

# Room-Level Indoor Localization with Artificial Neural Networks

Ahmet Serdar Karadeniz and Mehmet Önder Efe

Hacettepe University, Ankara, Turkey  
ahmet.karadeniz@hacettepe.edu.tr  
onderefe@hacettepe.edu.tr

**Abstract.** Indoor localization system determines the location of the users or some assets in indoor environments. There are important applications of indoor localization including smart home systems, indoor navigation and tracking systems. In this work, a reliable neural network model is developed for localizing users in room level. Model is based on Wi-Fi signals received by the users' devices at different rooms from various Wi-Fi access points. A neural network with two hidden layers with sigmoid activation functions is trained with back-propagation optimizing collected signal data. Some of the signals are set to 0 during the training process, which gives significant stability to the model under the conditions where some of the data required for prediction are not available. An additional dataset is collected for the evaluation in addition to the existing datasets. Performance of the model on the existing datasets as well as the new collected dataset is discussed and evaluated. Results are promising in terms of reliability and accuracy.

## 1 Introduction

As Global Positioning System (GPS) has plenty of applications that can easily be developed, determining the location of a user in an indoor environment is still a challenging problem as it requires a system with high accuracy and reliability with cost-effective technology. Proposed systems generally require different technologies such as GPS, Bluetooth beacons, RFID tags and Wi-Fi signals. There are some disadvantages of some of these technologies in terms of accessibility, cost and accuracy. GPS is only reliable in outdoor environments and may not give the desired accuracies [1]. Bluetooth beacons can only be used for short ranges [2]. RFID tags require users to carry an additional tag and may be expensive in terms of price [3]. Using the received signal strength (RSS) values of Wi-Fi access points (AP) makes use of the existing infrastructures by using smartphones and Wi-Fi APs which are highly available these days. Smartphones can be used to collect RSS values of the near Wi-Fi sources. Initially, a machine learning algorithm can be used to train with the RSS data and location of the users can be determined later with the trained algorithm. If a well-generalizing machine learning model can be developed, Wi-Fi RSS values can remove the need of additional cost and technology. Therefore, using RSS of Wi-Fi access points

can be considered as a feasible solution to this problem and an accurate, stable machine learning model needs to be developed for indoor localization systems.

Indoor localization as a machine learning problem have been considered as both supervised and unsupervised problem. For example, Chen *et al.* proposed a subarea localization scheme based on unsupervised clustering and subarea matching [4]. Indoor localization can also be described as a supervised classification or regression problem. Most of the existing solutions for solving indoor localization problem with RSSI try to find the 2D coordinate locations of the users [5, 6]. However, such precise localization may not be needed for some applications. For instance, an indoor navigation system can be developed by using only room-level position of the user by calculating the paths between different rooms. Another example is a smart home system where the room-level location of the users may improve efficiency and intelligence.

Room-level classification of indoor locations have been studied by various scholars and different solutions are proposed. Rezgui *et al.* have used SVM for room-level classification and introduced normalized rank transformation to reduce the effect of signal fluctuations [7]. Buchman *et al.* have used overlapping rings method which is based on the location of the access points [8]. Rohra *et al.* have used fuzzy hybrid of particle swarm optimization and gravitational search algorithm with Artificial Neural Networks (ANN) for room level indoor localization [2]. Their model has high classification accuracy and also includes fuzzy branch for dealing with uncertainties [2]. Furthermore, dataset used by Rohra *et al.* have been used in other works as well. For instance, Gomes *et al.* obtained a high accuracy with the dataset used in [2] by training a Random Forest classifier [9]. Altay *et al.* also obtained high classification accuracy with Linear Discriminant Analysis (LDA) classification [10]. Sabanci *et al.* compared different classifiers on the same dataset and obtained that k-Nearest Neighbors (k-NN) was the most successful classifier. Most of the works for room-level indoor localization give decent accuracies but few of them give stable performances of the proposed solutions under disturbances such as lack of a signal from an AP on which the model was trained. In this paper, a simple yet efficient solution to this problem is proposed and the performance of the model under different conditions is provided [11].

In the following sections, implementation details, behavior and the performance of the model is explained. In section 2, the datasets used in the experiments are described. In section 3, ANN implementation details and its behavior with respect to different parameters are discussed. Moreover, an algorithm for robustness is proposed. Finally, the model is evaluated on different datasets with classification accuracies and confusion matrices. Results show that proposed model is accurate and reliable which can be used for an indoor localization system.

## 2 Datasets

For training and evaluation, two different datasets are used which will be called Dataset1 and Dataset2. Both datasets include RSS values of the APs and the corresponding room labels. Dataset1 is collected by Rohra *et al.*, where the indoor environment is located in an office in United States [2]. In this dataset, there are seven APs and the data is collected in four different rooms with 1 second intervals [2]. Dataset2 is collected in a house located in Ankara, Turkey. In the collection process for Dataset2, a simple Python script, which calls `iwconfig` command from an Ubuntu terminal, for receiving RSS values of the routers is executed every 500 milliseconds in different rooms. In Dataset2, there are two APs and four different rooms. This can be seen as a more challenging dataset when compared to Dataset1 as four rooms are covered by only two APs. Wi-Fi RSS signals have values changing from -30 to -90. There are two thousand samples in each dataset. For preprocessing, we scaled the data to have zero mean and unit variance.

In Fig. 1, distribution of the training data is plotted for Dataset1. Note that Principal Component Analysis (PCA) for dimensionality reduction is applied as there are seven APs for Dataset1 and the explained variance ratio of PCA is 94.45%.  $x_1$  and  $x_2$  are input signals in the projected feature space. Different shapes correspond to the four rooms in Dataset1.

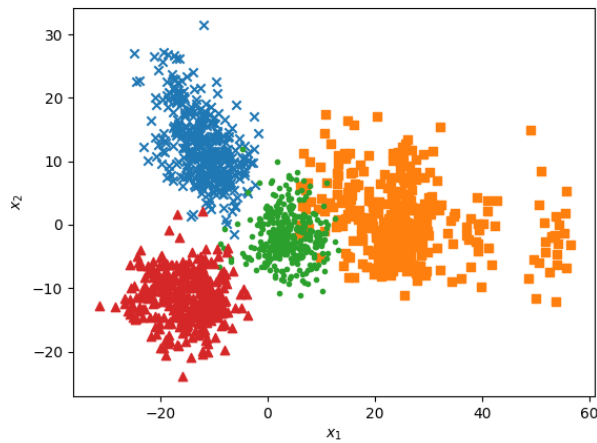


Fig. 1: Distribution for Dataset1

### 3 Proposed Method

ANNs have been powerful tools in solving many tasks. These tasks include identification of nonlinear systems, medical diagnosis, financial applications and many others. A neural network with 2 hidden layers is implemented in this work. In all layers, sigmoid function is used as it is one of the mostly chosen for classifying nonlinear data. 20% of the Dataset1 is used for parameter optimization. Furthermore, to validate the convergence of the model, all the tuning process is made only for Dataset1 and the model is tested on Dataset2 without changing any parameters.

There are four layers including the input and output layers. Let  $w^k$  be the weight matrix between the layers  $k$  and  $k + 1$ ,  $o^k$  be the output of the  $k$ -th layer and  $o^0$  represent the input layer.  $w^k$  is initialized to random small numbers and  $o^0$  corresponds to the input data. Then, the feedforward operation used can be represented as

$$s^{k+1} = w^k o^k \quad (1)$$

$$o^{k+1} = f(s^{k+1}) \quad (2)$$

where  $f$  is the sigmoid function  $f: \mathbb{R} \rightarrow [0, 1]$  which is defined as  $f(x) = \frac{1}{1+e^{-x}}$ . We used mean squared error  $J: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  defined as  $J(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$  where  $\hat{y}$  is the output vector,  $y$  is the true labels vector and  $n$  is the number of samples. To prevent overfitting,  $L_2$  regularization is used. After  $L_2$  regularization, loss function becomes

$$\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 + R(W) \quad (3)$$

where  $R(W) = \frac{\lambda}{2} \sum_k \sum_l (W_{k,l}^2)$ . Thus, we add  $\lambda w$  to the gradients in the backpropagation. Then, backpropagation with the gradient descent algorithm becomes

$$\Delta w_t^k = -\eta \frac{\partial J}{\partial w_{t-1}^k} + \lambda w_{t-1}^k \quad (4)$$

where  $t$  represents the iteration,  $\lambda$  and  $\eta$  are the regularization and the learning rate terms.

As we discussed in the introduction part, the model has to be stable in situations where the device is not able to receive the signals from some routers. We simulated this situation by randomly setting the RSS values from random access points to 0. Thus, an average of 519 signals out of 3500 signals are set to zero in the validation set, which corresponds to the 14.1% of all validation data. In practice, when a device cannot receive the signal, the same 0 value can be used instead of the expected RSS value. To be precise, the simulation algorithm is illustrated in Algorithm 1.

---

**Algorithm 1** RSS Disturbance Simulation

---

```
1:  $p \leftarrow 0.2$  ▷ Disturbance probability
2: for  $i = 0 \dots n_{samples}$  do
3:   for  $j = 0 \dots n_{features}$  do
4:      $r \leftarrow \text{random}(0, 1, p)$  ▷ Choose 0 or 1 with probability  $p$ 
5:     if  $r == 0$  then
6:        $\hat{X}[i, j] \leftarrow 0$ 
7:     end if
8:   end for
9: end for
```

---

An intuitive solution to this problem is to set some of the features in the input vector to zero randomly during the training. At each iteration, we set different signals to zero with probability  $p = 0.20$ . In this way, the network learns the cases where some signals are missing in the input data. Thus, feedforward equation is changed as initial  $o^0$  is replaced by the disturbed input vector  $\tilde{o}^0$ . Note that  $\tilde{o}^0$  changes at each iteration due to the randomness in the disturbance algorithm. While this operation can decrease the overall accuracy slightly, it may be tolerated when the stability is important.

A proper initialization and tuning of the parameters is essential for training neural networks. Therefore, we optimized the parameters of the network such as learning rate, number of epochs, regularization parameter and disturbance probability parameter. Results of the learning rate experiments are shown in Fig. 2 where loss graphs with different learning rates are plotted.  $\eta = 0.01$  gives a fast and smooth drop on the training loss curve. All the other chosen parameters are shown in Table 1.

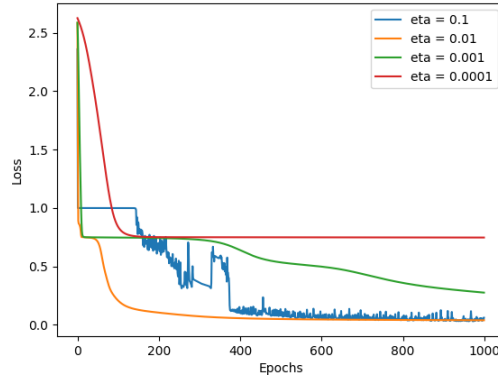


Fig. 2: Training Loss with Different Learning Rates

Table 1: Parameters of the ANN and Training Algorithm

<i>Parameter</i>	<i>Value</i>
Learning Rate	0.01
Regularization	0.001
Number of Epochs	500
Number of Neurons in the 1st Hidden Layer	7
Number of Neurons in the 2nd Hidden Layer	7
Disturbance Probability	20%

## 4 Performance Evaluation

In this section, performance of the model is evaluated. First, we tested the model for Dataset1 and Dataset2. We used 10-fold cross validation in the reported accuracies. Then, impact of the proposed algorithm is discussed. Moreover, we compared our model with other classifiers where around 20% of the signals in the test sets of Dataset1 were set to zero. Finally, confusion matrices for both datasets are provided.

As shown in Table 2, 96.44% and 91.45% classification accuracies were obtained on Dataset1 and Dataset2, respectively. Although all the parameter optimization process was made on the validation set of Dataset1, the model was also able to perform well for the Dataset2 with the same parameters. This shows that our model can be used in different indoor environments without any modification.

Table 2: Classification Accuracies

<i>Dataset</i>	<i>Accuracy</i>
Dataset1	96.44%
Dataset2	91.45%

In Fig. 4, training and validation accuracies were plotted to illustrate the effect of the proposed algorithm. In this figure, blue and orange curves are training and validation accuracies of ANN without the proposed modification, red and green curves represent the training and validation accuracies of ANN with the proposed modification. From the figure, it is deduced that the proposed method makes the model more robust against disturbances.

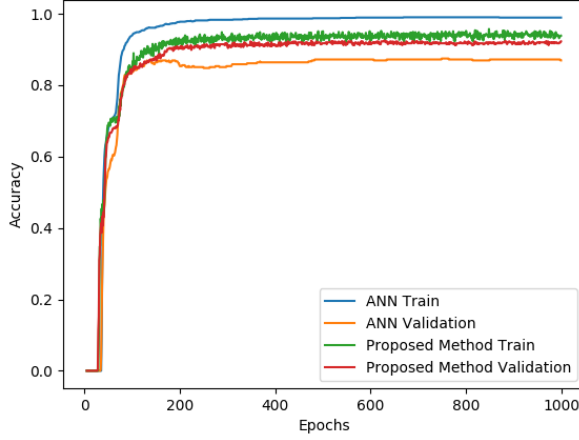


Fig. 3: Impact of Proposed Method

In Table 3, classification accuracies are compared with various classifiers. Classifiers used in other works were chosen to compare our model [9–11]. k-NN was used in [11], Random Forest was used in [9] and LDA was used in [10]. Although reported accuracies were high in [9–11], it is observed that they can significantly drop when some of the data are missing if there is no proper modification. Table 3 shows that our method yielded the highest accuracy under the condition that 20% of the test data were missing.

Table 3: Classification Accuracies

<i>Model</i>	<i>Accuracy</i>
k-NN [11]	90.70 %
Random Forest [9]	84.35 %
LDA [10]	87.35 %
Proposed Method	<b>93.05 %</b>

In Figs. 4, average confusion matrices with 10-fold cross validation are illustrated. Diagonal elements in the confusion matrices correspond to per class accuracy. Note that test sets were not disturbed in the reported confusion matrices. In Fig. 4a, confusion matrix for Dataset1 is illustrated. We have above 98% accuracy for room 0, room 1 and room 3. For room 2, we have 89% accuracy. In Fig. 4b, class accuracies are around 90% except room 1 which have 98% classification accuracy.

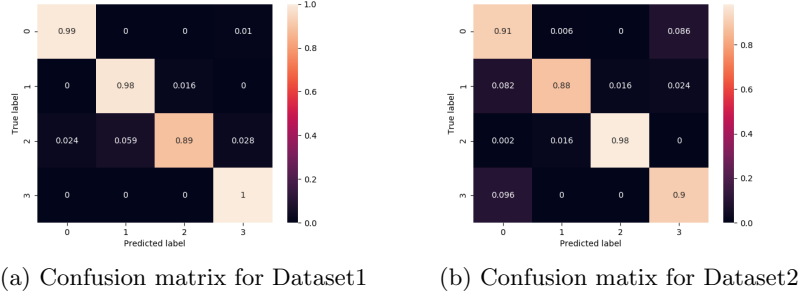


Fig. 4: Confusion Matrices

## 5 Conclusion

Room-level indoor localization problem is challenging especially when there is a lack of data at test time. Furthermore, the number of Wi-Fi access points is another important factor obtaining high accuracies. Main concentration in this work was to develop a room-level indoor localization algorithm which is resistant to lack of signals and generalizable into different indoor environments. An accuracy of 96.44% and 91.45% was obtained in two different datasets. Moreover, 93.05% classification accuracy was obtained where around 20% of the collected data were assumed to be missing at prediction time, which is around 3% higher than the closest classifier. In the future, more advanced architectures can be used to improve the overall performance of the proposed algorithm.

## References

1. Eladio Martin, Oriol Vinyals, Gerald Friedland, and Ruzena Bajcsy. Precise indoor localization using smart phones. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 787–790. ACM, 2010.
2. Jayant G Rohra, Boominathan Perumal, Swathi Jamjala Narayanan, Priya Thakur, and Rajen B Bhatt. User localization in an indoor environment using fuzzy hybrid of particle swarm optimization & gravitational search algorithm with neural networks. In *Proceedings of Sixth International Conference on Soft Computing for Problem Solving*, pages 286–295. Springer, 2017.
3. Azkario Rizky Pratama, W Widyawan, A Lazovik, and M Aiello. Indoor self-localization via bluetooth low energy beacons. *IDRBT J. Bank. Technol*, 1:1–15, 2017.
4. Qiuyun Chen and Bang Wang. Fincm: fingerprint crowdsourcing, clustering and matching for indoor subarea localization. *IEEE Wireless Communications Letters*, 4(6):677–680, 2015.
5. Krishna Chintalapudi, Anand Padmanabha Iyer, and Venkata N Padmanabhan. Indoor localization without the pain. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, pages 173–184. ACM, 2010.



6. Chin-Heng Lim, Yahong Wan, Boon-Poh Ng, and Chong-Meng Samson See. A real-time indoor wifi localization system utilizing smart antennas. *IEEE Transactions on Consumer Electronics*, 53(2):618–622, 2007.
7. Yasmine Rezgui, Ling Pei, Xin Chen, Fei Wen, and Chen Han. An efficient normalized rank based svm for room level indoor wifi localization with diverse devices. *Mobile Information Systems*, 2017, 2017.
8. Attila Buchman and Claudiu Lung. Received signal strength based room level accuracy indoor localisation method. In *2013 IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom)*, pages 103–108. IEEE, 2013.
9. Rahul Gomes, Mostofa Ahsan, and Anne Denton. Random forest classifier in sdn framework for user-based indoor localization. In *2018 IEEE International Conference on Electro/Information Technology (EIT)*, pages 0537–0542. IEEE, 2018.
10. Osman Altay and Mustafa Ulas. Location determination by processing signal strength of wi-fi routers in the indoor environment with linear discriminant classifier. In *2018 6th International Symposium on Digital Forensic and Security (IS-DFS)*, pages 1–4. IEEE, 2018.
11. Kadir Sabanci, Enes Yigit, Deniz Ustun, Abdurrahim Toktas, and Muhammet Fatih Aslan. Wifi based indoor localization: Application and comparison of machine learning algorithms. In *2018 XXIIIrd International Seminar/Workshop on Direct and Inverse Problems of Electromagnetic and Acoustic Wave Theory (DIPED)*, pages 246–251. IEEE, 2018.