Neural Network Training Algorithm for Improved Wi-Fi Fingerprint Indoor Positioning

PROJECT PRESENTATION

Nolan McQueen
Jaeyong Kwack

Overview

- Developed a neural network that is capable of improving upon the typical approach to Wi-Fi Fingerprint positioning
- Fingerprint through the generation of a Wi-Fi signal strength database of Riley Hall
- Research has shown Wi-Fi positioning can be statistically improved with the introduction of neural nets improving on the nonlinear, highly complex Wi-Fi signal propagation patterns
- Created Android application for collecting large amounts of data
- Trained and formed a neural net using the Neuromorph IDE

Overview

Data Collection

Neural Net

Input

Access Point 1

Signal Strength | Latitude | Longitude

Access Point 2

Signal Strength | Latitude | Longitude

Access Point 3

Signal Strength | Latitude | Longitude

Access Point 4

Signal Strength | Latitude | Longitude

Output

Test Location

1	2	3	4	5
0 to 1	0 to 1	0 to 1	0 to 1	0 to 1

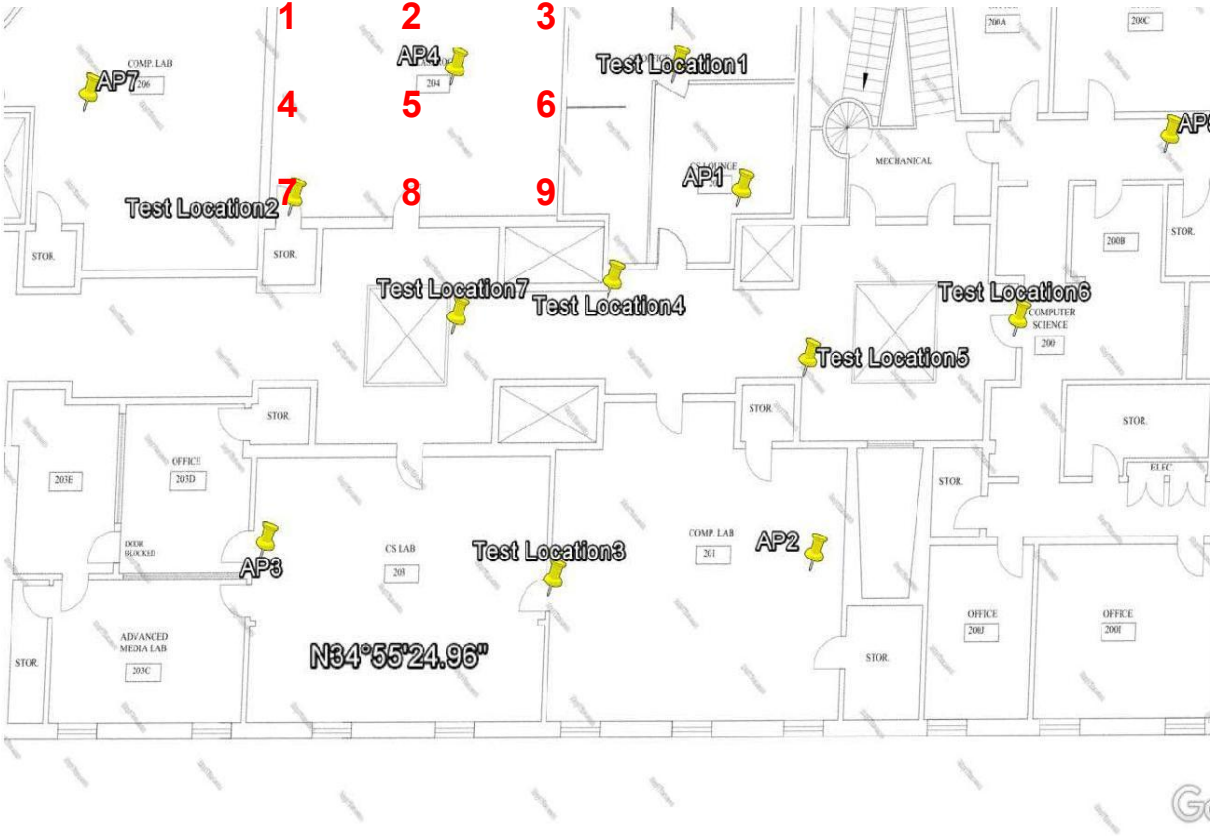
6	7	8	9
0 to 1	0 to 1	0 to 1	0 to 1

Data Collection -Georeference

- Overlay Riley Hall Floor Plan with Google Map

Data Collection -Georeference

- Pinpoint AP Locations



Data Collection
-WiFi Fingerprint

1. Triggered when WiFi Is Scanned
2. Create JSON Object
3. Sort BSSID by Signal Strength
4. Put Them in Order
5. Send JSON Object to Server
6. Repeat 50 Times

Android

```
mBroadcastReceiver = (BroadcastReceiver) (c, intent) → {
    if (buttonClicked) {
        JSONObject dataPackage = new JSONObject();
        try {
            dataPackage.put("Location", mLocationNumberEdit.getText());
        } catch (JSONException e) {
            e.printStackTrace();
        }
        mWifiList = mWifiManager.getScanResults();
        if (!mWifiList.isEmpty()) {
            Map<String, Integer> sortedMap = sortByComparator(mergeBSSID(mWifiList));
            if (sortedMap.size() < 4) {
                mWifiManager.startScan();
            }
            Set<Map.Entry<String, Integer>> entrySet = sortedMap.entrySet();
            int count = 0;
            for (Map.Entry entry : entrySet) {
                if (count < 4) {
                    try {
                        if (count == 0) {
                            dataPackage.put("BSSID1", entry.getKey());
                            dataPackage.put("SS1", entry.getValue());
                            count++;
                        } else if (count == 1) {
                            dataPackage.put("BSSID2", entry.getKey());
                            dataPackage.put("SS2", entry.getValue());
                            count++;
                        } else if (count == 2) {
                            dataPackage.put("BSSID3", entry.getKey());
                            dataPackage.put("SS3", entry.getValue());
                            count++;
                        } else if (count == 3){
                            dataPackage.put("BSSID4", entry.getKey());
                            dataPackage.put("SS4", entry.getValue());
                            count++;
                        }
                    } catch (JSONException e) {
                        e.printStackTrace();
                    }
                } else {
                    break;
                }
            }
        }
        Toast.makeText(RecordActivity.this, dataPackage.toString(), Toast.LENGTH_SHORT).show();
        AsyncT asyncT = new AsyncT(dataPackage);
        asyncT.execute();

        mRecordButton.setEnabled(true);
        buttonClicked = false;
        if (totalRecord < 50) {
            mRecordButton.callOnClick();
            totalRecord++;
```

Data Collection
-Data Insertion

1. Receive JSON Object and Assign Variables with Values from JSON
2. Assign Latitude and Longitude according to Each Test Location
3. Insert Data into Database

1

```php
$get = json_decode($_POST['req'], true);

$bssid1 = $get['BSSID1'];
$ss1 = $get['SS1'];
$lat1;
$long1;

$bssid2 = $get['BSSID2'];
$ss2 = $get['SS2'];
$lat2;
$long2;

$bssid3 = $get['BSSID3'];
$ss3 = $get['SS3'];
$lat3;
$long3;

$bssid4 = $get['BSSID4'];
$ss4 = $get['SS4'];
$lat4;
$long4;
```

2

```php
if ($bssid1 == "20:37:06:f6:4e"){ //AP1
  $lat1 = 34.552549;
  $long1 = 82.261959;
}
else if ($bssid1 == "d0:c2:82:67:fa"){ //AP2
  $lat1 = 34.552532;
  $long1 = 82.261922;
}
else if ($bssid1 == "d0:c2:82:2d:90"){ //AP3
  $lat1 = 34.552492;
  $long1 = 82.261969;
}
else if ($bssid1 == "d0:c2:82:7e:03"){ //AP4
  $lat1 = 34.552535;
  $long1 = 82.261993;
}
else if ($bssid1 == "d0:c2:82:67:2a"){ //AP5
  $lat1 = 34.552585;
  $long1 = 82.261928;
}
else if ($bssid1 == "d0:c2:82:2d:fc"){ //AP6
  $lat1 = 34.552572;
  $long1 = 82.261899;
}
else if ($bssid1 == "d0:c2:82:2d:b2"){ //AP7
  $lat1 = 34.552506;
  $long1 = 82.262022;
}
```

3

```php
$checkDB = "SELECT BSSID1, SS1, Lat1, Long1, BSSID2, SS2, Lat2, Long2, BSSID3, SS3, Lat3, Long3, BSSID4, SS4, Lat4, Long4, Location, LocLat, LocL
  FROM WiFi_Fingerprint_Training_Set WHERE Location = '$location'";
$result = mysqli_query($conn, $checkDB);

if (!$result) {
    die('Query failed to execute for some reason');
}
$sql = "INSERT INTO WiFi_Fingerprint_Training_Set (Location, BSSID1, SS1, Lat1, Long1, BSSID2, SS2, Lat2, Long2, BSSID3, SS3, Lat3, Long3, BSSID4, SS4, Lat4, Long4, LocLat, LocLong)
      VALUES ('$location','$bssid1', '$ss1', '$lat1', '$long1', '$bssid2', '$ss2', '$lat2', '$long2', '$bssid3', '$ss3', '$lat3', '$long3', '$bssid4', '$ss4', '$lat4', '$long4', '$locLat', '$locLong');";
if ($conn->multi_query($sql) === TRUE) {
  echo "successful\n";
}
else {
  echo "Error: " . $sql . "<br>" . $conn->error;
}
```

PHP

Data Collection
-Database

- About 450 Data
- 50 per Each Location

BSSID1	SS1	Lat1	Long1	BSSID2	SS2	Lat2	Long2	BSSID3	SS3	Lat3	Long3	BSSID4	SS4	Lat4	Long4	Location
d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-74	34.552532	82.261922	20:37:06:f6:4e	-77	34.552549	82.261959	6
d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:2d:fc	-65	34.552572	82.261899	d0:c2:82:67:fa	-69	34.552532	82.261922	20:37:06:f6:4e	-77	34.552549	82.261959	6
d0:c2:82:67:2a	-63	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-75	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-73	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:2d:fc	-65	34.552572	82.261899	d0:c2:82:67:fa	-74	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:2d:fc	-64	34.552572	82.261899	d0:c2:82:67:2a	-66	34.552585	82.261928	d0:c2:82:67:fa	-75	34.552532	82.261922	20:37:06:f6:4e	-76	34.552549	82.261959	6
d0:c2:82:2d:fc	-64	34.552572	82.261899	d0:c2:82:67:2a	-66	34.552585	82.261928	d0:c2:82:67:fa	-68	34.552532	82.261922	20:37:06:f6:4e	-76	34.552549	82.261959	6
d0:c2:82:2d:fc	-63	34.552572	82.261899	d0:c2:82:67:2a	-66	34.552585	82.261928	d0:c2:82:67:fa	-72	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:2d:fc	-61	34.552572	82.261899	d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:67:fa	-75	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:2d:fc	-62	34.552572	82.261899	d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:67:fa	-74	34.552532	82.261922	20:37:06:f6:4e	-76	34.552549	82.261959	6
d0:c2:82:2d:fc	-63	34.552572	82.261899	d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:67:fa	-73	34.552532	82.261922	20:37:06:f6:4e	-75	34.552549	82.261959	6
d0:c2:82:2d:fc	-62	34.552572	82.261899	d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:67:fa	-70	34.552532	82.261922	20:37:06:f6:4e	-75	34.552549	82.261959	6
d0:c2:82:2d:fc	-64	34.552572	82.261899	d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:67:fa	-71	34.552532	82.261922	20:37:06:f6:4e	-76	34.552549	82.261959	6
d0:c2:82:2d:fc	-63	34.552572	82.261899	d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:67:fa	-71	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:2d:fc	-64	34.552572	82.261899	d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:67:fa	-70	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
20:37:06:f6:4e	-56	34.552549	82.261959	d0:c2:82:67:2a	-66	34.552585	82.261928	d0:c2:82:2d:fc	-67	34.552572	82.261899	d0:c2:82:67:fa	-70	34.552532	82.261922	6
d0:c2:82:67:2a	-63	34.552585	82.261928	d0:c2:82:2d:fc	-64	34.552572	82.261899	[illegible]	[illegible]	[illegible]	82.261922	20:37:06:f6:4e	-77	34.552549	82.261959	6
d0:c2:82:67:2a	-63	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	[illegible]	[illegible]	[illegible]	82.261922	20:37:06:f6:4e	-77	34.552549	82.261959	6
d0:c2:82:67:2a	-63	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-74	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-74	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-71	34.552532	82.261922	20:37:06:f6:4e	-79	34.552549	82.261959	6
d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-73	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:2d:fc	-65	34.552572	82.261899	d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:67:fa	-70	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:2d:fc	-65	34.552572	82.261899	d0:c2:82:67:2a	-66	34.552585	82.261928	d0:c2:82:67:fa	-72	34.552532	82.261922	20:37:06:f6:4e	-77	34.552549	82.261959	6
d0:c2:82:2d:fc	-65	34.552572	82.261899	d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:67:fa	-73	34.552532	82.261922	20:37:06:f6:4e	-77	34.552549	82.261959	6
d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-71	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:67:2a	-63	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-73	34.552532	82.261922	20:37:06:f6:4e	-79	34.552549	82.261959	6
d0:c2:82:67:2a	-62	34.552585	82.261928	d0:c2:82:2d:fc	-67	34.552572	82.261899	d0:c2:82:67:fa	-74	34.552532	82.261922	20:37:06:f6:4e	-77	34.552549	82.261959	6
d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-73	34.552532	82.261922	20:37:06:f6:4e	-78	34.552549	82.261959	6
d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-74	34.552532	82.261922	20:37:06:f6:4e	-76	34.552549	82.261959	6
d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:2d:fc	-65	34.552572	82.261899	d0:c2:82:67:fa	-73	34.552532	82.261922	20:37:06:f6:4e	-75	34.552549	82.261959	6
d0:c2:82:67:2a	-64	34.552585	82.261928	d0:c2:82:2d:fc	-65	34.552572	82.261899	d0:c2:82:67:fa	-72	34.552532	82.261922	20:37:06:f6:4e	-75	34.552549	82.261959	6
d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-73	34.552532	82.261922	20:37:06:f6:4e	-75	34.552549	82.261959	6
d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-72	34.552532	82.261922	20:37:06:f6:4e	-76	34.552549	82.261959	6
d0:c2:82:67:2a	-65	34.552585	82.261928	d0:c2:82:2d:fc	-66	34.552572	82.261899	d0:c2:82:67:fa	-71	34.552532	82.261922	20:37:06:f6:4e	-76	34.552549	82.261959	6

MySQL

Data Collection

-Data Processing

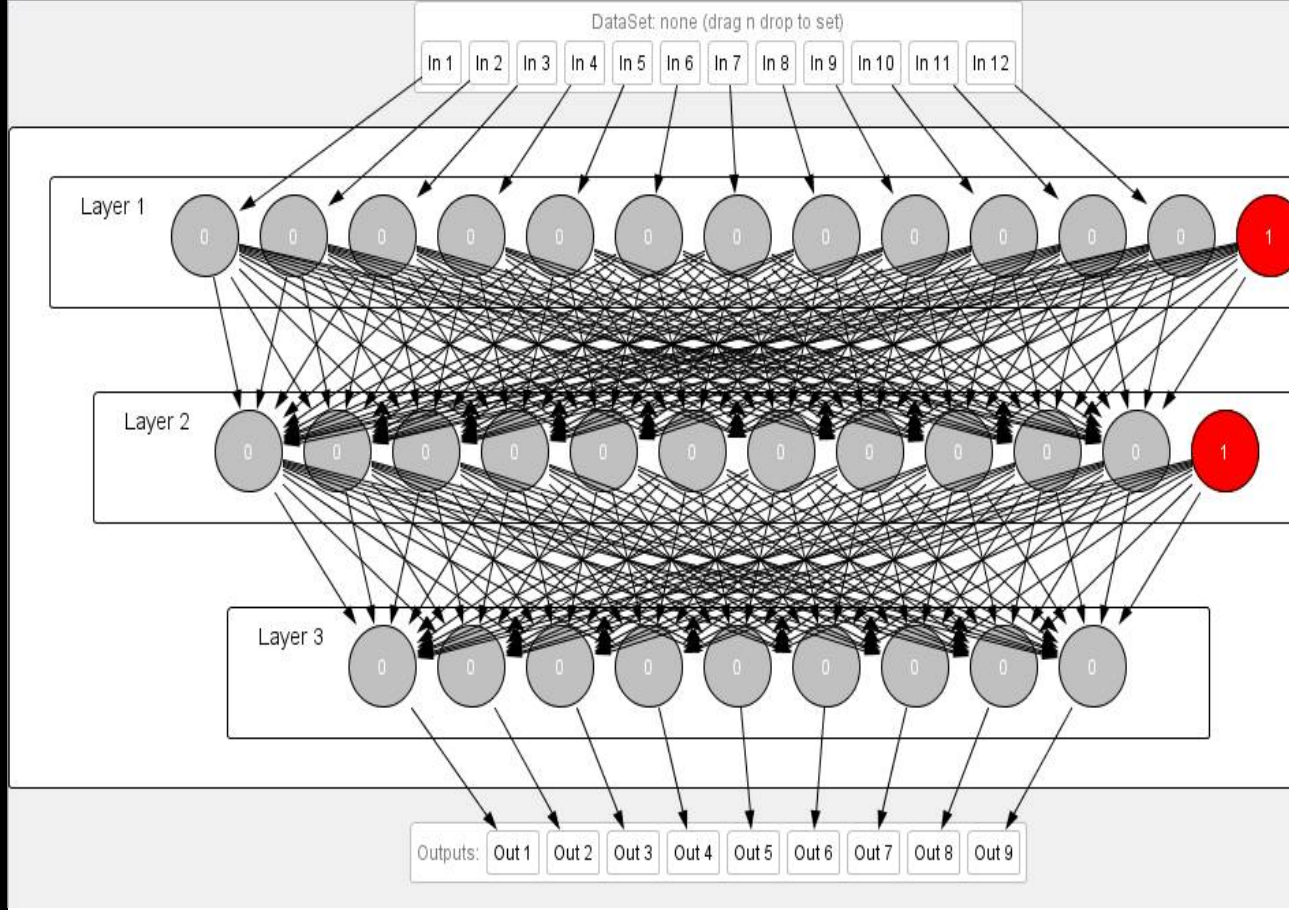
- Export as CSV
- Delete BSSID
- Format Output
- Normalize data between 0 and 1

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	0.66666667	0.46235435	0.76421693	0.61666667	0.15054388	1	0.41666667	0.61289823	0.48780988	0.3	0	0.5691164	1	0	0	
2	0.63333333	0.15054388	1	0.58333333	0.46235435	0.76421693	0.4	0.61289823	0.48780988	0.11666667	0	0.5691164	1	0	0	
3	0.63333333	0.15054388	1	0.6	0.46235435	0.76421693	0.45	0.61289823	0.48780988	0.08333333	0	0.5691164	1	0	0	
4	0.65	0.15054388	1	0.61666667	0.46235435	0.76421693	0.45	0.61289823	0.48780988	0.1	0	0.5691164	1	0	0	
5	0.65	0.15054388	1	0.6	0.46235435	0.76421693	0.4	0.61289823	0.48780988	0.13333333	0	0.5691164	1	0	0	
6	0.66666667	0.15054388	1	0.61666667	0.46235435	0.76421693	0.41666667	0.61289823	0.48780988	0.11666667	0	0.5691164	1	0	0	
7	0.66666667	0.15054388	1	0.63333333	0.46235435	0.76421693	0.43333333	0.61289823	0.48780988	0.11666667	0	0.5691164	1	0	0	
8	0.63333333	0.46235435	0.76421693	0.5	0.15054388	1	0.45	0.61289823	0.48780988	0.1	1	0.23578307	0	1	0	
9	0.63333333	0.46235435	0.76421693	0.48333333	0.15054388	1	0.4	0.61289823	0.48780988	0.15	0	0.5691164	0	1	0	
10	0.63333333	0.46235435	0.76421693	0.48333333	0.15054388	1	0.36666667	0.61289823	0.48780988	0.08333333	0.43010878	0.1869933	0	1	0	
11	0.66666667	0.46235435	0.76421693	0.5	0.15054388	1	0.38333333	0.61289823	0.48780988	0.1	0.43010878	0.1869933	0	1	0	
12	0.61666667	0.46235435	0.76421693	0.5	0.15054388	1	0.4	0.61289823	0.48780988	0.08333333	1	0.23578307	0	1	0	
13	0.63333333	0.46235435	0.76421693	0.48333333	0.15054388	1	0.38333333	0.61289823	0.48780988	0.08333333	1	0.23578307	0	1	0	
14	0.65	0.46235435	0.76421693	0.46666667	0.15054388	1	0.4	0.61289823	0.48780988	0.11666667	0	0.5691164	0	1	0	
15	0.63333333	0.46235435	0.76421693	0.45	0.15054388	1	0.38333333	0.61289823	0.48780988	0.11666667	0	0.5691164	0	1	0	
16	0.65	0.46235435	0.76421693	0.45	0.15054388	1	0.38333333	0.61289823	0.48780988	0.1	1	0.23578307	0	1	0	
17	0.66666667	0.46235435	0.76421693	0.46666667	0.15054388	1	0.4	0.61289823	0.48780988	0.1	0	0.5691164	0	1	0	
18	0.55	0.46235435	0.76421693	0.46666667	0.15054388	1	0.43333333	0.61289823	0.48780988	0.11666667	0	0.5691164	0	0	1	
19	0.6	0.46235435	0.76421693	0.48333333	0.15054388	1	0.45	0.61289823	0.48780988	0.13333333	0	0.5691164	0	0	1	
20	0.6	0.46235435	0.76421693	0.5	0.15054388	1	0.41666667	0.61289823	0.48780988	0.15	0	0.5691164	0	0	1	
21	0.6	0.46235435	0.76421693	0.5	0.15054388	1	0.41666667	0.61289823	0.48780988	0.1	1	0.23578307	0	0	1	
22	0.58333333	0.46235435	0.76421693	0.46666667	0.15054388	1	0.43333333	0.61289823	0.48780988	0.1	1	0.23578307	0	0	1	
23	0.58333333	0.46235435	0.76421693	0.46666667	0.15054388	1	0.45	0.61289823	0.48780988	0.1	1	0.23578307	0	0	1	
24	0.6	0.46235435	0.76421693	0.46666667	0.15054388	1	0.45	0.61289823	0.48780988	0.1	1	0.23578307	0	0	1	
25	0.6	0.46235435	0.76421693	0.48333333	0.61289823	0.48780988	0.46666667	0.15054388	1	0.11666667	0	0.5691164	0	0	1	
26	0.58333333	0.46235435	0.76421693	0.48333333	0.61289823	0.48780988	0.45	0.15054388	1	0.11666667	0	0.5691164	0	0	1	
27	0.58333333	0.46235435	0.76421693	0.45	0.15054388	1	0.43333333	0.61289823	0.48780988	0.11666667	1	0.23578307	0	0	1	
28	0.58333333	0.46235435	0.76421693	0.41666667	0.15054388	1	0.4	0.61289823	0.48780988	0.11666667	1	0.23578307	0	0	1	
29	0.58333333	0.46235435	0.76421693	0.56666667	0.15054388	1	0.45	0.61289823	0.48780988	0.13333333	1	0.23578307	0	0	0	
30	0.58333333	0.15054388	1	0.55	0.46235435	0.76421693	0.46666667	0.61289823	0.48780988	0.11666667	1	0.23578307	0	0	0	
31	0.56666667	0.15054388	1	0.55	0.46235435	0.76421693	0.46666667	0.61289823	0.48780988	0.1	0	0.5691164	0	0	0	
32	0.58333333	0.46235435	0.76421693	0.56666667	0.15054388	1	0.5	0.61289823	0.48780988	0.11666667	0	0.5691164	0	0	0	
33	0.56666667	0.15054388	1	0.56666667	0.46235435	0.76421693	0.48333333	0.61289823	0.48780988	0.15	0	0.5691164	0	0	0	
34	0.58333333	0.15054388	1	0.56666667	0.46235435	0.76421693	0.46666667	0.61289823	0.48780988	0.13333333	0	0.5691164	[illegible]	[illegible]	0	
35	0.58333333	0.15054388	1	0.56666667	0.46235435	0.76421693	0.5	0.61289823	0.48780988	0.13333333	0	0.5691164	[illegible]	[illegible]	0	
36	0.58333333	0.15054388	1	0.56666667	0.46235435	0.76421693	0.48333333	0.61289823	0.48780988	0.11666667	0	0.5691164	0	0	0	
37	0.58333333	0.15054388	1	0.56666667	0.46235435	0.76421693	0.46666667	0.61289823	0.48780988	0.11666667	0	0.5691164	0	0	0	
38	0.58333333	0.15054388	1	0.56666667	0.46235435	0.76421693	0.46666667	0.61289823	0.48780988	0.13333333	0	0.5691164	0	0	0	
39	0.58333333	0.46235435	0.76421693	0.56666667	0.15054388	1	0.46666667	0.61289823	0.48780988	0.13333333	0	0.5691164	0	0	0	
40	0.65	0.46235435	0.76421693	0.5	0.15054388	1	0.41666667	0.61289823	0.48780988	0.2	0	0.5691164	0	0	0	
41	0.65	0.46235435	0.76421693	0.48333333	0.15054388	1	0.4	0.61289823	0.48780988	0.2	0	0.5691164	0	0	0	

CSV

Neural Net -Training

- Multi Layer Perceptron
- Bias Neurons
- Sigmoid Function
- Full Connectivity



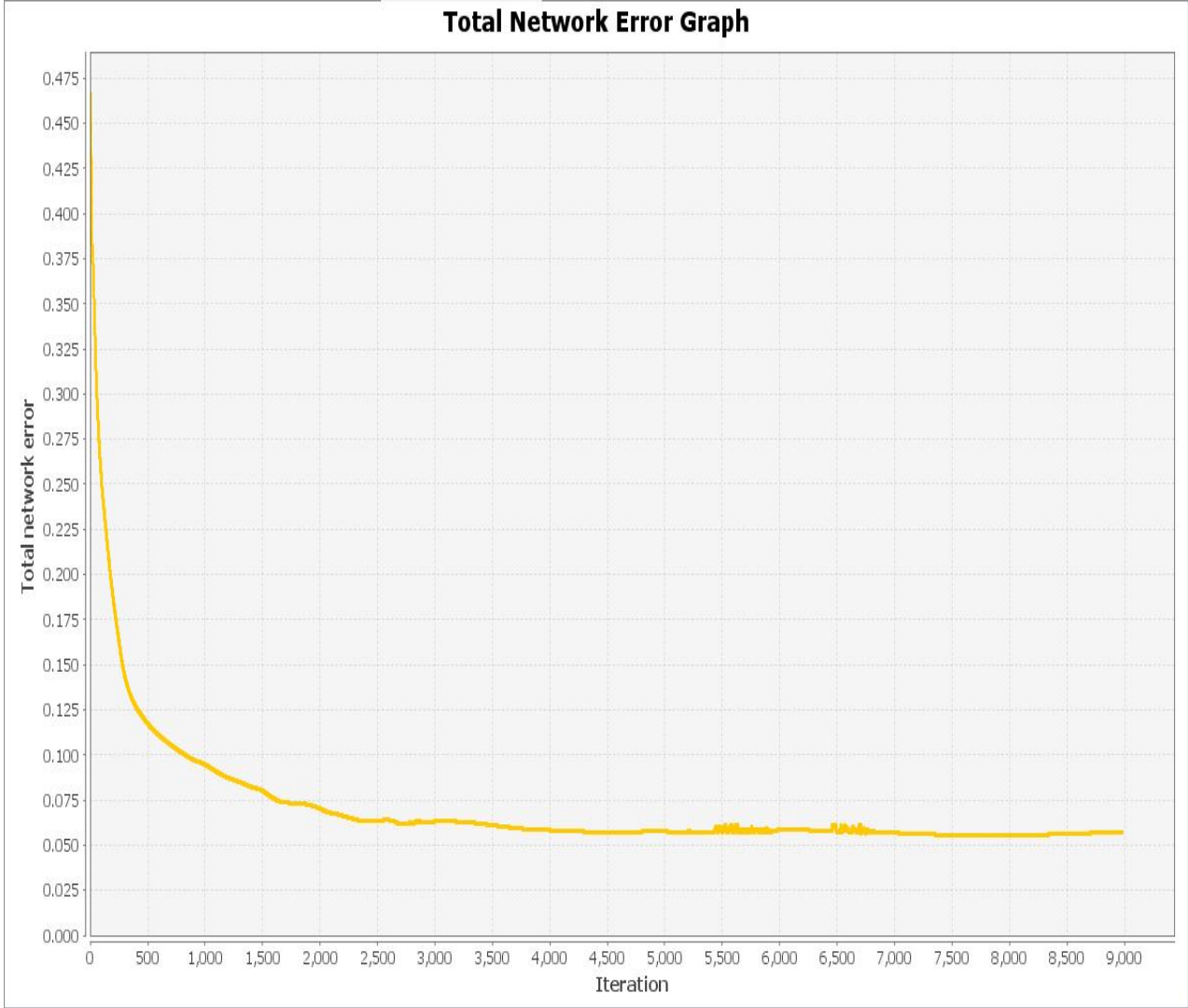
Neuroph Studio

Neural Net
-Training Result

- Network Training Graph
- Mean Square Error

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - y_{di}\right)^2$$

[Equation 4]



Neural Net -Testing with Training Data

- Total Mean Square Error: 0.017149080670527145

```
Output: 0.997; 0.0017; 0; 0.0001; 0; 0; 0; 0.0487; 0; Desired output: 1; 0; 0; 0; 0; 0; 0; 0; 0;  Erro

tput: 0; 0.0126; 0; 0; 0.9968; 0; 0; 0; 0.0002; Desired output: 0; 0; 0; 0; 1; 0; 0; 0; 0;  Error: 0; 0.
 Output: 1; 0; 0; 0.0041; 0; 0; 0.0006; 0; 0; Desired output: 1; 0; 0; 0; 0; 0; 0; 0; 0;  Error: -0; 0;
 Output: 0; 0; 0; 0.0011; 0; 0; 0.9519; 0; 0; Desired output: 0; 0; 0; 0; 0; 0; 1; 0; 0;  Error: 0; 0;
put: 0; 0.0001; 0.0965; 0; 0.0009; 0.0175; 0; 0; 0.9986; Desired output: 0; 0; 0; 0; 0; 0; 0; 0; 1;  Err

 Output: 0; 0; 0.0665; 0; 0.0007; 0.0428; 0.0001; 0; 0.9991; Desired output: 0; 0; 0; 0; 0; 0; 0; 0; 1;

1;  Output: 0.998; 0.0014; 0; 0.0002; 0; 0; 0; 0.0681; 0; Desired output: 1; 0; 0; 0; 0; 0; 0; 0; 0;  Er

1;  Output: 0; 0.0121; 0.294; 0; 0.0195; 0; 0; 0.0002; 0.3991; Desired output: 0; 0; 0; 0; 0; 0; 0; 0; 1

1;  Output: 0; 0.9886; 0; 0; 0.016; 0.0048; 0; 0.0009; 0; Desired output: 0; 1; 0; 0; 0; 0; 0; 0; 0;  Er

put: 0.7871; 0.0014; 0; 0.0004; 0; 0; 0; 0.1015; 0; Desired output: 1; 0; 0; 0; 0; 0; 0; 0; 0;  Error: -

1;  Output: 0.3407; 0; 0; 0.6937; 0; 0; 0.0665; 0; 0; Desired output: 0; 0; 0; 1; 0; 0; 0; 0; 0;  Error:

 Output: 0; 0.9844; 0; 0; 0.0104; 0.0116; 0; 0.0005; 0; Desired output: 0; 1; 0; 0; 0; 0; 0; 0; 0;  Err

utput: 0; 0.0001; 0.0938; 0; 0.0009; 0.0162; 0; 0; 0.9983; Desired output: 0; 0; 0; 0; 0; 0; 0; 0; 1;  E

t: 0; 0; 0; 0.9998; 0; 0; 0; 0; 0; Desired output: 0; 0; 0; 1; 0; 0; 0; 0; 0;  Error: 0; 0; 0; -0.0002;
 Output: 0; 0.1272; 0; 0; 0.0005; 0; 0.0064; 0.9895; 0; Desired output: 0; 0; 0; 0; 0; 0; 0; 1; 0;  Err

1;  Output: 0; 0.078; 0; 0; 0.0001; 0; 0.0033; 0.9826; 0; Desired output: 0; 0; 0; 0; 0; 0; 0; 1; 0;  Er
```

Neural Net -Testing with Testing Data

- Total Mean Square Error: 0.08063635897671174

```
.5691;  Output: 0; 0.0174; 0.0088; 0; 0; 0.9988; 0; 0; 0.0115; Desired output: 0; 0; 0; 0; 0; 1; 0; 0; 0;  Er
.5691;  Output: 0.0197; 0; 0; 0.9736; 0.0008; 0; 0; 0.0109; 0; Desired output: 0; 0; 0; 0; 0; 0; 1; 0; 0;  Er
691;  Output: 0; 0; 0; 0.404; 0; 0; 0.9811; 0.0005; 0; Desired output: 0; 0; 0; 1; 0; 0; 0; 0; 0;  Error: 0;
;  Output: 0.0029; 0; 0.0001; 0.1505; 0.0003; 0; 0.0001; 0.862; 0; Desired output: 0; 0; 0; 0; 0; 0; 1; 0; 0;
.5691;  Output: 0; 0; 0.6367; 0; 0.0001; 0; 0.1719; 0.0089; 0; Desired output: 0; 0; 0; 0; 0; 0; 0; 1; 0;  Er
 Output: 0.0351; 0.0001; 0; 0.0002; 0.0001; 0; 0.0016; 0.9507; 0; Desired output: 1; 0; 0; 0; 0; 0; 0; 0; 0;
691;  Output: 0.0324; 0; 0; 0.9735; 0; 0; 0.0024; 0; 0; Desired output: 0; 0; 0; 0; 0; 0; 1; 0; 0;  Error: 0.
;  Output: 0; 0.0101; 0.0004; 0; 0.0001; 0.9359; 0; 0; 0.0276; Desired output: 0; 0; 0; 0; 0; 1; 0; 0; 0;  Er
691;  Output: 0.0259; 0; 0; 0.8416; 0; 0; 0.0142; 0; 0; Desired output: 0; 0; 0; 0; 0; 0; 1; 0; 0;  Error: 0.
691;  Output: 0.8377; 0; 0.0001; 0; 0; 0; 0.0003; 0.516; 0; Desired output: 1; 0; 0; 0; 0; 0; 0; 0; 0;  Error
1;  Output: 0; 0.9996; 0; 0; 0; 0.0951; 0; 0.0199; 0; Desired output: 0; 1; 0; 0; 0; 0; 0; 0; 0;  Error: 0; -
 Output: 0; 0; 0.0001; 0; 0.002; 0; 0; 0; 0.9996; Desired output: 0; 0; 0; 0; 0; 0; 0; 0; 1;  Error: 0; 0; 0.
 Output: 0; 0; 0; 0; 0.0024; 0; 0; 0; 0.9985; Desired output: 0; 0; 0; 0; 0; 0; 0; 0; 1;  Error: 0; 0; 0; 0;
.5691;  Output: 0; 0.43; 0.0004; 0; 0; 0.9792; 0; 0; 0.0003; Desired output: 0; 0; 0; 0; 0; 1; 0; 0; 0;  Erro
691;  Output: 0; 0.9989; 0; 0; 0.0007; 0.0064; 0; 0.009; 0; Desired output: 0; 1; 0; 0; 0; 0; 0; 0; 0;  Error
.5691;  Output: 0; 0; 0.1038; 0; 0.0006; 0; 0.0005; 0.0175; 0.0004; Desired output: 0; 0; 1; 0; 0; 0; 0; 0; 0
1;  Output: 0.1114; 0; 0; 0.2526; 0; 0; 0; 0.2388; 0; Desired output: 1; 0; 0; 0; 0; 0; 0; 0; 0;  Error: -0.8
.5691;  Output: 0; 0.0204; 0.0025; 0; 0; 0.998; 0; 0; 0.0031; Desired output: 0; 0; 0; 0; 0; 1; 0; 0; 0;  Err
.5691;  Output: 0.0007; 0; 0.0659; 0.0007; 0.0023; 0; 0.0026; 0.6879; 0; Desired output: 0; 0; 0; 0; 0; 0; 0;
;  Output: 0; 0; 0.0006; 0; 0.0003; 0.0016; 0; 0; 0.974; Desired output: 0; 0; 0; 0; 0; 0; 0; 0; 1;  Error: 0
 Output: 0; 0.9955; 0; 0; 0.0003; 0.0003; 0; 0.0379; 0; Desired output: 0; 1; 0; 0; 0; 0; 0; 0; 0;  Error: 0;
.5691;  Output: 0.7201; 0; 0; 0.0003; 0; 0; 0.0001; 0.638; 0; Desired output: 1; 0; 0; 0; 0; 0; 0; 0; 0;  Err
.5691;  Output: 0; 0.0187; 0.0517; 0; 0.0045; 0.0001; 0; 0; 0; Desired output: 0; 0; 1; 0; 0; 0; 0; 0; 0;  Er
;  Output: 0; 0.0002; 0.0346; 0; 0.0007; 0.0001; 0; 0; 0.5375; Desired output: 0; 0; 1; 0; 0; 0; 0; 0; 0;  Er
91;  Output: 0; 0.4469; 0.0076; 0; 0; 0.3322; 0; 0; 0; Desired output: 0; 1; 0; 0; 0; 0; 0; 0; 0;  Error: 0;
1;  Output: 0; 0.9831; 0; 0; 0.0005; 0.1209; 0; 0.0009; 0; Desired output: 0; 0; 0; 0; 0; 1; 0; 0; 0;  Error:
;  Output: 0; 0.9099; 0.0001; 0; 0.0474; 0; 0; 0.0113; 0.003; Desired output: 0; 1; 0; 0; 0; 0; 0; 0; 0;  Er
691;  Output: 0.0024; 0; 0; 0.0079; 0; 0; 0.9529; 0.0002; 0; Desired output: 0; 0; 0; 0; 0; 0; 1; 0; 0;  Erro
691;  Output: 0.0014; 0.0014; 0.0001; 0.0001; 0.0001; 0; 0.0102; 0.9967; 0; Desired output: 1; 0; 0; 0; 0; 0;
.5691;  Output: 0; 0.0029; 0.0004; 0.0003; 0.0113; 0; 0; 0.9284; 0; Desired output: 0; 0; 0; 0; 1; 0; 0; 0; 0
.5691;  Output: 0.001; 0; 0; 0.9002; 0; 0; 0.1612; 0; 0; Desired output: 0; 0; 0; 1; 0; 0; 0; 0; 0;  Error: 0
```

Neural Net -Accuracy

Pinpoint vs. Neural Net

- Classifying a Correct Inner Location
- Improvement by about 15%

Neural Network	Pinpoint
69.2%	54.4%

Changes Along the Way

Real Number Prediction vs. Classification

Normalization

How many layers?

Neural Net
-Next Move

Scaling to Larger Buildings

Combine Algorithms

First NN then Pinpoint

Integrate Neural Network into Mobile App

Use Language like TensorFlow

Things learned

- How to design a neural network for a real life classification problem
- How to preprocess and normalize data for testing and learning
- The complexities of getting accurate measurements to ensure accuracy
- Developing in PHP and on Android
- Using GIS Tools to make overlay building maps to get the latitude and longitude of the WiFi APs

Sources

Mok, E., and G. Retscher. "Location Determination Using WiFi Fingerprinting versus WiFi Trilateration." *Journal of Location Based Services* 1.2 (2007): 145-59. Print.

Mok, Esmond, and Bernard Cheung. "An Improved Neural Network Training Algorithm for Wi-Fi Fingerprinting Positioning." *ISPRS International Journal of Geo-Information* 2.3 (2013): 854-68. Print.