Enabling accurate indoor localization for different platforms for smart cities using a transfer learning algorithm

Received: Added at production | Revised: Added at production | Accepted: Added at production
DOI: xxx/xxxx

ARTICLE TYPE

Enabling Accurate Indoor Localization for Different Platforms for Smart Cities Using a Transfer Learning Algorithm†

Halgurd S. Maghdid[1] | Kayhan Zrar Ghafoor*[2,3] | Abdulbasit Al-Talabani[1] | Ali Safaa Sadiq[3] | Pranav Kumar Singh[4] | Danda B. Rawat[5]

[1]Department of Software Engineering, Faculty of Engineering, Koya University, Kurdistan Region-F.R.Iraq. First.Last@koyauniversity.org.
[2]Department of Software Engineering, Salahaddin University-Erbil, Iraq kayhan@ieee.org.
[3]School of Mathematics and Computer Science, University of Wolverhampton, Wulfruna Street, Wolverhampton, WV1 1LY, UK; Centre for Artificial Intelligence Research and Optimisation, Torrens University Australia, Fortitude Valley, Brisbane, 4006 QLD, Australia. Email: Ali.Sadiq@wlv.ac.uk
[4]Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Assam, 781039, India. snghprana@gmail.com.
[5]Department of Electrical Engineering & Computer Science, Howard University, Washington, DC, 20059, USA. db.rawat@ieee.org

Correspondence
*Kayhan Zrar Ghafoor, Department of Software Engineering, Salahaddin University-Erbil, Iraq; School of Mathematics and Computer Science, University of Wolverhampton, Wulfruna Street, Wolverhampton, WV1 1LY, UK. Email: kayhan@ieee.org

Present Address
This is sample for present address text this is sample for present address text

Summary

Indoor localization algorithms in smart cities often use Wi-Fi fingerprints as a database of Received Signal Strength (RSS) and its corresponding position coordinate for position estimation. However, the issue of fingerprinting is the use of different platform-devices. To this end, we propose a Long Short-Term Memory (LSTM)-based novel indoor positioning mechanism in smart city environment. We used LSTM, a type of recurrent neural network to process sequential data of users' trajectory in indoor buildings. The proposed approach first utilizes a database of normalizing fingerprint landmarks to calculate WiFi Access Points (WAPs) RSS values to mitigate the fluctuation issue and then apply the normalization parameters on the RSS values during the online phase. Afterwards, we constructed a transfer model to adapt the RSS values during the offline phase and then applying it on the RSS values from the different smartphones during the online phase. Thorough simulation results confirm that the proposed approach can obtain 1.5 to 2 meters positioning accuracy for indoor environments, which is 60 % higher than traditional approaches.

KEYWORDS:
Deep Learning, Indoor Positioning, Transfer Learning, WiFi Signal

1 | INTRODUCTION

Wi-Fi is widely used as an infrastructure to position users when they are indoors in smart cities. Particularly, Wi-Fi fingerprint-based localization utilize channel status and RSS to create fingerprint repository, which contains the exact position of pre-selected locations and corresponding RSS of the nearby AP[1,9]. This phase of fingerprinting-based technique is called offline training. Then, machine learning algorithms such as k-Nearest Neighbour (k-NN) can be applied on collected RSS and fingerprint repository to estimate the position of target positions. Although fingerprint-based localization has many useful features, creating fingerprint landmarks for a large area is at the expenses of a significant amount of time and human resources[6]. Further, multi-path propagation (fluctuation) of RSS readings due to movable objects, complex indoor environment structures[6], large area or multi-building datasets for the survey[7], using different device-platforms to collect RSS values[8], and time efficiency of using different matching-algorithms are the main challenges of using RSS-based fingerprinting technique.

Besides k-NN, deep learning is widely used as a machine learning technique for indoor positioning. This domain of research has attracted many researchers in both academia and industry. With this technique, the accuracy of indoor positioning can be improved as it can efficiently handle sequential datasets and associate RSS values with the geolocation positions[2,10]. More precisely, Long Short-Term Memory (LSTM) model is able to find long-term correlation in the Wi-Fi RSS and utilize them in position predictions. LSTM architecture is efficient for feature classification and prediction based on time-series analysis. Thus, LSTM can make an accurate prediction of people trajectory when they are indoors and use the same device platform[3,13].

In indoor positioning environment, however, it is known that data can be frequently outdated because collected data in a specific time interval is not equivalent to the same distribution in the subsequent period. Further, it is very costly to collect RSS values to build positioning in large-scale scenarios, because it requires a large lookup table of RSS values with exact ground locations. This process needs to be repeated when the distribution of RSS values is varied with time and type of device (such as iPhone and Huawei devices). For instance, using deep learning to train a model in one device and in a specific time interval may lead to low accuracy of position estimation in different time period and device. Distribution of Wi-Fi RSS values at two different time intervals are not the same even if the mobile device remains in the same position[4,5].

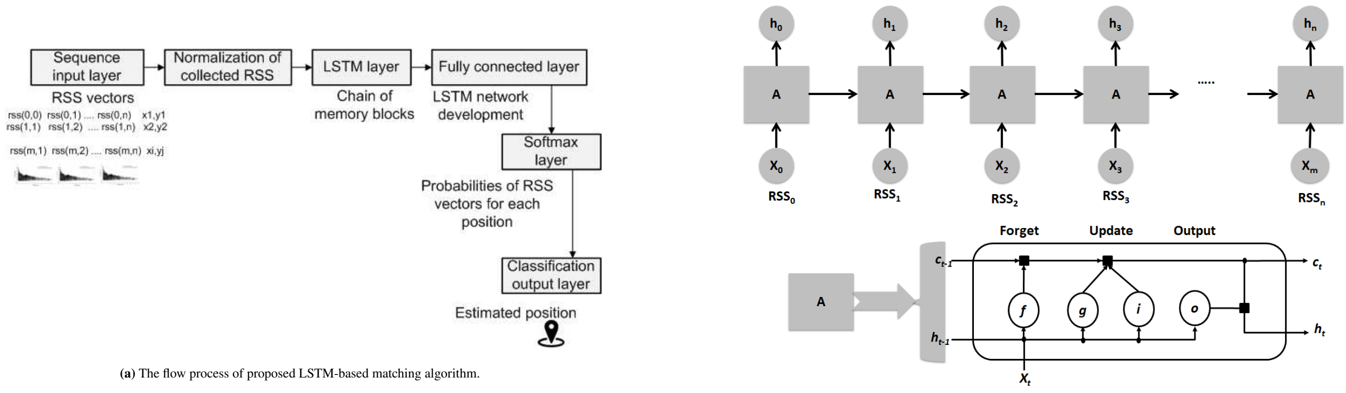
However, the issue of fingerprinting technique, which has not been well addressed in the literature, is the use of different platform-devices to train and test the dataset. This is because; using different platform-devices will measure the RSS readings very differently for many reasons. It is noteworthy that the term of different platform-devices refers to the fact that the devices use different API-operating system, measurement criteria and materials. These different platform-devices are a realist demand of indoor positioning solutions. Christos et al.[8] proposed a new calibration method of the RSS readings to mitigate such issue. The calibration method is called self-calibration which provides better mapping between the reference device and the tested devices by using the histograms of the RSS records. Further, the calibration method doesn't need any user interaction due to updating histogram RSS values concurrently during the positioning process. Although their proposed method provided a comprehensive comparison among a set of existing calibration methods, the evaluated methods did not offer enough positioning accuracy. Transferred learning model (TLM), as a new approach, has been mentioned as a future trend in[7,1] to reduce the effectiveness of using different platform-devices on fingerprinting technique.

In this paper, we first review the various efforts made to improve the accuracy of the target trajectory in a dynamic indoor environment. Then, we present an LSTM-based novel indoor positioning mechanism. We used LSTM, which is a type of recurrent neural network, to process sequential data of peoples trajectory in indoor buildings. We utilize a database of fingerprint landmarks to calculate WAPs RSS values that are normalized to mitigate the fluctuation issue and apply the normalization parameters on the RSS values during the online phase. Then, we constructed a model that transfer extracted parameters during the offline phase to adapt the RSS values when applying on the RSS values from the different smartphones during the online phase. Simulation results confirm that the proposed approach can obtain 1.5 to 2 meters positioning accuracy for indoor environments, which is 60 % higher than traditional protocols.

The rest of this paper is arranged as follows: Section 2 highlight an overview of the proposed protocol and also discusses the detail of designed algorithm. This is followed by performance validation and evaluation in section 3, where we highlight the feasibility of our proposed approach by considering different platform-devices. Finally, section 4 concludes the paper.

[†]This is an example for title footnote.

[0]**Abbreviations:** ANA, anti-nuclear antibodies; APC, antigen-presenting cells; IRF, interferon regulatory factor



(a) The flow process of proposed LSTM-based matching algorithm.

(b) The proposed architecture of LSTM Layer.

FIGURE 1 The flow process and architecture of the proposed approach.

2 | PROPOSED APPROACH DESIGN

This section presents the motivation and the detailed algorithm of the proposed approach. The proposed approach elaborates the process of creating a transfer learning model[7], which includes normalization of fingerprint landmarks and design of LSTM layers.

Recently, a new challenge of RSS-based localization techniques was raised, which is related to the materials and the positioning techniques, together. Particularly, when a fingerprinting technique is used, the smartphones used to construct the RSS dataset should be the same as the smartphones for testing during the online phase. Otherwise, the accuracy of the calculated position will be worse. Therefore, to overcome this issue, we propose a new approach to transfer or adapt the nature of the calculated RSS values among multi-smartphones. The proposed approach adopts the transfer learning model to estimate the accurate positions. The model is trained based on different type of smartphone measurements using a machine learning algorithm. In this paper, a RNN based model is adopted, specifically long-short-term-memory (LSTM) algorithm as a transfer learning model. This is to mitigate the issue of multi-platform devices.

The design of the LSTM is illustrated in Fig. 1a. As noted, the process of creating the transfer learning model includes two main steps. In the first step, collected RSS values are normalized at every known nearby positions by subtracting the mean of the device(s) involved in training the model and dividing by standard deviation of the same device(s). Further, the sequence in the input layer, which is a set of RSS values of each reference position (fingerprint), will be input into the network. Afterwards, the LSTM layer develops long-range connections between the RSS values of the sequence data. For more understanding of the LSTM layer, Fig. 1b depicts the structure of LSTM model. Further, the cell and hidden states calculations are expressed in equations 1 and 2, respectively.

$$c_t = f(t) \odot c_{t-1} + i_t \odot g_t \tag{1}$$

$$h_t = o_t \odot \sigma_c(c(t)) \tag{2}$$

where $\odot$ indicates the Hadamard product, σ_c indicate the state activation function which is the hyperbolic tangent function. For this study, i_t is the input gate, f_t is the forget gate, c_t is the cell candidate gate, and finally o_t is the output gate.

The LSTM layer is, then, connected to the fully Connected Layer. This layer is responsible to provide connection between their nodes and all nodes in the LSTM layer. Each node in this layer applied an activation function on a linear combination of all the features learned by the previous layers across the RSS-vector to identify the larger patterns, where the parameters of the linear combination representation are the set of weights at each layer. The last fully connected layer maps the node outputs into a number of nodes equal to the number of classes available in the classification problem. In other words, it combines the features to classify the RSS vector to the label or a known position. Therefore, the output size parameter in the last fully connected layer is equal to the number of classes in the target data. However, due to having a multi-classes classification, i.e., multi-positions in the area, the Softmax function is applied to compute the probabilities for each class (each position) to the output layer from its network inputs[11].

3 | PERFORMANCE EVALUATION

3.1 | Dataset & Simulation Setup

We have used the public KIOS WiFi RSS dataset[8]. The KIOS dataset includes the collected WiFi RSS values from the KOIS research center at the University of Cyprus. The area of KIOS research center is around 560 m2, and it has a complex indoor structure which is composed of labs, conference rooms, corridors, private offices, and open-cubical-style. This dataset is ideal for testing our proposed approach. This is because the RSS values are collected from five different WiFi-enabled commercial mobile-devices within different platforms including three Android-based smartphones (namely HTC Desire, Samsung Nexus S, and HTC Flyer tablet), HP iPAQ hw6915 PDA with Windows Mobile, and an Asus eeePC T101MT laptop running Windows 7. In our experiments, to simplify mobile device names, we have called them as follow: device1, device2, device3, device4, and device5. Further, the dataset is divided into two parts: train dataset and test dataset. The train dataset contains 2100 records of RSS values within the labeled position from 9 WAPs in the vicinity. While the test dataset includes 960 records of RSS values, again, from 9 WAPs, both datasets are generated by each mobile-device, separately.

The performance of the proposed transfer learning model using LSTM was evaluated via the MATLAB toolbox. Three different scenarios are tested where LSTM and k-NN are used as matching algorithms. The first scenario uses matching algorithms without applying the RSS normalization parameters (mean and standard deviation). The second scenario uses matching algorithms by conducting the RSS normalization process. Note: in both scenarios, the train datasets of all the devices are contributed to the train model, we named main train dataset. This is followed by testing each device, using its own test dataset, separately. While the third scenario is to test the devices, however, when the device is tested, the train dataset of that device is excluded. This is to show how the performance of the LSTM offers positioning accuracy, even the test device is not used during the offline phase, i.e., during constructing train dataset. Thus, this is a real demand for the current indoor positioning solutions.

3.2 | Simulation Setup

The performance of the proposed transfer learning model using LSTM was evaluated via the MATLAB toolbox. The size of sequence input is initialized to 9, since 9 WAPs RSS values are available at each position in the vicinity. Since the number of the fingerprints (or the number of labelled positions) during the survey of the building is equal to 105 positions, therefore, the number of classes is also initialized to 105 classes. The Adam activation function is used for all input, hidden, and output nodes). The learning rate, number of hidden of nodes, and the number epochs are configured to 0.01, 180, and 300, respectively.

To show the performance of the proposed approach (specifically to test the matching algorithm), we also test the same datasets by using k-NN technique. With the k-NN technique, the number of neighbours k is initialized to 3, as a rule of thumb and according to[12], that setting the values of k to 3 is more reliable for the classification purpose.

Three different scenarios are tested where LSTM and k-NN are used as matching algorithms. The first scenario uses matching algorithms without applying the RSS normalization parameters (mean and standard deviation). The second scenario uses matching algorithms by conducting the RSS normalization process. Note: in both scenarios, each device has its own train and test dataset, and they are used separately. While the third scenario is to test the devices, all train data sets are combined into a main dataset, however, when the device is tested, the train dataset of that device is excluded. This is to show how the performance of the LSTM offers positioning accuracy, even the test device is not used during the offline phase, i.e., during constructing the main train dataset. Thus, this is a real demand for the current indoor positioning solutions.

3.3 | Simulation Results

Positioning accuracy is the most reliable metric to evaluate positioning solutions. There are several methods to calculate the accuracy such as root-mean-square-error (RMSE), absolute-mean-error (AME), Median of the absolute mean error (MAME), and Median of the absolute maximum error (MAME). In this study, MAME is selected to calculate the positioning accuracy due to the comparison issue with the state-of-art solutions.

Table 1(left) shows the experiment results for the first scenario. The average positioning accuracy obtained for all the devices tested, when LSTM is used, is worse than the k-NN results because the RSS fluctuation issue from the trained dataset is not adopted. Also, the k-NN technique only works on measuring the similarity of RSS vectors; therefore, positioning accuracy via k-NN is better.

TABLE 1 Positioning accuracy of proposed approach and k-NN algorithm when normalization is not (left) and applied (middle) as well as positioning accuracy using the proposed approach when normalization is applied and the train dataset of the test device is excluded (right)

Using LSTM		Using Test Data set of the devices				
		Device 1	Device 2	Device 3	Device 4	Device 5
Using Training data set of the devices	Device 1	3.65	3.37	3.51	5.06	4.13
	Device 2	1.99	4.40	2.89	4.20	3.78
	Device 3	5.68	5.67	2.37	3.32	2.96
	Device 4	3.03	2.94	2.60	2.22	2.59
	Device 5	3.68	5.08	4.01	4.55	5.40
Using KNN		Using Test Data set of the devices				
		Device 1	Device 2	Device 3	Device 4	Device 5
Using Training data set of the devices	Device 1	1.442	2.117	1.505	4.223	1.6119
	Device 2	2.279	1.4594	1.88	3.193	1.8686
	Device 3	1.484	1.8511	1.214	3.931	1.6378
	Device 4	6.27	4.3775	5.214	1.538	5.4292
	Device 5	1.564	1.6791	1.726	4.207	1.3487

Using LSTM		Using Test Data set of the devices				
		Device 1	Device 2	Device 3	Device 4	Device 5
Using Training data set of the devices	Device 1	1.54	1.76	1.53	1.79	1.71
	Device 2	1.58	1.50	1.40	1.49	1.58
	Device 3	1.39	1.64	1.23	1.79	1.63
	Device 4	1.51	1.70	1.81	1.67	1.46
	Device 5	1.46	1.65	1.72	1.73	1.46
Using KNN		Using Test Data set of the devices				
		Device 1	Device 2	Device 3	Device 4	Device 5
Using Training data set of the devices	Device 1	1.44	1.65	1.53	1.71	1.45
	Device 2	1.51	1.46	1.38	1.65	1.56
	Device 3	1.35	1.56	1.21	1.78	1.50
	Device 4	1.43	1.63	1.75	1.54	1.52
	Device 5	1.41	1.54	1.71	1.65	1.35

Using LSTM		Using Test Data set of the devices				
		Device 1	Device 2	Device 3	Device 4	Device 5
Using Training data set of the all devices, except:	Device 1	**1.58**	1.49	1.36	1.74	1.34
	Device 2	1.43	**1.67**	1.25	1.76	1.26
	Device 3	1.56	1.43	**1.64**	1.75	1.39
	Device 4	1.47	1.46	1.31	**2.01**	1.35
	Device 5	1.42	1.52	1.36	1.74	**1.52**
Using KNN		Using Test Data set of the devices				
		Device 1	Device 2	Device 3	Device 4	Device 5
Using Training data set of the all devices, except:	Device 1	**1.6316**	1.4857	1.4117	1.5712	1.3664
	Device 2	1.4123	**1.8879**	1.2683	1.5826	1.4027
	Device 3	1.5416	1.5041	**1.6612**	1.5734	1.3858
	Device 4	1.4959	1.4991	1.3358	**3.6284**	1.4377
	Device 5	1.5156	1.5065	1.2917	1.5888	**1.6363**

To show how the normalization helps the LSTM training model, we experimented with the second scenario. The result of the second scenario is presented in Table 1(middle). The accuracy obtained by both LSTM and KNN is better than the first scenario, as it avoids the problem of fluctuation of the RSS feed through the normalization process.

To present the promised LSTM result when the train data set is not included in the main train dataset, we experimented with the third scenario in comparison with k-NN. The positioning accuracy of this scenario for both applied LSTM and KNN is shown in Table 1 (in the right). As can be seen, the LSTM always offers enough positioning accuracy in comparison to k-NN. Note: the maximum error of the LSTM is up to 2.01 meters while the maximum error via kNN is around 3.6 meters. This is because k-NN will lose the nearest RSS records when the training dataset of the test devices is excluded. However, the exclusion of the train datasets of the test devices does not affect on the learned model via the LSTM algorithm.

To further validate the performance of the proposed approach, we compare the accuracy measure obtained via LSTM with the results in [8]. In [8], several calibration methods have been tested. We have chosen the modified self-calibration (SCmed) method, since it offers better positioning accuracy among existing methods. The comparison is presented in Fig. 2. Both Device1 and Device2 are tested and their train datasets are excluded. The selection of these two devices is referred to in the experimental setup of [8]. As it is shown in Fig. 2, the average positioning accuracy of SCmen is up to 2.5 meters. In contrast, benefiting from transfer learning and normalization, the proposed approach achieves accurate positioning estimation, up to 1.5 meter. This

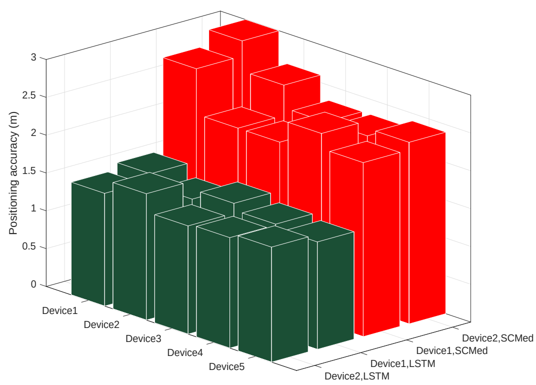


FIGURE 2 Positioning accuracy of the proposed approach and the SCMen when the first and second devices are tested.

result illustrates that our proposed approach-based on LSTM provides 60 % (equivalent to 1 meter) less positioning error. This improvement will help indoors positioning solutions to be more applicable, commercially.

4 | CONCLUSION

we have shown that the positioning error increases when the fingerprinting technique is utilized in different platform-devices to train and test the datasets. Further, RSS measurements by different platform-devices are most likely differ from each other for the same position and time. To overcome this limitation, we propose an LSTM-based novel indoor positioning mechanism for accurate position estimation. The proposed approach uses normalized fingerprint landmarks to calculate APs RSS values to address the problem of RSS fluctuation and then apply the normalization parameters on the RSS values during the online phase. Then, the proposed approach incorporates with a transfer model to match the RSS values in the offline phase to RSS values of different devices in the online phase. Through extensive simulations, we have shown an accuracy gain, up to 1.5 to 2 meters, of the proposed approach. Because the proposed approach uses transfer learning and normalized fingerprint points, it performs well in a realistic indoor environment when different platform-devices are used.

References

1. HS Maghdid, KZ Ghafoor, A Al-Sherbaz, L Kong, N Al-Jawad, Cooperative Strategies for Localization and Navigation, Cooperative Localization and Navigation: Theory, Research, and Practice, 2019, 173.
2. A. Sahar and D. Han, “An lstm-based indoor positioning method using wi-fi signals,” in Proceedings of the 2nd International Conference on Vision, Image and Signal Processing. ACM, 2018, p. 43. 5
3. G. Hussain, M. S. Jabbar, J.-D. Cho, and S. Bae, “Indoor positioning system: A new approach based on lstm and two stage activity classification,” Electronics, vol. 8, no. 4, p. 375, 2019.
4. S. J. Pan and Q. Yang, “A survey on transfer learning,” IEEE Transactions on knowledge and data engineering, vol. 22, no. 10, pp. 1345–1359, 2010.
5. H. S. Maghdid, I. A. Lami, K. Z. Ghafoor, and J. Lloret, “Seamless outdoors-indoors localization solutions on smartphones: implementation and challenges,” ACM Computing Surveys (CSUR), vol. 48, no. 4, p. 53, 2016.
6. R. Gorak and M. Luckner, “Automatic detection of missing access points in indoor positioning system,” Sensors, vol. 18, no. 11, p. 3595, 2018.
7. M. Mohammadi and A. Al-Fuqaha, “Enabling cognitive smart cities using big data and machine learning: Approaches and challenges,” IEEE Communications Magazine, vol. 56, no. 2, pp. 94–101, 2018.
8. C. Laoudias, R. Piche, and C. G. Panayiotou, “Device self-calibration in location systems using signal strength histograms,” Journal of Location Based Services, vol. 7, no. 3, pp. 165–181, 2013.
9. M. Ibrahim, M. Torki, and M. ElNainay, “Cnn based indoor localization using rss time-series,” in 2018 IEEE Symposium on Computers and Communications (ISCC). IEEE, 2018, pp. 01 044–01 049.
10. Y. Lukito and A. R. Chrismanto, “Recurrent neural networks model for wifi-based indoor positioning system,” in 2017 International Conference on Smart Cities, Automation & Intelligent Computing Systems (ICONSONICS). IEEE, 2017, pp. 121–125.
11. L. Wan, M. Zeiler, S. Zhang, Y. Le Cun, and R. Fergus, “Regularization of neural networks using dropconnect,” in International conference on machine learning, 2013, pp. 1058–1066.
12. H. S. Maghdid, “Web news mining using new features: A comparative study,” IEEE Access, vol. 7, pp. 5626–5641, 2019.
13. K. Z. Ghafoor, M. Guizani, L. Kong, H. S. Maghdid and K. F. Jasim, “Enabling Efficient Coexistence of DSRC and C-V2X in Vehicular Networks,” IEEE Wireless Communications, vol. 27, no. 2, pp. 134-140, April 2020.

How to cite this article: Williams K., B. Hoskins, R. Lee, G. Masato, and T. Woollings (2016), A regime analysis of Atlantic winter jet variability applied to evaluate HadGEM3-GC2, *Q.J.R. Meteorol. Soc.*, *2017;00:1–6*.