A Bayesian Optimization Based Deep Learning Model for Wi-Fi Fingerprinting based Indoor Positioning

Duc Khoi Nguyen [1], Le Cuong Nguyen [2], and Manh Kha Hoang [1,*]

[1] Faculty of Electronics Engineering, Hanoi University of Industry, Hanoi, Vietnam
[2] Faculty of Electronic and Telecommunications, Electric Power University, Hanoi, Vietnam
Email: khoind@haui.edu.vn (D.K.N); cuongnl@epu.edu.vn (L.C.N); khahoang@haui.edu.vn (M.K.H)
*Corresponding author

Abstract—**Location-based services for various indoor applications are often built upon the results offered by indoor positioning systems. Among many positioning approaches, Wi-Fi received signal strength indicator fingerprinting based techniques are of particular interest because of wide Wi-Fi network deployment in many indoor environments. With the rapid development of computing resources, many deep learning models have been proposed for determining indoor mobile objects showing their superiorities compared to traditional models. However, the hyperparameter tuning procedure for obtaining the most suitable model is very challenging and time-consuming. To relax this circumstance, this paper presents a utilization of Bayesian Optimization to reach the best hyperparameters of a long short-term memory regression model for indoor positioning solutions. In addition, a combination of two well-known dimensionality reduction techniques namely Truncated Singular Value Decomposition and Linear Discriminant Analysis is proposed to enhance the positioning accuracy. The results produced on a public dataset show a considerable improvement of the proposed solution over the others in terms of positioning accuracy, i.e., the mean distance error improved by 3%, 9%, and 24% compared to three state-of-the-art studies.**

Keywords—**indoor positioning, long short-term memory, Bayesian optimization, dimensionality reduction**

I. Introduction

Recently, Location-Based Services (LBS) have played an increasingly important role in many applications such as logistics, people tracking, advertisement, etc. Since the LBS are built upon the object location, various positioning systems have been developed to enhance the accuracy of location estimation. For outdoor positioning, satellite based positioning systems have demonstrated their obvious advantages compared to other techniques in supporting outdoor LBS systems. However, most human activities are in indoor environments where satellite signals are often not available. Therefore, developing indoor positioning schemes has attracted considerable interest from researchers [1–5].

Many indoor positioning approaches have been proposed by utilizing various signal sources such as radio signals, visible light, acoustics, inertial sensors, etc. [1–6]. A very challenging task when developing a positioning solution is obtaining an accurate estimated position while maintaining the system at a low cost. Among the existing approaches, Wi-Fi Received Signal Strength Indicator (RSSI) fingerprinting based techniques are of particular interest because of the widespread deployment of Wi-Fi networks in many indoor environments as well as personal Wi-Fi devices [7]. Wi-Fi RSSI fingerprinting based methods operate in two phases: offline training phase and online positioning phase. In the training phase, Wi-Fi RSSI data are collected at the Reference Points (RPs) in the deployment area from nearby Wi-Fi Access Points (APs) to establish the radio map. During the positioning phase, by comparing the online observed data with the training data, the object position is determined based on the similarity between them. Although this scheme preserves the requirement of low cost for most civil indoor positioning systems, achieving accurate positioning remains an open challenge for researchers and engineers.

Traditional indoor fingerprinting based methods often make use of deterministic classification techniques, e.g., k nearest neighbors [8] or probabilistic based approaches [9–11]. Because of the fluctuations of the Wi-Fi RSSI data in the indoor environment, the latter seems to outperform the former because it can efficiently take this phenomenon into account. One of the major problems of traditional methods is that their computational time for delivering positioning estimates changes monotonically with the deployment area of the indoor positioning systems. This could lead to unexpected positioning errors caused by the computational time and the movement of the mobile object in real-time applications. Therefore, besides the requirement of keeping the positioning systems at a low cost, all real-time applications also require short execution time to ensure the system scalability while achieving accurate location estimates.

Manuscript received May 7, 2024; revised July 12, 2024; accepted July 19, 2024; published October 8, 2024.

Recently, with the rapid development of computing systems, many solutions have been proposed for Wi-Fi Fingerprinting based Indoor Positioning System (WF-IPS) by using Artificial Neural Networks (ANN) [12-20]. The presented results showcase the effectiveness of ANN based positioning models compared to the traditional methods. The reason is ANN based models can efficiently approximate the nonlinear relationship between the Wi-Fi RSSI data and the positions of the mobile object. It seems that Deep Learning (DL) based models are the most promising approach for performance enhancement of WF-IPS. However, the manual hyperparameter tuning procedure for obtaining the most suitable model is a very challenging and time-consuming task.

Besides the fluctuations due to the changes in the indoor environment, Wi-Fi RSSI data are also often affected by operations or hardware limitations of Wi-Fi APs as well as the Wi-Fi capturing devices to be located. In addition, the dimensionality of Wi-Fi data, especially in a wide deployment area, is very high leading to a phenomenon called "curse of dimensionality" which might degrade the performance of the DL model. Therefore, an essential task that has to be considered when utilizing DL for WF-IPS is data preprocessing such as data cleaning, outlier removal, data dimensionality reduction, etc. Several studies have shown that dimensionality reduction techniques, e.g., Principal Component Analysis (PCA) [21, 22], Truncated Singular Value Decomposition (TSVD) [23, 24], or autoencoders, can help to improve not only computational time but also positioning accuracy. It is worth noting that choosing the best dimensionality reduction technique depends on the data and specific requirements.

Inspired by the effectiveness of the DL model and data dimensionality reduction, a Long Short-Term Memory (LSTM) model with a series of data preprocessing techniques is proposed in this study to enhance positioning accuracy. In addition, to obtain the best hyperparameters for the LSTM model, Bayesian optimization is utilized for the tuning procedure. The contributions of this paper are as follows:

- The proposed combination of dimensionality reduction techniques, i.e., TSVD and Linear Discriminant Analysis (LDA), helps to avoid the "curse of dimensionality" as well as to exploit the supervised learning information which consequently enhances the positioning accuracy.
- Bayesian optimization is utilized for robust and efficient hyperparameter tuning of the LSTM model.
- The proposed positioning scheme is compared to the state-of-the-art studies using the same public dataset showing considerable improvement in positioning accuracy.

The paper is organized as follows. In Section II, the related works are presented. The data pre-processing procedure and hyperparameter tuning of the LSTM model are presented in Section III. The experimental results of the proposed approach and discussions are drawn in Section IV. The conclusions of the article are given in Section V.

II. Literature Review

Over the last decade, indoor positioning has increasingly attracted the attention of researchers and engineers. Among many approaches, deep learning models are of particular interest. It has to be noted that Wi-Fi RSSI signals have been the most popular choice for developing indoor positioning systems [5]. In a large positioning system, Wi-Fi RSSI data often experience high dimensionality and signal fluctuation. In addition, there might be some unintended APs present in the captured data due to the sharing network by people for their own purposes, e.g., mobile hotspots turning on/off, during measurement campaigns. Therefore, in developing a robust WF-IPS with a reasonable inference time, data pre-processing techniques should be employed to remove noisy data and reduce feature dimension.

Recently, various deep learning based approaches for WF-IPS have been developed as reported in [1–6]. Among them, Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) were the most favorable models to be employed in many positioning approaches. In [14], two data pre-processing methods for large scale and small scale scenarios were proposed, then a CNN model with four convolutional layers was developed to enhance the positioning accuracy. In [15], a WF-IPS model for edge devices was developed by combining a light convolutional auto-encoder for feature extraction and a light CNN model for classification. In [16], deep learning models were developed based on RNN and LSTM for indoor positioning. The presented results show the similarity of floor classification and location estimates between RNN and LSTM models. In addition, the number of RNN or LSTM layers for the deep learning model was evaluated showing a slight improvement by utilizing more layers compared to using only one layer. In [17], to improve positioning estimation accuracy, an LSTM model was developed based on the results of a robust local feature extractor by using sliding windows. In [20], a combination of SAE and an attention-based LSTM model was proposed for WF-IPS. The Stacked Autoencoder (SAE) was employed for feature selection and the LSTM was utilized as the position predictor delivering robust location estimation. In [19], for improving localization robustness and accuracy, a spatial-temporal positioning solution was built upon a residual network and LSTM. The spatial features are extracted from the Wi-Fi signal of each time slice by the residual network while the temporal features are extracted by LSTM.

In order to reduce the computation time of the positioning model, several approaches have been presented [18, 21–25]. In [21, 22], PCA was chosen as the data dimensionality reduction technique to pre-process the Wi-Fi RSSI before applying a classification/regression model for positioning estimation. The results demonstrated that utilizing PCA helps to improve the performance of WF-IPS both in positioning accuracy and system complexity. In [23, 24], authors have presented approaches in the same fashion as in [21, 22], however, using TSVD instead of PCA. The experiments illustrated that TSVD is more suitable for reducing the dimensionality of Wi-Fi RSSI data which often have sparsity characteristics. In [26], during the training phase, the localization area was divided into subareas by employing fuzzy C-means algorithm, only reliable APs in subareas were chosen to reduce the feature dimensions. In the positioning phase, the nearest neighbor technique was utilized based on the selected APs to determine the subareas, the location was then estimated by employing the relative distance fuzzy localization algorithm. In [18], an AP selection scheme was proposed to reduce the computational cost and noise impact while enhancing the positioning accuracy of the positioning model. In [25], data dimensionality reduction technique, i.e., selection of informative APs based on K-means and Fuzzy C-means clustering, was proposed for Multiple Service Set Identifiers signals.

As discussed above, deep learning based WF-IPS seems to be the most promising approach for the enhancement of positioning accuracy. To further improve the performance of real-time positioning applications, data dimensionality reduction should be conducted before applying deep learning models to lower the inference time.

III. Materials and Methods

This section presents the system model of the proposed WF-IPS. The data pre-processing procedure, i.e., data normalization, data dimensionality reduction, and Bayesian optimization for hyperparameter tuning of the LSTM regression model are discussed.

A. *Wi-Fi RSSI Dataset and Data Pre-processing*

1) *Wi-Fi RSSI dataset*

The dataset provided by Mendoza-Silva *et al.* [27] was gathered on the 3rd and 5th levels of a library building in a university. The data collecting process required orienting oneself in specified directions and obtaining six fingerprints at each position. Six successive samples were taken at each position to eliminate any initial values. The datasets for training, Test-01, Test-02, and Test-03 encompassed the directions of "Up" and "Down", whereas Test-04 and Test-05 specifically targeted the directions of "Left" and "Right". The collection proceeded in the following order: (1) "Up" direction 3rd floor, (2) "Down" direction 3rd floor, (3) "Up" direction 5th floor, and (4) "Down" direction 5th floor. The datasets for Training, Test-01, and Test-05 consistently encompassed data captured with all directions for the month positions corresponding to training. The data collected for Test-04 originated from horizontal corridors, hence the data collecting directions are "Left" and "Right". Test-02 and Test-03 were only focused on "Up" and "Down" directions corresponding to walking directions between bookshelves. The Wi-Fi RSSI data collection campaign is conducted in a total area of 308.4 square meters over both levels. The datasets were categorized into 15 collecting months, yielding a total of 16,704 training samples and 46,800 test samples. In the process of data preprocessing, any values corresponding to undiscovered Access Points (APs) are substituted with −100 dBm representing the weakest signal strength in the whole dataset which will be discussed in the subsequent sections of this research.

2) *Data normalization*

As discussed in the above sections, Wi-Fi RSSI observed data often experience variations over time as well as fluctuation because of fast fading in indoor environments. Therefore, the measured data should be scaled/normalized to reduce the noise while maintaining data information and structure before applying any dimensionality reduction technique. In this paper, several data normalized techniques, i.e., standard normalization and max-min normalization, as presented in Eq. (1) and Eq. (2), respectively, have been utilized to find the best method for delivering accurate position estimates.

$$RSSI_{jStdNorm} = \frac{RSSI_j - RSSI_\mu}{RSSI_\sigma} \quad (1)$$

$$RSSI_{jMaxMinNorm} = \frac{RSSI_j - RSSI_{min}}{RSSImin_{max}} \quad (2)$$

where, $RSSI_\mu, RSSI_\sigma, RSSI_{\max}, RSSI_{\min}$ are the average, standard deviation, maximum, and minimum Wi-Fi RSSI values, respectively, of each sample. $RSSI_j, RSSI_{jStdNorm}, RSSI_{jMixMinNorm}$ are the measured, standard normalized and max-min normalized RSSI values, respectively, of the j-th AP.

3) *Data dimensionality reduction*

This subsection presents an approach for data dimensionality reduction by combining two well-known techniques namely TSVD and Linear Discriminant Analysis (LDA). This approach is proposed based on the principles of fingerprinting techniques which are supervised training methods. As reported in [23, 24], TSVD was applied successfully to reduce the data dimensions while improving positioning accuracy. However, TSVD is an unsupervised dimensionality reduction technique trying to preserve as much data information as possible. It does not provide any extra information to enhance the performance of the regression/classification model. Therefore, in this paper, we utilized LDA, a supervised dimension reduction technique, to obtain some extracted features that help the regression/classification model to have more informative features related to the training data labels. The results of TSVD and LDA are then concatenated to form the final dimensional reduction data. As a result, it is expected that the positioning accuracy should be improved.

TSVD is a method devised to reduce the number of feature dimensions in a dataset. It is frequently employed to address diverse issues involving the presence of high-dimensional data. The problem of high dimensional data referred to as the "curse of dimensionality" often negatively affects the performance of deep learning systems. TSVD is based on the notion of Singular Value Decomposition (SVD). The SVD tries to decompose a matrix A into three distinct matrices Σ, U, V which are, respectively, singular values, left singular vectors, and right singular vectors of the matrix A, as shown in Eq. (3).

$$A_{M\times N} = U_{M\times M}\Sigma_{M\times N}(V_{N\times N})^T \quad (3)$$

TSVD preserves the k highest singular values and their corresponding singular vectors. The primary objective of TSVD is to find a reduced-dimensional representation of the original matrix that retains the maximum amount of data information, including data patterns and correlations. In order to efficiently decrease the dimensionality of data based on a particular problem, it is crucial to determine the optimal value of k. The mathematical representation of TSVD is given in Eq. (4).

$$A_{M\times N} \approx A_{k\times k} = U_{k\times k}\Sigma_{k\times k}(V_{k\times k})^T \qquad (4)$$

The LDA is a very popular and successful feature extraction technique that utilizes the supervised membership of the data. The LDA tries to find an optimal linear projection where the between-class scatter is maximized while the within-class scatter is minimized. For the data with C classes, the between-class scatter matrix and the within-class scatter are computed as Eq. (5) and Eq. (6).

$$S_B = \sum_{c=1}^{C} N_c(m_c - m)(m_c - m)^T \qquad (5)$$

$$S_W = \sum_{c=1}^{C} \sum_{x\in X_c}(x - m_c)(x - m_c)^T \qquad (6)$$

where, N_c, N are the number of samples in class c and total number of samples, respectively. m_c, m are the mean of the data in class c and the global mean, respectively. X_c is the set of data samples that belong to the class c.

Let's denote the number of TSVD components and LDA components is k and d, respectively. The number of features of the data after conducting dimensionality reduction is $(k+d)$.

B. System Model

The proposed WF-IPS is separated into two phases including the offline phase and the online phase as illustrated in Fig. 1. In the offline phase, Wi-Fi RSSI data captured from the seen APs at RPs are pre-processed by cleaning, normalization, and dimensionality reduction techniques. The processed training data with the accompanying location labels are then utilized to train the LSTM regression model. Once the training procedure is completed, the LSTM regression model is ready to be used in the positioning phase. In the positioning phase, the real-time captured Wi-Fi RSSI data by a mobile object is pre-processed in the same fashion as described in the training phase before feeding to the trained LSTM regression model for estimating the object location.

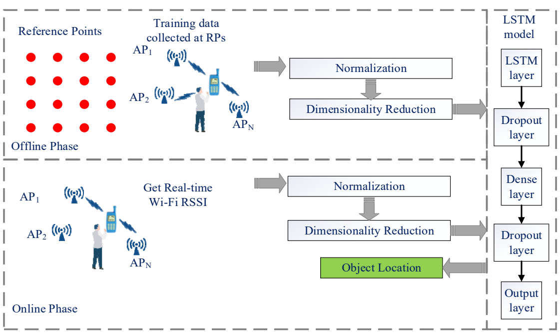

Fig. 1. Note how the caption is centered in the column.

C. LSTM Model and Bayesian Optimization

1) Introduction to LSTM model

This article employs the LSTM regression model to estimate mobile object position. The LSTM model exhibits the ability to selectively retain or discard information when processing lengthy sequences of data. The LSTM is designed to effectively capture long-term relationships in order to describe context and sequential patterns. An LSTM cell consists of a memory cell and three gates: an input gate i_t, a forget gate f_t, and an output gate o_t, as depicted in Fig. 2. The input gate controls the flow of information that is sent to the cell. The forget gate determines the amount of information that should be preserved and conveyed to the cell. The output gate controls the output and the hidden state. The memory cell is tasked with retaining information over a period of time within the network. The operation at each time step t of the LSTM can be mathematically expressed by Eq. (7) to Eq. (12).

$$i_t = \sigma[(W_{i,x}x_t + W_{i,h}h_{t-1}) + b_i] \qquad (7)$$

$$f_t = \sigma[(W_{f,x}x_t + W_{f,h}h_{t-1}) + b_f] \qquad (8)$$

$$\tilde{C}_t = tanh[(W_{c,x}x_t + W_{c,h}h_{t-1}) + b_c] \qquad (9)$$

$$C_t = f_tC_{t-1} + i_t\tilde{C}_t \qquad (10)$$

$$o_t = \sigma[(W_{o,x}x_t + W_{o,h}h_{t-1}) + b_o] \qquad (11)$$

$$h_t = o_t\, tanh(C_t) \qquad (12)$$

where, the input, output, cell state, and updated cell state at the time step t are denoted as x_t, h_t, C_t, and $\tilde{C}_t$, correspondingly, whereas the previous cell state and hidden state are presented as C_{t-1}, h_{t-1}. The weight matrices and bias vectors of the input, forget, updated cell state, and output gate layers are denoted as $W_i, W_f, W_c, W_o, b_i, b_f, b_c, b_o$, respectively. The activation functions used in the LSTM cells at each gate as illustrated in Fig. 2 are σ and tanh.

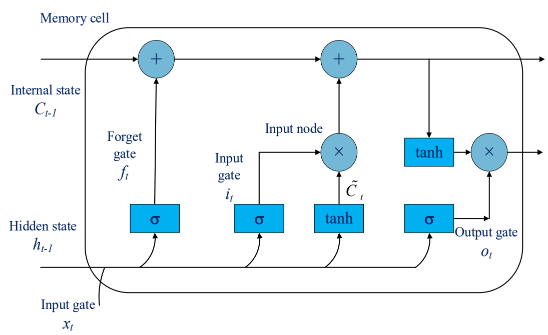

Fig. 2. The structure of an LSTM cell.

2) Bayesian optimization

The performance of the deep learning models highly depends on the setup of hyperparameters. Manually tuning hyperparameters, especially in the case of complex networks, is a time-consuming task. Recently, several hyperparameter optimization methods have been introduced, e.g., grid search, randomized search, Bayesian Optimization (BO), etc. to relax the difficulty of manual tuning procedure. Grid search is an exhaustive searching technique which evaluates all the combinations of the hyperparameters. Therefore, it will come up with the best solution once the search process is complete. However, it is obviously the most time-consuming compared to other search algorithms. On the other hand, randomized search utilizes statistics of the hyperparameters during the searching process which could be more efficient compared to grid search in several situations. However, its effectiveness in finding optimal hyperparameters could degrade when the search space is large. It is worth noting that both grid search and randomized search do not consider the past results during the searching process.

In contrast, BO is the hyperparameter tuning technique based on Bayes theorem [28]. It keeps track of the past results to build a probabilistic model of the loss function. The model is then utilized for determining the next combination of hyperparameters to be evaluated. This helps to reduce the time required for obtaining the optimal hyperparameters. The BO technique operates in as follows:

- BO uses a surrogate (probability) model to approximate the loss function. A Gaussian process is often selected for the surrogate model which is utilized for determining the promising combination of hyperparameters to be examined in the true loss function.
- An acquisition function is utilized by BO to employ the posterior information for determining the best set of hyperparameters in each iteration and identifying the most promising combination of hyperparameters to be evaluated in the next iteration. The exploration and exploitation of the searching process are balanced by the acquisition function. Exploration is the strategy to escape local optima by selecting a combination of hyperparameters in the less explored regions while exploitation is focused on the regions with a higher probability of improving the current solution.

As discussed above, BO seems to be more effective than the other two mentioned searching techniques. Several studies have demonstrated the effectiveness of utilizing BO for determining the best hyperparameter set for deep learning models [29, 30]. Therefore, BO is chosen as the hyperparameter tuning for the LSTM model in this study.

D. Positioning Performance Evaluation Metrics

In order to assess the performance of the proposed positioning approach, Mean Distance Error (MDE) and Root Mean Squared Error (RMSE) are used as performance metrics to compare the positioning accuracy of different systems. Let's denote the positioning distance error, the coordinates of the ground truth and the estimated location of the i-th RSSI test sample asd_i,$(x_{i,true}, y_{i,true})$ and $(x_{i,pred}, y_{i,pred})$, respectively. The positioning distance error can be calculated by Eq. (13). Eq. (14) and Eq. (15) are then utilized to compute the values of MDE and RMSE, respectively.

$$d_i = \sqrt{\left(x_{i,true} - x_{i.pred}\right)^2 + \left(y_{i,true} - y_{i.pred}\right)^2} \quad (13)$$

$$MDE = \frac{\sum_{i=1}^{N_{test}} d_i}{N_{test}} \quad (14)$$

$$RMSE = \sqrt{\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} d_i^2} \quad (15)$$

IV. RESULT AND DISCUSSION

This section presents the experimental results conducted on a public dataset to demonstrate the effectiveness of the proposed WF-IPS. Since the objective of this study is to improve the positioning estimates, various setups have been produced to examine this target. The computational cost of the proposed approach is also briefly discussed.

A. Data Dimensionality Reduction

For data dimensionality reduction, the number of features to be kept should be considered carefully. If it is too small or too large, some essential information might be lost or some unnecessary information might still present in the data, respectively, leading to degradation of the performance of the classification/regression models. In the following, the procedure for determining the number of LDA and TSVD components is explained in detail.

As mentioned in Subsection III.A.3, the LDA technique can extract features in a supervised manner which intuitively helps the classification/regression model perform better. Therefore, the LDA was first utilized and the maximum number of LDA components was chosen to be kept. In the training data, there are 24 different coordinates of RPs representing 24 data classes. As a result, the maximum number of LDA components that can be kept is $d = 23$.

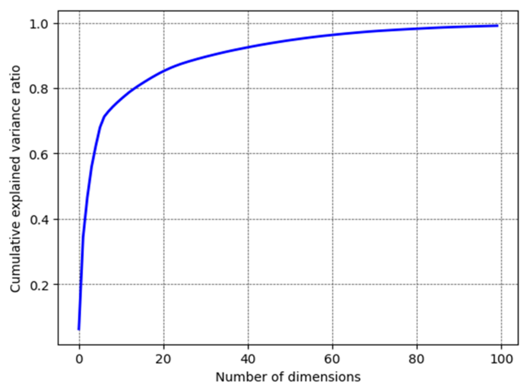

Fig. 3. The preserved information vs. number of STVD dimensions.

Since we intended to preserve as much as possible the information of data when performing dimensionality reduction using TSVD, the cumulative explained variance ratio versus the number of TSVD components to be kept was plotted as illustrated in Fig. 3. It can be seen that when the number of TSVD dimensions is above 80, almost all information in the data is preserved. If 100 TSVD dimensions are selected, it is clear from Fig. 3 that approximately 100% of data information is retained. Therefore, in this article, the number of TSVD dimensions is selected in the range of 80 to 100 with a step size of 10 to evaluate the positioning accuracy.

For determining the most suitable data normalization and the number of components being kept after dimensionality reduction, the baseline LSTM model as presented in [21] was utilized. In [21], the positioning model is in the same fashion as the proposal in this study. The differences are the dimensionality reduction technique and the hyperparameter tuning process. Table I illustrates the model structure with hyperparameters being used in [21].

TABLE I. BASELINE LSTM MODEL

Characteristics	Appearance
Num. of LSTM units	100
Input droprate for LSTM layer	0.3
Act. Func. for LSTM layer	sigmoid
Num. of Dense layer units	100
Act. Func. for Dense layer	sigmoid
Num. of Output layer units	2
Act. Func. for Output layer	linear
Learning rate	0.001
Optimizer	Adam
Batch size	32
Training epoch	100

The positioning accuracy of different normalization schemes and the number of components is presented in Table II. As can be seen from the Table, standard normalization is much better than max-min normalization for all different setups. Looking in more detail, for each type of normalization, combining TSVD with LDA for data dimensionality reduction helps the regression model perform better than using only TSVD. This demonstrates the importance of LDA in data dimensionality reduction. It has to be noted that, compared to the positioning results presented in [21], e.g., MDE = 2.18 m, without any modification of the LSTM regression model, using the proposed data dimensionality technique is able to produce a considerable improvement result. As indicated in Table II, the standard normalization is chosen in this study. Furthermore, for dimensionality reduction, 20 LDA components are concatenated with 80 TSVD components to form the data with 100 features which will be fed to the LSTM regression model for position estimation.

TABLE II. EVALUATION OF DATA NORMALIZATION AND NUMBER OF COMPONENTS FOR DIMENSIONALITY REDUCTION

Normalization	Dimensionality Reduction		MDE [m]
	LDA	TSVD	
Standard	***20***	***80***	***2.041***
Standard	10	90	2.043
Standard	0	100	2.087
Min-Max	20	80	2.159
Min-Max	10	90	2.160
Min-Max	0	100	2.231

B. Hyperparameter Tuning of LSTM Model

This paper aims to improve the positioning accuracy while maintaining the model complexity, i.e., maintaining the inference time of the system. Therefore, the main structure of the LSTM model is similar to the one proposed by [21]. However, with the help of the BO, this article examines the larger set of hyperparameters in order to come up with the most promising model for obtaining high positioning accuracy. Especially, we added two dropout layers after the LSTM layer and the dense layer which might help the model to have better generalization resulting in better positioning accuracy for the unseen test data.

As presented in Table III, the column "BO results" is the final optimal set of hyperparameters for the proposed LSTM regression model. It can be seen that the number of units for the LSTM layer and dense layer is the same as the model proposed by [21]. However, other hyperparameters such as drop rates and activation functions are different from the parameters provided by [21].

It has to be noted that the optimal hyperparameters have been obtained with ease because of the utilization of BO. During the BO based searching process, we have implemented a strategy for early stopping if the performance of the LSTM regression model is not improved in 20 consecutive epochs. This also helps to shorten the searching time of the BO. For this study, BO needs only a few hours to deliver the optimal set of hyperparameters.

To further tune the hyperparameters of the proposed LSTM model, the number of training epochs was varied in the range of 20 to 100 while keeping all other determined hyperparameters. The best results were obtained when the number of training epochs was 30.

TABLE III. HYPERPARAMETER TUNING FOR THE LSTM MODEL

Characteristics	Appearance	BO results
Num. of LSTM units	[40:10:100]	100
Input drop rate for LSTM layer	[0.0:0.1:0.3]	0.0
Act. Func. for LSTM layer	[relu, tanh, sigmoid]	tanh
Drop rate for Dropout layer	[0.0:0.1:0.5]	0.4
Num. of Dense layer units	[40:10:100]	100
Act. Func. for Dense layer	[relu, tanh, sigmoid]	tanh
Droprate for Dropout layer	[0.0:0.1:0.5]	0.0
Num. of Output layer units	2	2
Act. Func. for Output layer	linear	linear
Learning rate	[0.01, 0.001, 0.0001]	0.001
Optimizer	[Adam, Nadam]	Adam
Training epoch	100	100

C. Positioning Performance Evaluation

To demonstrate the superior performance of the proposed approach, for a fair comparison, three previous studies have been reproduced in this work on the same public data set provided by [27]. Fig. 4 presents the experimental results of the proposed method and three other solutions. The results demonstrate a considerable improvement in positioning accuracy of the proposed model compared to the others. As illustrated in the figure, with the aid of two unsupervised dimensionality reduction techniques, i.e., PCA [21] and TSVD [24], and the fine-tuning LSTM model, the positioning accuracy was considerably improved. The results also revealed that TSVD performed better than PCA in the case of Wi-Fi data where sparsity is present in the data. Furthermore, with the help of supervised dimensionality reduction, i.e., LDA, the positioning accuracy was further improved as expected. The MDE and RMSE obtained by our model are 1.99 m and 1.69 m, respectively, which are the lowest values among all presented methods. For instance, the MDE of the proposed approach improved by roughly 3%, 9%, and 24% compared to the results obtained by [16, 21, 24], respectively. Table IV shows the differences between our study and other benchmark works.

TABLE IV. COMPARISON BETWEEN DIFFERENT APPROACHES

Models	Hyperparameter Tuning Method	Dimensionality Reduction Techniques	Advantages	Drawbacks
LSTM [16]	Manual	No	Simple	Slow and possibly not optimal hyperparameter determination Curse of dimensionality experience
PCA-LSTM [21]	Manual	PCA	Curse of dimensionality avoidance	Slow and possibly not optimal hyperparameter determination Low computational cost
TSVD-LSTM [24]	Manual	TSVD	Curse of dimensionality avoidance	Slow and possibly not optimal hyperparameter determination Low computational cost
LDA-TSVD-LSTM (Proposed)	BO based	LDA-TSVD	Fast and robust hyperparameter determination Curse of dimensionality avoidance Better positioning accuracy based on utilization of supervised dimensionality reduction	Slightly higher computational cost compared to [21] and [24]

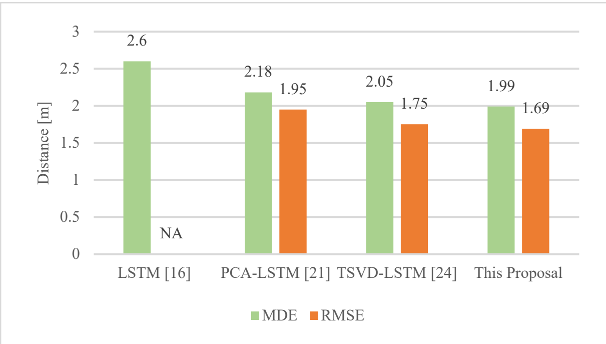

Fig. 4. Comparison of positioning accuracy.

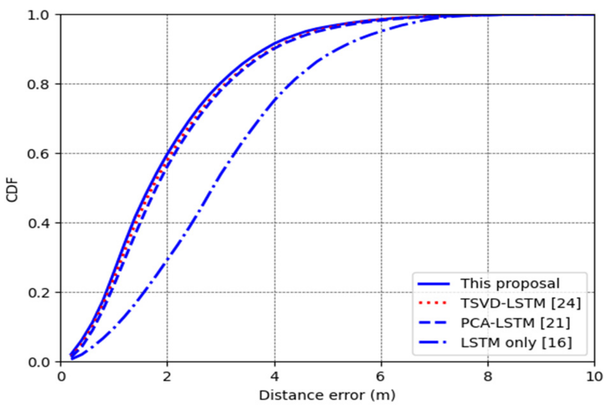

Fig. 5. CDF of positioning error distance.

For a better understanding of the positioning accuracy of the proposed approach compared to the other benchmark solutions, Fig. 5 shows the Cumulative Distribution Function (CDF) of the positioning error distance. The solid blue line, dashed dot blue line, dashed blue line, and dotted red line represent the CDF of the proposed model, [16, 21, 24], respectively. It is clear that data dimensionality reduction based LSTM regression models perform better than the model using data with the original number of features. Among the approaches utilizing data dimensionality reduction, our proposed model delivers better positioning results in the whole range of the CDF. As illustrated in Fig. 5, 90% of the estimated positions have a distance error of less than 4 m which is suitable for civil LBS applications. It should be noted that the majority of the test data provided by [27] were collected at locations different from the RPs.

In terms of computational cost, since the proposed LSTM regression model has the same structure as the model presented in [21], the inference time of the two LSTM models are the same. However, there are two dimensionality reduction techniques applied to the online testing data in this work, therefore, the inference time of the whole system of the proposed model is slightly higher than that of the system presented in [21].

V. CONCLUSION

This paper presents an approach for WF-IPS to enhance positioning accuracy while maintaining the system complexity at a low cost. The model is built upon the combination of two well-known dimensionality reduction techniques, namely LDA and TSVD, to employ the advantages of each technique. In addition, to ease the procedure of hyperparameter tuning of the LSTM regression model, the BO is chosen because of its robust performance with exploitation and exploration abilities. The experimental results of different approaches conducted on the same public dataset demonstrate the effectiveness of the proposed solution.

It is noted that the dataset used in the study is remarkably similar to the real situations where the number of RPs is limited while the test locations can be anywhere in the area of interest. In the future, effective data augmentation and interpolation techniques for enriching the training data should be investigated to further improve the performance of WF-IPS.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

D.K.N, L.C.N, and M.K.H designed the study. D.K.N and M.K.H performed the experiment and simulation. D.K.N, L.C.N, and M.K.H analyzed the data and verified the result of the simulation. D.K.N and M.K.H drafted the manuscript and wrote the manuscript with input from all authors. All authors had approved the final version.

REFERENCES

[1] L. Qi, Y. Liu, Y. Yu, and L. Chen *et al.*, "Current status and future trends of meter-level indoor positioning technology: A review," *Remote Sens*., vol. 16, no. 398. https://doi.org/10.3390/rs16020398

[2] M. D. Jovanovic and S. M. Djosic, "Analysis of indoor localization techniques," in *Proc. 2023 58th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST)*, Nis, Serbia, 2023, pp. 219–222. doi: 10.1109/ICEST58410.2023.10187323

[3] I. Ashraf, S. Hur, and Y. Park, "Recent advancements in indoor positioning and localization," *Electronics*, vol. 11, no. 2047, 2022. https://doi.org/10.3390/electronics11132047.

[4] J. Dai, M. Wang, and B. Wu *et al.*, "A survey of latest Wi-Fi assisted indoor positioning on different principles," *Sensors*, vol. 23, no. 7961, 2023. https://doi.org/10.3390/s23187961

[5] V. Bellavista-Parent, J. Torres-Sospedra, and A. Pérez-Navarro, "Comprehensive analysis of applied machine learning in indoor positioning based on Wi-Fi: An extended systematic review," *Sensors*, vol. 22, p. 4622, 2022. https://doi.org/10.3390/s22124622

[6] J. Kunhoth, A. Karkar, S. Al-Maadeed *et al.*, "Indoor positioning and wayfinding systems: A survey," *Hum. Cent. Comput. Inf. Sci.* vol. 10, no. 1, 2020. https://doi.org/10.1186/s13673-020-00222-0

[7] B. Sulaiman, S. Tarapiah, E. Natsheh *et al.*, "Radio map generation approaches for an rssi-based indoor positioning system," *Systems and Soft Computing*, vol. 5, 200054, 2023.

[8] P. Bahl and V.N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, 2000, vol. 2, pp. 775–784.

[9] M. K. Hoang and R. Haeb-Umbach, "Parameter estimation and classification of censored gaussian data with application to WiFi indoor positioning," in *Proc. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 3721–3725.

[10] T. K. Vu, M. K. Hoang, and H. L. Le, "Performance enhancement of Wi-Fi fingerprinting-based IPS by accurate parameter estimation of censored and dropped data", *Radioengineering*, vol. 28, no. 4, pp. 740–748, 2019.

[11] T. K. Vu, M. K. Hoang, and H. L. Le, "An EM algorithm for GMM parameter estimation in the presence of censored and dropped data with potential application for indoor positioning," *ICT Express*, vol. 5, no. 2. pp. 120–123, 2019. https://doi.org/10.1016/j.icte.2018.08.001

[12] E. Scavino, M. A. A. Rahman, and Z. Farid, "An improved hybrid indoor positioning algorithm via QPSO and MLP signal weighting," *Computers, Materials & Continua*, vol. 74, no. 1, pp. 379–397, 2023. https://doi.org/10.32604/cmc.2023.023824

[13] Z. E. Khatab, A. H. Gazestani, S. A. Ghorashi *et al.*, "A fingerprint technique for indoor localization using autoencoder based semi-supervised deep extreme learning machine," *Signal Processing*, vol. 181, 107915, 2021. https://doi.org/10.1016/j.sigpro.2020.107915

[14] H. Zhu, L. Cheng, and X. Li *et al.*, "Neural-network-based localization method for Wi-Fi fingerprint indoor localization," *Sensors*, vol. 23, 6992, 2023. https://doi.org/10.3390/s23156992

[15] A. Kargar-Barzi, E. Farahmand, and A. Mahani *et al.*, "CAE-CNNLoc: An edge-based WiFi fingerprinting indoor localization using convolutional neural network and convolutional auto-encoder", 2023. arXiv preprint, arXiv:2303.03699.

[16] H. Y. Hsieh, S. W. Prakosa, and J. S. Leu, "Towards the implementation of recurrent neural network schemes for WiFi fingerprint-based indoor positioning," in *Proc. 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, 2018, pp. 1–5.

[17] Z. Chen, H. Zou, and J. Yang *et al.*, "WiFi fingerprinting indoor localization using local feature-based deep LSTM," *IEEE Systems Journal*, vol. 14, no. 2, pp. 3001–3010, June 2020. doi: 10.1109/JSYST.2019.2918678

[18] X. Shi, J. Guo, and Z. Fei, "WLAN fingerprint localization with stable access point selection and deep LSTM," in *Proc. 2020 IEEE 8th International Conference on Information, Communication and Networks (ICICN)*, Xi'an, China, 2020, pp. 56–62. doi: 10.1109/ICICN51133.2020.9205086

[19] R. Wang, H. Luo, and Q. Wang *et al.*, "A spatial-temporal positioning algorithm using residual network and LSTM," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 11, pp. 9251–9261, Nov. 2020. doi: 10.1109/TIM.2020.2998645

[20] S. L. Ayinla, A. A. Aziz, and M. Drieberg, "SALLoc: An accurate target localization in WiFi-enabled indoor environments via SAE-ALSTM," *IEEE Access*, vol. 12, pp. 19694–19710, 2024. doi: 10.1109/ACCESS.2024.3360228

[21] T. H. Duong, A. V. Trinh, and M. K. Hoang, "An enhancement of indoor localization using PCA-aided LSTM approach," in *Proc. 2023 International Conference on Advanced Technologies for Communications (ATC)*, Da Nang, Vietnam, 2023, pp. 512–516. doi: 10.1109/ATC58710.2023.10318863

[22] S. -H. Fang and T. Lin, "Principal component localization in indoor WLAN environments," *IEEE Transactions on Mobile Computing*, vol. 11, no. 1, pp. 100–110, Jan. 2012. doi: 10.1109/TMC.2011.30

[23] H. Duong, K. Hoang, and V. Trinh *et al.*, "Dimensionality reduction with truncated singular value decomposition and k-nearest neighbors regression for indoor localization" *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 14, no. 10, 2023. http://dx.doi.org/10.14569/IJACSA.2023.0141034

[24] D. K. Nguyen, T. H. Duong, and L. C. Nguyen *et al.*, "Performance enhancement of Wi-Fi fingerprinting based indoor positioning using truncated singular value decomposition and LSTM model," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 15, no. 5, 2024. https://dx.doi.org/10.14569/IJACSA.2024.0150529

[25] A. Abed and I. Abdel-Qader, "RSS-fingerprint dimensionality reduction for multiple service set identifier-based indoor positioning systems," *Appl. Sci*., vol. 9, 3137, 2019. https://doi.org/10.3390/app9153137

[26] J. Li, J. Tian, and R. Fei *et al.*, "Indoor localization based on subarea division with fuzzy C-means," *International Journal of Distributed Sensor Networks*, vol. 12, no. 8, 2016. doi: 10.1177/1550147716661932

[27] G. M. Mendoza-Silva, P. Richter, and J. Torres-Sospedra *et al.*, "Long-Term WiFi fingerprinting dataset for research on robust indoor positioning," *Data,* vol. 3, no. 3, 2018. https://doi.org/10.3390/data3010003

[28] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[29] S. H. Almuhaini and N. Sultana, "Bayesian-optimization-based Long Short-Term Memory (LSTM) super learner approach for modeling long-term electricity consumption," *Sustainability*, vol. 15, 13409, 2023. https://doi.org/10.3390/su151813409

[30] M. Masum, H. Shahriar, and H. Haddad *et al.*, "Bayesian hyperparameter optimization for deep neural network-based network intrusion detection," in *Proc. 2021 IEEE International Conference on Big Data (Big Data)*, Orlando, FL, USA, 2021, pp. 5413–5419. doi: 10.1109/BigData52589.2021.9671576