- **Device differences:** Different devices (smartphones, tablets) may report slightly different signal strengths, making it harder to determine an accurate location.
- **Data collection:** Gathering accurate Wi-Fi signal data for each location within the building (like lobbies) is time-consuming and requires effort.

This problem is important because predicting a person's location in lobbies or other indoor spaces can enhance navigation, safety, and other smart building applications. The project aims

to solve this by using a machine learning method (weighted ensemble classifier) to predict the location (lobby) more accurately, even when Wi-Fi signals vary across devices and time.

Our team tackled the indoor localization problem by focusing on predicting the specific locations, such as **lobbies**, based on Wi-Fi signal strength (RSSI) data. Here's how we approached the solution:

1. **Data Collection:** We collected Wi-Fi signal data (RSSI values) from multiple Wi-Fi access points in different locations within the building, particularly the lobbies. This data was gathered using multiple handheld devices to account for device differences and various environmental conditions. The dataset included the RSSI values for each access point, collected at different times of the day and with different devices.

2. **Challenges Considered:** We addressed challenges like:

- **Signal variability:** Wi-Fi signals fluctuate due to obstacles, interference, and other environmental factors.
- **Device heterogeneity:** Different devices (smartphones, tablets) have different signal reception characteristics.
- **Temporal variation:** RSSI values change over time due to movement or other dynamic factors.

3. **Preprocessing the Data:** The raw Wi-Fi signal data was preprocessed to handle missing values and noise. This involved filling in missing RSSI values and removing outliers that could affect accuracy. We also calculated features like the **mean** and **variance** of RSSI signals to account for short-term fluctuations and environmental changes.

4. **Machine Learning Approach:** To predict the lobby locations based on the Wi-Fi signals, we used a **weighted ensemble classifier** approach. The ensemble method combined multiple machine learning models (base classifiers) to improve accuracy. We trained different models for various conditions (e.g., different times of day).

5. **Weighted Ensemble Decision Making:** The ensemble classifier worked by combining the predictions of several base classifiers. We used **weighted voting** to make the final prediction, where classifiers that were more accurate in certain conditions (e.g., with a specific device or time of day) were given more weight. This helped improve accuracy even when test conditions differed from the training conditions.

6. **Evaluation and Results:** The performance of our method was evaluated using several datasets, for predicting lobby locations. The method was particularly effective in handling the challenges of device heterogeneity and environmental changes.

II. WORK DISTRIBUTION

The project was collaboratively executed with the following division of responsibilities among the team members:

A. Abhay:

Abhay was responsible for the data preprocessing tasks, which included cleaning, merging, and structuring the dataset. Additionally, he handled the implementation and evaluation of the classification task for predicting the floor with lobby task, ensuring the model accurately mapped input data to the combined labels.

B. Shrut:

Shrut focused on understanding the foundational concepts and preparing the groundwork for the project. He thoroughly studied the relevant research papers and collaborated with the data collection team to ensure the availability of accurate and well-structured data. Shrut also gained expertise in implementing and evaluating KNN and SVM models, contributing to the understanding and integration of these classifiers in the project.

C. Dixit:

Dixit concentrated on understanding the Random Forest, Decision Tree models and XGBoost including its fine tuning, In addition to that he implemented and evaluated the task for prediction the lobby . His work ensured accurate predictions for the individual lobbies within the dataset.

III. METHODOLOGY

A. Block Diagram

The block diagram below illustrates the flow of our implementation.

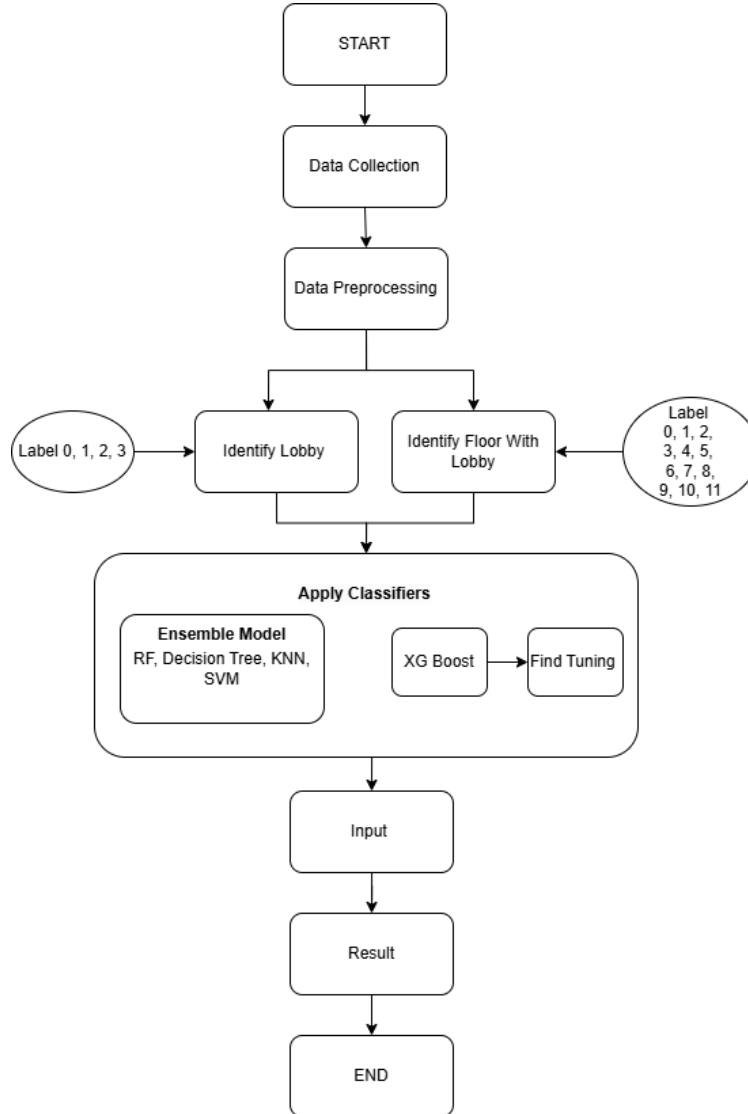


Fig. 1: Block Diagram illustrating the flow of the system

B. Wi-Fi Signal Measurement Dataset Description

The dataset provided by the data collection team contains Wi-Fi signal measurements collected across multiple locations. It includes detailed information about several visible Wi-Fi networks, capturing key attributes such as:

- **SSID (Service Set Identifier):** The name of the Wi-Fi network, used to identify different access points.
- **BSSID (Basic Service Set Identifier):** A unique identifier for each access point, representing the MAC address of the Wi-Fi device broadcasting the signal.
- **Signal Strength (RSSI):** The received signal strength indicator, which reflects the strength of the Wi-Fi signal at the given location.
- **Operating Frequency:** The frequency band used by the Wi-Fi network, typically either 2.4 GHz or 5 GHz.

This dataset serves as a comprehensive record of the Wi-Fi environment, helping to analyze the signal characteristics of different networks in various indoor spaces. It is crucial for building a Wi-Fi-based localization system, where the signal strength measurements are used to estimate the user's location within the building.

A	B	C	D	E	F
Timestamp	Location	SSID	BSSID	Signal Strength	Frequency
14-11-2024 15:54	LAB_0_3PM_A_1	Attendance	58:b6:33:7a:1c:d8	-68	2437MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DAIICT_STAFF	58:b6:33:3a:1c:d8	-70	2437MHz
14-11-2024 15:54	LAB_0_3PM_A_1	Lab_211_5G	24:c9:a1:20:e7:4c	-81	5280MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DAIICT_Student	60:d0:2c:df:63:9c	-56	5220MHz
14-11-2024 15:54	LAB_0_3PM_A_1	TP-Link_13F8_5G_20	9c:53:22:e7:13:fa	-83	5765MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DA_Public	18:4b:0d:a3:75:38	-67	2412MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DAIICT_Student	18:4b:0d:e3:75:38	-67	2412MHz
14-11-2024 15:54	LAB_0_3PM_A_1	dasolarpnl	c0:c5:20:24:87:18	-76	2467MHz
14-11-2024 15:54	LAB_0_3PM_A_1		9e:53:22:a7:13:f8	-78	2417MHz
14-11-2024 15:54	LAB_0_3PM_A_1	Dev	b6:16:63:ff:73:29	-86	5745MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DA_Public	58:b6:33:ba:1c:d8	-68	2437MHz
14-11-2024 15:54	LAB_0_3PM_A_1	VM_31	4a:7a:79:08:30:db	-77	2412MHz
14-11-2024 15:54	LAB_0_3PM_A_1		52:91:e3:25:ac:c9	-75	2422MHz
14-11-2024 15:54	LAB_0_3PM_A_1	Dell-Gujcost	78:98:e8:7a:2c:1f	-83	2472MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DAIICT_Student	44:1e:98:fb:55:d8	-72	2462MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DAIICT_Student	24:79:2a:34:ea:0d	-43	5745MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DAIICT_STAFF	1c:b9:c4:3d:84:e8	-70	2462MHz
14-11-2024 15:54	LAB_0_3PM_A_1	Attendance	18:4b:0d:63:60:7c	-70	5805MHz
14-11-2024 15:54	LAB_0_3PM_A_1	Lab_211	24:c9:a1:20:e7:48	-72	2457MHz
14-11-2024 15:54	LAB_0_3PM_A_1	Attendance	1c:b9:c4:7d:84:e8	-70	2462MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DAIICT_Student	58:b6:33:fa:1c:d8	-68	2437MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DAIICT_STAFF	18:4b:0d:23:75:3c	-82	5600MHz
14-11-2024 15:54	LAB_0_3PM_A_1	Attendance	18:4b:0d:63:75:3c	-83	5600MHz
14-11-2024 15:54	LAB_0_3PM_A_1	DAIICT_Student	18:4b:0d:dc:30:9c	-64	5540MHz

Fig. 2: Data Collected

C. Data Pre-Processing

The following data pre-processing steps were performed to clean and structure the Wi-Fi signal dataset:

- 1) **Data Merging:** All data collected from different locations and at various timestamps were combined into a single dataset. This step provided a unified view of the Wi-Fi networks across the entire dataset, allowing for a comprehensive analysis.
- 2) **Filtering Specific SSIDs:** The dataset was filtered to retain only records corresponding to the DAIICT_Student and DA_Public SSIDs. This focused the analysis specifically on these two Wi-Fi networks, as they were of primary interest in the study.
- 3) **Pivoting the Data:** After filtering the dataset, it was pivoted to create columns for each unique MAC address (BSSID). This resulted in a total of 54 columns, each representing the signal strength (RSSI) of a specific MAC address across the dataset.

- 4) **Handling Missing Values:** The dataset contained some missing values (NaN). These missing values were replaced with -100 dBm, which is considered the lowest possible signal strength, to ensure consistency in the dataset.

These preprocessing steps ensured that the dataset was properly structured, cleaned, and ready for further analysis.

	Location	SSID	AP01	AP02	AP03	AP04	AP05	AP06	AP07	AP08	...	AP46	AP47	AP48	AP49	AP50	AP51	AP52	AP53	AP54
0	LAB_0_12PM_A_1	DAICT_Student	-100.0	-100.0	-100.0	-100.0	-100.0	-100.0	-52.0	-68.0	...	-100	-100	-100	-100	-100	-100	-100	-100	-100
1	LAB_0_12PM_A_1	DA_Public	-52.0	-68.0	-79.2	-100.0	-83.0	-100.0	-100.0	-100.0	...	-100	-100	-100	-100	-100	-100	-100	-100	-100
2	LAB_0_12PM_A_10	DAICT_Student	-100.0	-100.0	-100.0	-100.0	-100.0	-100.0	-75.0	-100.0	...	-100	-100	-100	-100	-100	-100	-100	-100	-100
3	LAB_0_12PM_A_10	DA_Public	-100.0	-100.0	-85.0	-100.0	-100.0	-85.0	-100.0	-100.0	...	-100	-100	-100	-100	-100	-100	-100	-100	-100
4	LAB_0_12PM_A_11	DAICT_Student	-100.0	-100.0	-100.0	-100.0	-100.0	-100.0	-100.0	-100.0	...	-100	-100	-100	-100	-100	-100	-100	-100	-100

Fig. 3: Data Preprocessing

D. Lobby Prediction

To categorize the data into predefined lobbies (Lobby_0, Lobby_1, Lobby_2, Lobby_3), the following logic was applied based on the `Location` field:

- 1) **Extracting Cell Information:** The cell identifier within the `Location` string (e.g., `A_1` in `LAB_0_3PM_A_1`) was extracted using a regular expression. This step helped to isolate the relevant cell information for further mapping to specific lobbies.
- 2) **Mapping to Lobbies:** The extracted cell details were mapped to specific lobbies based on predefined rules:
 - **Lobby_0:** Locations with cell letters A or B and numbers ranging from 1 to 14.
 - **Lobby_1:** Locations with cell letters P or Q and numbers ranging from 1 to 14.
 - **Lobby_2:** Locations with cell letters in C to O (excluding P and Q) and numbers 1 or 2.
 - **Lobby_3:** Locations with cell letters in C to O and numbers 13 or 14.
- 3) **Implementation:** A function called `categorize_lobby` was developed to implement the mapping logic. This function was applied to the `Location` column of the dataset. The result was stored in a new column called `Lobby`, which contains the predicted lobby categories (Lobby_0, Lobby_1, Lobby_2, Lobby_3).

These steps allowed the categorization of locations into predefined lobbies based on the cell information in the `Location` field.

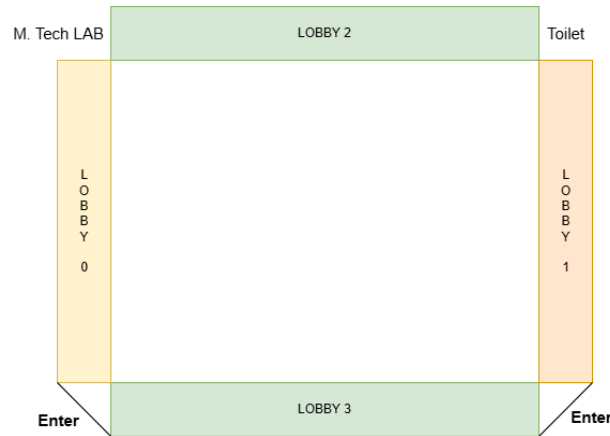


Fig. 4: Lobby Prediction

E. Preparing Data for Training and Testing

The following steps were followed to prepare the dataset for training and testing the machine learning models:

- 1) **Extracting Unique MAC Addresses and Lobby Labels:** To prepare the dataset for machine learning, the unique MAC addresses were used as features (denoted as X), and the predicted lobby labels ($Lobby$) were used as the target variable (denoted as y). The MAC addresses represented the signal strengths from different access points, which were used to classify the locations into lobbies.
- 2) **Splitting the Dataset:** The dataset was divided into training and testing sets using the `train_test_split` function from the `sklearn` library.
 - **Training Set:** 80% of the data was used to train the model.
 - **Testing Set:** 20% of the data was reserved for evaluating the model's performance.

A random state of 42 was used to ensure reproducibility of the splits.

- 3) **Model Training and Evaluation:** Four individual classifiers were utilized to predict the lobby based on the signal data:
 - **Random Forest Classifier**
 - **Decision Tree Classifier**
 - **K-Nearest Neighbors (KNN) Classifier**
 - **Support Vector Machine (SVM) Classifier**

Each classifier's performance was evaluated using 5-fold cross-validation, where the dataset was divided into five subsets. This ensures a reliable estimate of accuracy by evaluating the model on different portions of the data. The accuracy scores for individual classifiers were calculated, highlighting their contributions to the model's performance.

- 4) **Ensemble Model (Voting Classifier):** An ensemble model was created using a `Voting Classifier` with hard voting, which combines the predictions of the four individual classifiers. Hard voting selects the label that receives the majority vote from the individual classifiers.
 - The ensemble model underwent 5-fold cross-validation, and the mean accuracy score was recorded.
 - The model was trained on the training data (X_{train} , y_{train}) and saved using the `pickle` library for future use.
- 5) **Evaluation on the Test Set:** After training, the ensemble model was evaluated on the test dataset (X_{test} , y_{test}). The accuracy score was calculated to assess the model's performance on unseen data.

These steps ensured that the dataset was properly prepared and that the model was trained and evaluated effectively using multiple classifiers and an ensemble approach.

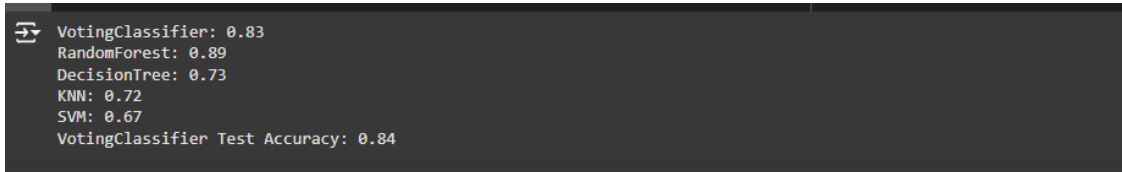


Fig. 5: Model-wise Accuracy for the Lobby Prediction

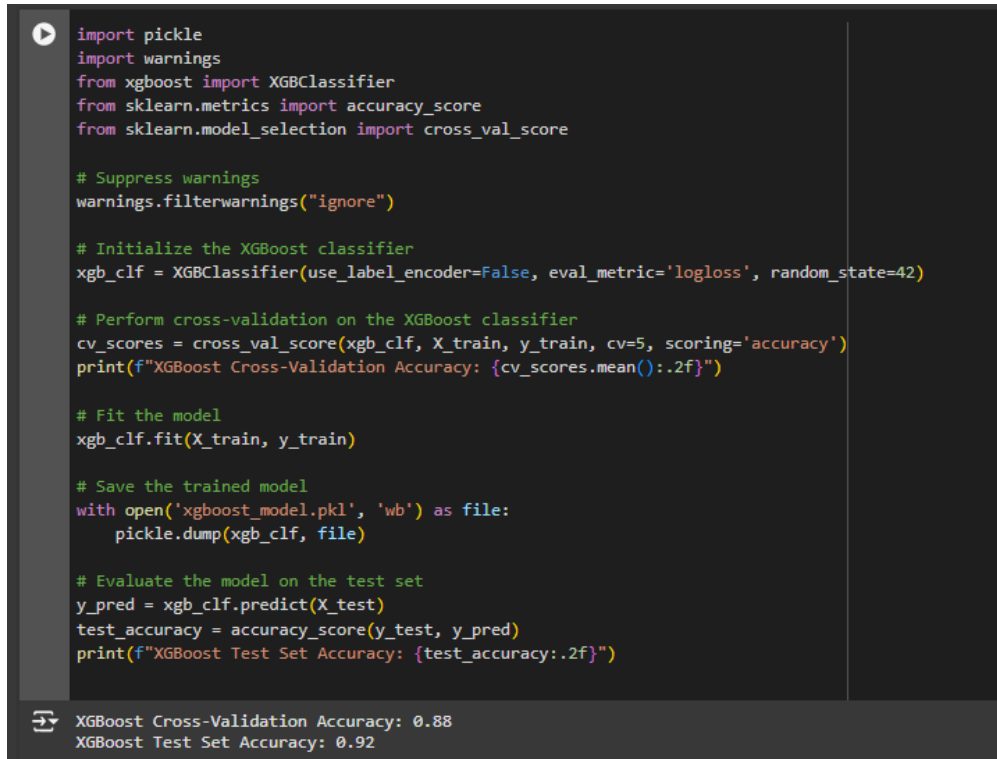
F. XGBoost Classifier: Training and Evaluation

1) **Introduction to XGBoost:** To enhance model performance and capture complex patterns in the data, we utilized the `XGBoost Classifier`. XGBoost is a powerful gradient boosting

algorithm known for its efficiency, scalability, and superior predictive capabilities. It is particularly effective in handling large datasets and complex models, making it a popular choice for classification tasks.

2) *Cross-Validation*: The model's performance was first evaluated using **5-fold cross-validation** to ensure reliability. This approach splits the training dataset into five subsets. The model is iteratively trained on four of the subsets and validated on the remaining one. The mean accuracy score obtained from cross-validation was 88%, indicating strong predictive capabilities.

3) *Test Set Evaluation*: After training, the model achieved a test set accuracy of 92%, demonstrating its effectiveness in predicting the lobby based on the input data. This high accuracy further emphasizes the model's ability to generalize well on unseen data.



```
import pickle
import warnings
from xgboost import XGBClassifier
from sklearn.metrics import accuracy_score
from sklearn.model_selection import cross_val_score

# Suppress warnings
warnings.filterwarnings("ignore")

# Initialize the XGBoost classifier
xgb_clf = XGBClassifier(use_label_encoder=False, eval_metric='logloss', random_state=42)

# Perform cross-validation on the XGBoost classifier
cv_scores = cross_val_score(xgb_clf, X_train, y_train, cv=5, scoring='accuracy')
print(f"XGBoost Cross-Validation Accuracy: {cv_scores.mean():.2f}")

# Fit the model
xgb_clf.fit(X_train, y_train)

# Save the trained model
with open('xgboost_model.pkl', 'wb') as file:
    pickle.dump(xgb_clf, file)

# Evaluate the model on the test set
y_pred = xgb_clf.predict(X_test)
test_accuracy = accuracy_score(y_test, y_pred)
print(f"XGBoost Test Set Accuracy: {test_accuracy:.2f}")
```

XGBoost Cross-Validation Accuracy: 0.88
XGBoost Test Set Accuracy: 0.92

Fig. 6: XGBoost Model and Accuracy

G. XGBoost Classifier: Hyperparameter Tuning and Evaluation

1) *Hyperparameter Tuning with GridSearchCV*: To improve the performance of the XGBoost Classifier, hyperparameter tuning was performed using GridSearchCV. We defined a grid of possible values for the model's hyperparameters, including:

- **n_estimators**: The number of boosting rounds (estimators) in the model.
- **learning_rate**: The step size used to update the model during training.
- **max_depth**: The maximum depth of the decision tree.

The grid search was executed with 5-fold cross-validation, selecting the best combination of hyperparameters based on the highest accuracy achieved during the cross-validation process.

2) *Model Retraining and Results*: After fine-tuning, the best combination of hyperparameters was selected, and the model was retrained on the entire training dataset. The model was then evaluated on the test set, which resulted in an improvement in performance.

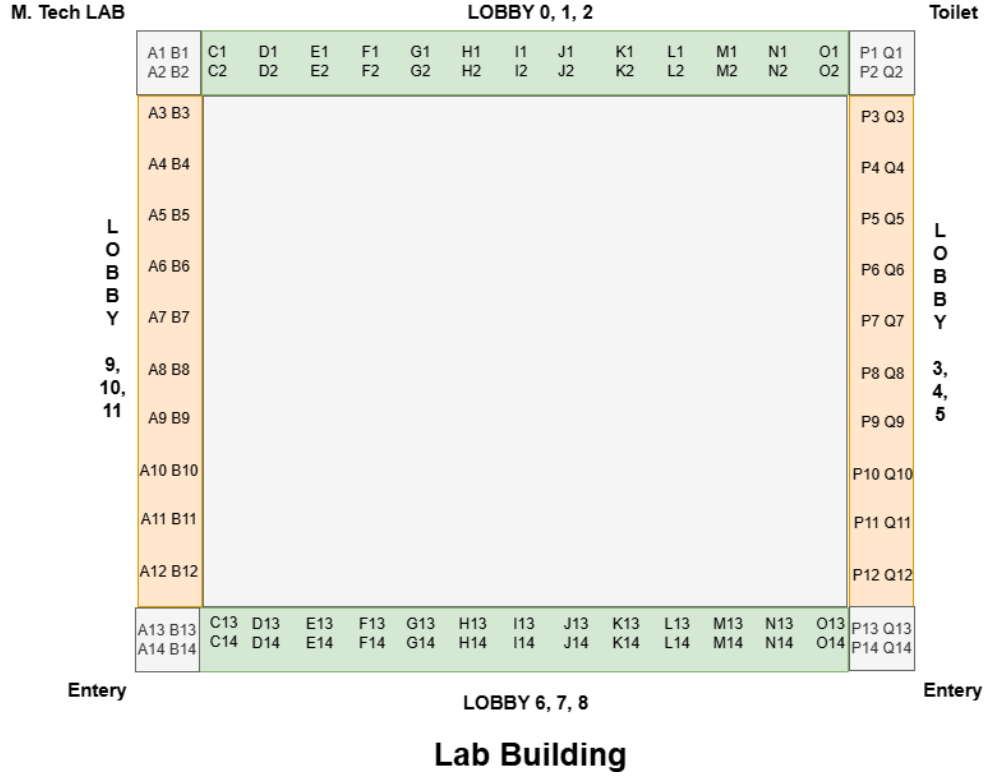


Fig. 9: Example of Floor and Lobby Labeling

I. Floor with Lobby Prediction

The dataset includes Wi-Fi signal data collected from three floors, each having four lobbies. This setup provides 12 unique location labels, where each label corresponds to a specific floor and lobby combination (e.g., "Floor_1_Lobby_0", "Floor_1_Lobby_1", ...).

We have created a new column, `Label`, in the dataset. This column assigns a unique numeric label to each floor-lobby combination based on predefined rules. For example:

- `Floor_1_Lobby_0` might be assigned label 0,
- `Floor_1_Lobby_1` might be assigned label 1,
- And so on, until all 12 combinations are covered.

This labeling scheme ensures that the floor and lobby are uniquely encoded for each data point, making it easier for the model to classify locations within multi-floor buildings.

J. Label Distribution

We have this much data for each label. Refer to the image below for a visualization of the distribution of labels across the dataset.

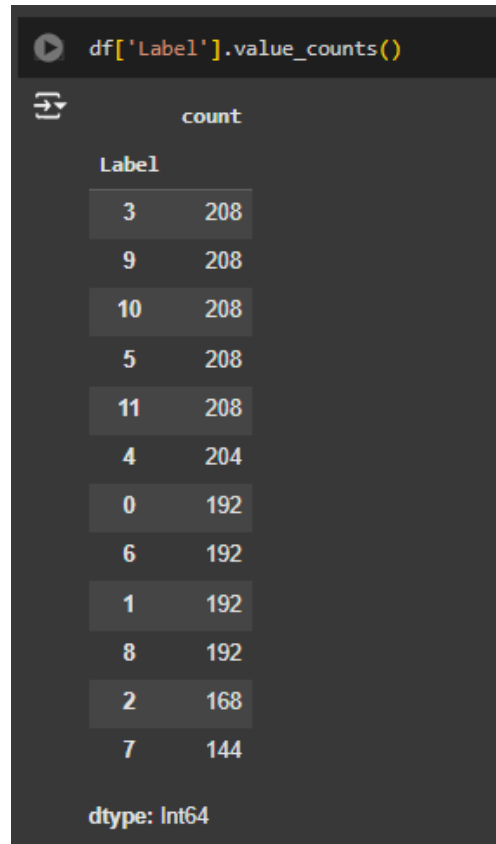


Fig. 10: Distribution of Floor and Lobby Labels in the Dataset

K. Data Splitting: Features (X) and Labels (Y)

After creating the `Label` column, the dataset was split into two parts:

- 1) **Features (X):** X contains the signal strength values corresponding to all unique MAC addresses in the dataset. These features represent the RSSI (Received Signal Strength Indicator) readings from different Wi-Fi access points, which the model uses to predict the location (specific floor and lobby combination). These features are the input data for the classification model.
- 2) **Labels (Y):** Y contains the newly created `Label` column, which uniquely encodes the floor and lobby combination as a numeric value. For example, each label corresponds to a specific combination like "Floor_1_Lobby_0" or "Floor_2_Lobby_3." This column serves as the target variable for the classification model.

The dataset was thus structured with X as the input features and Y as the target labels, preparing the data for model training.

IV. CODE LINK

<https://drive.google.com/drive/folders/1V67TPNekgMDquJCTJyCN8WhmeImA3eT8?usp=sharing>

V. RESULTS AND DISCUSSION

A. Model Training and Evaluation

The classification task to predict floor with lobby combinations was performed using individual classifiers and a Voting Classifier (ensemble model). Below are the results from cross-validation:

- **Random Forest: 88%**
- **Decision Tree: 72%**
- **K-Nearest Neighbors (KNN): 71%**
- **Support Vector Machine (SVM): 73%**
- **Voting Classifier (Ensemble): 82%**

```

VotingClassifier: 0.82
RandomForest: 0.88
DecisionTree: 0.72
KNN: 0.71
SVM: 0.73
VotingClassifier Test Accuracy: 0.82

```

Fig. 11: Accuracy of the classifiers

The Random Forest Classifier showed the highest individual accuracy during cross-validation. However, the Voting Classifier, which combined the strengths of all individual models, achieved a solid accuracy of 82%.

1) *Test Set Accuracy:* After training on the full dataset, the Voting Classifier was evaluated on the test set and achieved an accuracy of 82%, which was consistent with its cross-validation performance.

2) *XGBoost Classifier:* The XGBoost classifier was used to predict the combined floor-lobby labels. Below are the results from the cross-validation and test set evaluation:

- **Cross-Validation Accuracy: 86%**
- **Test Set Accuracy: 87%**

```

XGBoost Cross-Validation Accuracy: 0.86
XGBoost Test Set Accuracy: 0.87

```

Fig. 12: Test Accuracy

After fine-tuning the hyperparameters, the accuracy increased to 87%, demonstrating the effectiveness of the XGBoost classifier in predicting the floor-lobby labels.

```

Fitting 5 folds for each of 27 candidates, totalling 135 fits
Best Parameters: {'learning_rate': 0.2, 'max_depth': 5, 'n_estimators': 150}
XGBoost (Tuned) Cross-Validation Accuracy: 0.87
XGBoost (Tuned) Test Set Accuracy: 0.87

```

Fig. 13: Fine Tuning Accuracy

3) *Model Prediction:* After training, the model was used to predict the floor-lobby combination for new input data. By providing RSSI signal strength values to the model, it successfully predicted the corresponding floor-lobby label. Based on the input data, the model predicted the following result:

Predicted Floor-Lobby Combination: Floor_0_Lobby_0

