Wi-fi fingerprinting location

Contents

[Goal 2](#_Toc111556180)

[Data description 2](#_Toc111556181)

[Data management 2](#_Toc111556182)

[Models trained 3](#_Toc111556183)

[Recommendations 3](#_Toc111556184)

[Model summary 4](#_Toc111556185)

## Goal

Determine a person’s location in indoor spaces using nothing but wi-fi fingerprints.

## Data description

The database covers three buildings with 4 or more floors in a 110.000m2 area. It consists of 19,937 records and 529 attributes where 520 of them are the Wireless Access Points (WAP) and the corresponding Received Signal Strength Intensity (RSSI) that make the wi-fi fingerprint. The intensity of the signal is denoted by negative values ranging from -104dBm (extremely poor signal) to 0dbM. If no WAP is detected, it is denoted as the positive value of 100.

The other 9 attributes are the latitude, longitude, floor, buildingid, spaceid, relativeposition, userid, phoneid, and timestamp. The spaceid is an integer attribute that refers to a particular space (e.g., office, corridor, classroom, etc.). The relativeposition is an integer binary attribute where 1 represents inside and 2 represents outside of a particular space. The buildingid, userid and phoneid are the identifiers of the respective buildings, users, and phones. The timestamp is an integer attribute in UNIX time. And finally, longitude is a negative real value and latitude is a positive real value.

## Data management

The approach to achieving the goal will be through the implementation of several different classification models. The response variable will be a categorical variable made through the combination of buildingid, floor, spaceid, and relative position. Then the data will be sampled to train the models. In this case, the focus will be on floor 1 in building 2 with a sample size of 2,162 or 10.8% of the original database. Furthermore, the features selected to do the analysis will only be the WAP attributes that have some variance. In other words, the WAP attributes that have zero variance will be dropped.

## Models trained

There were three models trained. A C5.0 decision tree, KNN, and Random Forest. The model that had the best metrics was the Random Forest (**Graph 1**).

**Graph 1.**

Chart, bar chart

Description automatically generated

## Recommendations

Some recommendations to solve this problem are to use a simple algorithm first so that it is faster to get feedback on the model in case of errors. A starting algorithm I’d recommend would be the KNN because it was the fastest model trained. However, as stated before the random forest algorithm has the best metrics of the three. Therefore, it is time saving to start with a simple algorithm and then progress to a more complex one.

The results of the models might be improved if Bluetooth signals were included in the database.

## Model summary

Accuracy

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model | Min | 1st Qu. | Median | Mean | 3rd Qu. | Max. | NA's |
| C50 | 0.65 | 0.73 | 0.75 | 0.75 | 0.77 | 0.85 | 0 |
| KNN | 0.56 | 0.63 | 0.65 | 0.64 | 0.66 | 0.71 | 0 |
| RF | 0.81 | 0.83 | 0.84 | 0.85 | 0.86 | 0.90 | 0 |

Kappa

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model | Min | 1st Qu. | Median | Mean | 3rd Qu. | Max. | NA's |
| C50 | 0.64 | 0.72 | 0.75 | 0.74 | 0.77 | 0.85 | 0 |
| KNN | 0.56 | 0.62 | 0.64 | 0.64 | 0.66 | 0.71 | 0 |
| RF | 0.81 | 0.83 | 0.84 | 0.85 | 0.86 | 0.90 | 0 |