# Random Forest Algorithm:

The Random Forest algorithm is an ensemble learning method that builds multiple decision trees during training and outputs the class that is the mode of the classes (classification) or the mean prediction (regression) of the individual trees. Here's a mathematical representation of the Random Forest effectively addresses overfitting in decision trees by employing bagging techniques. This involves randomly selecting samples, constructing decision trees for each sample, and training the model by aggregating the results through a voting mechanism among the decision trees.

**Mathematical Representation:**

Given a dataset {(X_1, Y_1), (X_2, Y_2), ..., (X_N, Y_N)}:

1. Training:

- For each tree t in the forest N:
  - Sample a bootstrap sample S_t from the dataset with replacement.
  - Randomly select k features from the K features.
  - Train a decision tree t using S_t and the selected features.

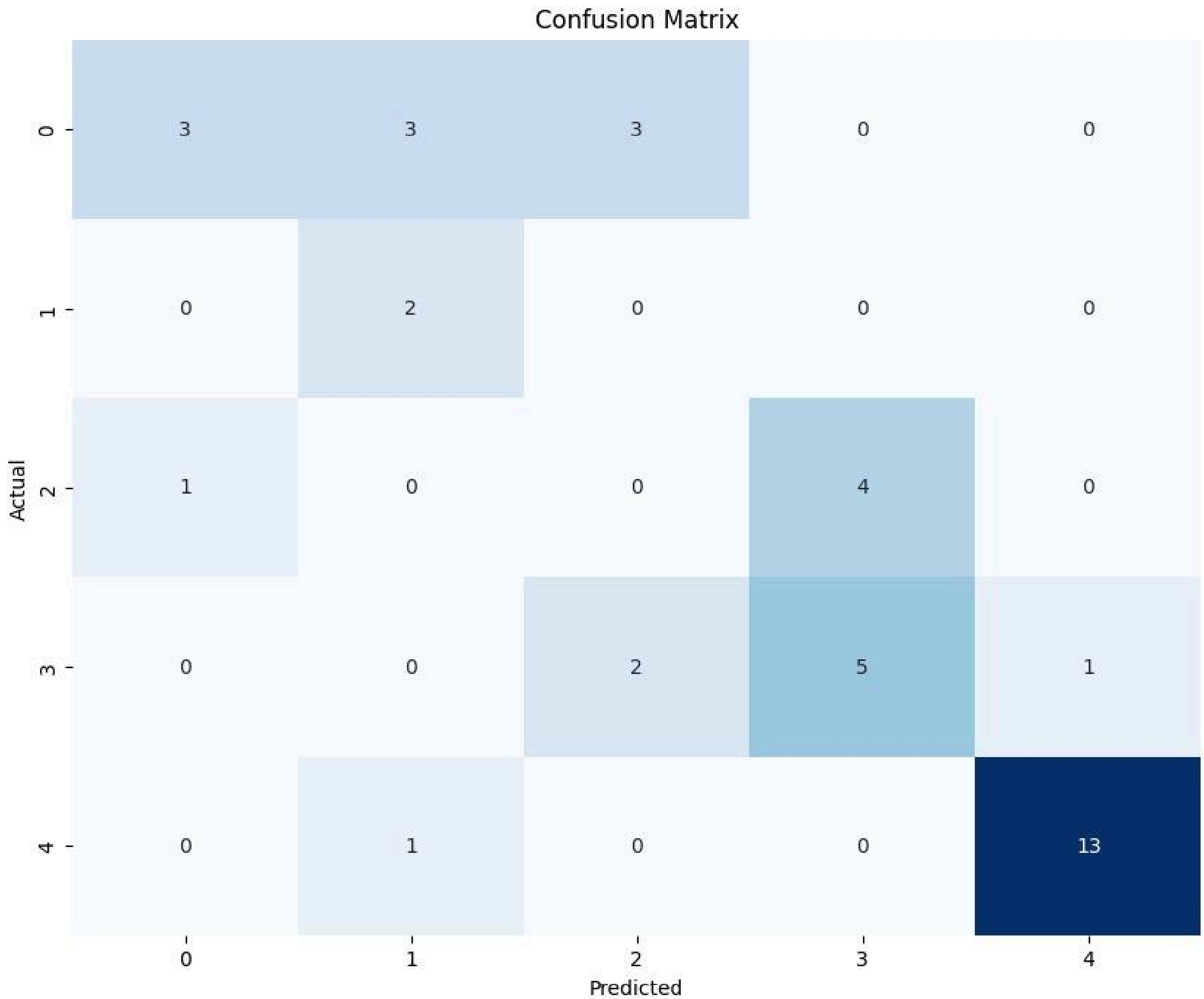
2. Prediction:

- Given a new data point X_new:
  - For each tree t in the forest N:
    - Traverse the decision tree t using the features of X_new.
    - Predict the class or regression value at the leaf node.
  - Aggregate the predictions from all trees to obtain the final prediction.

This mathematical representation outlines the key steps involved in the Random Forest algorithm for both training and prediction.

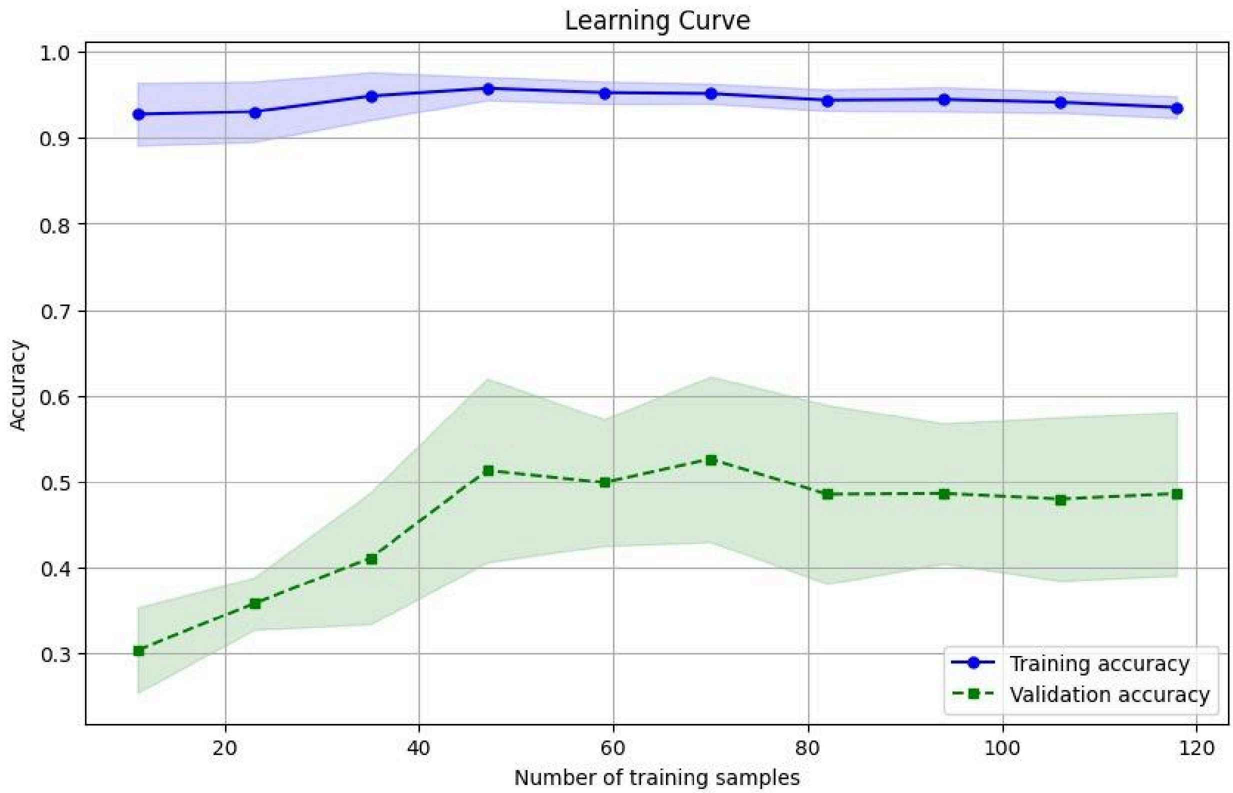
## Accuracy Score of Model:

Confusion Matrix:




Add constraint of medium to low accuracy due to overlapping paths

**Learning Curve:**




Graph representing the accuracy of the model with respect to the number of samples taken in the Random forest algorithm.

## Accuracy Metrics:

**Accuracy**: The accuracy of the model was measured to be 60.53%. This metric reflects the proportion of correctly classified instances out of the total instances evaluated.

**Precision**: Precision, indicating the model's ability to accurately classify positive instances, was calculated at 65.42%. This metric is crucial in scenarios where the cost of false positives is significant.

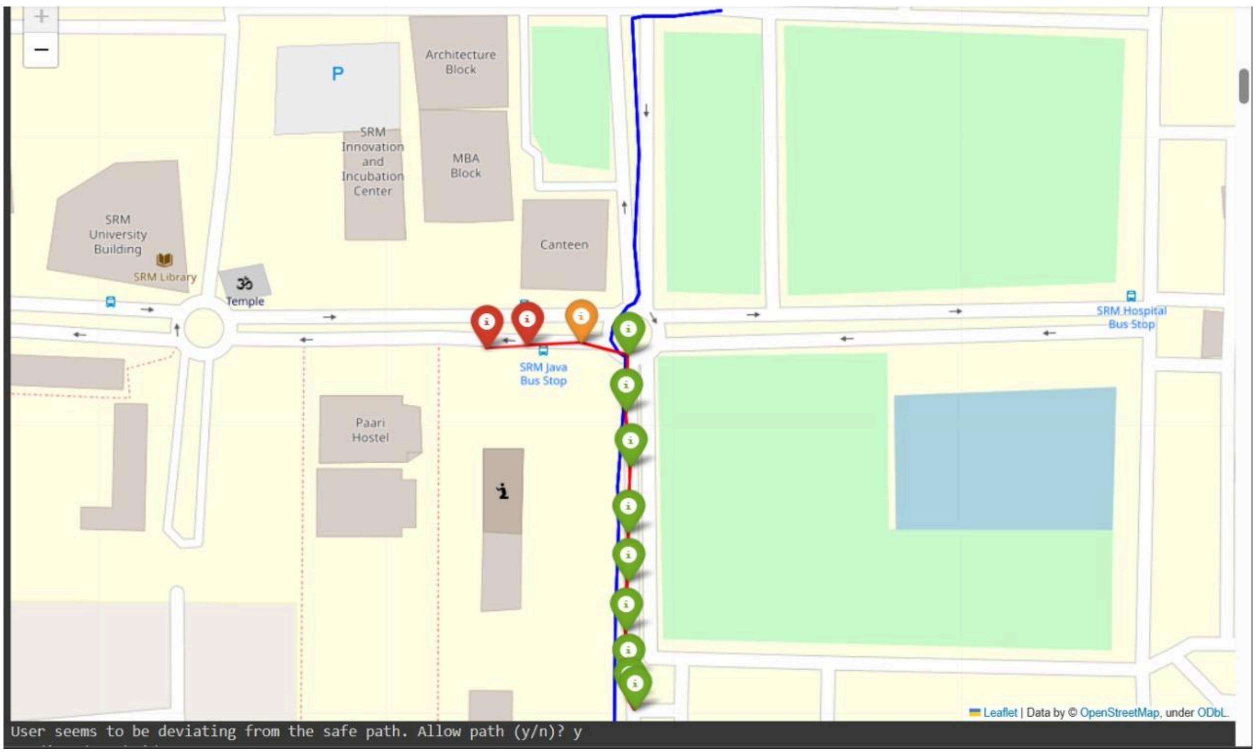
**Recall**: The recall of the model was observed to be 60.53%. This metric, also known as sensitivity, measures the proportion of actual positive instances correctly identified by the model.

**F1-score**: The F1-score, which balances precision and recall, was computed at 60.16%. This metric provides a harmonic mean of precision and recall, offering a single value to assess the overall performance of the model.

## ML Model Demonstration:

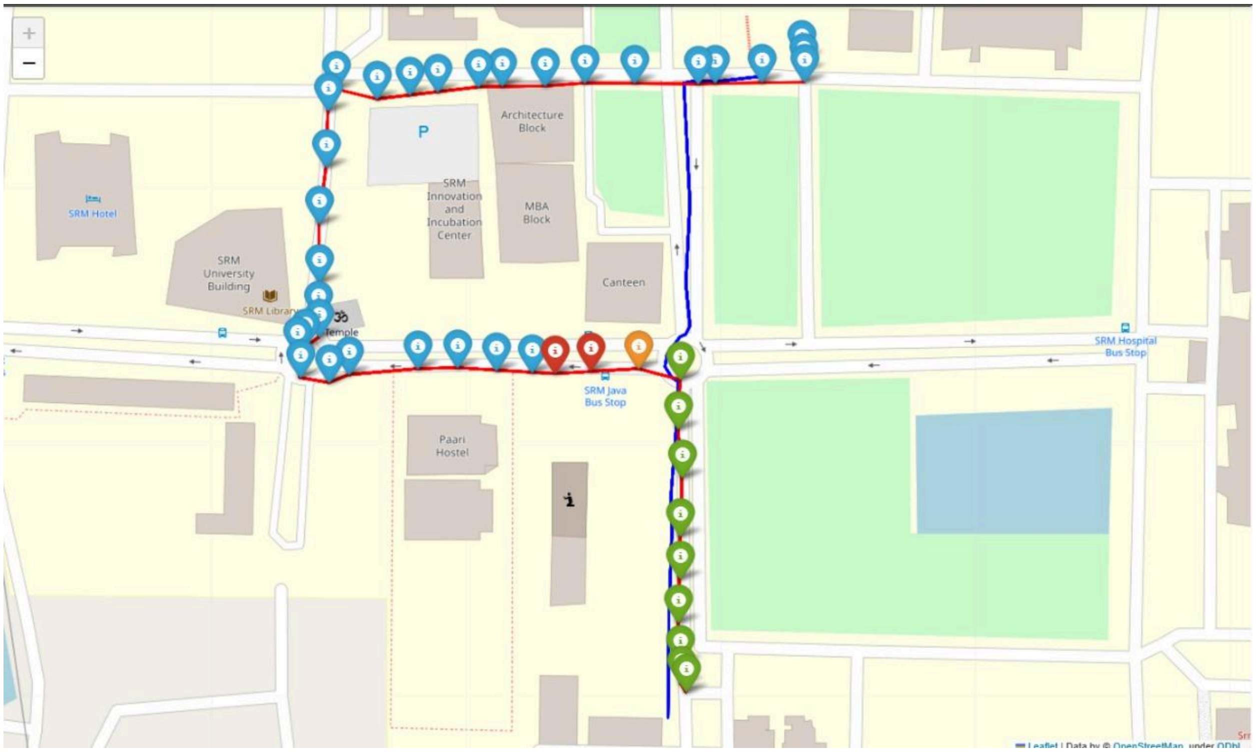
The utilized algorithm primarily revolves around predicting the path_id of the user's traversal. The ML model is tasked with predicting the path_id, following which deviation analysis is conducted based on the predicted path. This analysis aims to ascertain whether the user is straying from a safe path.

Green markers signify that the user is on a safe path, where the deviation is less than 15 meters. Yellow markers indicate potential deviation, falling within the range of 15 to 25 meters. Red markers are indicative of significant deviation, exceeding 25 meters from the predicted safe path.

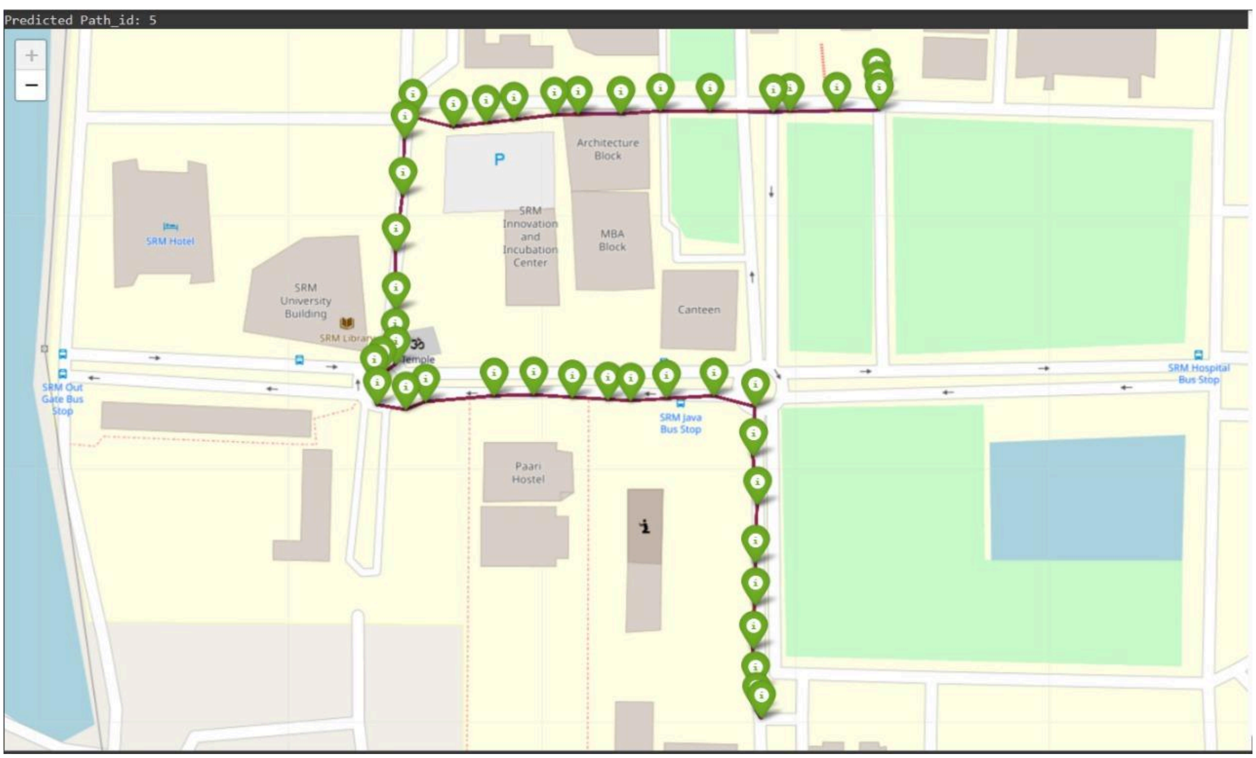



If the user appears to deviate from the designated path, the guardian is alerted. Upon approval by the guardian, a new path is incorporated into the dataset, and the machine learning model undergoes retraining with the updated data.

The user is deemed to be deviating from a safe path when there is a consistent increase in deviation from the predicted path.




New paths, represented by blue markers, are appended to the database.




Following model retraining, the path is classified as safe upon subsequent traversal by the user.