ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 7, Issue 4, pp: (138-145), Month: October - December 2019, Available at: www.researchpublish.com

Identification of buildings and floors based on SCAE-DNN

Chunkai Chen[1], Baocong Liu[2]

[12]Electronic and Information Engineering

[12]TianGong University, Tianjin, China

Abstract: **Outdoor positioning mainly relies on the GPS system, but GPS signals can be severely affected indoors and in dense building group. Many indoor areas have coved with WiFi due to the development of WLAN technology, so no additional beacons are needed for positioning. However, due to the complex indoor environment and the susceptibility of WiFi signals, indoor positioning based on WiFi fingerprints is facing great challenges. Therefore, we proposed a method to identify the buildings and its floors based on SCAE-DNN. The Stacked Contractive Auto-Encoder network (SCAE) is added to the Deep Neural Network (DNN). In the offline phase, SCAE is used to extract the characteristics of the WiFi signal strength as the input to the DNN. In the online phase, the network is trained to identify buildings and floors. The SCAE-DNN network is robust to the fluctuations of input signal, so the recognition accuracy is high. By using the public data set UJIIndoorLoc, the results shows that the accuracy achieves 99.7% by utilizing SCAE-DNN.**

Keywords **Indoor location, WiFi location fingerprint, Deep Neural Network, Stacked Contractive Auto-encoders.**

I. INTRODUCTION

In the past decades, indoor positioning has been widely studied, mainly in industrial environment, wireless sensor network and robot technology. With the development of various emerging industries, there is a great demand for indoor positioning navigation in the medical industry, business, disaster management and construction management. According to a report from Market Watch, the demand for indoor navigation is expected to grow by 30% in 2022.

At present, the floor positioning mainly depends on sensors, such as the air pressure sensor floor determination [1] proposed by Teng in 2019. In order to achieve accurate positioning, the main solution depends on filtering, manual data analysis and time-consuming parameter adjustment. In large buildings, the most accurate position estimation is obtained by laser scanner, passive camera or active RDB-D sensor [2]. In addition, in 2016, Zhou Proposed a WLAN floor location method based on hierarchical clustering, which can improve the accuracy of floor location and can quick locate the user's floor [3]. In 2019, based on hierarchical clustering (HC), Mao and others proposed an improved HC algorithm [4]. According to the survey, the positioning of buildings and floors is usually correctly predicted in 85% - 95% cases [5], but if the buildings are large and can obtain a large amount of data, these solutions are difficult to adjust. Therefore, the location method based on deep learning is a very promising solution, because in a larger environment, the parameter adjustment is less and the scalability is better, such as the random forest method [6] has achieved some success.

II. RELATED WORK

A. Auto Encoder

Auto encoder (AE) is mainly used in the data processing in the early stage of neural network training. The original purpose of auto encoder is to reduce the dimension of data, which is similar to PCA, but the principle is different. AE is a method to reconstruct the input vector by using neural network method, so as to obtain the dimension reduction vector of hidden layer as the input vector of training network, which is equivalent to instead of processing the original data manually. It can extract features from the data with unclear features. The schematic is shown in Fig.1. It includes the encoding and decoding part. The decoding part will be removed after the training, leaving the hidden layer of the two nodes. The whole workflow is an unsupervised learning process without label data. The input vector maps the input to the hidden layer through the encoder. The encoder is a mapping function f about the input $x \in R^{dx}$ and the hidden layer $h(x) \in R^{dh}$:

$$h = f(x) = s_f(W_x + b_h)$$

Where s_f is a nonlinear activation function, the activation function is generally S-shaped. The weight is a $d_h \times d_x$ matrix, $b_h \in R^{dh}$ is a deviation vector. The decoder reconstructs the input vector through the low dimension vector of the hidden layer. The decoder function g :

$$x' = g(h) = s_g(W'h + b_{x'})$$

The parameters are similar to those of the previous layer, but they can also be different, but in general $W' = W^T$. The training process is the process of continuously adjusting the weights and deviation matrix of the above parameters, which is realized by minimizing the error function. The objective function is expressed as follows:

$$J_{AE}(\theta) = \sum_{x \in D_n} L(x, x')$$

Where $\theta = \{W, b_h, b_y\}$, D_n is the sample set, $L(x, x') = \|x - x'\|^2$ is the square error. If the activation function is an S-shaped function, it can be expressed as:

$$L(x, x') = -\sum_{i=1}^{d_x} x_i \log(y_i) + (1 - x_i)\log(1 - y_i)$$

Through the above analysis, the x' can be expressed as $g(f(x))$.

There are many varieties of Auto encoder, which are basically achieved by adding noise to the input or by changing the objective function. The simplest form of regularization is attenuation weight, which is smaller by replacing the optimization of the following regularization target priority weight:

$$J_{AE+wd}(\theta) = \left(\sum_{x \in D_n} L(x, g(f(x))) \right) + \lambda \sum_{ij} W_{ij}^2$$

It is used λ to control the strength of the limit, where the penalty function directly controls the weight. It can be seen that it is possible to replace the weight of the control input layer with the activation unit of the control hidden layer, so a sparse coding algorithm appears.

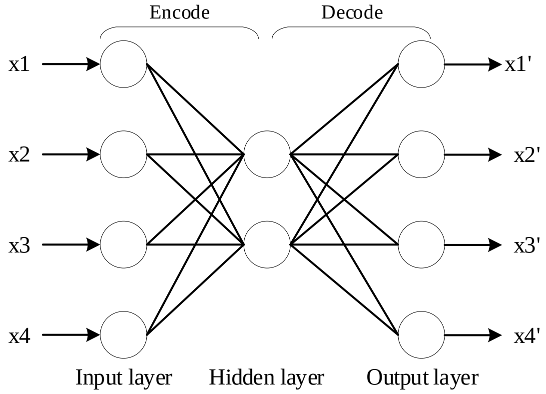

Fig.1 the structure of Auto encoder

Inspired by regularized auto encoder compressed auto encoder was proposed [7]. Unlike RAE, in the case of linearity, CAE maintains the only way to shrink the weight when it is small. When nonlinear, it can be achieved by moving the hidden layer Its saturation state to achieve shrinkage and robustness. This is achieved by adding a penalty function to its input sensitivity, which is represented by the F-norm of the non-linear mapping of the Jacobian norm:

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 7, Issue 4, pp: (138-145), Month: October - December 2019, Available at: www.researchpublish.com

$$\|J_f(x)\|_F^2 = \sum_{ij}\left(\frac{\partial h_j(x)}{\partial x_i}\right)$$

Where h is the hidden layer reconstruction function obtained by the input training set x through the encoding function $f(x)$. This penalty term is the sum of the squares of all the partial derivatives of the extracted features relative to the input dimensions. Its actual role is to shrink the mapping of the feature space near the training data, which is also the origin of "Contractive". The objective function of CAE can be expressed as:

$$J_{CAE}(\theta) = \sum_{x \in D_n}\left(L\left(x, g\left(f(x)\right)\right) + \lambda \|J_f(x)\|_F^2\right)$$

CAE can improve the robustness of the reconstruction function from the input training set, that is, when the input is slightly disturbed, the reconstruction function is invariant regardless of the direction of the disturbance. The reason for this characteristic is related to the Jacobian matrix. Here are some understandings about CAE.

B. The reconstruction process of SCAE

The reconstruction effect of single-layer CAE is not good, so in this study SCAE is used. As mentioned in Sec A, AE is a single-layer instant form $x->h->x$. What we needed are only the input layer and the hidden layer. After the output of the hidden layer is obtained, the new encoder can be input as the original information to get a new feature expression. Similar to a stack, the whole system diagram is shown in Fig.2.

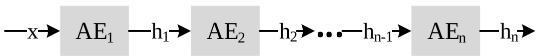

Fig.2 the structure of SAE

The flow chart of RSSI reconstruction method in this paper is shown in Fig.3. The left side of the dotted line in the figure is the structural flow of trestle compression auto encoder, which is the process of reconstructing and outputting multiple RSSI measured at the same location. After unsupervised learning, the hidden layer of SCAE after reconstruction is the representation of RSSI's high-dimensional features. The role of SCAE in WiFi signal feature extraction is that when the signal strength at the same location is different When input, the reconstructed eigenvectors are the same or similar, which effectively solves the problem of signal volatility. Therefore, when the unknown location is input, the high-dimensional connection with the fingerprint in the fingerprint database can be found after reconstruction, and the recognition accuracy is higher. On the right side of the dotted line is the working principle of single-layer CAE, the same as AE principle, which is to modify network parameters by minimizing reconstruction error.

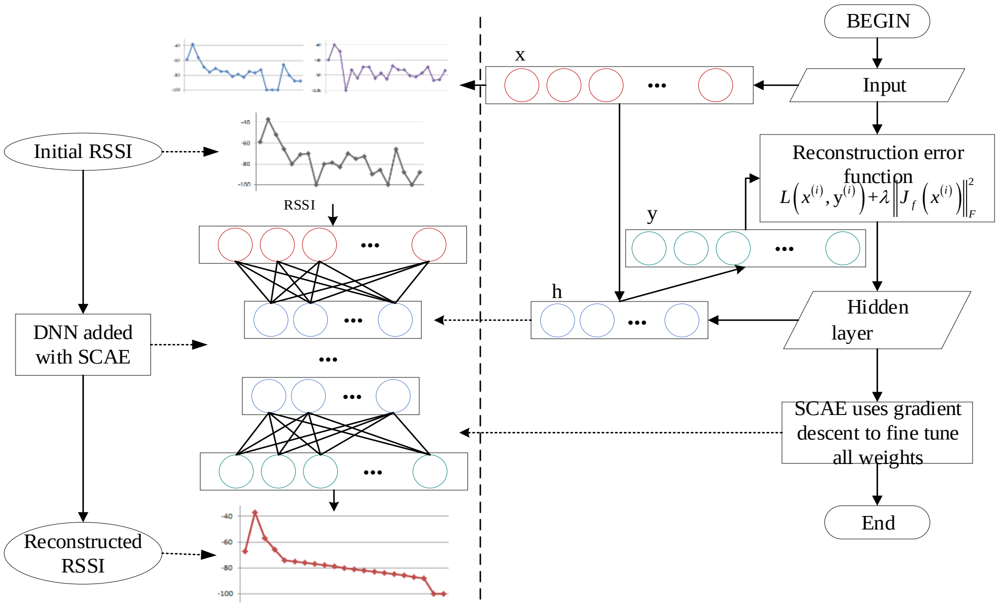

Fig.3 Schematic diagram of SCAE reconstruction RSSI

III. SYSTEM MODEL

Each RP can measure the signal strength of several available APS nearby, but the detected AP is only a small part of the AP in the whole WiFi network environment, and the fingerprint of each RP has a certain correlation, but not a complete correlation. Therefore, it is difficult to use machine learning or traditional algorithms to achieve reduced high-dimensional features. As discussed above, SCAE can complete the feature extraction task, as shown in Fig.4. The red part is the encoder, and the green part is the decoder. The purpose of learning SCAE during unsupervised training is to train the encoder decoder to obtain the same information at the output as provided input. Since the dimension of the layer between the encoder and decoder is smaller than the size of the input vector, the network must learn the high-dimensional representation of the information provided at the input.

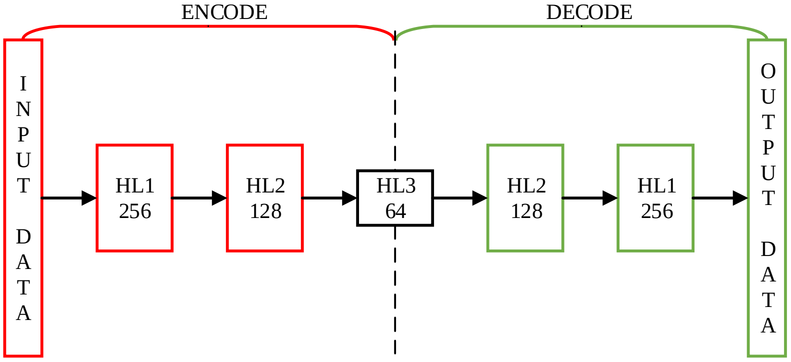

Fig.4 SCAE before added DNN

The input of SCAE is the RSSI detected in RP scan, the input of decoder is from the reduced signal strength, and the output is the reconstructed fingerprint vector. HL represents the hidden layer, and the number represents the number of neurons in the hidden layer.

When the unsupervised learning of SCAE weights is completed, the decoder part of the network is disconnected, and a typical deep network DNN is connected to the output of the encoder, as shown in Fig.5. The blue part is also the classifier. DNN classifier consists of two hidden layers, and the number of hidden neurons must be selected based on the complexity of the problem. In addition, Dropout is used between the hidden layers of DNN classifier. During training, the connections between layers are randomly discarded to force the network to learn redundant representation, so as to achieve better generalization and avoid over fitting. The final output layer is the softmax layer, whose output belongs to the probability of analyzing the current sample as a certain type.

This part of fig.5 is used to identify buildings and floors according to the provided input WiFi signal intensity scanning, extract features with SCAE, and output the recognition results with DNN classifier. The decoding part of the pre trained trestle compression auto encoder is removed, and the output of the encoder is connected to the two-layer DNN classifier. The number of each layer in the figure also represents the number of neurons. The output is the result of building and floor recognition.

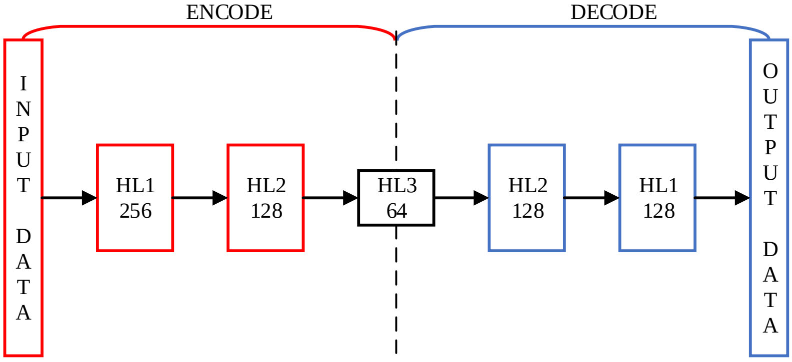

Fig.5 SCAE-DNN

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 7, Issue 4, pp: (138-145), Month: October - December 2019, Available at: www.researchpublish.com

IV. EXPERIMENT

A. Experimental Data

In order to compare and evaluate the proposed algorithm with other research methods, we need a large data set which contains the location of the markers and is available publicly. In 2014, UJIIndoorLoc public data set [8] created by Torres et al. The data set is measured in the campus of the University of Jaime I. there are three buildings in the measured environment, with an area of nearly 110000 square meters. The building plan is shown in Fig.6, including 21049 reference points, which are divided into 19937 training sets and 1111 verification sample sets. In addition, the test set is provided by EvALL. The training set, test set and verification set are collected at different times. The collected data comes from 25 different Android devices and 20 different users.

Fig.6 Data collection source building plan

In the training set of open data set, each reference point contains 529 attributes, among which 520 different APS are collected in three buildings, and 16-20 AP signals can be received at each reference point. In order to maintain the unity of fingerprint database, the value of AP signals not received is set to 100. The other nine attributes contain the longitude and latitude corresponding to the reference point, floor number, building ID, space ID, relative location, user ID, phone ID and measurement timestamp information. The latitude and longitude is the latitude and longitude of the actual map. The floor number is the floor of the building, and the building ID is the building, which is represented by 0,1,2. The reference point of space ID belongs to the information of office or laboratory, and the relative position is in the building (represented by 1) or in the corridor (represented by 0). Time stamp is the time to acquire the signal value, because according to the discussion in the previous chapters, the signal strength will be different with different acquisition time.

B. Experimental environment

The experiment is implemented in the environment of Python 3.7. It is more convenient to use the IDE environment of pycharm. In the experiment, keras library1 is used to build the deep neural network, tensorflow2 is used for numerical calculation, data flow graph and scikit learn 3 are used for typical machine learning operations, and finally, MATLAB and pylab are used to draw the experimental results.

There are 520 nodes in the input of unsupervised learning SCAE, and each input in the data set contains 520 dimensions. There are 5 hidden layers in all the hidden layers of SCAE-DNN, and the number of neurons in each layer is different. SCAE contains three hidden 256, 128 and 64 neuron layers, and the decoding part is consistent with the coding part; the classifier has two hidden layers, including 128 neurons measured. The output node is the combination of building label and floor. In the collection environment, two of the three buildings collect the data of four floors, and one collects the data of five floors. Therefore, there are 13 nodes in the output layer. The parameter settings of the network, such as the training learning rate and the number of iterations, are shown in Table I.

TABLE I: Neuron parameters and network parameter values

SCAE input	SCAE-DNN hidden					DNN output	Training epoch	Learning rate	Batch size
	1	2	3	4	5				
520	256	128	64	128	128	13	10E3	10E-4	15

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 7, Issue 4, pp: (138-145), Month: October - December 2019, Available at: www.researchpublish.com

Before performing SCAE supervised learning, tag data is divided into training, validation, and test sets. Scae-dnn network learns from the training data, analyzes the performance through the validation data, and finally tests the final performance according to the test set. In the process of training, the weights of SCAE and DNN classifier are modified, so that the whole network can achieve the best recognition effect. The training process uses classification cross entropy error and Adam optimizer. Data normalization is needed before training.

C. Results and Analysis

Fig.7 shows the change curve of recognition accuracy under different iteration times. The abscissa represents the iteration times, and the ordinate represents the accuracy. The training time is 3 minutes and 17 seconds, the best accuracy of the verification set is 0.9990999, and the best accuracy of the test set is 0.99729973. It is verified that SCAE-DNN network has a good recognition effect for buildings and floors.

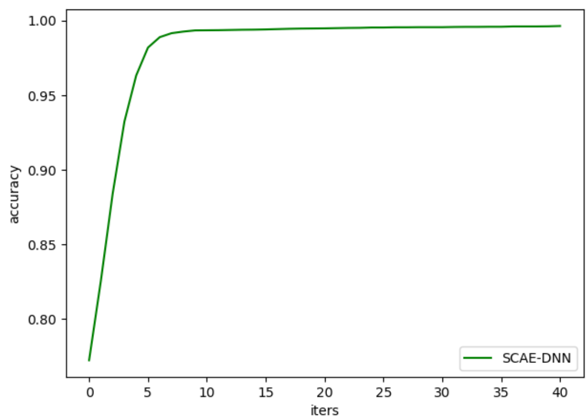

Fig.7 Accuracy under different iterations

For the signal value of the missing AP, UJIIndoorLoc set to 100 in the original data set, and compare it with the recognition result that the missing signal value is set to -100dB and -110dB, as shown in Fig.8. It can be seen in the figure that the result of using the original data is the worst, the highest recognition result is 0.982, the best result is 0.999 when using -110dB to indicate the missing AP, and the best result is 0.996 when using -100dB to indicate the missing AP. Because the received signal value is negative and the missing signal value is set to 100dB, the data is discontinuous. It is better to use -110dB value to express missing AP than -100dB value because it represents missing AP signal strength. The signal strength of AP received at many positions is -95dB to -99dB, which is similar to -100dB, but it is totally different in practice. It can receive AP signal with weak signal, which means that the distance is not far, but if the signal of AP is not received, it means the distance of geography is different It's far away, so using -110dB can solve this subtle problem.

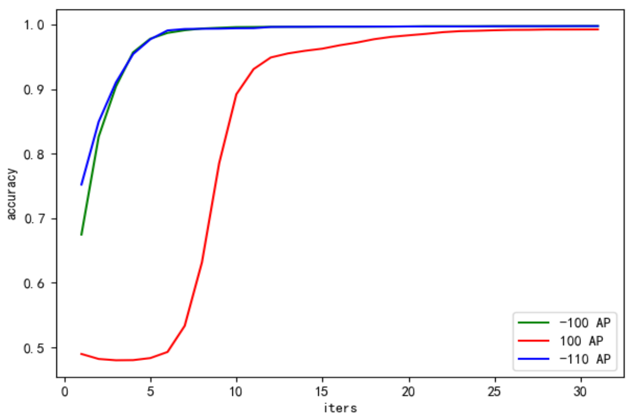

Fig.8 The influence of different representation missing AP value on recognition

In the experiment, the recognition accuracy of different network is compared shown in Table II. The verification sets of different networks have good positioning results, but the test sets are quite different. This is partly because as mentioned earlier, the test set is collected 4 months from the verification set, and the signal strength is different, but the SCAE(256-128-64) + DNN(256-128) network has achieved good results. More than 90% of the accuracy rate has achieved good results in positioning, but because it is the detection of buildings and floors, which is a large-scale positioning, so small detection error will also cause great inconvenience in practice, such as positioning to (1,2), which means building 1, floor 2, if the actual floor is 3, in the case of large space, such distance is also far away, which will cause great inconvenience for users.

TABLE II: Recognition accuracy of different network

Different network structures	SCAE(256-128-64) DNN(256-128)	SCAE(256-128) DNN(256-128)	DNN(256-128)	SAE(256-128-64) DNN(256-128)
Validation	99.9%	97.2%	91.6%	95.9%
Testing	99.7%	91.2%	71.3%	85.2%

Therefore, we decided to utilize SCAE to effectively reduce the dimension of input signal strength from 520 to 256128 and 64 features. Then the decoding part is removed and the encoding output of SCAE is connected to a typical DNN recognition network. The proposed architecture produces 99.9% correct recognition on the validation data set, and the correct recognition rate of the testing set is increased to 99.7%.

V. CONCLUSION

In this paper we proposed improvements on the algorithm of indoor positioning on the basis of previous studies. At present, the positioning of buildings and floors mostly depends on sensor equipment. In view of the instability of sensor data and the complex modeling and analysis of received data. This paper proposes to use SCAE-DNN network for recognition, and discusses the reasons and advantages of adding SCAE, effectively reducing the artificial data analysis, and robust to the interference of WiFi signal strength. The experimental results show that the recognition accuracy of the test set is 99.7%. At the same time, using the open data set for experiments, it is convenient to compare with other algorithms, and the accuracy is improved by 2% - 20%, which is obviously better than other algorithms.

The location based on deep learning needs a lot of off-line reference points as training samples, and it still faces many problems and high cost when applied to the actual location. Therefore, in the future, we will study the WiFi signal acquisition method based on crowdsourcing, so that each user can become a collector. In the process of positioning, not only can we enrich the fingerprint database continuously, but also can update the fingerprint database in time, so that the positioning will become more and more accurate with the passage of time.

REFERENCES

[1] Teng fan. “Study on floor discrimination method based on air pressure altimetry”. Xuzhou: China University of Mining and Technology, 2019.

[2] Liang Jian, Wang Jian. “Heuristic height constrained indoor positioning algorithm based on floor matching”. Chinese science and technology paper, 2018, 13 (15): 1805-1810.

[3] Zhou W F, Jie Y. “WLAN Floor Location Method Based on Hierarchical Clustering”. 3rd International Conference on Computer and Computing Science, Hanoi, VIETNAM, 2016:41-44.

[4] Mao wankui, Wu Fei, Zhang Yujin, et al. “WiFi indoor floor recognition method based on improved HC algorithm”. Sensors and Microsystems, 2019 (09): 43-46.

[5] Moreira A, Nicolau M J, Meneses F, et al. “Wi-Fi Fingerprinting in the Real World-RTLS@UM at the EvAAL Competition”. International Conference on Indoor Positioning and Indoor Navigation, Banff, Canada, 2015.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 7, Issue 4, pp: (138-145), Month: October - December 2019, Available at: www.researchpublish.com

[6] Elbasiony R, Gomaa W. "WiFi Localization for Mobile Robots based on Random Forests and GPLVM". In 13th International Conference on Machine Learning and Applications, Detroit, MI, 2014:225-230.

[7] Salah R, Vincent P, Muller X, et al. "Contractive Auto-Encoders: Explicit Invariance During Feature Extraction". ICML, 2011.

[8] Sospedra J T, Montoliu R, Uso A M, et al. "UJIIndoorLoc: A New Multi-building and Multi-floor Database for WLAN Fingerprint-based Indoor Localization Problems". In International Conference on Indoor Positioning and Indoor Navigation, Buan, South Korea, 2014:261-270.