Wifi indoor Locationing Report

by

Qing Wu

November 30th, 2018

**Background and Business goal:**

Our client would like us to investigate the feasibility of using "Wi-Fi fingerprinting" to determine a person's location in indoor spaces. We have been provided with a large database of Wi-Fi fingerprints for a multi-building industrial campus with a location associated with each fingerprint. We would need to build up the best machine learning model and take advantage of WLAN/Wi-Fi fingerprint to help develop indoor positioning system. The client can also make technology widely used by more people and incorporate it into a smartphone app for indoor locationing.

**Description of data sources:**

This database is focused on WLAN fingerprint-based ones (also know as WiFi Fingerprinting).  The UJIIndoorLoc database covers three buildings of Universitat Jaume I with 4 or more floors and almost 110.000m2. It can be used for multiclassification, e.g. actual building and floor identification, actual longitude and latitude estimation. It was created in 2013 by means of more than 20 different users and 25 Android devices. The 529 attributes contain the WiFi fingerprint, the coordinates where it was taken, and other useful information.

Location of data sources: The data for this project is currently stored on a database consisting of 19937 training/reference records (trainingData.csv file) and 1111 validation/test records (validationData.csv file). The database is stored by the host which address is <http://archive.ics.uci.edu/ml/machine-learning-databases/00310/>

**Attribute Information:**

|  |  |
| --- | --- |
| Attribute | Description |
| Attribute 001 (WAP001)~ Attribute 520 (WAP520) | Intensity value for WAP001 to Intensity value for WAP520. Negative integer values from -104 to 0 and +100. Positive value 100 used if WAP001 was not detected. |
| Attribute 521 (Longitude) | Longitude. Negative real values from -7695.9387549299299000 to -7299.786516730871000 |
| Attribute 522 (Latitude) | Latitude. Positive real values from 4864745.7450159714 to 4865017.3646842018. |
| Attribute 523 (Floor) | Altitude in floors inside the building. Integer values from 0 to 4. |
| Attribute 524 (BuildingID) | ID to identify the building. Measures were taken in three different buildings. Categorical integer values from 0 to 2. |
| Attribute 525 (SpaceID) | Internal ID number to identify the Space (office, corridor, classroom) where the capture was taken. Categorical integer values. |
| Attribute 526 (RelativePosition) | Relative position with respect to the Space (1 - Inside, 2 - Outside in Front of the door). Categorical integer values. |
| Attribute 528 (PhoneID) | Android device identifier (see below). Categorical integer values. |
| Attribute 529(Timestamp) | UNIX Time when the capture was taken. Integer value. |

**Data Management:**

We use this indoor localization database for classification, e.g. actual building and floor identification, latitude and longitude. No missing data are detected in these two datasets. I combined the building, floor, latitude and longitude attributes into a single unique identifier for each observation and convert this unique identifier to categorical variable. I used attribute 001 (WAP001) to attribute 520 (WAP520) as predictors and unique identifier as output variables. I used 19937 observations from training Data to train and validate the model. I used 1111 test records from validation Data to evaluate the model to the unseen data. Originally, WAP001 to WAP520, Negative integer values from -104 to 0 and +100. Positive value 100 used if WAP001 was not detected. I transformed the the value (-104 to 0) of WAP to 1 to 105 (weak to strong), 0 for no signal. For Neural Network algorithms, One-hot encoding method is used to transform dependent(y) variable into dummy variable, which can be applied to softmax function for multiclass classification problem.

**Model tuning and Model selection：**

k-nearest neighbors, Random Forest, Neural Network algorithms are selected to develop model.

For k-nearest neighbors, cross-validation is applied and two important hypermeters are tuned. Hyperparameter k means the number of neighbors. The other hyperparameter is a distance metric. Similarity is defined according to a distance metric between two data points where the distance metric commonly considered is the Euclidean distance but other measures include the Manhattan, Chebyshev and Hamming distance. Distance metric is set to Manhattan distance and k is set to a list of 1,5,9. After cross validation, the best parameters are selected. The best model with highest accuracy is that K is set to 1 and the distance metric is set to Manhattan.

For Random Forest, cross-validation is applied and three important hypermeters are set. Criterion is set to gini, 'n\_estimators' that means number of trees in the forest is set to 100, and 'max\_features' that means max number of features considered for splitting a node is set to sqrt.

For Neural Network, cross-validation is not applied due to computing capacity. The training Data is split into training set and validation set. Validation set can help to evaluate the accuracy of the model. Several important hypermeters are set. 'Optimizer' is set to Adam, 'hidden\_layers' is set to 1, 'hidden\_units' is set to 1500, 'batch\_size' is set to 256, 'epochs' is set to 120, 'losses' is set to 'categorical\_crossentropy', 'activation' of hidden lay is set to relu, 'last\_activation' is set to softmax.

**Model evaluation：**

For k-nearest neighbors, Random Forest, metrics of accuracy and kappa are selected to evaluate performance of training set and Cross validation. For Neural Network, accuracy is selected to evaluate performance of training set and validation set. Mean Error,25th percentile ,50th percentile ,75th percentile ,95th percentile,100th percentile, Building Hit rate, Floor Hit rate are selected to valuate the performance of test set, which can help to determine how well the model generalize to the unseen data.

The results of performance of Training set and Validation set are shown below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Classifier | Training set | | Validation set | |
| Accuracy | Kappa | Accuracy | Kappa |
| KNN | 0.966 | 0.966 | 0.818 | 0.8178 |
| Random Forest | 0.968 | 0.968 | 0.863 | 0.863 |
| Neural Network | 0.959 |  | 0.801 |  |

As we can see, the accuracy and kappa of KNN, Random Forest, Neural Network in training set are all very high. The accuracy and kappa of KNN, Random Forest, Neural Network in training set are all quite high. Random Forest has the best cross-validation accuracy, which is 4.5% higher than KNN, and 6.2% higher than Neural Network.

The results of performance of test set of these three model are shown in the Table below.

|  |  |  |  |
| --- | --- | --- | --- |
|  | KNN | Random Forest | Neural Network |
| Mean Error | 12.394 m | 8.313 m | 9.774 m |
| 25th percentile | 1.884 m | 1.268 m | 1.708 m |
| 50th percentile | 6.016 m | 5.127 m | 5.831 m |
| 75th percentile | 12.308 m | 10.711 m | 12.507 m |
| 95th percentile | 37.314 m | 27.516 m | 32.849 m |
| 100th percentile | 369.158 m | 112.144 m | 320.809 m |
| Building Hit rate | 98.6% | 99.9% | 99.7% |
| Floor Hit rate | 88% | 90.1% | 87.7% |

As we can see, Random Forest Model provided best result with the least error and highest hit rate among these three models. Compared to the baseline, Random Forest model has higher performance since expect for 95th percentile and 100th percentile, other error metrics of Random Forest model is lower. Building Hit rate is only 0.1% lower than baseline and Floor Hit rate is 4.76% higher than baseline. Floor Hit rate of KNN and Neural Network is also higher than baseline. Even compared to the competing teams result, Random Forest has comparable quite good result.

**Baseline Result of The 2015 EvAAL-ETRI Competition**

**Baseline**

|  |  |
| --- | --- |
| Mean Error | 8.46m |
| 25th percentile | 3.39m |
| 50th percentile | 6.50m |
| 75th percentile | 11.72m |
| 95th percentile | 21.41m |
| 100th percentile | 73.30m |
| Building Hit rate | 100% |
| Floor Hit rate | 85.34% |

**The results of the competing teams of The 2015 EvAAL-ETRI Competition**

**are shown in the Table below.**

**Competition and Ensemble Results**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | HFTS | MOSAIC | RTLS@UM | ICSL | Ensemble |
| Mean Error | 8.49m | 11.64m | 6.20m | 7.67m | 6.10m |
| 25th percentile | 3.69m | 3.26m | 2.51m | 3.10m | 2.51m |
| 50th percentile | 6.99m | 6.72m | 4.57m | 5.88m | 4.56m |
| 75th percentile | 11.60m | 12.12m | 8.34m | 10.87m | 8.24m |
| 95th percentile | 19.93m | 21.54m | 15.81m | 19.68m | 15.41m |
| 100th percentile | 40.70m | 313.33m | 52.27m | 39.14m | 52.27m |
| Building Hit rate | 100% | 98.65% | 100% | 100% | 100% |
| Floor Hit rate | 96.25% | 93.86% | 93.74% | 86.93% | 96.43% |

**Conclusion：**

According to our analysis, Random Forest Model is selected as best model due to highest accuracy and kappa of cross validation among these three model and comparable good performance of test set compared to baseline of the competition. "Wi-Fi fingerprinting" method is feasible for indoor location because of high accuracy. We can help our client to incorporate this model into mobile app for indoor locationing.

**Reference:**

Joaquín Torres-Sospedra, Adriano Moreira, Stefan Knauth, Rafael Berkvens, Raul Montoliu, Oscar Belmonte, Sergio Trilles, Maria João Nicolau, Filipe Meneses, António Costa, Athanasios Koukofikis, Maarten Weyn and Herbert Peremans. A Realistic Evaluation of Indoor Positioning Systems Based on Wi-Fi Fingerprinting: The 2015 EvAAL-ETRI Competition. Article in Journal of Am- bient Intelligence and Smart Environments Vol.9(2):263-279, February 2017.