BLE Environment Detection

Team 4 (Himani, Darpan & Harsha)

Motivation

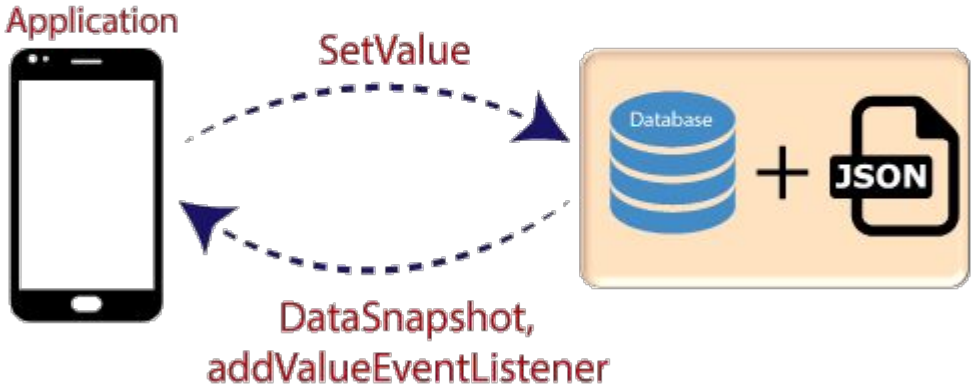
- Some applications need to be configured based on location
- Use cases:
  - Corona contact tracing
  - Cell Phone volume settings

Background - BLE

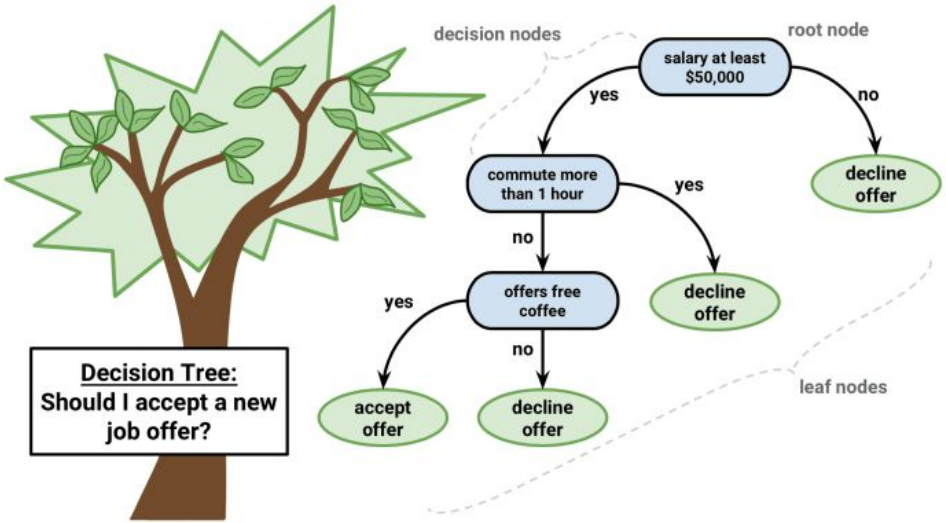
- Android built-in support for Bluetooth Low Energy (BLE) in the central role.
- Doesn’t need to connect or pair.

Background - Firebase

- The Firebase Realtime Database is a cloud-hosted database.
- Data is stored as JSON
- Synchronized in realtime to every connected client
- Data is persisted locally, and even while offline.



Background - Decision Tree




```
['advertise_flag', 'transmission_power', 'rssi', 'event_type']
|--- feature_0 <= 25.00
|    |--- feature_2 <= -69.50
|    |    |--- feature_2 <= -76.50
|    |    |    |--- feature_2 <= -92.50
|    |    |    |    |--- feature_2 <= -95.50
|    |    |    |    |    |--- class: indoors
|    |    |    |    |--- feature_2 >  -95.50
|    |    |    |    |    |--- class: outdoors
|    |    |    |--- feature_2 >  -92.50
|    |    |    |    |--- feature_3 <= 21.50
|    |    |    |    |    |--- class: outdoors
|    |    |    |    |--- feature_3 >  21.50
|    |    |    |    |    |--- class: public_transport
|    |    |--- feature_2 >  -76.50
|    |    |    |--- feature_0 <= 4.50
|    |    |    |    |--- feature_1 <= 2.00
|    |    |    |    |    |--- class: indoors
|    |    |    |    |--- feature_1 >  2.00
|    |    |    |    |    |--- class: indoors
|    |    |    |--- feature_0 >  4.50
|    |    |    |    |--- feature_0 <= 15.00
|    |    |    |    |    |--- class: outdoors
|    |    |    |    |--- feature_0 >  15.00
|    |    |    |    |    |--- class: outdoors
|    |--- feature_2 >  -69.50
|    |    |--- feature_3 <= 21.50
|    |    |    |--- feature_2 <= -59.50
|    |    |    |    |--- feature_2 <= -63.50
|    |    |    |    |    |--- class: indoors
|    |    |    |    |--- feature_2 >  -63.50
|    |    |    |    |    |--- class: indoors
|    |    |    |--- feature_2 >  -59.50
|    |    |    |    |--- feature_2 <= -55.50
|    |    |    |    |    |--- class: indoors
```

Goals

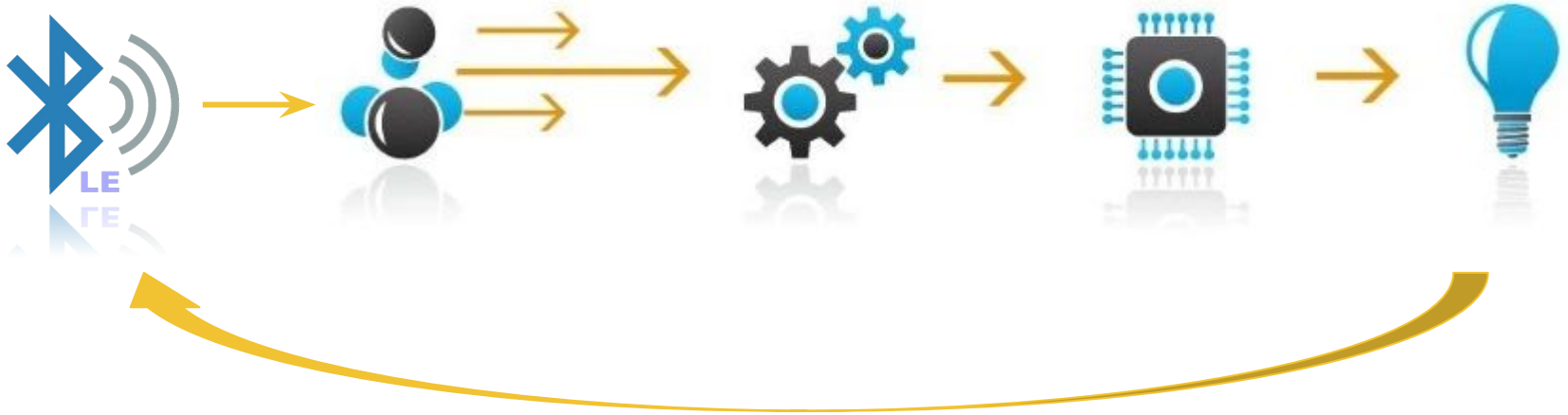
- Development of a BLE Beacon Tracer
- Build a classifier determining a location type.
- Evaluation of the classifier in different environments

Results

- BLE beacon scanner built on Android
- A Decision tree is built to identify location
- Classifier identifies the location with 92% accuracy

Approach

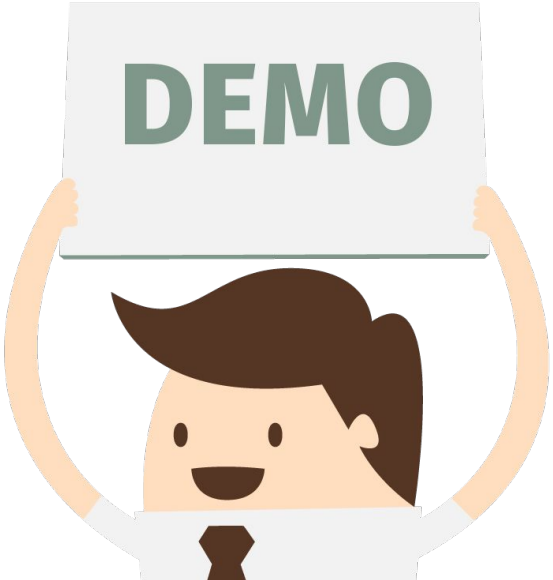
Scan for beacons | Prepare data | Engineer features | Build model | Draw insights



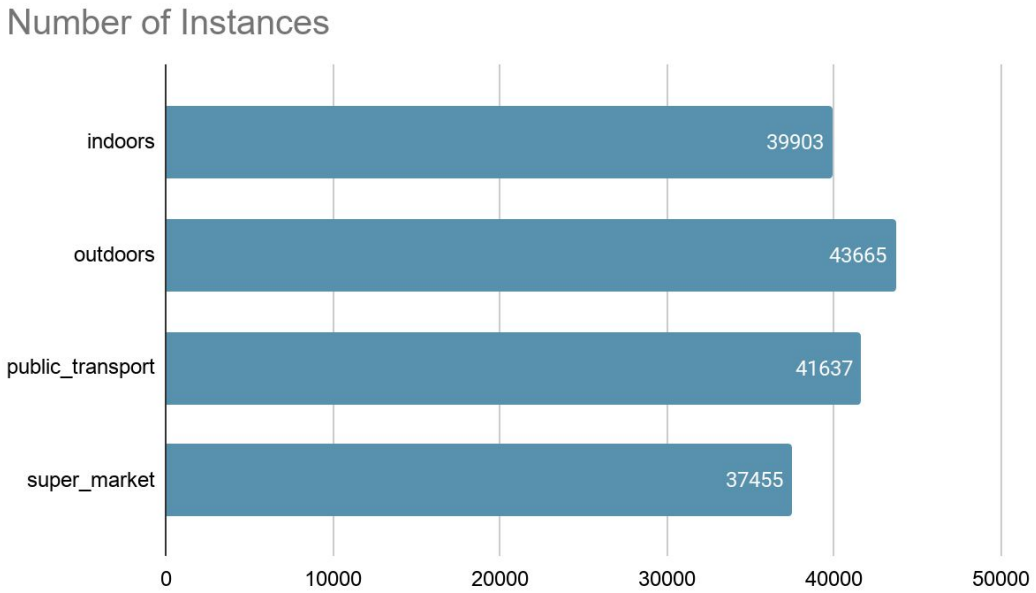
Demo

- Application
- Classifier

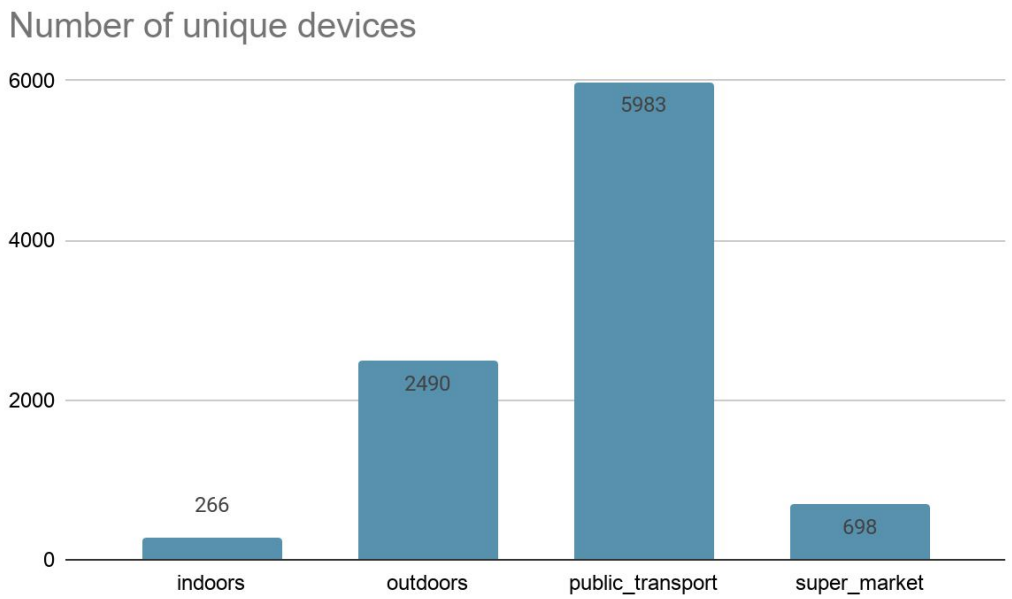

```
{
    "ashokroom" :
    {
      "1191941605105945@75:EE:7A:B9:12:CF" :
      {
          "device" : "75:EE:7A:B9:12:CF",
          "scanRecord" :
          {
              "mAdvertiseFlags" : -1,
              "mServiceUuids" : null,
              "mServiceSolicitationUuids" : [],
              "mManufacturerSpecificData" : [1, 9, 32, 2,
              "mServiceData" : [],
              "mTxPowerLevel" : -2147483648,
              "mDeviceName" : null,
              "mTransportBlocks" : [],
              "rssi" : -88,
              "timestampNanos" : 1191941605105945,
              "eventType" : 16,
              "primaryPhy" : 1,
              "secondaryPhy" : 0,
              "advertisingSid" : 255,
              "txPower" : 127,
              "periodicAdvertisingInterval" : 0
          }
      },
```

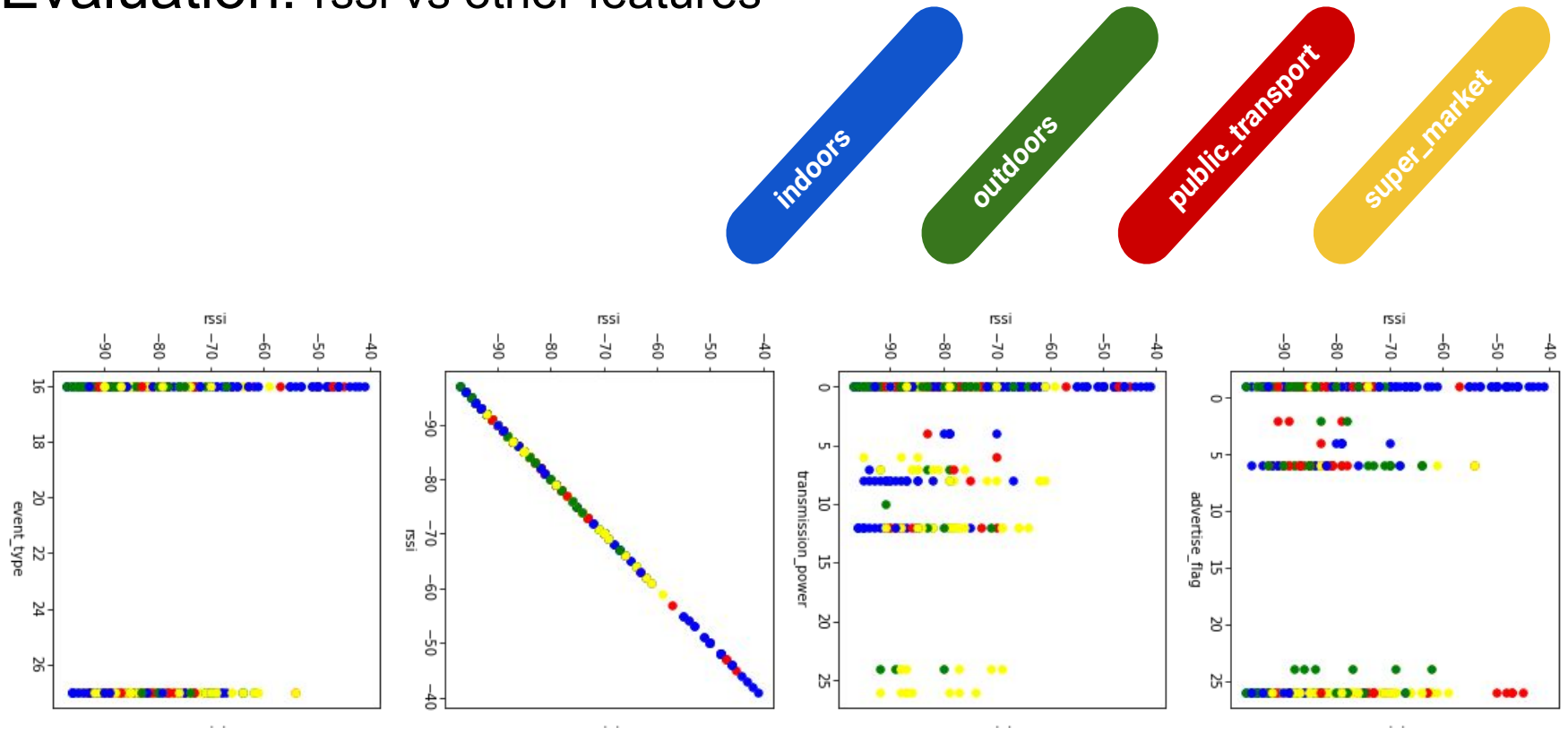
Evaluation: no. of instances

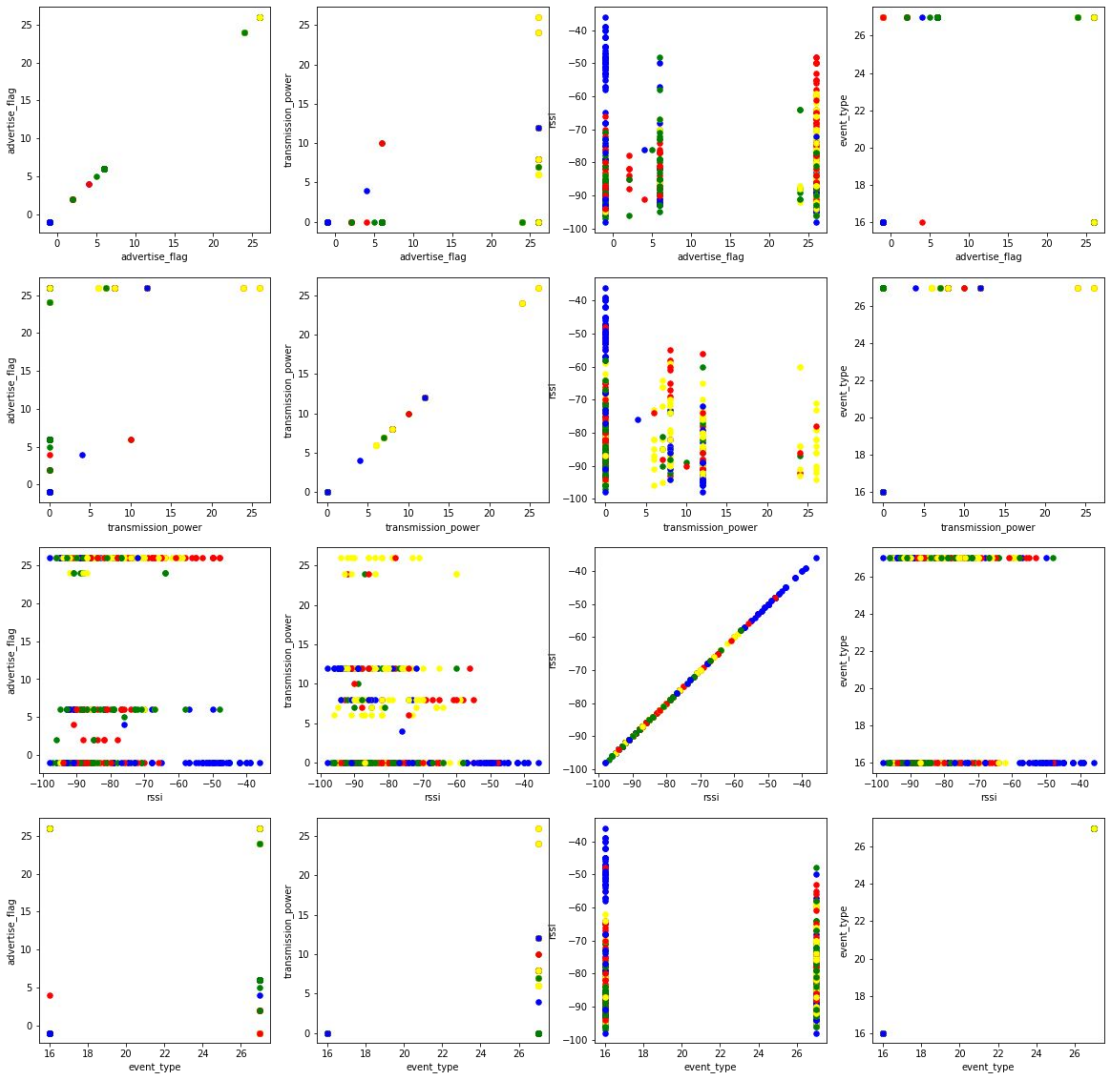


Evaluation: no. of unique devices



Evaluation: rssi vs other features





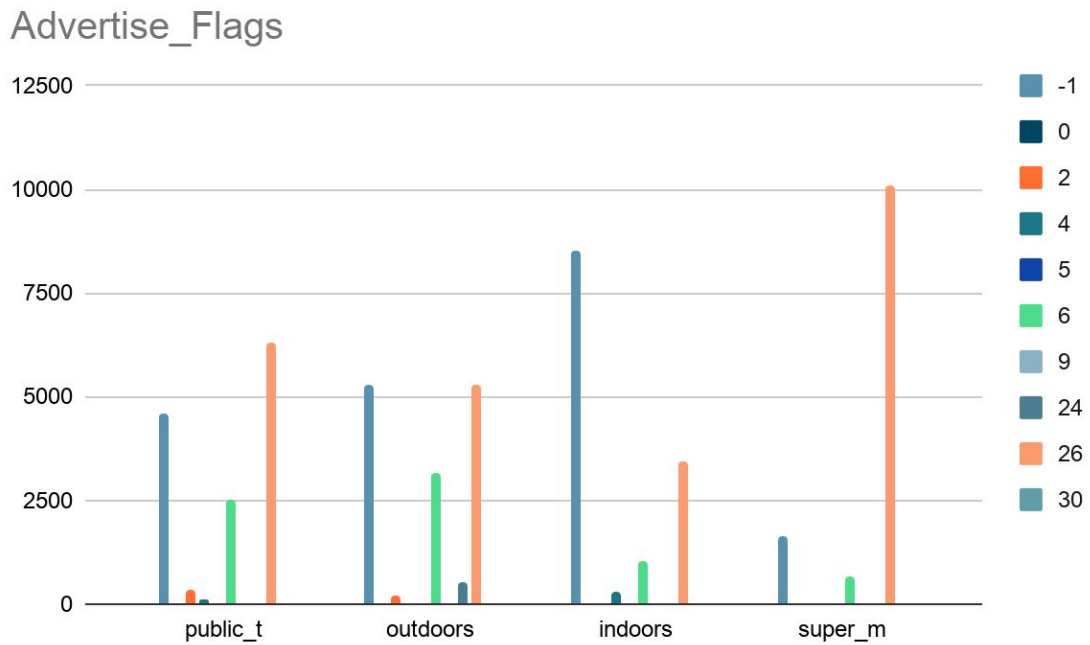
public_transport

super_market

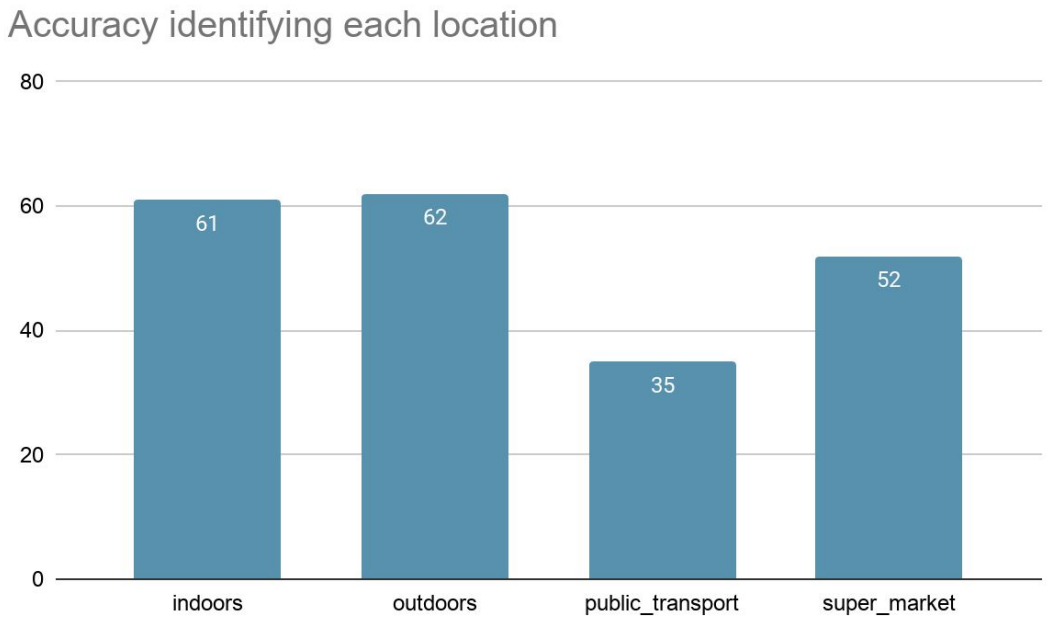
indoors

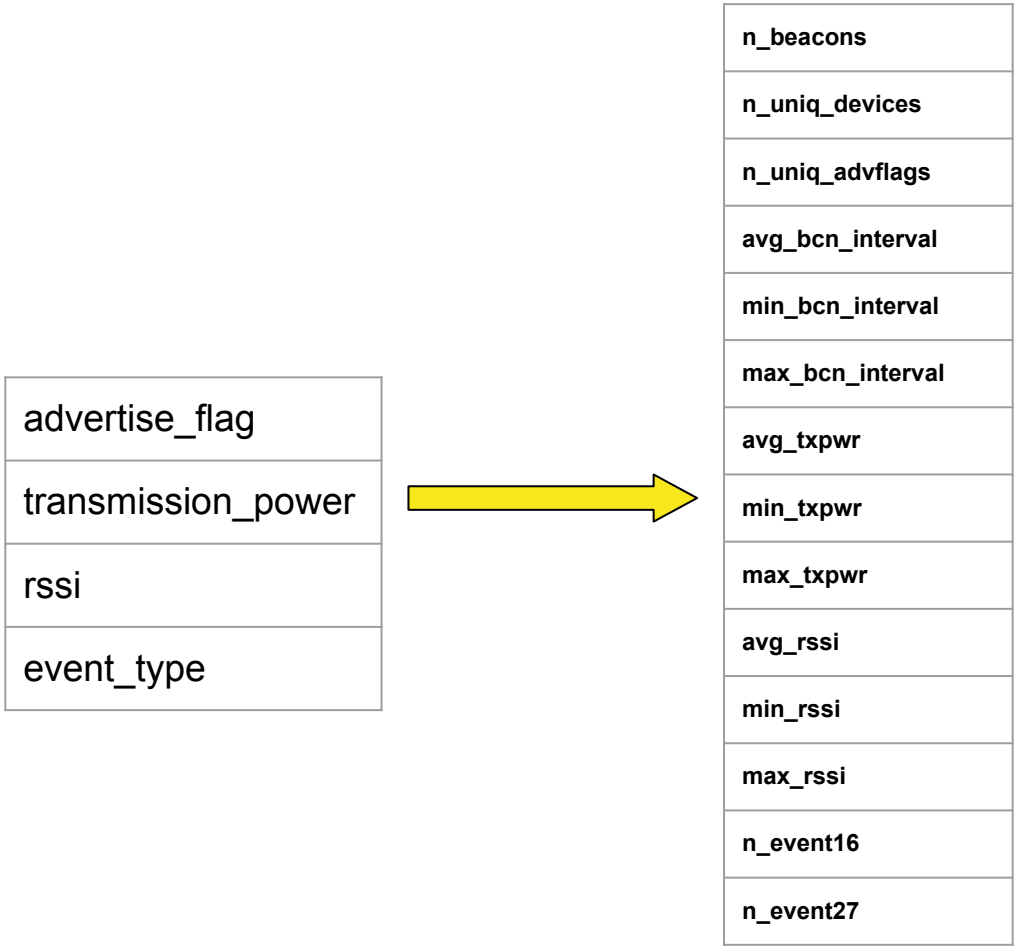
outdoors

Evaluation: advertise-flags in locations



Evaluation: accuracy identifying locations (in percentage)




advertise_flag
transmission_power
rssi
event_type
n_beacons
n_uniq_devices
n_uniq_advflags
avg_bcn_interval
min_bcn_interval
max_bcn_interval
avg_txpwr
min_txpwr
max_txpwr
avg_rssi
min_rssi
max_rssi
n_event16
n_event27

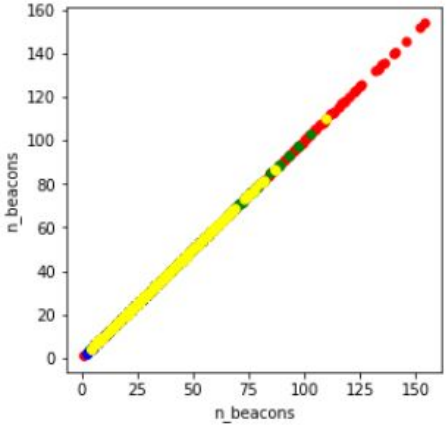
Evaluation: n_beacons vs others

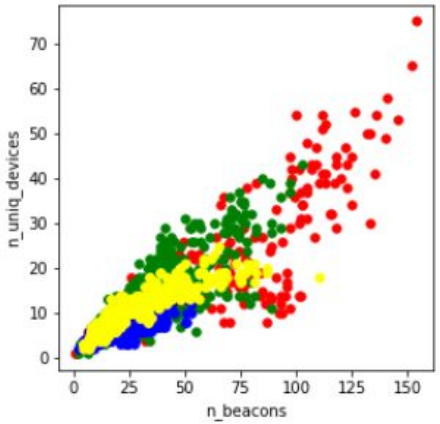
indoors

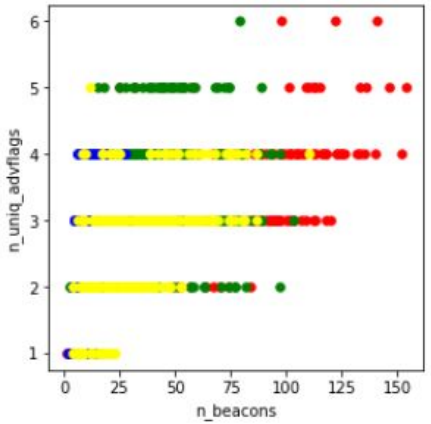
outdoors

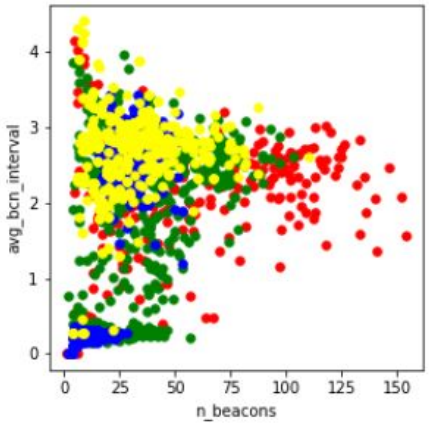
public_transport

super_market







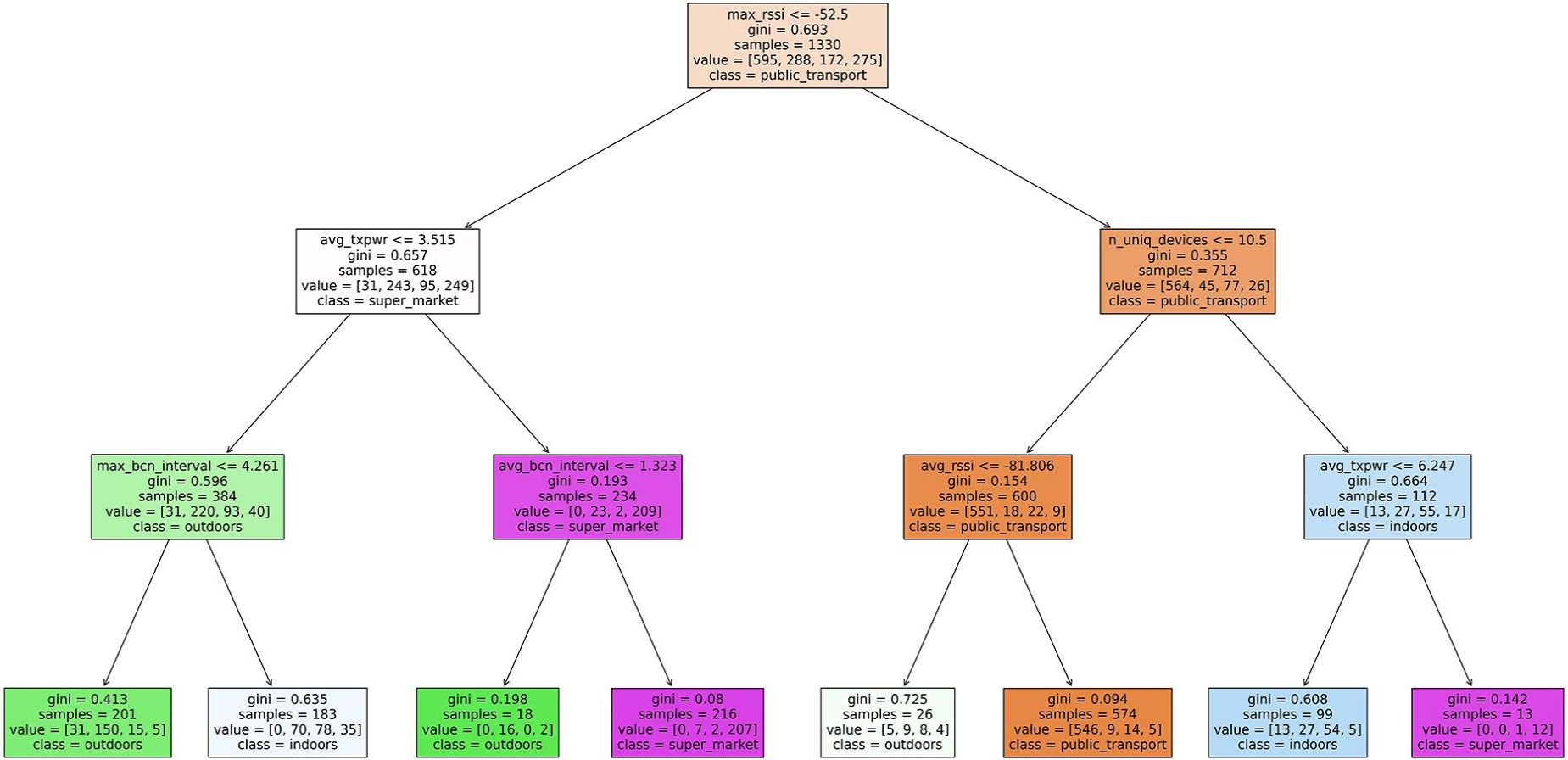


Evaluation: accuracy

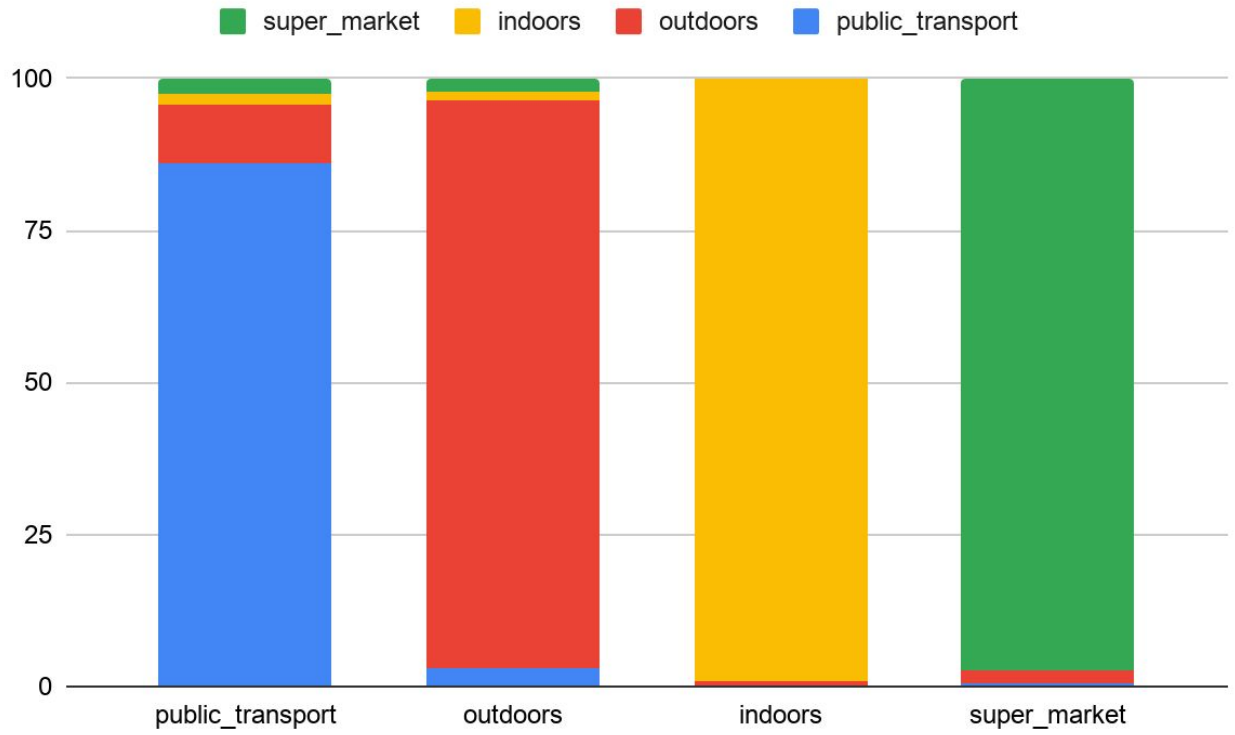
- Nearest Neighbors : 82%
- Linear SVM : 85%
- **Random Forest : 92%**
- **Decision Tree : 91%**

Evaluation: important features

- Top 3 Important features:
  - can identify a location 8 out of 10 times
  - max_rssi
  - avg_txpwr
  - n_unique_devices

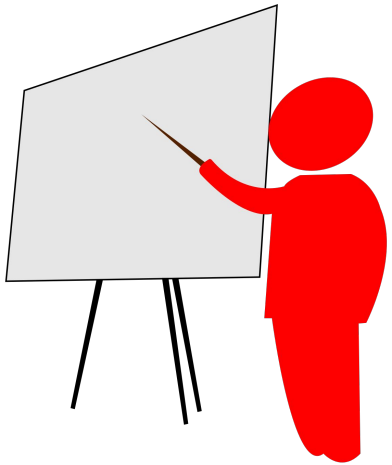

max_rssi <= -52.5
gini = 0.693
samples = 1330
value = [595, 288, 172, 275]
class = public_transport
avg_txpwr <= 3.515
gini = 0.657
samples = 618
value = [31, 243, 95, 249]
class = super_market
n_uniq_devices <= 10.5
gini = 0.355
samples = 712
value = [564, 45, 77, 26]
class = public_transport
max_bcn_interval <= 4.261
gini = 0.596
samples = 384
value = [31, 220, 93, 40]
class = outdoors
avg_bcn_interval <= 1.323
gini = 0.193
samples = 234
value = [0, 23, 2, 209]
class = super_market
avg_rssi <= -81.806
gini = 0.154
samples = 600
value = [551, 18, 22, 9]
class = public_transport
avg_txpwr <= 6.247
gini = 0.664
samples = 112
value = [13, 27, 55, 17]
class = indoors
gini = 0.413
samples = 201
value = [31, 150, 15, 5]
class = outdoors
gini = 0.635
samples = 183
value = [0, 70, 78, 35]
class = indoors
gini = 0.198
samples = 18
value = [0, 16, 0, 2]
class = outdoors
gini = 0.08
samples = 216
value = [0, 7, 2, 207]
class = super_market
gini = 0.725
samples = 26
value = [5, 9, 8, 4]
class = outdoors
gini = 0.094
samples = 574
value = [546, 9, 14, 5]
class = public_transport
gini = 0.608
samples = 99
value = [13, 27, 54, 5]
class = indoors
gini = 0.142
samples = 13
value = [0, 0, 1, 12]
class = super_market

Evaluation: performance in each location



Lessons learned

- Different locations have significantly different BLE environments
- RSSI of a signal provides considerable amount of discriminability
- BLE data can be used to identify the location type if it’s not mission critical.



Summary

- We can identify the location type using BLE data 9 out of 10 times
- Any queries?
- Thank you