Study on WiFi-based Indoor Positioning Prediction using Machine Learning Techniques

Kyounghyun Park
Mobility UX
Research Section
ETRI
Daejeon, Korea
hareton@etri.re.kr

Yangkoo Lee
Mobility UX
Research Section
ETRI
Daejeon, Korea
yk_lee@etri.re.kr

Seonghun Seo
Mobility UX
Research Section
ETRI
Daejeon, Korea
SSH@etri.re.kr

Min Jung Kim
Mobility UX Research
Section
ETRI
Daejeon, Korea
minjkim@etri.re.kr

Giyoung Lee
Mobility UX
Research Section
ETRI
Daejeon, Korea
giyoung@etri.re.kr

Daesub Yoon
Mobility UX
Research Section
ETRI
Daejeon, Korea
eyetracker@etri.re.kr

JaeJun Yoo
Mobility UX
Research Section
ETRI
Daejeon, Korea
jjryu@etri.re.kr

***Abstract*— With the proliferation of smartphones and advancements in artificial intelligence, WiFi-based indoor positioning technology continues to evolve. The fingerprinting approach generates a fingerprint map using RSSI (Received Signal Strength Indicator) from WiFi Access Points (APs) for indoor positioning. However, WiFi signals are susceptible to environment, leading to the challenge of rebuilding the fingerprint map whenever the environment changes. Machine learning techniques can overcome the drawback of the fingerprint method and therefore enhance indoor positioning accuracy. Ultimately, machine learning techniques can improve the accuracy, cost-effectiveness, and scalability of the indoor positioning system compared to traditional statistical methods. In this paper, we examine representative machine learning algorithms applicable to indoor positioning system and discuss the performance of the algorithms.**

Keywords*— *indoor positioning, machine learning, RSSI, WiFi

I. Introduction

With the advancement of WiFi technology and smartphones, many studies are being conducted on Indoor Positioning Systems [1][2][3]. Among them, WiFi, UWB (Ultra-Wideband), BLE (Bluetooth Low Energy) have been used for indoor positioning. WiFi-based positioning, although relatively more prone to positioning errors compared to UWB or RFID (Radio Frequency Identification), allows for easy positioning without additional devices, as most buildings already have WiFi access points (APs).

One of WiFi-based approaches is fingerprinting. The fingerprinting involves two phases: constructing a fingerprint map and utilizing it for positioning using RSSI (Received Signal Strength Indicator) received from multiple WiFi APs. This technique estimate the user's position by comparing the signal pattern with the fingerprint map. The fingerprinting technique using WiFi relies on both MAC(Media Access Control) and RSSI information at that point. However, a drawback of this approach is that we need to construct a fingerprint map whenever the surrounding environment changes. Machine learning techniques can deal with this issue and, furthermore, improve the performance of the positioning system.

As smartphones become ubiquitous and enable the collection of large amounts of data, machine learning algorithms can effectively learn from large-scale datasets, leading to more accurate and reliable location prediction. Consequently, machine learning techniques for indoor positioning offer various advantages such as accuracy, flexibility, cost-effectiveness, and scalability compared to traditional statistical methods.

In this paper, we study machine learning techniques for indoor positioning and examine the performance of the algorithms. The structure of the paper is as follows. Firstly, we discuss WiFi-based indoor positioning technologies and machine learning algorithms. Next, we briefly describe the indoor positioning system (IPS) and data model, and examine the performance of the machine learning algorithms. Finally, we summarize our study and describe the directions for the further research.

II. Related Work

Indoor positioning approaches using WiFi can be categorized into four types: Angle of Arrival (AoA), Time of Arrival (ToA), hybrid, and fingerprinting. Each positioning technology has its own advantages and disadvantages [3]. Firstly, WiFi signals are widely distributed everywhere making them convenient for indoor positioning. Furthermore, WiFi signals are less affected by Non-Line-of-Sight (NLOS) conditions in indoor environments. However, the signal strength is contingent on the hardware composing the mobile device, leading to difference in signal strength across difference devices. It means that these mobile devices, including smartphones, are influenced by factors such as WiFi chips, antennas, hardware drivers, operating systems, etc. As a result, different RSS at the same location due to heterogeneous devices can negatively impact positioning accuracy [4]. WiFi-based positioning can also be classified into active and passive methods. Active positioning involves users actively searching for and collecting signals from nearby access points (APs), while passive positioning uses changes in signal propagation when users move to determine their location. Active positioning generally offers higher accuracy compared to passive positioning.

To achieve robust indoor positioning, we have studied machine learning algorithms. In this section, we describe representative supervised learning algorithms that can be used to indoor positioning. The k-Nearest Neighbors (k-NN) algorithm is one of the simple supervised learning methods that classifies data points in a feature space. k-NN classifies large datasets without extensive training[5]. This method is commonly used in applications such as pattern recognition, test classification, and object recognition. It relies only on the training data for classification [6][7]. In the k-NN, the class of a test data point is determined by examining the classes of the k nearest training data points. To achieve this, the distances to the k nearest neighbors are calculated, and the class assigned to the test data point is based on the majority class among these neighbors [8].

Support Vector Machine (SVM) is the representative high-performance algorithm widely used in many applications[9]. SVM generates a hyperplane in a multi-dimensional space to separate different classes. This hyperplane should be positioned at the most stable point among the farthest members of different classes. SVM provides a solution for how to draw this hyperplane [10]. Features close to the hyperplane are represented as Support Vectors.

Decision Tree (DT) is a supervised learning method that allows for fast and efficient classification of large datasets [11]. This method can be applied in both classification and regression tasks. The DT algorithm generates a tree, consisting of decision nodes, branches, and leaves. The DT is constructed in two steps: tree construction and pruning. Once the tree is generated, the dataset is divided into subsets. The splitting process stops after all subsets belong to the same class. Tree pruning is performed to reduce overfitting and enhance the regression and classification accuracy [12].

Random Forest is a commonly-used ensemble learning method used for classification and regression analysis. During the training process, it creates multiple decision trees to perform classification or regression tasks. Random Forest generates trees randomly, resulting in each tree having different features. The predictions from these trees are uncorrelated, leading to an overall improvement in generalization performance. Since the trees are generated proportionally to the size of the data, Random Forest forms numerous trees, leading to longer computing times during prediction. Additionally, it is difficult to interpret the result because all the generated tree models are too many to examine [13].

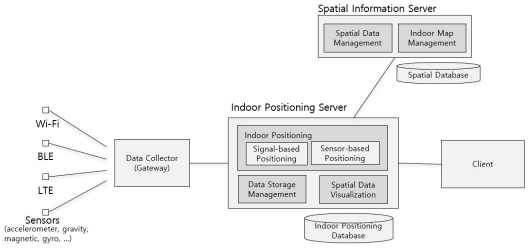

Fig. 1. Indoor Positioning System Architecture

III. Data Management for Indoor Positioning

We designed the indoor positioning system to examine machine learning algorithms and Fig.1 illustrates the architecture of the indoor positioning system [14]. The system consists of data collector, indoor positioning server, spatial information server, and client. The data collector gathers various signals such as WiFi, BLE, and LTE from mobile devices and also collects data from various sensors. The sensor data includes accelerometer, gravity, magnetic, gyro, pressure, and light. The spatial information server is a system that stores and manages spatial data for the maps. The server constructs indoor maps and sends them to the indoor positioning server. The indoor positioning server is the core subsystem which predicts the location using positioning algorithms, supporting both signal-based and sensor-based positioning. The server also manages indoor positioning database and visualizes the map data.

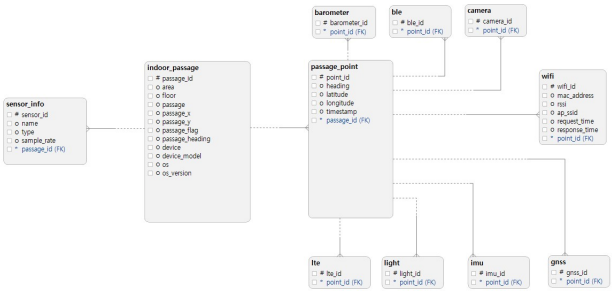

Fig. 2. Indoor Positioning Data Model

The system has databases for indoor positioning data and spatial data. Fig. 2 shows simple data model for indoor positioning and focuses on WiFi data. In the data model, the class "indoor_passage" is the most central class which manages overall information about buildings, floors, and passages. The class "indoor_passage" has multiple point(location) information, having relationships with the “passage_point” class. The class "passage_point" manages location information within passages and has a 1:N relationship with signal and sensor classes (barometer, BLE, camera, WiFi, LTE, light, IMU, GNSS). The class "WiFi" stores the signal strength of WiFi signals and includes properties such as AP's MAC address, RSSI, and timestamp. It establishes a relationship with the class "passage_point"

IV. Data Collection and Preprocessing

We collected data using Samsung Galaxy S21+ in a 7th-floor building, specifically on the 4th floor with 5 passages. Table I shows the number of data collection points per passage. The distance interval between adjacent points is 2m and we collected data 20 times at each location point. The gateway transmits data to the positioning server in the JSON format and the positioning server stores data in the positioning database.

TABLE I. THE NUMBER OF LOCATION POINTS

Floor	Passage No.	the Number of Points
4	1	11
4	2	14
4	3	13
4	4	9
4	5	4

Due to the limited coverage of WiFi signals, we cannot guarantee the detection of all Access Points (APs) when

collecting WiFi data. As a result, It could be that APs exist in the training data but not in the testing data, and vice versa. In that case, we need to deal with missing data. For missing data, we should either remove the data or transform it into appropriate values. In this paper, we used the one-hot encoding technique in the preprocessing step to handle missing and invalid data.


```
{
  "area_name": "ETRI12",
  "base_point_x": 232948.78,
  "base_point_y": 420253.8,
  "current_time": "2023-06-02-16-28-05",
  "device_name": "Galaxy S21+ 5G",
  "floor": "AF04",
  "id": "dfc5bb5cfb86829a",
  "link_flag": "F",
  "link_heading": 360.0,
  "link_id": 5,
  "model_name": "SM-G996N",
  "os_name": "13",
  "os_version": 33,
  "payload": [
    {
      "barometer": [
        {
          "pressure": 995.369873046875,
          "timestamp": 1685690916106
        }
```


Fig. 3. Data Example in the JSON format

V. Indoor Positioning Prediction and Performance Evaluation

We studied the performance of machine learning algorithms for indoor positioning. Among the machine learning algorithms, we tested k-NN, Linear Regression, SVM, Decision Tree, Random Forest, Gradient Boosting, and XGBoost.

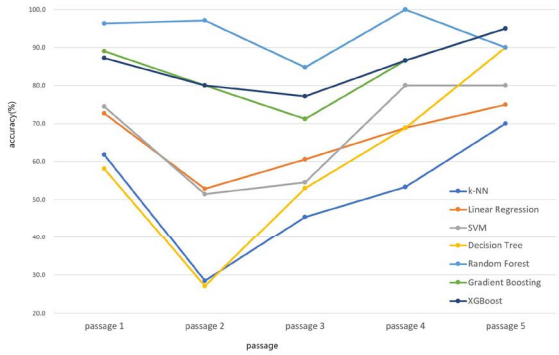

Fig. 4. Accuracy of Machine Learning Algorithms

Fig. 4 represents the accuracy of machine learning algorithms. Accuracy is an important metric that indicates the system's ability to accurately estimate the user's location. The accuracy is calculated as follows [15].

$$\text{Accuracy} = \frac{Number\ of\ correct\ prediction}{Total\ number\ of\ prediction} = \frac{TP+TN}{TP+FP+TN} \quad (1)$$

In Fig. 4, we observed that Random Forest performed the best with a range of 84.8% to 100% at accuracy, followed by the boosting algorithms XGBoost and Gradient Boosting, which also exhibited good performance. Interestingly, the commonly used k-NN algorithm showed relatively lower performance with a range of 28.5% to 70% at accuracy. The reason for the poorer performance of k-NN can be attributed to the high dimension of indoor positioning data. k-NN is highly sensitive to the dimensionality of data and is not well-suited for high-dimensional datasets. As a result, its performance tends to degrade in the case of high-dimensional data such as positioning data. Therefore, the results indicate that Random Forest, XGBoost, and Gradient Boosting outperformed k-NN likely due to their ability to handle higher dimensional data more effectively.

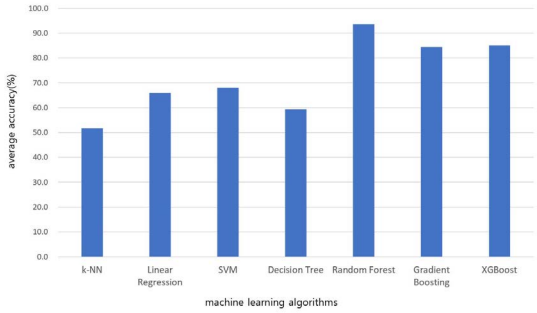

Fig. 5. Average Accuracy

Fig. 5 shows the average accuracy ($A_{avg\text{-}acc}$) of the machine learning algorithms. The average accuracy of an algorithm is calculated by summing up the accuracies ($A(P_i)_{acc}$) for each passage and then dividing it by the total number of passages ($N(P_i)$). In other words, it represents the overall accuracy of the algorithm across all passages, providing a comprehensive measure of its performance.

$$A_{agv-acc} = \frac{\sum_i A(P_i)_{acc}}{N(P_i)} \quad (2)$$

When examining the results by passage, we observe different levels of accuracy for different algorithms across passages. Some passages show better performance, while others show lower accuracy. For example, passage 2 displayed the lowest accuracy, whereas passage 5 demonstrated the highest accuracy.

Furthermore, we could find the specific passages which exhibited significant differences in accuracy depending on the algorithms. For example, in passage 2, the k-NN achieved an accuracy of 28.5%, while the Random Forest achieved an accuracy of 97%, indicating a substantial disparity in accuracy between these two algorithms for this particular passage.

TABLE II. TABLE 1. MIN/MAX ACCURACY

Passage No.	Features	Dataset(row)	minimum		maximum	
			accuracy	classification	accuracy	classification
1	207	220	58.1	Decision Tree	96.3	Random Forest
2	122	280	27.1	Decision Tree	97.1	Random Forest
3	188	261	45.4	k-NN	84.8	Random Forest
4	143	179	53.3	k-NN	100	Random Forest
5	116	80	70	k-NN	95	Gradient Boosting XGBoost

TABLE II. summarizes the performance of algorithms, displaying the minimum and maximum accuracies per passage. We can also find that the Random Forest and boosting families exhibit stronger performance.

Fig. 6 represents localization errors of machine learning algorithms. Localization error is a metric that measures the gap between the user's actual position and the estimated position [11]. As observed in the figure, except for passage 5, Random Forest shows the lowest error, while k-NN and Decision Tree show higher localization errors. Particularly the passage 2 shows a significant gap in errors among algorithms.

Fig. 7 illustrates the average localization error. Overall, it is evident that Random Forest shows lower error values, while Decision Tree and k-NN show the largest errors.

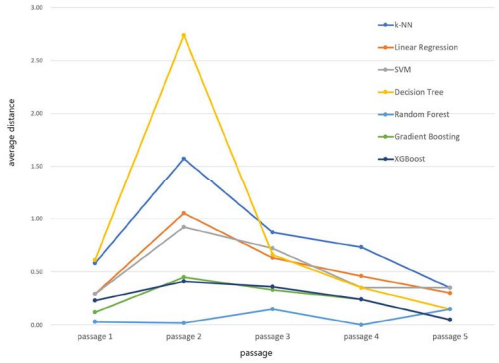

Fig. 6. Localization Error of Machine Learning Algorithms

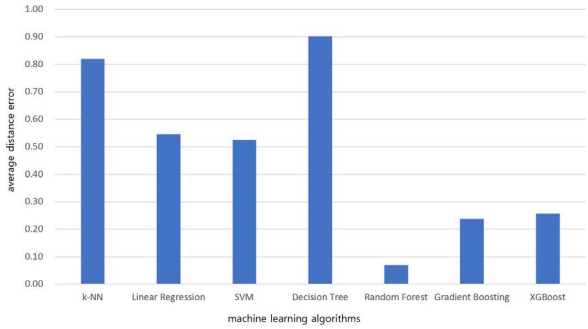

Fig. 7. Average Localization Error

TABLE III. TABLE 2. MIN/MAX LOCALIZATION ERROR

assage N	Features	ataset(rov	minimum		maximum	
			localization error	classification	localization error	classification
1	207	220	0.03	Random Forest	0.61	Decision Tree
2	122	280	0.02	Random Forest	2.74	Decision Tree
3	188	261	0.15	Random Forest	0.87	k-NN
4	143	179	0	Random Forest	0.73	k-NN
5	116	80	0.05	Gradient Boosting	0.35	k-NN SVM

TABLE III summarizes the minimum and maximum localization errors of the passages. In this table, we can also conclude that Random Forest and Gradient Boosting shows the favorable performance among the algorithms.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have studied machine learning algorithms for indoor positioning using WiFi data. The experimental results revealed that the performance of the Random Forest and boosting algorithms was favorable, while the k-NN and Decision Tree algorithms did not perform well. In this experiment, the dataset was insufficient considering the number of features, and adequate preprocessing, such as normalization, was not carried out. Hence, it is essential to enhance the experimental environment and then proceed with the analysis of machine learning algorithms. In addition, we focused on WiFi data points. In the future research, we need to extend to composite data, including signal data and sensor data. This would involve working with composite data from various sources within the indoor positioning system, necessitating a more comprehensive analysis of algorithm performance.

ACKNOWLEDGMENT

This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant RS-2022-00141819).

REFERENCES

[1] L. Hui, H. Darabi, P. Banerjee, J. Liu, “Survey of wireless indoor positioning techniques and systems”, IEEE Trans. Syst. Man Cybern. C 37 (6) (2007) 1067–1080.

[2] Y. Gu, A. Lo, I. Niemegeers, “A survey of indoor positioning systems for wireless personal networks”, IEEE Commun. Surv. Tutor. 11 (1) (2009) 13–32.

[3] C. Yang and H.-r. Shao, “WiFi-based indoor positioning,” IEEE Communications Magazine, vol. 53, no. 3, pp. 150–157, Mar. 2015

[4] F. Liu et al., “Survey on WiFi‐based indoor positioning techniques,” IET Communications, vol. 14, no. 9, pp. 1372‐1383, Jun. 2020

[5] J. Kim, B.-S. Kim, and S. Savarese, "Comparing image classification methods: K-nearest-neighbor and support-vector-machines," Ann Arbor, vol. 1001, pp. 48109-2122, 2012.

[6] N. Bhatia, "Survey of nearest neighbor techniques," arXiv preprint arXiv:1007.0085, 2010.

[7] N. Bhatia, Vandana, "Survey of nearest neighbor techniques," International Journal of Computer Science and Information Security, vol.8, no. 2, pp. 302-305, 2010.

[8] N. Suguna and K. Thanushkodi, "An improved k-nearest neighbor classification using genetic algorithm," International Journal of Computer Science Issues, vol. 7, no. 2, pp. 18-21, 2010.

[9] A. Ben-Hur and J. Weston, "A user's guide to support vector machines,",in Data mining techniques for the life sciences: Springer, 2010, pp. 223-239.

[10] N. Reljin and D. Pokrajac, "Classification of performers using support vector machines," in Neural Network Applications in Electrical Engineering, 2008. NEUREL 2008. 9th Symposium on, 2008, pp. 165-169

[11] D. M. Farid, L. Zhang, C. M. Rahman, M. A. Hossain, and R. Strachan, "Hybrid decision tree and naive Bayes classifiers for multi-class classification tasks," Expert Systems with Applications, vol. 41, no. 4, pp.1937-1946, 2014.

[12] A. Priyama, R. G. Abhijeeta, A. Ratheeb, and S. Srivastavab, "Comparative analysis of decision tree classification algorithms," International Journal of Current Engineering and Technology, vol. 3, no.2, pp. 334-337, 2013.

[13] M. Rathnayake, P. Maduranga, “RSSI and Machine Learning-Based Indoor Localization Systems for Smart Cities”, 3rd KDU-FOC Student Symposium, pp.1468-1494, Jan. 2023.

[14] Y. Lee, M. Kim, G. Lee, J. Yoo, D. Yoon, “Integrated Raw Data Collection and Validation System for Indoor Positioning”, International Conference on Electrical, Computer, Communications and Mechatronics Engineering(ICECCME), pp.1-6, 2023.

[15] Obeidat, H., Shuaieb,W., Obeidat, O., Abd-Alhameed, R, “A Review of Indoor Localization Techniques andWireless Technologies-Wireless Personal Communications”, Wireless Personal Communications, pp.289-327, February 2021