Project Title: Machine Learning based Distance Estimation Using RSSI Measurements

Authors

Yifan Ye <yifany@kth.se>

Weaam Bayaa <bayaa@kth.se>

Abstract

With the rise of the Internet of Things and location-based services paradigms, indoor localization has become an interesting research topic. Received Signal Strength Indicator (RSSI) has frequently been used in distance estimation for ranging-based localization due to its availability without the need for investment in new infrastructure. The tradition way to estimate distance is based on the signal propagation model, however, temporal variations of RSSI are the main reason for RSSI's reduced accuracy of distance estimation. In this research, the fingerprinting concept is reused and complemented with 3 machine learning (ML) algorithms (Decision Tree Classifier, Random Forest Classifier, and K-Nearest Neighbor) to estimate the distance of devices from a WiFi access point (AP). Instead of using the RSSI values directly in the modeling, statistical properties (mean, standard deviation, minimum, maximum) of RSSI measurements have been used to overcome the uncertainty. The accuracy of the proposed models are evaluated, experimental results are presented to show that ML-based models have better accuracy than the propagation-based model, and RFC showed a high performance comparing to the other 2 ML algorithms.

keywords: Distance Estimation, Wi-Fi, RSSI, Fingerprinting, IoT, Location - based Services, Machine Learning.

Introduction

The Global Position System (GPS) has been widely used for positioning in outdoor environments. Since GPS relies mainly on signal propagation in the air, the accuracy of GPS is degraded severely in indoor environments as the buildings' infrastructure will impact the propagation [1]. Accordingly, all efforts have been redirected to reuse existing Wi-Fi infrastructure in localization and distance estimation. In the recent decade, many researches have adopted different approaches to indoor localization. Node localization is used to measure the distance using two methods: Range-based and Range free. The ranging-free method estimate position of the device without knowing the device's distance from APs first, Wi-Fi fingerprinting is one one of the most ranging-free positioning technologies, which builds a database with fingerprinting of all the coordinate points, and estimates position by matching measurements with all the fingerprinting in the database, this method has high accuracy but low efficiency, and is sensitive to environment' s change [2,3]. On the other hand, the range-based method need estimation of devices' distance from APs, several techniques can be used to estimate distance such as Time of Arrival (ToA) [4], Time Difference of Arrival (TDOA) [5], Angle of Arrival (AOA), or Receive Signal Strength Indicator (RSSI) Log-normal shadowing model (LNSM). Combining LNSM with TOA, TDOA and AOA methods have results with high accuracy but require high investments [6]. RSSI-LNSM was very attractive approach as it does not imply high cost [7], however, the main problem with RSSI is the uncertainty of measured values due to multi-path propagation, fading and reflections, using propagation model to estimate distance sometimes can have more than 50% distance error [8]. Accordingly, many researchers focused on increasing the accuracy of distance estimation and reducing errors. This research proposes a novel ML-based distance estimation method, and and we exploit statistical properties of RSSI rather. The remainder of this research is constructed as follows: section 2 analyses the related work. Section 3 presents the proposed method of using machine learning on different datasets. Results and discussions are shown in section 4. Finally, conclusions are listed in section 5.

Related work

There are several techniques to use RSSI measurements in localization and distance estimation. Trilateration is a popular method to determine the location of a device from two reference points (RPs) by using the intersection of the circles around the RPs [9]. On the other hand, the triangulation technique determines the location of a device by measuring the angles to it from known access points at either end of a fixed baseline [10]. The problem with these methods is that they require the knowledge of APs or RPs positions. Fingerprinting technique was the most widely used for localization which involves creating radio map by collecting measurements in different locations and store it in a database[3], this technique doesn't require prior knowledge of APs locations. K-Nearest Neighbor (KNN) is used to estimate the location of users [11]. In the probabilistic approach, the probability of each grid point is calculated and the user's location is estimated using Bayesian inference [12]. The accuracy is enhanced in the probabilistic approach but the main drawback is the need to collect a large number of points to create a distribution.

This research focuses on using machine learning approach to estimate the distance. it investigates the relationship between RSSI measurements and distance in different environments. To overcome the uncertainty of RSSI values, mean, minimum, maximum and of RSSI are used as input features for the ML algorithms. Then, different ML algorithms are investigated to compare accuracy.

Research questions, hypotheses

Before conducting this research, we proposed the following questions.

1. Is the accuracy of propagation model based distance estimation indoors as poor as was pointed out in [7,11]?
2. Can we find some certain statistic properties of RSSI measurements at each distance, while there is so much interference indoors that RSSI measurements can fluctuate drastically?

3. Do DTC, RFC and KNN algorithms perform well on statistic feature vectors? Do these algorithms can so improve the accuracy of estimating devices' distance from APs in complicated indoor environments with much interference?

4. How parameter K of KNN algorithm can impact on the accuracy of KNN-based model?

5. Near an AP, if the signal propagation based model can also estimate as accurately as ML-based models? In another word, can we use signal propagation model to estimate distance nearby an AP?

Method

The framework of our research is shown in Figure 1. We use collected RSSI measurements in a complicated indoor environment to build a signal propagation based model and several ML-based models, and the performance of these models will be evaluated and compared. Especially, it consists of the following modules: RSSI measurements collection, generate statistic feature vectors, data partition, apply ML algorithms, data processing, curve fitting, distance estimation.

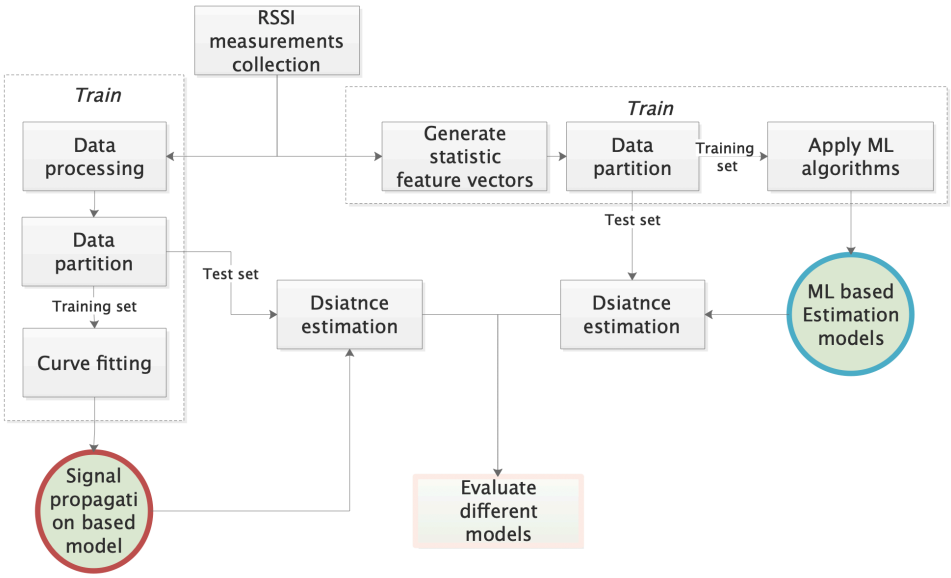

Figure 1: Framework

(1) RSSI measurements collection. We collect enough RSSI data in a complicated indoor space, which will be discussed in detail in the experiment data section.

(2) Generate statistic feature vectors. Rather than just using RSSI as the feature, we focus on statistic properties of RSSI measurements. For every 100 consecutive RSSI measurements, we calculate their mean, standard deviance, maximum and minimum, then construct a vector with these 4 features.

(3) Data partition. We split data into two parts, the first one serves as the training dataset to build either signal propagation based model or ML-based model. The other is for testing.

(4) Apply ML algorithms. We apply 3 different ML algorithms on training data: decision tree classifier (DTC), random forest classifier (RFC), and KNN.

DTC is a very popular machine learning classification algorithm with low algorithm complexity. A decision tree consists of internal nodes, leaf, and directed edges. Every internal node represents a test condition that is used to separate data, and one leaf represents one category. After we successfully build a DTC, then it is very easy to category data by let data pass through the root all the way to a leaf. [13]

RFC is an ensemble algorithm that evolves from DTC. In RFC, a collection of decision trees are generated and the tree with the highest vote is chosen, RFC takes weights based on the input as a parameter that resembles the number of decision trees. Those weights will be formed in the collaborative forest classifier without the conventional tree pruning process. [14] Many researchers have done some work to compare the performance between RFC and DTC, in most cases, RFC is less likely to overfit, thereby RFC performs better than DTC [14, 15-17], we also compare their performance on our dataset.

KNN is one of the easiest ML classifier algorithms that even “do not have train phase”, the core of KNN is sample A belongs to the category which has the most samples among A’ s k nearest neighbors[18]. The parameter k need to be set manually, in our research, we analyze parameter k’ s impact on the accuracy of distance estimation.

(5)Data processing. We pre-process the data for building a propagation model to make the performance better.

(6)Curve fitting. Chen, et al. [19] proposed the following equation can be used to estimate distance from RSSI.

$$d = d_0 \cdot 10^{(RSSI_0 - RSSI)/(10 \cdot n)} \quad (1)$$

where:

$RSSI$– read-out returned by the receiver, in [dBm],

d – distance between receiver and transmitter , in [m],

n– propagation constant,

$RSSI_0 - RSSI$ read-out at reference distance d_0, in [dBm].

Using Equation (1) to fit collected data, we can obtain value of n and $RSSI_0$, thereby a correlation between distance and $RSSI$ is built.

(7)Distance estimation. We separately estimate distance based on propagation model and ML-based models, then evaluate performance of these models.

1. Experiment setup

The experiment is conducted in the corridor of KTH campus in Kista with lots of sources of attenuation, reflection, interference, etc. due to walls, obstacles, pillars, ceiling, and so many people that walked around. An Apple iPhone is used as an AP, and it is located on one sofa to add more interference as shown in Figure 2 and Wireshark is used in monitoring mode to collect RSSI measurements.

2. Experiment data

As is shown in Figure 3, we have 21 distance points, each point is defined in one dimension (horizontally) and the distance between each point is 0.5 meter. At each point, 5000 RSSI measurements are collected. So we can get a raw dataset (RDS) shown in the following equation.

$$RDS = \begin{Bmatrix} (d_0, y_1^0) & (d_0, y_2^0) & (d_0, y_3^0) & \dots & (d_0, y_n^0) \\ (d_1, y_1^1) & (d_1, y_2^1) & (d_1, y_3^1) & \dots & (d_1, y_n^1) \\ (d_2, y_1^2) & (d_2, y_2^2) & (d_2, y_3^2) & \dots & (d_2, y_n^2) \\ \dots & \dots & \dots & \dots & \dots \\ (d_m, y_1^m) & (d_m, y_2^m) & (d_m, y_3^m) & \dots & (d_m, y_n^m) \end{Bmatrix} \quad (2)$$

Where m is the number of collection points($m = 20$), n refers to times that RSSIs are measured at each point ($n = 5000$), d_i represents i-th point's distance from AP ($0 \leqslant i \leqslant m$), and y_j^i is the j-th of $RSSI$ measurements at point i (($0 \leqslant i \leqslant m, 1 \leqslant j \leqslant n$).

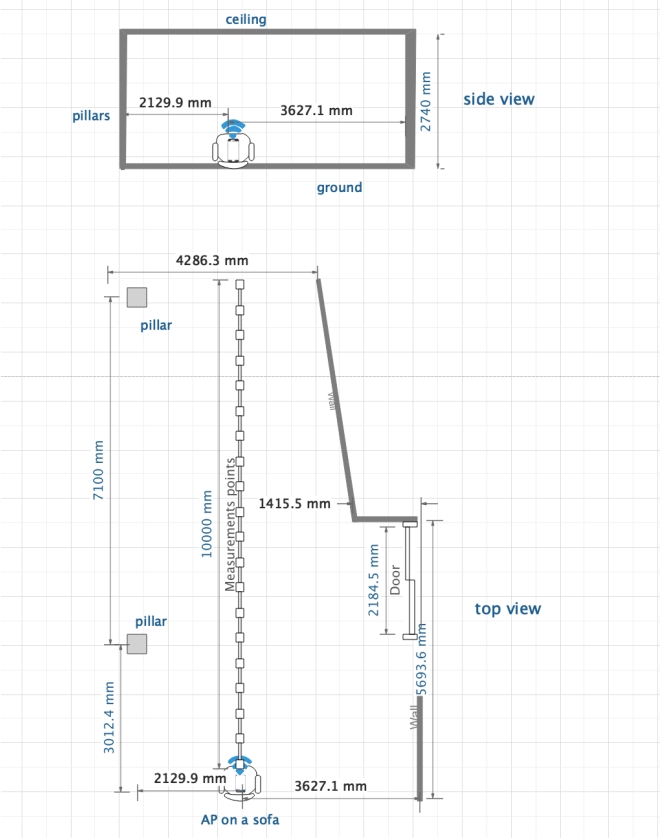

Figure 2: Floor plan

Figure 3: Experimental scene

3. Statistic feature vectors expression

We construct feature vectors out of collected RSSI measurements, which are fed to ML algorithms. For every 100 consecutive RSSI measurements, the mean, var, maximum, and minimum are calculated to be assembled into a vector as is shown is the following equation.

$$\overrightarrow{V_j^i} = \left[\begin{array}{cccc} mean_j^i & var_j^i & max_j^i & min_j^i \end{array} \right], (0 \leqslant i \leqslant m, 1 \leqslant j \leqslant \frac{n}{100}) \quad (3)$$

Where :

$\overrightarrow{V_j^i}$ — j-th feature vectors at i-th points.

$mean_j^i$— mean of j-th 100 consecutive RSSI measurements at i-th points

var_j^i— var of j-th 100 consecutive RSSI measurements at i-th points

max_j^i— maximum of j-th 100 consecutive RSSI measurements at i-th points

min_j^i— minimum of j-th 100 consecutive RSSI measurements at i-th points

4. Signal propagation based model

Matlab curve fitting tool provides an easy and intuitional way to fit, as is shown in Figure 4.

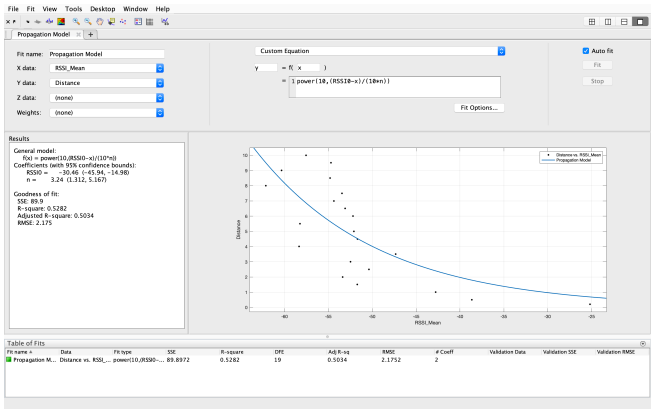

Figure 4: Matlab curve fitting toolbox

We choose mean of RSSI measurements at each point as X data, and distance as Y data, then customize the fitting equation the same as Equation (1). The Matlab will fit the curve and give the parameter n and $RSSI_0$ automatically.

5. ML based models

We use python sklearn library to build DTC, RFC, and KNN models, the training dataset is an array with 630 ((3000/100) * 21) 4-statistic-feature vectors, and a 630*1 array which contains 21 different distance. The test dataset is an array with 420 ((2000/100) * 21) 4-statistic-feature vectors, and a 420*1 array which contains 21 different distance.

6. Evaluation index

Accuracy and efficiency are used as indexes to evaluate the performance of different models [8].

For evaluation of the accuracy of different methods of distance estimation, we adopt average distance error (ADE) as indicated in equation(4).

$$ADE = \sum_{i=1}^{n_t} (\left| \hat{d}_i - d_i \right| / n_t) \quad (4)$$

where $\hat{d}_i$ and d_i are estimation distance and real distance for i-th test data, and n_t refers to the size of test dataset. The smaller ADE is, the better accuracy the model has.

As to evaluation of efficiency, two indexes are used. The first one is T_1, which is training time, the second one is T_2 representing to testing time.

Results and analysis

1. Parameter K' s impact on accuracy of KNN based distance estimation model

K is the only hyper-parameter of KNN algorithm, it denotes how many nearest neighbors a sample will choose. Value of K can have a high impact on KNN

algorithm, a way too small k can easily lead to overfitting, while the model may under-fit the dataset if the k is too large. We want to know what is the best k for our dataset. Based on Equation(4) for k ranging from 2 to 29, we separately compute average distance error within 10m. We can see from Figure 5 That KNN based model have the highest performance when K =4.

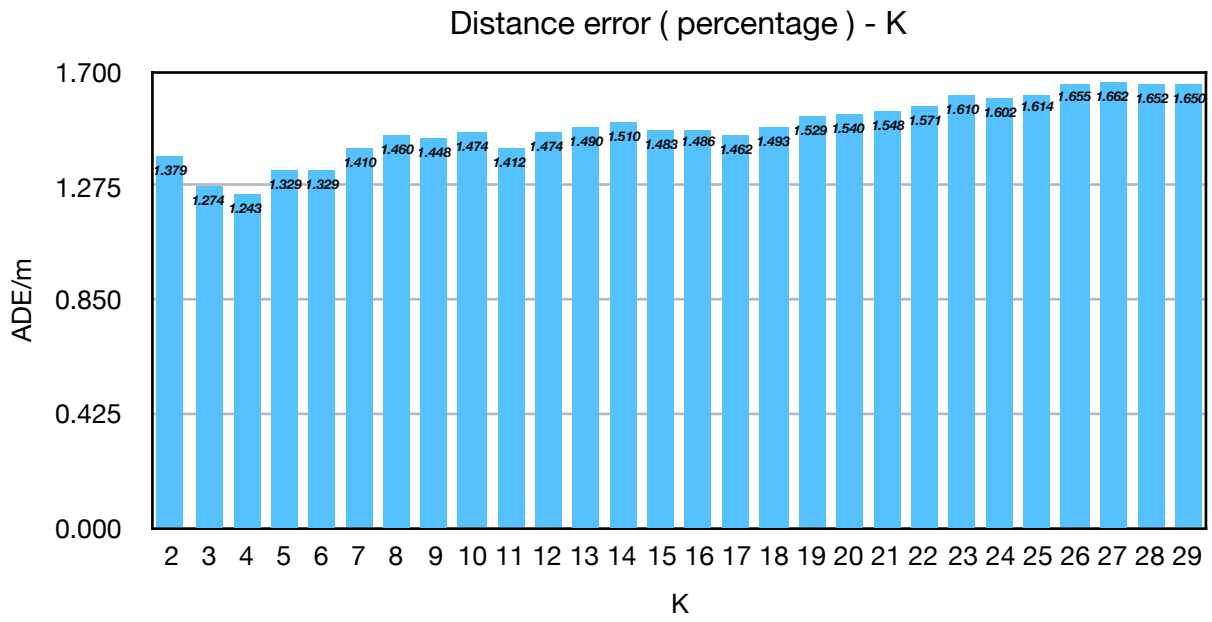

Figure 5: Distance error - K

2. Evaluation of accuracy of the models

The train set is fed into 3 different ML algorithms, KNN, DTC, and RFC, using Equation (4), we evaluate their accuracy with distance as is shown in Figure 6. Obviously all the 3 ML-based models have less average distance error than the propagation-based model, and on the whole, accuracy decrease with the increase of distance. Near the AP, the accuracy of ML-based models is extremely high, while the propagation-based model perform badly, we think one reason is Log-normal shadowing model itself fails to work well in such a complicated environment with so much interference, the other reason is the space between every two sampling point is way too long to curve fitting.

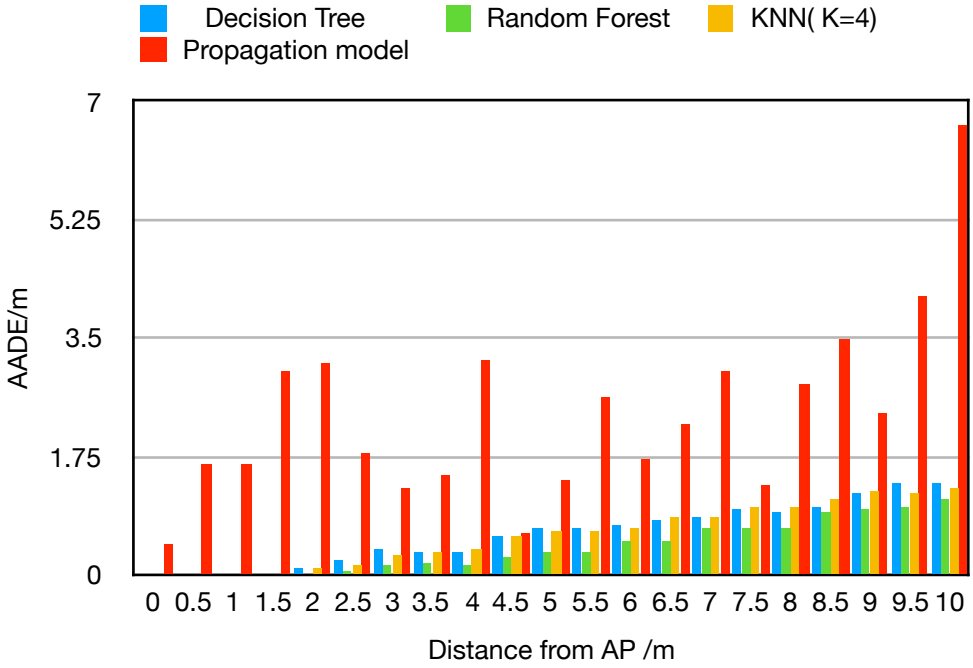

Figure 6: Comparison of 4 models

Figure 7 gives a clearer comparison of 3 ML-based models, the error curve of KNN and DTC almost overlap, and both of them have less accuracy than RFC, we think it is because that RFC overcome overfitting of DTC to some degree, and "KNN often fails to work well with inappropriate choice of distance metric or due to the presence of numerous irrelevant features." [20,21]

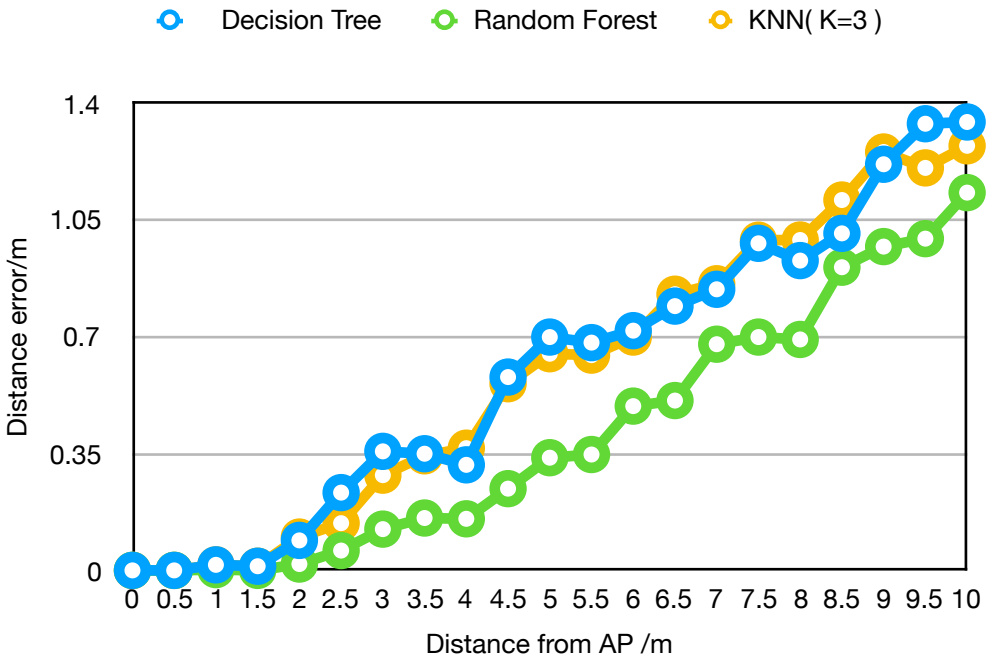

Figure 7: comparison of 3 ML-based models

Conclusion

In this report, we have proposed a novel ML-based distance estimation scheme incorporating the use of statistical properties of collected RSSI measurements. In proposed models, the dataset is a 4-features vector with mean, standard deviation, minimum and maximum values of RSSI measurements. Statistical properties proved that it can be used to overcome the uncertainty issue of RSSI measurements. The vectors are used as an input to three ML algorithms: KNN, DTC, and RFC to build three ML-based models, as a comparison, a propagation-based model is also built. Experiment results show that our ML-based models can improve accuracy compared with the propagation-based model, and RFC performs best among 3 ML-based models.

Future work

1. It is reasonable to take time and computation into account and evaluate the efficiency of these models;
2. Decrease space between every two sampling points and test in more different places may help to improve accuracy;
1. More factors can be considered and giving vectors more features to be fed to ML algorithms.
1. Deep neural learning may perform excellently with more data and features.

Reference:

[1] Subhan, F., Hasbullah, H., Rozyyev, A., & Bakhsh, S. T. (2011, April). Indoor positioning in bluetooth networks using fingerprinting and lateration approach. In Information Science and Applications (ICISA), 2011 International Conference on (pp. 1-9). IEEE. DOI: 10.1109/ICISA.2011.5772436

[2] Mok, E., & Retscher, G. (2007). Location determination using WiFi fingerprinting versus WiFi trilateration. Journal of Location Based Services, 1(2), 145 -15 9.DOI: 10.1080/17489720701781905

[3] Farshad, A., Li, J., Marina, M. K., & Garcia, F. J. (2013, October). A microscopic look at WiFi fingerprinting for indoor mobile phone localization in diverse environments. In International Conference on Indoor Positioning and Indoor Navigation (Vol. 28, p. 31st). DOI: 10.1109/IPIN.2013.6817920

[4] Alavi, B., & Pahlavan, K. (2006). Modeling of the TOA-based distance measurement error using UWB indoor radio measurements. IEEE communications letters, 10(4), 275-277. DOI: 10.1109/LCOMM.2006.1613745

[5] Kim, Y. G., An, J., & Lee, K. D. (2011). Localization of mobile robot based on fusion of artificial landmark and RF TDOA distance under indoor sensor network. International Journal of Advanced Robotic Systems, 8(4), 52. DOI: 10.5772/45698

[6] Seet, B. C., Zhang, Q., Foh, C. H., & Fong, A. C. (2012). Hybrid RF mapping and Kalman filtered spring relaxation for sensor network localization. IEEE Sensors Journal, 12(5), 1427-1435. DOI: 10.1109/JSEN.2011.2173190

[7] Peng, Yu & Luo, Qing-Hua & Wang, Dan & Peng, Xi-Yuan. (2012). WSN Localization Method Using Interval Data Clustering. Acta Automatica Sinica. 38. DOI: 1190. 10.3724/SP.J.1004.2012.01190.

[8] Luo, Q., Yan, X., Li, J., Peng, Y., Tang, Y., Wang, J., & Wang, D. (2016). DEDF: lightweight WSN distance estimation using RSSI data distribution-based fingerprinting. Neural Computing and Applications, 27(6), 1567-1575. DOI: 10.1007/s00521-015-1956-2

[9] Shchekotov, M. (2014, October). Indoor localization method based on Wi-Fi trilateration technique. In Proceeding of the 16th conference of fruct association (pp. 177-179). Available at: https://www.fruct.org/publications/abstract16/files/Shc1.pdf

[10] Y. Wang, X. Yang, Y. Zhao, Y. Liu, and L. Cuthbert, “Bluetooth positioning using RSSI and triangulation methods,” in 2013 IEEE Consumer Communications and Networking Conference (CCNC), 2013, pp. 837–842. DOI: 10.1109/CCNC.2013.6488558

[11] Gualda, D., Urena, J., Garcia, J. C., Garcia, E., & Ruiz, D. (2013, October). RSSI distance estimation based on Genetic Programming. In Indoor Positioning and

Indoor Navigation (IPIN), 2013 International Conference on (pp. 1-8). IEEE. DOI: 10.1109/IPIN.2013.6817881

[12] Yang, B., Lu, Y., Wang, J., Zhang, Y., & Ma, Y. (2017, July). An improved RBF neural network algorithm to mitigate the distance error based on RSSI. In Control Conference (CCC), 2017 36th Chinese (pp. 3759-3764). IEEE. DOI: 10.23919/ChiCC. 2017.8027945

[13] Wu, X., Kumar, V., Quinlan, J.R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G.J., Ng, A., Liu, B., Philip, S.Y. and Zhou, Z.H., 2008. Top 10 algorithms in data mining. Knowledge and information systems, 14(1), pp.1-37. DOI: 10.1007/ s10115-007-0114-2

[14] L. Breiman, “Random Forests,” Machine Learning, vol. 45, no. 1, pp. 5–32, 2001. DOI: 10.1023/A:1010933404324

[15] Ali, J., Khan, R., Ahmad, N., & Maqsood, I. (2012). Random forests and decision trees. International Journal of Computer Science Issues (IJCSI), 9(5), 272. ISSN (Online): 1694-0814

[16] Pal, M. (2005). Random forest classifier for remote sensing classification. International Journal of Remote Sensing, 26(1), 217-222. DOI: 10.1080/01431160412331269698

[17] Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R news, 2(3), 18-22. Available at:https://www.researchgate.net/profile/Andy_Liaw/ publication/228451484_Classification_and_Regression_by_RandomForest/links/ 53fb24cc0cf20a45497047ab/Classification-and-Regression-by-RandomForest.pdf

[18] Zhang, M. L., & Zhou, Z. H. (2007). ML-KNN: A lazy learning approach to multi-label learning. Pattern recognition, 40(7), 2038-2048. Available at: https:// www.sciencedirect.com/science/article/pii/S0031320307000027

[19] Chen, L., Li, B., Zhao, K., Rizos, C., & Zheng, Z. (2013). An improved algorithm to generate a Wi-Fi fingerprint database for indoor positioning. Sensors, 13(8), 11085-11096. DOI: 10.3390/s130811085

[20] Liang, X., Gou, X., & Liu, Y. (2012, September). Fingerprint-based location positoning using improved KNN. In Network Infrastructure and Digital Content (IC-NIDC), 2012 3rd IEEE International Conference on (pp. 57-61). IEEE. DOI: 10.1109/ICNIDC.2012.6418711

[21] Li, Baoli, Shiwen Yu, and Qin Lu. "An improved k-nearest neighbor algorithm for text categorization." arXiv preprint cs/0306099 (2003). Arxiv ID: cs/0306099