Abstract

This report will investigate the accuracy detection of indoor localization and navigation with BLE (Bluetooth Low Energy) RSSI iBeacon devices which were installed on the first floor of Waldo Library, Western Michigan University. The dataset is collected from UCI Machine Learning Repository available online for download and it is a large dataset consist of 1420 instances (labelled dataset) with 13 iBeacons device. It is examined by the performance of each iBeacon installed in a real-world environment and applied machine learning algorithms to find the level of accuracy.

1. Introduction

Smart phones and mobile devices have been widely used in this global era. Mobile applications have been developed to connect contextual information such as tracking position/ localization and it has become increasingly powerful. The functions of this application can be used to assist in emergency response or as indoor navigation in large environments such as libraries, airports, or malls.

Machine Learning has been implemented to improve performance of the applications over time. Supervised learning algorithms such as Classification are trained on data which can provide excellent solutions for building models in a large data with many attributes; even for a task that is impossible by other means. In this report, Classification will be implemented as the analytical method to predict the unseen data using the experience or past data.

2. Methodology

2.1 Data Preparation

As a Data Scientist, it is important to first understand the data before deciding what method to use. The dataset is external and consists of 1420 instances and 15 attributes where 13 of them are the iBeacon devices, 1 attribute is the timestamp, and the other is location.

The RSSI measurements are negative values. Bigger RSSI values indicate closer proximity to a given iBeacon (e.g. RSSI of -65 represent a closer distance to a given iBeacon compared to RSSI of -85). For out-of-range iBeacons, the RSSI is indicated by -200. The location related to RSSI readings are combined in one column consisting a letter for the column and a number for the row of the position.

For a better understanding about the position of each iBeacon inside the library, below is the map of Waldo Library:

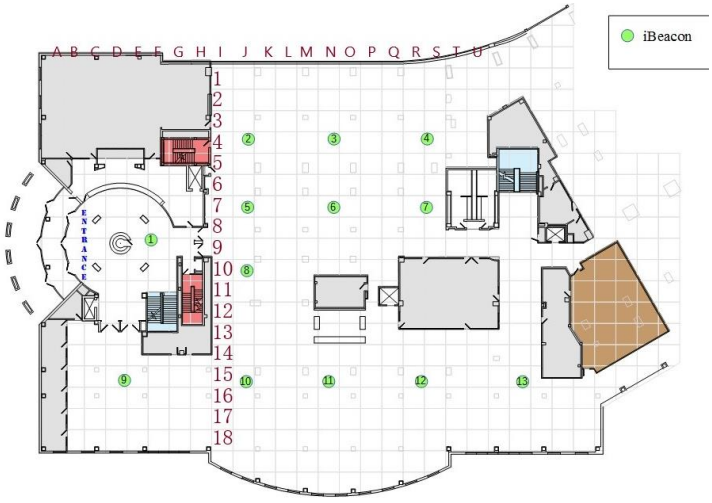

Figure 1. Waldo Library map with iBeacon installed in the grid location

As can be seen from figure 1, the locations related to RSSI readings are combined in one column consisting a letter for the column and a number for the row of the position. The location of receiving RSSIs from iBeacons (b3001 to b3013); symbolic values showing the column and row of the location on the map.

2.2 Data Exploration

Data cleaning is performed to check on any missing values or irrelevant data and adjust the values to be acceptable for data modelling .We need to understand the shape of the data which can be useful in considering the features that will be used and performed in the next step. In this step, we will count the distributions of all variables, check the data type of each attribute, find the missing value, and the correlations.

In this report, we are interested in finding the relationship between the location and the iBeacon. The features and labels are then identified as follow:

1. Location as the label
2. B3001 - B3013 as the features

Before training a model with the dataset, we need to handle the missing value problem that may impact the machine learning model's quality. The missing value which is indicated as -200 is first being replaced with NaN, then we calculate the mean of the features before replacing the NaN value with the average score of each feature.

Descriptive Statistics

Upon checking the descriptive statistics, it has been found that feature b3002 has a minimum value of -198 and b3012 has a minimum value of -199. This could be an error and can be replaced as NaN/empty value. And below is the result of removed of all missing values:

	b3001	b3002	b3003	b3004	b3005	b3006	b3007	b3008	b3009	b3010	b3011	b3012	b3013
count	25.000000	486.000000	280.000000	402.000000	247.000000	287.000000	50.000000	91.000000	31.000000	29.000000	25.000000	31.000000	44.000000
mean	-76.480000	-73.308642	-75.917857	-74.723881	-75.696356	-76.620209	-76.100000	-74.703297	-69.225806	-74.758621	-72.120000	-73.419355	-73.022727
std	4.134408	5.844321	5.794297	5.136625	4.695711	4.019012	6.952462	6.013863	8.849519	6.168209	7.562627	8.106681	7.963547
min	-81.000000	-87.000000	-88.000000	-88.000000	-83.000000	-87.000000	-85.000000	-83.000000	-82.000000	-81.000000	-85.000000	-82.000000	-87.000000
25%	-80.000000	-78.000000	-80.000000	-78.000000	-79.000000	-79.000000	-80.750000	-79.000000	-77.000000	-79.000000	-79.000000	-81.000000	-79.500000
50%	-78.000000	-74.000000	-78.000000	-76.000000	-77.000000	-77.000000	-79.000000	-77.000000	-72.000000	-78.000000	-72.000000	-77.000000	-75.000000
75%	-74.000000	-69.000000	-74.000000	-71.000000	-73.000000	-75.000000	-73.000000	-71.500000	-59.500000	-72.000000	-67.000000	-66.500000	-65.750000
max	-67.000000	-59.000000	-56.000000	-56.000000	-60.000000	-62.000000	-58.000000	-56.000000	-55.000000	-61.000000	-59.000000	-60.000000	-59.000000

Descriptive Statistic of iBeacons

The above Descriptive Statistics showing each mean of the feature, standard deviation, min and max value, and quantiles. This table will be useful for Data Modelling in the next step for the algorithms to calculate the best model to fit.

To explore each feature (b3001 – b3013), Histogram will be used to identify how the iBeacons are performing in the area compare to the other nearby iBeacons.

The Scatter Matrix visualisation will also be implemented to observe the similarities and differences between features visually.

To check the comparison between locations and iBeacons, it will be depicted using Scatter plots and the iBeacons are grouped based on the nearest location as per the map of Waldo library.

Histograms and Scatter Matrix of iBeacons

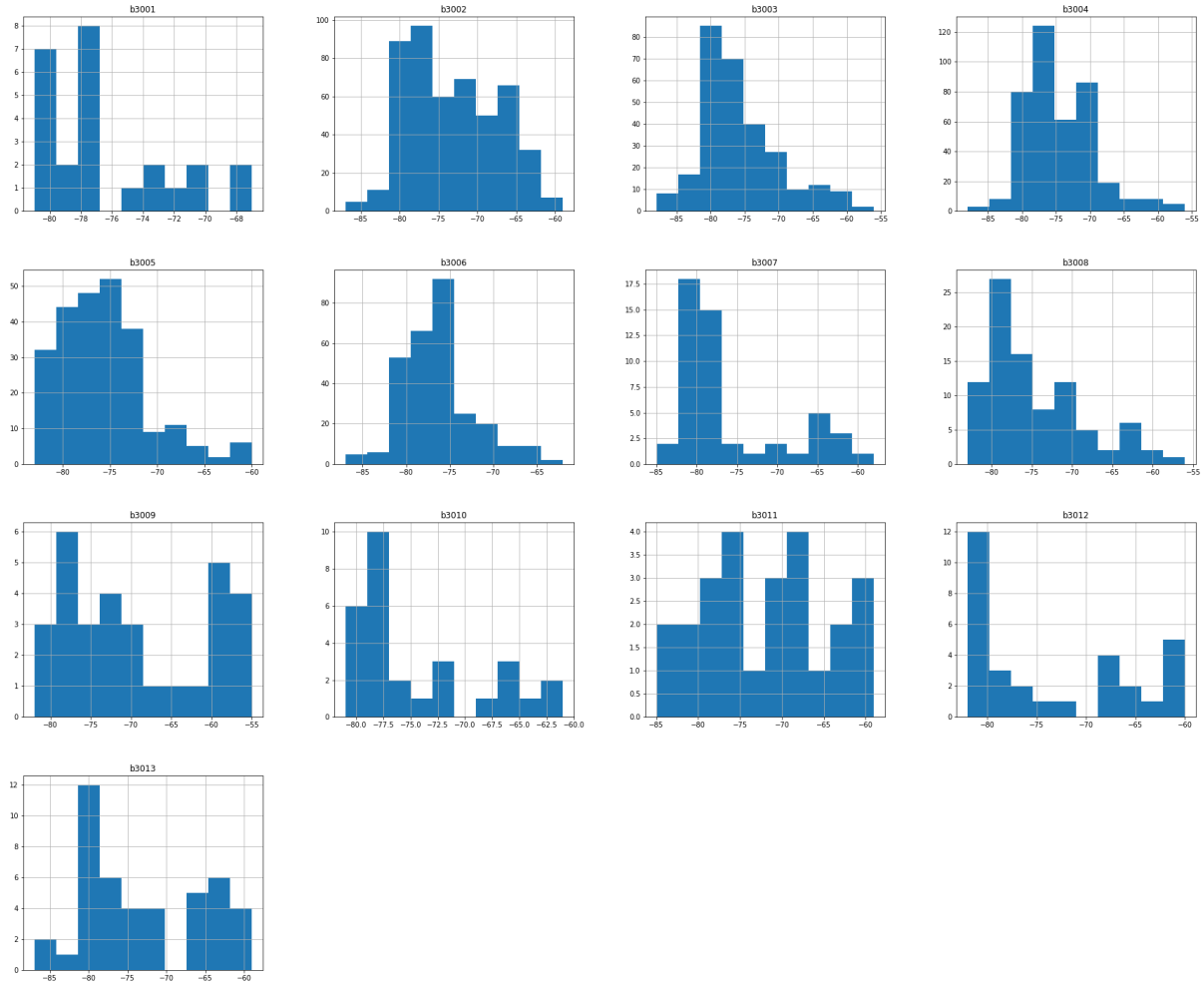

Figure 2. Histograms of iBeacons

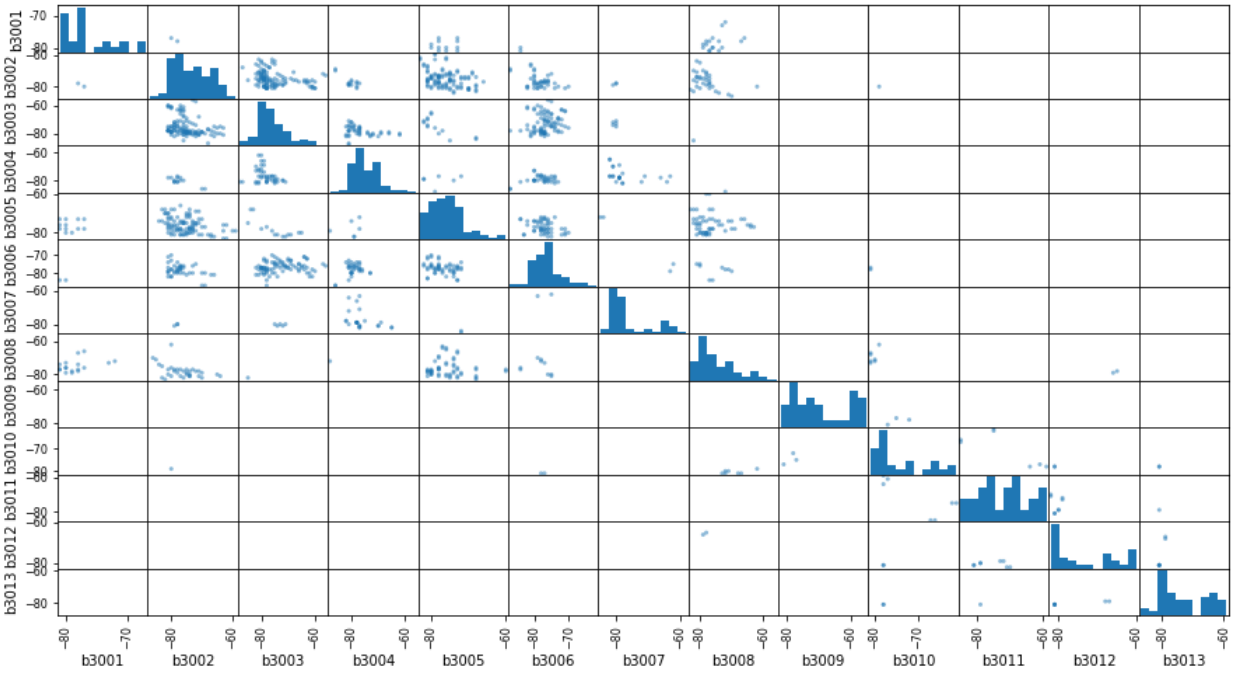

Figure 3. Scatter Matrix of iBeacons (pair relationship)

Scatter plots of pair relationship between location and iBeacon

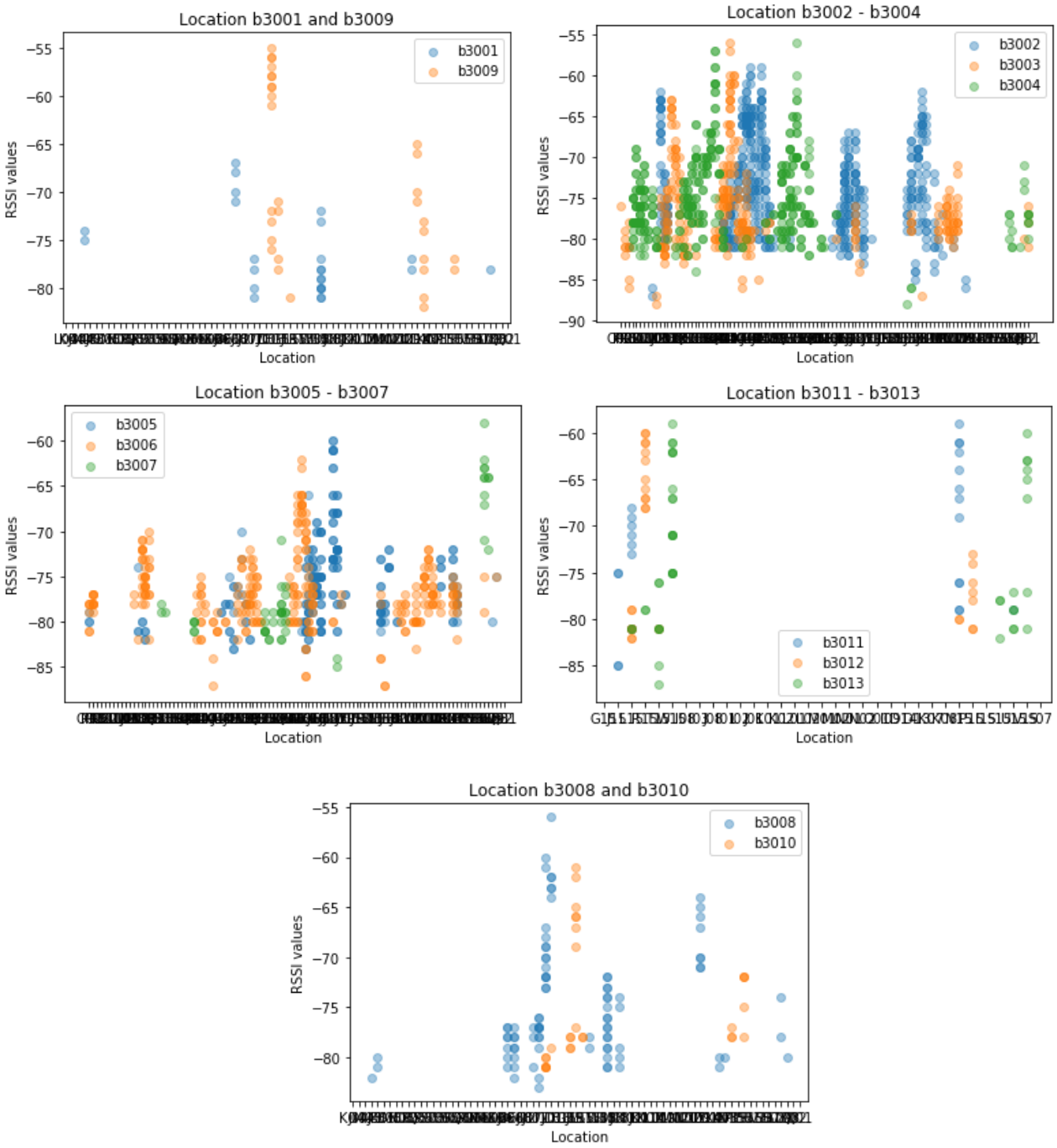

Figure 4. Scatter plot of pair relationship

Plausible Hypothesis

We are investigating the accuracy detection of the iBeacon in the location and compare the most/least visit location. For each graph shown above, b3002 to b3004 seems to be the busiest spots, followed by b3005 – b3007 as the second busiest, then for the rest of the groups are least visited.

Machine Learning technique will learn from the dataset, train the data, evaluate the data, and find the best models of algorithms to predict the unseen future data.

2.3 Data Modelling

The BLE RSSI dataset is considered as non-continuing and categorical data, the best method to implement is using Classification approach having K-Nearest Neighbors and Decision Tree as the classifiers.

KNN-Imputer

During data exploration, there have been found a large amount of -200 which indicates as missing values. In this report, we are investigating the accuracy detection of the iBeacon. Therefore, each missing value is imputed using the mean value from nearest eighbors found in the training set which is 5.

Train Test Split

The data will be trained and tested with 80% training value and 20% test value to find the best accuracy from the two models' comparison. Both models will use the same train_test_split value.

K-Nearest Neighbors (KNN) Classifier

The KNN method gives flexibility in tuning the parameters of the n_neighbors, weights, p, and the others before fitting the data. In this report, it will implement 3 different parameters set-up:

1. n_neighbors = 5, weights = 'uniform', and p = 2
2. n_neighbors = 3, weights = 'distance', and p = 1
3. n_neighbors = 1, weights = 'distance', and p = 1

Decision Tree Classifier

The Decision Tree have two types of criterions to choose before fitting the data which are Gini and Entropy. There are also other parameters that can be tuned, however, in this report, it will only compare between the criterion and the others will be left default.

3. Results

The results of both methods in Data Modelling are combined into one table as below:

K Nearest Neighbors Accuracy Result	
(n_neighbors = 5, weights = 'uniform', p = 2)	18 %
(n_neighbors = 3, weights = 'distance', p = 1)	29 %
(n_neighbors = 1, weights = 'distance', p = 1)	30 %
Decision Tree Accuracy Result	
(criterion = 'gini')	24 %
(criterion = 'entropy')	26 %

** The accuracy results are just an approximate value; it may differ time to time depending on the random training and testing set.*

Both results show the classification rate of 18% - 30% which considered as low accuracy.
This could be due to too many missing values that are not part of the iBeacons problem. The location detection is changing from one location to another and was fetched by the other iBeacons.

4. Discussion

We have performed the important steps of Data Science Process from setting the research goal which is to find the accuracy of the iBeacons detection in Waldo Library. We then retrieved and prepared the data in Jupyter notebook and checked that there are 1420 instances and 15 attributes in the data.

The information that is required in this is report is just the iBeacon devices and location, the date column is not much important, it was then being dropped during the data preparation. We then perform checking on each attribute's data type, understanding the attribute's function to the data, then check the Descriptive Statistic of each column to better understand the pattern of the data. The features are iBeacons 3001 to 3013 and the labels are the location.

During data exploration, there are missing values on the iBeacons that was indicated as -200, we replaced it with NaN value, because we are interested with the iBeacon detection. Histogram graph was implemented to compare each iBeacon performance, and Scatter Matrix was used to see the differences. We also checked on the relationship between iBeacons and the location using Scatterplot to identify which location has the most visit than the others.

In the data modelling process, we impute the empty value using KNN Imputer method. The KNN imputer calculate the mean of the nearest neighbors (5) then replacing the empty value for better training purposes.

Classification method has been applied on the data having iBeacons 3001 to 3010 as the features and location as the labels. We then used the method of train test split to split the training and test value into 80% and 20% and selected the two models of Classification which are K-Nearest Neighbors and Decision Tree to train and predict the unseen future data.

In KNN method we tuned the parameters with 3 different neighbors value from n = 5, n =3, and n = 1. Given the 3 parameters tuned on KNN method, we can see that the accuracy is improved when reducing the number of the n_neighbors. Although the n_neighbors 1 has the higher accuracy score, but in KNN algorithm, the lower the n_neighbors value, the lesser the error on the training set as well. The optimal value depends on the nature of the problem chosen.

In Decision Tree method, we tuned the parameters with 2 different criterions which is gini and entropy. The result for entropy predicted better accuracy on the unseen data compared to gini by at least 2 to 3%.

5. Conclusion

Overall, the result between the two classifiers are not much different, and we can tell that both models are performing well in identifying the best accuracy for supervised machine learning.
Due to limitation time and model used in this report, the optimal classification result is found to be 30% accuracy. The modelling could be done better by implementing other classifier, however, due to large dataset, the buffering time may take longer as well.

The iBeacon device has been helpful in navigating people, motion detection, track the most visit sites, and many more. For this dataset, the missing values are not considered as the iBeacons are used to detect the signal strength of the mobile devices such as iPhone 6S or any other similar smartphone to interact with. Majority people have smartphone with them, and Machine Learning will keep on improving the efficiency of our living.

6. References

- *RMIT Lecture Notes, Tutorials, Discussion Forums & Collaborate Ultra Recordings.*
- *Boschetti, Alberto, and Luca Massaron. Python Data Science Essentials : A Practitioner's Guide Covering Essential Data Science Principles, Tools, and Techniques, 3rd Edition, Packt Publishing, Limited, 2018.*
- *www.SciKit-Learn.org*
- *www.TowardsDataScience.com*