IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 21, 2020, accepted June 15, 2020, date of publication June 22, 2020, date of current version July 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3004039

Feature Fusion Using Stacked Denoising Auto-Encoder and GBDT for Wi-Fi Fingerprint-Based Indoor Positioning

HUA ZHANG, BIAO HU, SHIQI XU, BI CHEN, MIAN LI, AND BO JIANG
School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou 310018, China

Corresponding author: Hua Zhang (zerozhua@126.com).

This work was supported in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LY19F020003 and Grant LY18F020006, in part by the National Natural Science Foundation of China under Grant 61672459, and in part by the Key Research and Development Program Project of Zhejiang Province under Grant 2019c01004.

ABSTRACT It may be very difficult to receive the signals from satellite positioning systems due to the existing obstacles in indoor environment. Arising from the popularity of smart phones, Wi-Fi based indoor positioning technology has the advantages with convenient deployment and low hardware cost. In this study, we focused on indoor positioning using Wi-Fi fingerprint data that were collected in shopping malls. Due to the volatility of Wi-Fi signals and the high-dimensional sparseness of fingerprint data, we proposed a feature extraction algorithm, called joint multi-task stacked denoising auto-encoder (JMT-SDAE), aiming at reducing the dimensionality of the original fingerprint data and improving the indoor positioning performance in shopping malls. Furthermore, the features extracted by JMT-SDAE and gradient boosting decision tree (GBDT) were merged to construct a hybrid model, named as JMT-SDAE+GBDT. The experimental results based on 13 location datasets showed that the proposed feature fusion model had better positioning accuracy when compared with other existing positioners, and thus confirmed the effectiveness of our proposed feature extraction algorithm through multi-task learning.

INDEX TERMS Indoor positioning, Wi-Fi fingerprint, feature fusion, stacked denoising auto-encoder, multi-task learning, gradient boosting decision tree.

I. INTRODUCTION

The popularity of mobile devices has led to a greater usage of various location-based services (LBS). An increase in demand for LBS applications with high positioning accuracy is expected. The Global Navigation Satellite Systems (GNSS), such as GPS, BeiDou, GLONASS, and Galileo, are very mature outdoor positioning solutions that can locate targets and achieve high positioning accuracy in an open outdoor environment [1], [2]. However, the satellite signals are easily weakened as they travel through the building [3]. The signal strength is about 10 to 100 times weaker in an indoor environment [4]. It is almost impossible for a GNSS receiver to acquire signals from any satellites due to further attenuation arising from various factors such as no line-of-sight, people movement, multi-path effect, interference and noise, etc. [4], [5]. Even if the signals from a satellite are detected indoors, poor signal-strength, multi-path effects as well as lack of visible satellites can cause computing inaccurate distance from the satellite, leading to the estimated location far from the true location [4], [6], [7]. Although there exist multiple ways to enhance the signal strength, such as cloud-offloaded instant GPS [4] and longer coherent integration [8], they usually require additional costly computing and equipment. Therefore, the GNSS based positioning is not an appropriate choice for indoor environment, especially for large buildings.

At present, indoor positioning can be divided into two categories: infrastructure-based and non-infrastructure-based. One of typical non-infrastructure-based methods is the inertial navigation system (INS), such as Pedestrian dead-reckoning (PDR) [9]–[11]. However, such methods are highly susceptible to drift in gyroscope readings and fluctuations in a magnetic field in a room. Infrastructure-based positioning methods require preinstalled transmitters such as Bluetooth [12], [13], Infra-Red (IR) [14], ultrasonic [15], Ultra

The associate editor coordinating the review of this manuscript and approving it for publication was Senthil Kumar.

Wide Band (UWB) [16], Radio Frequency Identification (RFID) [17], Wi-Fi [18], [19]. Some studies also focused on indoor positioning using multimodal fingerprints [5] of those wireless signals. However, except for Wi-Fi, other signals typically require additional network architectures and dedicated hardware devices for positioning and signal reception. Moreover, wireless signals such as IR and ultrasonic are difficult to penetrate obstacles such as doors, walls, and floors, leading to limited positioning range. In contrast, Wi-Fi based indoor positioning technology is undoubtedly more feasible since Wi-Fi networks have been almost ubiquitous in many buildings, such as campuses, science museums, and shopping malls. Furthermore, Wi-Fi chips are already embedded in mobile devices such as smart phones, tablet computers and laptops. The indoor positioning based on Wi-Fi signals has the advantages with convenient deployment, low hardware cost, strong environmental adaptability and high real-time performance.

Wi-Fi based indoor positioning methods can be further divided into two categories: one is based on triangular geometry, called triangulation positioning method; the other is based on fingerprint matching, named as location fingerprinting [20]. The triangulation methods are usually carried out by measuring the angle of arrival (AOA) between the target and several reference points [21], [22] or calculating the distance between the target and the reference points in terms of the time difference of arrival (TDoA)[23]. However, even little time error between the target and the reference point may lead to huge precision errors. Therefore, the Wi-Fi fingerprints are preferred data that have been extensively investigated by researchers [24], [25]. The solution for fingerprint matching usually requires a preliminary system calibration procedure [26] (called *off-line phase*) for constructing the indoor floor plan by specialists [27]. This off-line phase consists of manual collections for the received signal strength (RSS) observations at certain reference points in predefined locations, regions or grid cells [28]. Then, the collected RSS vectors at each reference point will be stored in a database as training fingerprints for further pattern matching during the localization procedure (called *online phase*)[26]. However, it is very time-consuming and labor-intensive [27], [29], especially in large buildings, such as shopping malls. Although several studies [20], [30] have focused on the impact of reduced location fingerprints as well as the absent RSS data, based on (weighted) k-nearest neighbor, stochastic gradient descent and sparsity rank-singular value decomposition, these approaches may lead to additional localization errors and high computing complexity.

As an alternative to the schemes that aim at locating exact positions, the area classification has recently started to attract attention on estimating the current area of a user, such as the room in a building or the shop inside a mall. This is especially applicable to large-scale deployments with low cost or crowdsourced data with low quality that do not allow for accurate localization [28]. Liu *et al.* [31] proposed an area estimation algorithm by using indoor map information and user trajectories. Rezgui *et al.* [32] evaluated an area localization system on room level accuracy and proposed a normalized rank based support vector machine (SVM) to solve the problem caused by hardware variance and signal fluctuation. Chow *et al.* [33] focused on locality classification composed of two coarse-grained sequential tasks, i.e. inside/outside region decision and area classification. However, majority of them are still on basis of the costly indoor map or some labor-intensive preprocessing steps. Thus, these approaches are not applicable to crowdsourced data with high sparsity collected from smartphones of users in large malls where lots of access points (APs) are deployed.

In this work, we focus on shop-level indoor positioning based on Wi-Fi fingerprint data collected from only customers′ smart phones in shopping malls. Inside these large malls, the observed RSS vectors contain a large number of missing values due to the obstruction of out-of-range APs, random noise, signal fluctuation, momentary occlusion or scanning duration [33], which result in extremely data sparsity. Recently, rapid development of machine learning, especially deep learning[34], has made feature extraction and feature fusion more and more popular. We proposed a novel feature extraction algorithm, called Joint Multi-task Stacked Denoising Auto-Encoder (JMT-SDAE), aiming at reducing the sparseness of Wi-Fi fingerprint data and improving the positioning accuracy. The JMT-SDAE actually consists of two tasks: indoor positioning and reconstruction of RSS inputs. The purpose using SDAE is denoising by adding random noise against harmful effects on localization due to the missing or incorrect values in RSS vectors. Meanwhile, novel feature representations for the original RSS vectors with reduced sparseness can be extracted using JMT-SDAE. Moreover, we integrate the features extracted from the proposed JMT-SDAE and the gradient boosting decision tree (GBDT)[35], and construct a hybrid model to achieve more accurate indoor positioning.

The main contributions of our work are summarized as follows:

- This paper focused on shop-level indoor positioning for the crowdsourced RSS observations with extreme sparsity in large malls.
- We proposed a feature extraction algorithm, named as JMT-SDAE, based on a multi-task model to extract effective and reduced features from extremely sparse RSS vectors.
- A hybrid model, which combined the features extracted by JMT-SDAE and from the GBDT outputs, was further proposed to improve the indoor positioning.

The rest of this article is organized as follows: Section II reviews the background about autoencoder and GBDT. Section III introduces the models about JMT-SDAE and the procedures concerning the feature fusion of GBDT and JMT-SDAE. Section IV describes the datasets, experimental designs and parameters chosen for comparing learning algorithms. Section V includes the analysis of

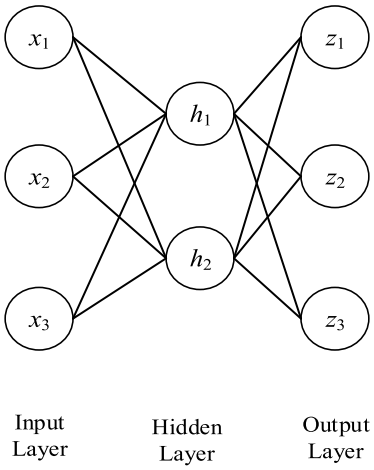


FIGURE 1. The structure of an autoencoder.

experimental results. Section VI presents some conclusions and perspectives.

II. BACKGROUND WORKS

A. AUTOENCODER

An autoencoder is a three-layer feedforward neural network composed of an input layer, a hidden layer and an output layer. The goal of an autoencoder is to recover the original input signal by setting equal dimensions of the input and the output, i.e. the output target is the original input signal itself. Then, the information encoded in the hidden layer can be utilized to reduce the data dimensionality and extract representative features [36]. An autoencoder in general includes two modules: the encoder from the input layer to the hidden layer and the decoder from the hidden layer to the output layer. The network structure of an autoencoder is shown in Figure 1 and the loss function is usually designed as follows:

$$h = f(Wx + b) \tag{1}$$
$$z = g(W'h + b') \tag{2}$$
$$L(x, z) = \|x - z\|^2 \tag{3}$$

where x is the input vector; h is the output of the encoder; f is the activation function of the encoder; W and b are the parameters of the encoder; g is the activation function of the decoder; W' and b' are the parameters of the decoder; z is the output of the decoder; $\mathrm{L}(x, \mathrm{z})$ is the loss function of the autoencoder.

B. STACKED DENOISING AUTOENCODER

The denoising auto-encoder (DAE) [37] is a variation of the standard auto-encoder, which randomly adds noise to the input data. Figure 2 shows the network architecture of DAE where parts of the neuron nodes in the original input layer are randomly set to be 0 with certain probability. The denoising objective of DAE is actually achieved by adding random noise to the input layer, which is similar to the Dropout [38] technique, but the difference is that Dropout sets the neurons

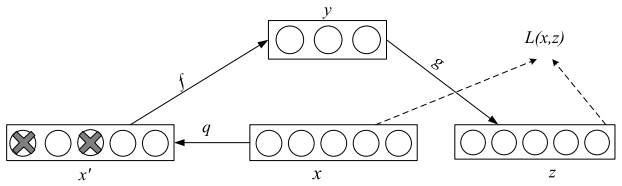


FIGURE 2. The architecture of the denoising auto-encoder.

of hidden layers to be zero. The loss function of the DAE network is defined as follows:

$$y = f(Wx' + b) \tag{4}$$
$$z = g\left(W'y + b'\right) \tag{5}$$
$$\mathrm{L}(x, \mathrm{z}) = \| x - \mathrm{z} \|^2 \tag{6}$$

where x is the input vector; x' is a random vector obeying $q(x'|x)$ distribution in terms of the denoising operation; f is the activation function of the encoder; W and b are the parameters of the encoder; y is the output of the encoder; g is the activation function of the decoder; W' and b' are the parameters of the decoder; z is the output of the decoder; $\mathrm{L}(x, \mathrm{z})$ is the loss function of the DAE network.

Moreover, the stacked denoising autoencoder (SDAE) is a deep neural network (DNN) stacked by multiple denoising auto-encoders. The deep DAE has the advantage to express complex functions, which is obviously superior to the shallow network in extracting effective features. Figure 3 shows a SDAE network with two hidden layers. When training the network of SDAE, we followed the greedy pre-training in layer-by-layer algorithm proposed by Bengio *et al.* [39]. The entire network of SDAE is pre-trained for the first hidden layer $h^{(1)}$ by removing the second hidden layer $h^{(2)}$ and the weights for the first layer are saved. Next, the output of the layer $h^{(1)}$ is further inputted to the second hidden layer $h^{(2)}$, and the weights for the second layer weight are also saved. This is an effective pre-training procedure for training deep networks to better initialize the weights of the network.

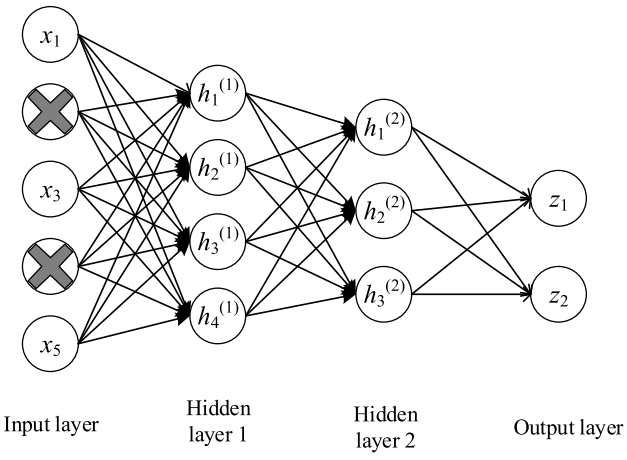


FIGURE 3. The architecture of the stacked denoising auto-encoder.

C. GRADIENT BOOSTING DECISION TREE

Gradient Boosting Decision Tree (GBDT) is a strong classifier by iteratively constructing M weak classifiers [35]. It is composed of multiple additive decision trees using a gradient boosting technique [40] and has been widely applied in various data mining topics [41]–[46], such as credit scoring [44], GPS signal reception classification [45] and feature selection [42]. This algorithm establishes a new decision tree iteratively along the gradient descent direction of the loss function of the previous decision tree model. It has the advantage of discovering a variety of distinguishing features and feature combinations. The GBDT model can be finally expressed as a combination of multiple decision trees as follows:

$$F_M(x) = F_0 + \beta_1 T_1(x) + \beta_2 T_2(x) + \cdots + \beta_M T_M(x) \quad (7)$$

where β_i is the weight, F_0 is the initial value, and T_i represents the decision tree constructed in the ith iteration, $i = 1, 2, \cdots, M$.

III. THE PROPOSED MODELS

A. FEATURE EXTRACTION ALGORITHM BASED ON MULTI-TASK LEARNING

The Wi-Fi signals for one location are usually fluctuating. It is impossible for one location to completely receive the Wi-Fi signals from all of the access points in shopping malls, which leads to sparse RSS vectors due to missing fingerprint data. We use multi-task learning to extract new features from the original sparse RSS vectors for positioning tasks to reduce noise and negative factors caused by fingerprint data volatility and high-dimensional sparsity. Below, the stacked denoising auto-encoder (SDAE) is first introduced, and then a multi-task learning algorithm, called Joint Multi-Task Stacked Denoising Auto-Encoder (JMT-SDAE), is proposed to effectively extract robust features from the original sparse fingerprint data.

Multi-task learning can share representations among related tasks, which may helpfully extract effective features from the original inputs and better improve the performance of the target task. We propose an end-to-end training algorithm, called Joint Multi-Task Stacked Denoising Auto-Encoder (JMT-SDAE), which is based on Stacked Denoising Auto-Encoder (SDAE).

The framework of the JMT-SDAE algorithm is shown in Figure 4. The encoder acts as a parameter sharing layer followed by two tasks including reconstruction and positioning. The reconstruction task is actually a decoder that aims at recovering the original input data ($RSS_1, RSS_2, \cdots, RSS_N$). The positioning task is to find a classifier for RSS based indoor positioning. These two tasks are combined in a way of linear weighted loss functions to perform multi-task learning. The features represented in the encoder after training the JMT-SDAE will be again extracted for further feature fusion task that may improve the generalization performance of the model.

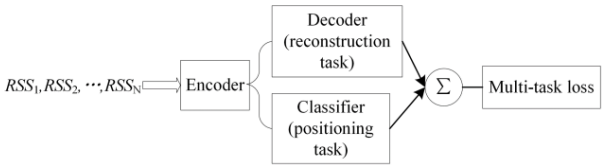


FIGURE 4. The framework of the proposed joint multi-task stacked denoising auto-encoder (JMT-SDAE).

Figure 5 shows the details about the network architecture of the proposed JMT-SDAE. This network is a multi-task learning architecture by combining the SDAE and the store positioning task in order to extract much more robust feature representation. The deep network for store positioning task in shopping malls has 10 layers, including the input layer (i.e. the input x), seven hidden layers (i.e. $h^{(1)}, h^{(2)}, \cdots, h^{(7)}$) and two output layers (i.e. the output y and the output z) as shown in Figure 5. However, the structure of the entire deep network can be divided into three modules, i.e. encoder, decoder, and positioner.

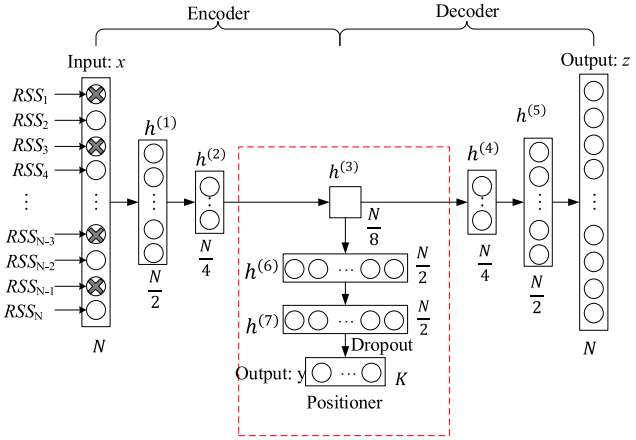


FIGURE 5. The architecture of the proposed JMT-SDAE.

For the encoder, the target location area contains N access points, and the RSS feature vector is x=($RSS_1, RSS_2, \cdots, RSS_N$) corresponding to N neuron nodes for the input layer. The numbers of nodes for the hidden layers $h^{(1)}$, $h^{(2)}$ and $h^{(3)}$ are set to be N/2, N/4 and N/8, respectively. The number of nodes is reduced from the first hidden layer to the third hidden layer by 0.5 times for the purpose of data dimensionality reduction. The value of one node in the input layer is randomly set to be 0 with a probability of 0.7 before the RSS vector $x = (RSS_1, RSS_2, \cdots, RSS_N)$ is inputted to the first hidden layer. The purpose is denoising by adding noise into the input data.

In the decoder module, the numbers of nodes for the fourth hidden layer $h^{(4)}$, the fifth hidden layer $h^{(5)}$ and the output layer z are set to be N/4, N/2 and N, respectively.

In the positioning module, the neuron numbers for the sixth hidden layer $h^{(6)}$, the seventh hidden layer $h^{(7)}$ and the output layer y are set to be N/2, N/2 and K, respectively. The number

K represents there are K stores in the shopping mall. In order to reduce the risk of overfitting, a layer concerning Dropout is connected behind the hidden layer $h^{(7)}$, where the values of the neuron nodes in the hidden layer $h^{(7)}$ are randomly set to be 0 with a probability of 0.5.

B. FEATURE FUSION FOR Wi-Fi FINGERPRINT BASED INDOOR POSITIONING

To achieve more accurate indoor positioning, we further propose a hybrid model by combining the features extracted by the JMT-SDAE network and the gradient boosting decision tree (GBDT). The GBDT models of Wi-Fi indoor localization can be finally expressed as $\{F_{k,M}(x)\}(k = 1, 2, \ldots, K)$, where K is the number of classes (i.e. the number of shops in one training dataset), M is the iteration number using the default value of 100 in practice, and $F_{k,M}(x)$ represents a score that the sample x belongs to the k-th class. All of these scores are mapped to the interval [0, 1] by the softmax function as follows:

$$p_k(x) = \frac{e^{F_{k,M}(x)}}{\sum_{l=1}^{K} e^{F_{l,M}(x)}} \tag{8}$$

where $p_k(x)$ is the probability that the sample x belongs to the k-th class. Thus, a new feature vector composed of K probability values for each sample can be extracted by the GBDT models.

Next, we propose a hybrid model by combining the features extracted by the JMT-SDAE network (*i.e.*, the DNN features for a short) and the GBDT features to improve the generalization ability of the model and reduce the over-fitting risk, as shown in Figure 6. The encoder in the hybrid model is pre-trained by the JMT-SDAE algorithm, and the weights are migrated to the hybrid model. The DNN feature vector is actually the output of the hidden layer $h^{(3)}$ after the JMT-SDAE model was established. The length of the DNN feature vector is N/8. For a sample with unknown label, the input vector $x = (RSS_1, RSS_2, \cdots, RSS_N)$ will be mapped into a feature vector with dimension of N/8 through the encoder of the JMT-SDAE as shown in Figure 5. On the other hand, the K-dimensional probability vector was generated by inputting x into the pre-trained GBDT model that was composed of multiple decision trees like a tree division procedure as shown in Figure 6. Next, this vector was combined with the DNN feature vector, and the fusion of DNN and GBDT features resulted in $(N/8 + K)$-dimensional feature vector. Finally, a new DNN model was established with $(N/8 + K) - N/2 - N/2 - K$ layer structure, including $(N/8 + K)$ input neurons, two hidden layers with N/2 neurons and K output neurons.

IV. EXPERIMENTS

A. DATASETS

The datasets used in this study comes from one of the topics of CCF Big Data and Computing Intelligence Competition in 2017, entitled 'Accurately locate the store where the user is located in the shopping mall'. They can be downloaded from Tianchi Big Data Competition Platform.[1] The platform introduced the details for 97 shopping malls, *i.e.* 97 datasets, including customer positioning behavior and stores in the mall. The goal of this competition is to utilize machine learning algorithm to accurately identify the stores where the customers are. We downloaded the datasets and conducted the computational experiments to verify the performance of the proposed algorithms.

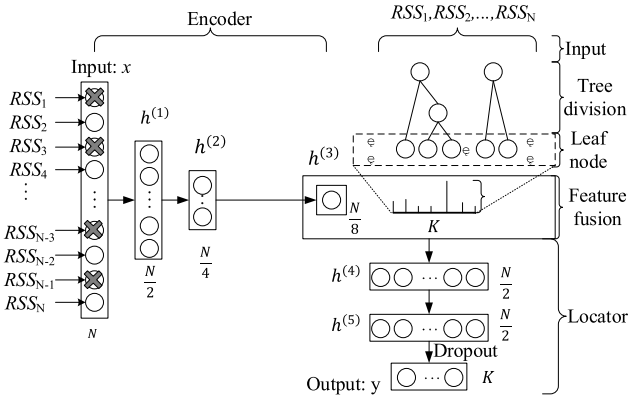


FIGURE 6. The hybrid model by combining DNN features and GBDT features.

TABLE 1. Brief descriptions for the 13 datasets concerning indoor location in shopping malls.

Mall ID	#Samples	#Access points	RSS range	#Shops
m_5076	21373	1364	-101dB~-8dB	132
m_2467	21430	851	-102dB~-3dB	96
m_1175	29894	979	-103dB~-5dB	125
m_3839	22673	1052	-103dB~-1dB	89
m_4094	25372	1063	-108dB~-1dB	78
m_4422	20741	977	-105dB~-4dB	94
m_1293	22280	1376	-110dB~-1dB	108
m_690	26816	1187	-106dB~-3dB	137
m_7168	26355	1054	-108dB~-5dB	99
m_1377	24462	976	-105dB~-1dB	82
m_4079	20619	1049	-107dB~-1dB	91
m_6337	25784	1041	-103dB~-5dB	123
m_3005	21919	1398	-103dB~-4dB	93

The contest provided two kinds of data. One is the basic information about the stores in the mall, and the other is the transaction data of customers in the mall during a time period. In order to protect the privacy of customers and merchants, their names are anonymous and only meaningless IDs are given. In addition, each dataset contains about 5000-30000 records. It is time-consuming to test all 97 datasets to verify the proposed method. In this work, we just selected the datasets with sample numbers larger than 20,000 that resulted in 13 store location datasets. As shown in Table 1, we can observe that the dimension of RSS vectors, *i.e.* the number of access points (APs), ranges from 851 to 1398. However, only 10 valid RSS values are at most included in each RSS vector. A few of samples even contain only one

[1]https://tianchi.aliyun.com/competition/entrance/231620/information?lang=en-us

or two valid RSS values. In other words, the missing rate of RSS values ranges from at least 98.82% (841/851) to at least 99.28% (1388/1398) for any RSS observation. It is concluded that the RSS vectors in these 13 datasets are extremely sparse. Furthermore, we counted the samples with varying numbers of valid RSS values (*see* Table 5 attached in Appendix), and found that the vast majority of samples (94.61% averaged over 13 datasets) include 7 to 10 valid RSS values.

In addition, we computed accuracy as the unique performance measure in this work. This is because our task is a multi-class classification problem, while other popular criteria, including confusion matrix, precision, recall, specificity, F1-score and ROC curve, are usually applicable to binary classification models [47]. The accuracy in this work is defined as follows:

$$Accuracy = \frac{Correctly\ predicted\ number\ of\ samples}{Total\ number\ of\ samples} \quad (9)$$

B. DATA PROCESSING

Irrelevant data, such as in-store transaction data, are not included in our computation. Only the RSS values and location information are retained for model training and test. The RSS values in the 13 datasets range from −110dB to −1dB as shown in Table 1. In general, the effective range of Wi-Fi RSS values is between −120dB and 0dB. The larger the value, the stronger the signal strength. RSS values below −100dB usually mean too weak signals to be detected by most devices [27]. Due to the extreme sparsity of RSS vectors as mentioned above, we replaced all of the missing RSS values by using the value of −105dB. This simple padding way is very fast and applicable to large-scale fingerprint data collected in large shopping malls.

Moreover, data normalization has become a common way to eliminate the influence of the different numerical scales of various features. We adopt the following formula to normalize the original data for each feature:

$$x' = \frac{x - \mu}{\sigma} \quad (10)$$

where μ is the mean value of the feature and σ is the corresponding standard deviation. This normalization method is adopted in both JMT-SDAE and JMT-SDAE+GBDT algorithms proposed in this paper.

C. LOSS FUNCTIONS

We utilized exponential linear unit (ELU) [48] as the activation function in both the DNN feature extraction step of JMT-SDAE and the feature fusion step of JMT-SDAE+GBDT. The ELU function is defined as follows:

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha\,(e^x - 1) & \text{if } x \le 0 \end{cases} \quad (11)$$

where the hyperparameter α is set to 1.

Moreover, two kinds of tasks in the JMT-SDAE network were designed using different types of loss functions. For the reconstruction task in the JMT-SDAE network, it is actually a regression problem. Thus, we adopted the squared error as the loss function, i.e., the sum of the squared distance between the input vector x_i and the output vector z_i of the decoder:

$$L_{reconstruction} = \sum_{i=1}^{N} (x_i - z_i)^2 \quad (12)$$

where N is the number of input samples for training, x_i is the input vector for the ith sample, z_i is the output vector by the decoder, and $L_{reconstruction}$ represents the loss function for the reconstructed task.

The positioner in the JMT-SDAE model actually corresponds to a multi-class classification task for indoor positioning in shopping malls. The Wi-Fi fingerprint information is collected in the mall to determine which store it is, rather than the spatial coordinate. Therefore, the output vector is mapped to a probability vector using the classical softmax function, and then the cross entropy is used as loss function. Suppose the output vector is V before the softmax conversion, and K represents the length of V, i.e. the number of classes in the positioning task. Then, the kth element of the final output vector after softmax conversion is:

$$P_k = \frac{e^{V_k}}{\sum_{i=1}^{K} e^{V_i}} \quad (13)$$

The cross entropy, namely the loss function for the positioning task, is defined as follows:

$$L_{positioning} = -\sum_{i=1}^{N} \sum_{k=1}^{K} y_{ik} log P_{ik} \quad (14)$$

where N is the number of all samples, K is the number of classes, y_{ik} represent the real label of class k of the ith sample in the training dataset, P_{ik} means the predicted probability value of the model for the ith sample belonging to the class k.

In order to extract more discriminative features, the positioning task and the reconstruction task are combined to mine the correlation between multiple tasks. Meanwhile, the discriminative features are extracted by sharing the encoder parameters to reduce the over-fitting risk and improve the generalization ability of the model. The final loss function of this joint multi-task learning is linearly combined as follows:

$$L = \alpha L_{reconstruction} + (1 - \alpha) L_{positioning} \quad (15)$$

where $L_{reconstruction}$ is the loss function of the reconstruction task, $L_{positioning}$ is the loss function of the positioning task, and α is the weight of the linear combination. In our computational experiments, we set α to be 0.5 optimized by a grid search on α values from 0.1 to 0.9 with step size of 0.1. In the feature fusion model, i.e., JMT-SDAE+GBDT, we also adopted cross entropy as the loss function.

D. TRAINING

In the feature fusion model, i.e. JMT-SDAE+GBDT as shown in Figure 6, the network weights for the encoder were initialized using the pre-trained parameters by training the JMT-SDAE model. Furthermore, we followed the initialization method proposed by Glorot and Bengio [49] to initialize

the network weights for the positioner. This method utilizes random parameter initialization according to a uniform distribution in terms of the numbers of input and output neurons of each layer. This uniform distribution can be written as follows:

$$\mathrm{W} \sim U[-\frac{\sqrt{6}}{\sqrt{n_{in}+n_{out}}}, \frac{\sqrt{6}}{\sqrt{n_{in}+n_{out}}}] \tag{16}$$

where n_{in} and n_{out} are the numbers of neurons for the input and output layers, respectively. This initialization scheme can also greatly reduce the computational time for the entire network training.

The entire network for JMT-SDAE+GBDT was finally trained by the Adam (Adaptive Moment Estimation) [50] algorithm, which can calculate adaptive learning rate for gradient back propagation. In order to avoid the overfitting risk of the proposed JMT-SDAE+GBDT, five-fold cross-validation was performed on all 13 location datasets.

V. RESULTS

A. PERFORMANCE COMPARISON OF JMT-SDAE WITH OTHER POSITIONERS

In order to verify the effectiveness of the proposed JMT-SDAE algorithm on extracting features from high-dimensional and sparse Wi-Fi fingerprint data, we compared our method (JMT-SDAE) with several other positioners, including Gaussian naïve Bayes (GNB), K-nearest neighbor (KNN), decision tree (DT), deep neural network (DNN), and SDAE+DNN. The DNN method has a network structure of $N-\mathrm{N}/2-\mathrm{N}/4-\mathrm{N}/8-\mathrm{N}/2-\mathrm{N}/2-K$, where N is the dimension of the input vector and K is the number of stores in one shopping mall. The difference between the DNN and the proposed JMT-SDAE is that the former does not include decoder module. The network structure of SDAE+DNN is similar to the JMT-SDAE, but their training ways are different. The proposed JMT-SDAE adopts joint multi-task training the training procedure of SDAE+DNN was divided into two stages. The first stage is to train the independent SDAE, and the second stage is to perform global fine-tuning for SDAE+DNN based on the pre-trained SDAE. The GNB, DT, and KNN algorithms are directly implemented by scikit-learn [51], while DNN, SDAE+DNN and JMT-SDAE are implemented using the Tensorflow [52]. These methods were compared based on 13 location datasets as listed in Table 1. In order to avoid overfitting risk, all computational experiments were tested with five-fold cross-validation. That is to say, the average accuracy evaluated on the five folds is used as the performance measure on the considered models. The accuracy values of the above-mentioned positioners on the 13 datasets are listed as shown in Table 2 as well as Figure 7 for intuitive comparison.

Table 2 and Figure 7 show that the positioning accuracies of the proposed JMT-SDAE on 13 datasets are higher than those of GNB by 20%∼50%, KNN by 5%∼20%, DT by 5%∼10%, DNN by 20%∼30%, and SDAE + DNN by 2%∼3%. Specially, the improvement achieved by the proposed JMT-SDAE when compared with SDAE+DNN implies that the multi-task learning in JMT-SDAE is effective.

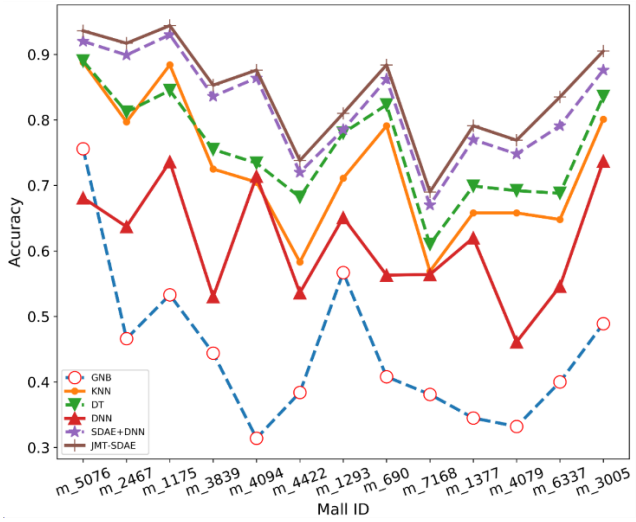


FIGURE 7. Performance comparison of the proposed JMT-SDAE with other different positioners.

TABLE 2. Performance comparison of the proposed JMT-SDAE with other different positioners, including GNB, KNN, DT, DNN and SDAE+DNN.

Mall ID	GNB	KNN	DT	DNN	SDAE+DNN	JMT-SDAE
m_5076	0.756	0.888	0.890	0.681	0.920	**0.936**
m_2467	0.466	0.797	0.812	0.637	0.899	**0.917**
m_1175	0.533	0.884	0.845	0.736	0.930	**0.944**
m_3839	0.444	0.725	0.755	0.530	0.836	**0.853**
m_4094	0.314	0.705	0.734	0.714	0.864	**0.876**
m_4422	0.384	0.583	0.682	0.536	0.720	**0.738**
m_1293	0.567	0.711	0.780	0.651	0.785	**0.810**
m_690	0.408	0.791	0.823	0.563	0.862	**0.884**
m_7168	0.381	0.569	0.610	0.564	0.670	**0.690**
m_1377	0.345	0.658	0.699	0.620	0.770	**0.791**
m_4079	0.332	0.658	0.692	0.461	0.748	**0.769**
m_6337	0.400	0.648	0.688	0.546	0.791	**0.835**
m_3005	0.489	0.801	0.836	0.737	0.876	**0.905**

B. HYBRID MODELS BY MIXING DNN AND GBDT FEATURES

To improve the positioning performance, we further proposed two hybrid models by fusing DNN and GBDT features, as shown in Figure 6. The first one, named as JMT-SDAE+GBDT, is the mixture of JMT-SDAE and GBDT where the encoder parameters were initialized by the pre-training procedure of the proposed JMT-SDAE. The second one, named as REN+GBDT, is a hybrid model of the random encoder (REN) and the GBDT where the parameters for the encoder part are randomly initialized. Therefore, we compared the positioning performance of four positioners,

including JMT-SDAE+GBDT, REN+GBDT, JMT-SDAE and GBDT. All computational experiments on 13 datasets were verified by five-fold cross validation. The accuracy values of JMT-SDAE+GBDT, REN+GBDT, JMT-SDAE and GBDT on the 13 datasets were shown in Table 3 as well as Figure 8 for intuitive comparison.

TABLE 3. Performance comparison of the proposed JMT-SDAE+GBDT with other positioners, including REN+GBDT, JMT-SDAE and GBDT.

Mall ID	GBDT	JMT-SDAE	REN+GBDT	JMT-SDAE +GBDT
m_5076	0.911	0.936	0.930	**0.947**
m_2467	0.908	0.917	0.910	**0.935**
m_1175	0.918	0.944	0.936	**0.953**
m_3839	0.84	0.853	0.855	**0.863**
m_4094	0.865	0.876	0.869	**0.886**
m_4422	0.714	0.738	0.721	**0.750**
m_1293	0.780	0.810	0.790	**0.820**
m_690	0.861	0.884	0.876	**0.889**
m_7168	0.664	0.690	0.683	**0.705**
m_1377	0.772	0.791	0.784	**0.801**
m_4079	0.753	0.769	0.772	**0.791**
m_6337	0.779	0.835	0.820	**0.840**
m_3005	0.877	0.905	0.885	**0.911**

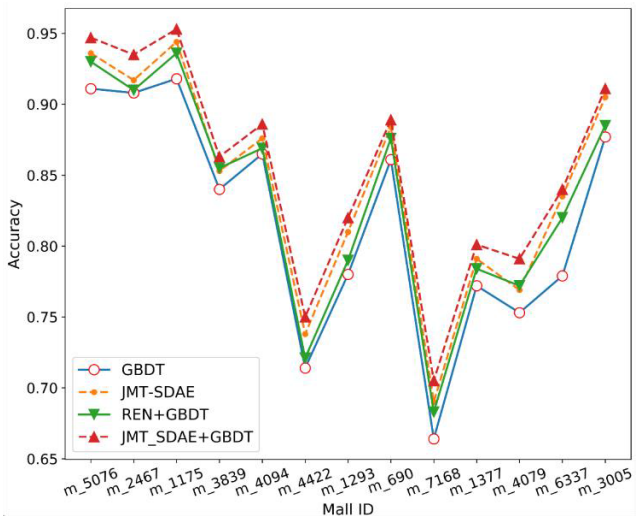


FIGURE 8. Performance comparison of four positioners including JMT-SDAE+GBDT, REN+GBDT, JMT-SDAE and GBDT that were validated on 13 location datasets.

Figure 8 shows that the classification accuracy of the hybrid model with pre-training, i.e. JMT-SDAE+GBDT, is higher than that of GBDT about 3%-5%, JMT-SDAE about 1%-2%. The improvements of JMT-SDAE+GBDT are consistent on the 13 location datasets, which implies that the fusion of DNN features and GBDT features is useful to enhance the indoor positioning performance. On the other hand, the improvement when comparing JMT-SDAE+GBDT with REN+GBDT again confirmed the effectiveness of the feature extraction using pre-training way in JMT-SDAE.

C. POSITIONING MODELS WITH PCA DIMENSIONALITY REDUCTION

The lately proposed AE algorithm has attract wide applications to nonlinear feature fusion studies [53]. Due to the advantage of the nonlinear dimensionality reduction ability, AE has been shown the superiority when compared with linear methods, such as principal component analysis (PCA) [54] in several studies [53]. In this work, we also designed two models by using PCA, as a representative of linear feature extraction methods, for practical parallel comparisons with the proposed models concerning the stacked denoising auto-encoder. The first one is named as PCNN with the meaning of PCA based DNN classifier that the new reduced features extracted by PCA are the inputs of the DNN classifier. This is intended to show a direct comparison with the proposed model JMT-SDAE. The other one is called PCA+GBDT meaning the combination of new features extracted by PCA and GBDT as the inputs of the DNN model. This second model is intended for a parallel comparison with the final proposed classifier, *i.e.* JMT-SDAE+GBDT. The accuracy values of the classification models PCNN and PCA+GBDT are listed in Table 4.

TABLE 4. Parallel performance comparisons of the proposed models with PCA based classifiers, including PCNN and PCA+GBDT.

Mall ID	PCNN	JMT-SDAE	PCA+GBDT	JMT-SDAE+GBDT
m_5076	0.932	0.936	0.939	0.947
m_2467	0.915	0.917	0.929	0.935
m_1175	0.937	0.944	0.944	0.953
m_3839	0.844	0.853	0.86	0.863
m_4094	0.872	0.876	0.871	0.886
m_4422	0.717	0.738	0.745	0.75
m_1293	0.812	0.81	0.816	0.82
m_690	0.877	0.884	0.887	0.889
m_7168	0.676	0.69	0.685	0.705
m_1377	0.788	0.791	0.798	0.801
m_4079	0.774	0.769	0.778	0.791
m_6337	0.84	0.835	0.841	0.84
m_3005	0.893	0.905	0.904	0.911

As shown in Table 4, the proposed method JMT-SDAE was achieved by higher accuracy values on most datasets except the comparable datasets from three malls m_1293, m_6337 and m_4079 when compared with the classification model PCNN. Moreover, the proposed model JMT-SDAE+GBDT outperformed the parallel method PCA+GBDT based on 12 datasets, except the comparable case of the mall m_6337. As a summary, the nonlinear feature fusion using SDAE is more recommendable when compared

with the linear feature extraction using PCA in context of Wi-Fi fingerprint data with extreme sparsity.

VI. CONCLUSION

Public Wi-Fi has been available just about everywhere, from the local coffee shops to the hotels and airports you visit while traveling. Hence, positioning technique based on Wi-Fi fingerprints has become one of the promising indoor positioning approaches. It has the advantages with low cost, wide signal coverage and high positioning accuracy. However, factors such as co-frequency interference, complex indoor environment and human activities bring various attenuation effects to the RSS signals, making Wi-Fi signals fluctuate which decreases the accuracy of indoor positioning. In this work, according to the fluctuation characteristics of Wi-Fi signals and the high-dimensional sparseness of fingerprint data in a wide range of localization areas, we extract effective DNN features and reduce data dimensionality with joint multi-task learning by combining an auto-encoder and a classification task. As a result, DNN features can be effectively extracted from this method called JMT-SDAE. Furthermore, a hybrid model by mixing DNN features and GBDT features is proposed in this study.

The proposed JMT-SDAE algorithm exploits the correlation between reconstruction task and positioning task. The robust features extracted by denoising auto-encoder could effectively reduce the noise and abundant information in the RSS data, and improve the performance of indoor positioning. The computational experiments show that the positioning performance of the proposed JMT-SDAE algorithm is significantly better than other methods, including GNB, KNN, DT, DNN, and SDAE+DNN.

Moreover, we proposed JMT-SDAE+GBDT, which is a feature fusion model that combines DNN features extracted by the proposed JMT-SDAE and GBDT features extracted by the GBDT algorithm. The computational experiments show that the proposed hybrid model JMT-SDAE+GBDT has higher positioning performance than other algorithms, such as GBDT, JMT-SDAE and REN+GBDT. When compared with the proposed JMT-SDAE and another feature fusion method REN+GBDT without pre-training in the encoder part, the proposed final model, named as JMT-SDAE+GBDT, significantly improve the positioning accuracy, which again confirms the effectiveness of data dimensionality reduction and feature extraction of the JMT-SDAE algorithm.

Finally, parallel comparisons of the proposed models with the PCA based classifiers were also performed. The experimental results show that the proposed nonlinear feature extraction method using SDAE exhibits better performance than the linear approach using PCA in majority of datasets considered. The present study provides new helpful guidelines for reducing the dimensionality and the noise of fingerprint data with extreme sparsity collected in large buildings.

APPENDIX

See Table 5.

TABLE 5. Counts of samples with valid RSS values of observable APs in the 13 shopping malls.

Mall ID	Count of samples with v RSS values of observable APs					Percent of samples for $v \geq 7$
	$v \leq 2$	$v = 3,4$	$v = 5,6$	$v = 7,8$	$v = 9,10$	
m_5076	468	801	2042	4401	13661	84.51%
m_2467	180	117	391	2625	18117	96.79%
m_1175	419	605	1582	3671	23617	91.28%
m_3839	298	93	292	1862	20128	96.99%
m_4094	175	108	288	2199	22602	97.75%
m_4422	168	68	267	2201	18037	97.57%
m_1293	308	413	1638	4376	15545	89.41%
m_690	452	178	730	4862	20594	94.93%
m_7168	174	94	506	3562	22019	97.06%
m_1377	157	157	363	2454	21331	97.23%
m_4079	211	104	397	4219	15688	96.55%
m_6337	242	343	754	3007	21438	94.81%
m_3005	371	166	550	3160	17672	95.04%

REFERENCES

[1] C. Cai, Y. Gao, L. Pan, and J. Zhu, "Precise point positioning with quad-constellations: GPS, BeiDou, GLONASS and Galileo," *Adv. Space Res.*, vol. 56, no. 1, pp. 133–143, Jul. 2015.

[2] L. Pan, X. Zhang, X. Li, X. Li, C. Lu, J. Liu, and Q. Wang, "Satellite availability and point positioning accuracy evaluation on a global scale for integration of GPS, GLONASS, BeiDou and Galileo," *Adv. Space Res.*, vol. 63, no. 9, pp. 2696–2710, May 2019.

[3] G. Dedes and A. G. Dempster, "Indoor GPS positioning–challenges and opportunities," in *Proc. VTC–Fall. IEEE 62nd Veh. Technol. Conf.*, vol. 1, Sep. 2005, pp. 412–415.

[4] S. Nirjon, J. Liu, G. DeJean, B. Priyantha, Y. Jin, and T. Hart, "COIN-GPS: Indoor localization from direct GPS receiving," in *Proc. 12th Annu. Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, Bretton Woods, NH, USA, Jun. 2014, pp. 301–314.

[5] B. Molina, E. Olivares, C. E. Palau, and M. Esteve, "A multimodal fingerprint-based indoor positioning system for airports," *IEEE Access*, vol. 6, pp. 10092–10106, 2018.

[6] S. Nirjon, J. Liu, B. Priyantha, and G. DeJean, "High-sensitivity cloud-offloaded instant GPS for indoor environments," in *Proc. 11th ACM Conf. Embedded Netw. Sensor Syst. (SenSys)*, Roma, Italy, Nov. 2013, pp. 1–2.

[7] P. Huang and Y. Pi, "Urban environment solutions to GPS signal near-far effect," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 26, no. 5, pp. 18–27, May 2011.

[8] N. S. Gowdayyanadoddi, A. Broumandan, G. Lachapelle, and J. T. Curran, "Indoor GPS positioning using a slowly moving antenna and long coherent integration," in *Proc. Int. Conf. Location GNSS (ICL-GNSS)*, Jun. 2015, pp. 1–6.

[9] W. Kang and Y. Han, "SmartPDR: Smartphone-based pedestrian dead reckoning for indoor localization," *IEEE Sensors J.*, vol. 15, no. 5, pp. 2906–2916, May 2015.

[10] Y. Zhuang, H. Lan, Y. Li, and N. El-Sheimy, "PDR/INS/WiFi integration based on handheld devices for indoor pedestrian navigation," *Micromachines*, vol. 6, no. 6, pp. 793–812, Jun. 2015.

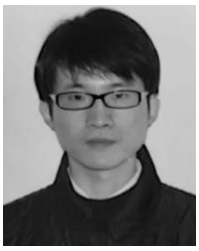
[11] Y. Li, Y. Zhuang, P. Zhang, H. Lan, X. Niu, and N. El-Sheimy, "An improved inertial/WiFi/magnetic fusion structure for indoor navigation," *Inf. Fusion*, vol. 34, pp. 101–119, Mar. 2017.

[12] L. Chen, H. Kuusniemi, Y. Chen, L. Pei, T. Kroger, and R. Chen, "Information filter with speed detection for indoor Bluetooth positioning," in *Proc. Int. Conf. Localization GNSS (ICL-GNSS)*, Jun. 2011, pp. 47–52.
[13] G. De Blasio, A. Quesada-Arencibia, C. R. Garcia, J. C. Rodriguez-Rodriguez, and R. Moreno-Diaz, "A protocol-channel-based indoor positioning performance study for Bluetooth low energy," *IEEE Access*, vol. 6, pp. 33440–33450, 2018.
[14] C. D. McGillem and T. S. Rappaport, "Infra-red location system for navigation of autonomous vehicles," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 2, Apr. 1988, pp. 1236–1238.
[15] M. M. Saad, C. J. Bleakley, T. Ballal, and S. Dobson, "High-accuracy reference-free ultrasonic location estimation," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 6, pp. 1561–1570, Jun. 2012.
[16] Z. Wang, S. Li, Z. Zhang, F. Lv, and Y. Hou, "Research on UWB positioning accuracy in warehouse environment," *Procedia Comput. Sci.*, vol. 131, pp. 946–951, May 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050918306112
[17] R. Tesoriero, R. Tebar, J. A. Gallud, M. D. Lozano, and V. M. R. Penichet, "Improving location awareness in indoor spaces using RFID technology," *Expert Syst. Appl.*, vol. 37, no. 1, pp. 894–898, Jan. 2010.
[18] A. Makki, A. Siddig, M. Saad, and C. Bleakley, "Survey of WiFi positioning using time-based techniques," *Comput. Netw.*, vol. 88, pp. 218–233, Sep. 2015.
[19] S. Eisa, J. Peixoto, F. Meneses, and A. Moreira, "Removing useless APs and fingerprints from WiFi indoor positioning radio maps," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Oct. 2013, pp. 1–7.
[20] K. Dong, Z. Ling, X. Xia, H. Ye, W. Wu, and M. Yang, "Dealing with insufficient location fingerprints in Wi-Fi based indoor location fingerprinting," *Wireless Commun. Mobile Comput.*, vol. 2017, Aug. 2017, Art. no. 1268515.
[21] J. Xu, M. Ma, and C. L. Law, "AOA cooperative position localization," in *Proc. IEEE GLOBECOM-IEEE Global Telecommun. Conf.*, Nov. 2008, pp. 1–5.
[22] M. Dakkak, A. Nakib, B. Daachi, P. Siarry, and J. Lemoine, "Indoor localization method based on RTT and AOA using coordinates clustering," *Comput. Netw.*, vol. 55, no. 8, pp. 1794–1803, Jun. 2011.
[23] J. Hu, L. Xie, J. Xu, and Z. Xu, "TDOA-based adaptive sensing in multi-agent cooperative target tracking," *Signal Process.*, vol. 98, pp. 186–196, May 2014.
[24] B. Wang, S. Zhou, W. Liu, and Y. Mo, "Indoor localization based on curve fitting and location search using received signal strength," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 572–582, Jan. 2015.
[25] B. Wang, S. Zhou, L. T. Yang, and Y. Mo, "Indoor positioning via subarea fingerprinting and surface fitting with received signal strength," *Pervasive Mobile Comput.*, vol. 23, pp. 43–58, Oct. 2015.
[26] P. Barsocchi, S. Lenzi, S. Chessa, and F. Furfari, "Automatic virtual calibration of range-based indoor localization systems," *Wireless Commun. Mobile Comput.*, vol. 12, no. 17, pp. 1546–1557, Dec. 2012.
[27] R. Santos, M. Barandas, R. Leonardo, and H. Gamboa, "Fingerprints and floor plans construction for indoor localisation based on crowdsourcing," *Sensors*, vol. 19, no. 4, p. 919, Feb. 2019.
[28] M. Laska, J. Blankenbach, and R. Klamma, "Adaptive indoor area localization for perpetual crowdsourced data collection," *Sensors*, vol. 20, no. 5, p. 1443, Mar. 2020.
[29] Y. Ye and B. Wang, "RMapCS: Radio map construction from crowdsourced samples for indoor localization," *IEEE Access*, vol. 6, pp. 24224–24238, 2018.
[30] Z. Gu, Z. Chen, Y. Zhang, Y. Zhu, M. Lu, and A. Chen, "Reducing fingerprint collection for indoor localization," *Comput. Commun.*, vol. 83, pp. 56–63, Jun. 2016.
[31] H.-X. Liu, B.-A. Chen, P.-H. Tseng, K.-T. Feng, and T.-S. Wang, "Map-aware indoor area estimation with shortest path based on RSS fingerprinting," in *Proc. IEEE 81st Veh. Technol. Conf. (VTC Spring)*, Glasgow, U.K., May 2015, pp. 1–5.
[32] Y. Rezgui, L. Pei, X. Chen, F. Wen, and C. Han, "An efficient normalized rank based SVM for room level indoor WiFi localization with diverse devices," *Mobile Inf. Syst.*, vol. 2017, pp. 1–19, Jul. 2017.
[33] K.-H. Chow, S. He, J. Tan, and S.-H.-G. Chan, "Efficient locality classification for indoor fingerprint-based systems," *IEEE Trans. Mobile Comput.*, vol. 18, no. 2, pp. 290–304, Feb. 2019.
[34] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
[35] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting (with discussion and a rejoinder by the authors)," *Ann. Statist.*, vol. 28, no. 2, pp. 337–407, Apr. 2000.
[36] J. Deng, Z. Zhang, F. Eyben, and B. Schuller, "Autoencoder-based unsupervised domain adaptation for speech emotion recognition," *IEEE Signal Process. Lett.*, vol. 21, no. 9, pp. 1068–1072, Sep. 2014.
[37] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, New York, NY, USA, 2008, pp. 1096–1103.
[38] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[39] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Proc. Neural Inf. Process. Syst.*, 2006, pp. 153–160.
[40] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, 2001.
[41] C. Krauss, X. A. Do, and N. Huck, "Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500," *Eur. J. Oper. Res.*, vol. 259, no. 2, pp. 689–702, Jun. 2017.
[42] H. Rao, X. Shi, A. K. Rodrigue, J. Feng, Y. Xia, M. Elhoseny, X. Yuan, and L. Gu, "Feature selection based on artificial bee colony and gradient boosting decision tree," *Appl. Soft Comput.*, vol. 74, pp. 634–642, Jan. 2019.
[43] L. Guelman, "Gradient boosting trees for auto insurance loss cost modeling and prediction," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 3659–3667, Feb. 2012.
[44] H. Zhang, H. He, and W. Zhang, "Classifier selection and clustering with fuzzy assignment in ensemble model for credit scoring," *Neurocomputing*, vol. 316, pp. 210–221, Nov. 2018.
[45] R. Sun, G. Wang, W. Zhang, L.-T. Hsu, and W. Y. Ochieng, "A gradient boosting decision tree based GPS signal reception classification algorithm," *Appl. Soft Comput.*, vol. 86, Jan. 2020, Art. no. 105942.
[46] W. Wang, T. Li, W. Wang, and Z. Tu, "Multiple fingerprints-based indoor localization via GBDT: Subspace and RSSI," *IEEE Access*, vol. 7, pp. 80519–80529, 2019.
[47] S. García, A. Fernández, J. Luengo, and F. Herrera, "A study of statistical techniques and performance measures for genetics-based machine learning: Accuracy and interpretability," *Soft Comput.*, vol. 13, no. 10, p. 959, Dec. 2008.
[48] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," 2015, *arXiv:1511.07289*. [Online]. Available: http://arxiv.org/abs/1511.07289
[49] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, Mar. 2010, pp. 249–256.
[50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[51] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and J. Vanderplas, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.
[52] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*. [Online]. Available: http://arxiv.org/abs/1603.04467
[53] D. Charte, F. Charte, S. García, M. J. del Jesus, and F. Herrera, "A practical tutorial on autoencoders for nonlinear feature fusion: Taxonomy, models, software and guidelines," *Inf. Fusion*, vol. 44, pp. 78–96, Nov. 2018.
[54] J. Lever, M. Krzywinski, and N. Altman, "Principal component analysis," *Nature Methods*, vol. 14, pp. 641–642, Jun. 2017.

HUA ZHANG is currently an Associate Professor with the School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou, China. His research interests include data mining, machine learning, natural language processing, and bioinformatics.

BIAO HU received the B.S. degree from the Wuhan Institute of Technology, Wuhan, China, in 2018. He is currently pursuing the master's degree with the School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou, China. His current research interests include data mining, indoor positioning, and natural language processing.

SHIQI XU received the B.S. degree from the Ningbo University of Technology, Ningbo, China, in 2016, and the M.S. degree from the School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou, China, in 2019. His research interests include deep learning and data mining.



BI CHEN received the Ph.D. degree in information science from Pennsylvania State University, USA, in 2012. He is currently is a Lecturer with Zhejiang Gongshang University, China. His research interests include natural language processing and so on.

MIAN LI received the B.S. degree from the Zhejiang University of Science and Technology, Hangzhou, China, in 2016, and the M.S. degree from Zhejiang Gongshang University, Hangzhou, in 2020. His research interests include data mining, natural language processing, and sentiment analysis.

BO JIANG is currently a Professor with the School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou, China. Her main research interests include machine learning and knowledge graph.