# Recognition of Human Activities in Smart Homes Using Stacked Autoencoders

Nour El Houda Mbarki, Ridha Ejbali and Mourad Zaied
RTIM: Research Team in Intelligent Machines
University of Gabes, National Engineering School of Gabes (ENIG)
nourelhouda.mbarki.tn@ieee.org, ridha_ejbali@ieee.org, mourad.zaied@ieee.org

*Abstract*—**There is a growing interest in the domain of smart homes. One of the most important tasks in this domain is the recognition of inhabitants' activities. To ameliorate the proposed approaches, we propose, in this paper, a Staked Autoencoder (SAE) algorithm based on a deep learning framework for recognizing activities in a smart home. Our approach is tested on the Washington State University (WSU) dataset. We will show that our proposed approach outperforms existing methods such as the Artificial Neural Networks (ANNs) in terms of recognition accuracy of activities. In particular, the SAE shows an accuracy of 87.5% in recognizing activities based on WSU smart home dataset while the ANN algorithm has shown an accuracy of 79.5% on the same dataset.**

***Keywords- smart home; recognition of human activities; deep learning; stacked auto-encoders.***

## I. INTRODUCTION

The idea of home automation was first used in the 20th century. The main aim of home automation is to improve the comfort of living. With the progress in computer sciences and sensors' technologies, the smart home system allows users to predefine settings to manage their house remotely and gather data from the environment, to analyze it and execute necessary commands [1]. Moreover, sensors placed in homes are used to find out semantically meaningful events or activities [2]. Furthermore, a home management system utilizes machine learning, makes use of experienced systems and adopts necessary services after learning to provide appropriate services according to a user's habits [3].

A system is called a smart system when it has the ability to learn and take necessary actions or makes decisions for us. Thus, an automated home environment with the capability of learning and making decisions may be called a smart home.

Apart from reducing waste power, the objective of smart homes as sensor-based systems is to create smart, secure and comfortable environment for the aged and disabled people [4]. Therefore, sensors are needed to monitor and collect required data such as motion, temperature, analog sensors, etc [5]. In this regard, Cook et al. [6] prepared a smart apartment testbed to study human daily living activities and behaviors. Their objective was to recognize human activities throughout the collected data. The dataset was collected from 20 volunteers who performed a series of activities in the smart apartment testbed. Today, smart home technologies have rapidly developed into a large number of productions of a smart home's ready appliances. There are many different types of smart home appliances such as heating, ventilation, air conditioning, entertainment, lighting, shading, home security systems, health care applications and the control of other household appliances. These appliances were designed based on the different specific services required.

In this paper, we propose a recognition system of human activities using deep learning. The idea of using deep learning comes from its effectiveness in the pattern recognition domain. Recently, deep learning has gained its popularity as a powerful tool for learning complex and large-scale problems [7]. The model for deep learning is typically constructed by stacking multiple auto-encoders (SAEs) [8]. This deep architecture has been successfully used as a feature extractor for text, image, and sound data and as a good initial training step for deep architectures [8][9]. In this paper, we propose novel activities' recognition algorithm using the deep learning architecture as an alternative to existing shallow architectures such as Artificial Neural Networks (ANNs) [4][10]. The performance of the proposed classification method is demonstrated using the Washington State University (WSU) smart home dataset. The proposed classification method achieves an accuracy of 87.5% for classifying activities based on WSU smart home dataset (see Figure 1) while the classification based on ANN algorithm has shown an accuracy of 79.5% on the same dataset.

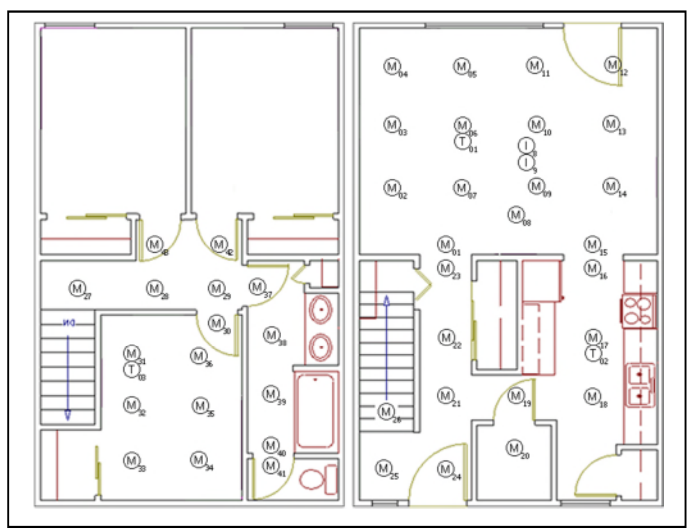

Figure 1. The installation of sensors used in the smart apartment testbed [6].

The remainder of this paper contains five sections. Section 2 includes an overview of related works. The proposed method is presented in Section 3. The experiments and the tests' results are mentioned in Section 4. As a final point, Section 5 concludes this paper.

## II. RELATED WORKS

Since their creation in the early 1940s [11], Artificial Neural Networks (ANNs) have been used to solve many types of problems in robotic processing [12], pattern recognition [13], speech and handwriting recognition [14], etc.

Despite its popularity, ANNs did not escape the central problem of Machine Learning: over-learning. To move forward, new ideas were needed. After several decades of stagnation, it was G. E. Hinton [15] and his team who, in 2006, made the main breakthrough in this field. This modern machine learning technique is called deep learning [16]. The main goal of deep learning algorithms is to develop computational models that can find an optimal weighing between the input variables (also called predictors) and their corresponding class labels.

Since 2009, deep neural networks have won many official international pattern recognition competitions such as handwriting competitions at ICDAR 2009 [17] and human actions in supervision videos [18] achieving the first superhuman visual pattern recognition results in limited domains [19]. Other successful deep learning applications include object detection [20], video classification [21], and neuro-imaging studies of psychiatric and neurological disorders [22].

Regardless of the activities of recognition, the task of classification of any type of data has benefited by the advent of deep architectures [23][24]. Previously existing methods of classification mostly relied on the usage of specific features always crafted manually by human experts. Finding the best features was the subject of various researches and the performance of the classifier was strongly dependent on their quality. The advantage of the deep learning is that it can learn such features by itself reducing the need for human experts.

## III. REVIEW OF METHODOLOGY

An autoencoder is a neural network that has three layers: an input layer, a hidden (encoding) layer, and a decoding layer. The network aims at reconstructing its inputs, which forces the hidden layer to try to learn good representations of the inputs [25].

In order to encourage the hidden layer to learn good input representations, certain variations on the autoencoder exist. A stacked autoencoder [7][26] is a neural network consisting of multiple hidden layers of neurons in which the outputs of each layer is wired to the inputs of the successive layer (see Figure 2).

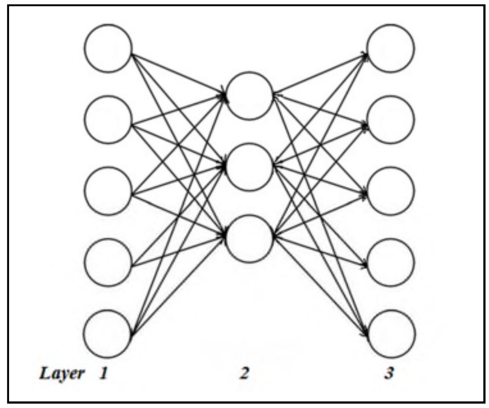


Figure 2. The structure of an autoencoder.

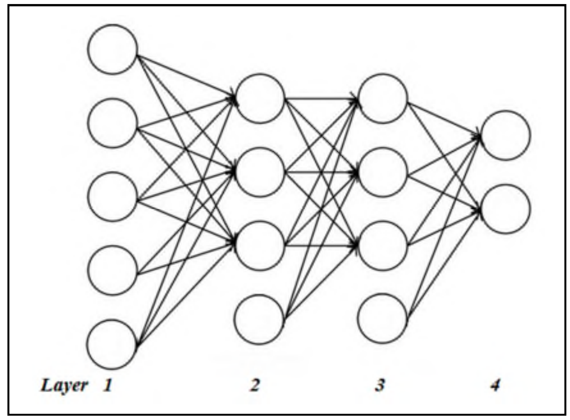


Figure 3. The structure of the stacked autoencoder used.

The SAE used in our study is constructed by two autoencoder layers and a softmax layer as shown in Figure 3. An autoencoder is the basic entity of a SAE classifier. It is composed of an encoder step (from Layer 1 to Layer 2 in Figure 2) and a decoder step (from Layer 2 to Layer 3 in Figure 2). This process can be formulated as (1) and (2), where $\boldsymbol{s}$ is a non-linearity function (the sigmoid function in our case), $\boldsymbol{W}$ and $\boldsymbol{W^T}$ are the weight matrices of this model, $\boldsymbol{b}$ and $\boldsymbol{b'}$ are two different bias vectors of this model, $\boldsymbol{y}$ is a latent variable representation of the input layer $\boldsymbol{x}$, and $\boldsymbol{z}$ represents a prediction of $\boldsymbol{x}$ when the value of $\boldsymbol{y}$ is givenand it has the same shape as $\boldsymbol{x}$.

$$y = s(Wx + b) \tag{1}$$

$$z = s(W^T x + b') \tag{2}$$

Various autoencoder layers are stacked together from an unsupervised pretraining stage (from Layer 1 to Layer 3 in Figure 3). The latent representation $\boldsymbol{y}$ obtained by an autoencoder is used as the input to its successive autoencoder layer. In these steps, the training is performed with one layer at a time and each layer is trained as an autoencoder by minimizing its reconstructing error. This reconstruction (Loss function: $\boldsymbol{L(x,z)}$ ) can be calculated in many ways. For our model, we use cross-entropy [27] to calculate the reconstruction error as shown in (3), where $\boldsymbol{x_k}$ and $\boldsymbol{z_k}$ denote the $\boldsymbol{k_{th}}$ element of $\boldsymbol{x}$ and $\boldsymbol{z}$, respectively.

$$L(x,z) = -\sum_{k=1}^{d}[x_k In z_k + (1-x_k) In(1-z_k)] \quad (3)$$

The reconstruction error can be minimized using the Gradient Descent method [28]. The weights in (1) and (2) should be updated according to (4), (5) and (6), where $\boldsymbol{\alpha}$ denotes the learning rate.

$$W = W - \alpha \frac{\partial L(x,z)}{\partial W} \quad (4)$$

$$b = b - \alpha \frac{\partial L(x,z)}{\partial b} \quad (5)$$

$$b' = b' - \alpha \frac{\partial L(x,z)}{\partial b'} \quad (6)$$

After this phase of training is complete, fine-tuning using back propagation is used to improve the results by tuning the parameters of all layers that are changed at the same time. In our model, the probability that an input vector $\boldsymbol{x}$ (in Layer 3 in Figure 3) belongs to class $\boldsymbol{i}$ can be obtained as (7), where $\boldsymbol{Y}$ is the predicted class of an input vector $\boldsymbol{x}$, $\boldsymbol{W}$ and $\boldsymbol{b}$ are respectively the weight matrices and the bias vectors of this layer, $\boldsymbol{W_i}$ and $\boldsymbol{W_j}$ are respectively the $\boldsymbol{i^{th}}$ and $\boldsymbol{j^{th}}$ row of matrix $\boldsymbol{W}$, $\boldsymbol{b_i}$ and $\boldsymbol{b_j}$ are respectively the $\boldsymbol{i^{th}}$ and $\boldsymbol{j^{th}}$ elements of vector $\boldsymbol{b}$, and the ***softmax*** is the used function (non-linear). In equation (8), the class with the highest probability is regarded as the predicted label $\boldsymbol{y_{pred}}$ of the input vector $\boldsymbol{x}$. The prediction error of sample data set ***DS*** (***Loss(DS)***) is calculated based on the true labels, as shown in (9), where $\boldsymbol{y_i}$ is the true label of $\boldsymbol{x_i}$. The reconstruction error can be minimized using the Gradient Descent method as described above.

$$P(Y=i|x,W,b) = softmax(Wx+b) = \frac{e^{W_i x + b_i}}{\sum_j e^{W_j x + b_j}} \quad (7)$$

$$y_{pred} = argmax\big(P(Y=i|x,W,b)\big) \quad (7)$$

$$Loss(DS) = -\sum_{i=0}^{D} In(P(Y=y_i|x_i,W,b)) \quad (8)$$

## IV. ACTIVITY RECOGNITION

We have used the dataset of Washington State University, obtained from the experimental study of "Assessing the quality of activities in a smart environment" [6] in the current study.

To create the dataset, 20 WSU undergraduate students recruited into the smart apartment and had them performed five activities:

- ✓ Make a phone call (5steps): in the dining room, the participant moves to the phone, looks specific number in the phone book, dials the number, and listens to the message. Then, the participant summarizes the recorded message (provides cooking directions) on a notepad.
- ✓ Wash hands (6 steps): In the kitchen, the participant moves into the sink and washes his/her hands. They use hand soap and dry their hands with a paper towel.
- ✓ Cook (7 steps): According to the directions given in the phone message, the participant cooks a pot of oatmeal. To cook it, the participant should measure water, pour the water into a pot and boil it, add oats, then put the oatmeal into a bowl with grapes and brown sugar.
- ✓ Eat (3 steps): The participant takes the oatmeal and a medicine container to the dining room and eats the food.
- ✓ Clean (5 steps): In the kitchen, the participant takes all of the dishes to the sink and cleans them with water and dish soap.

TABLE I. DISTRIBUTION OF EACH SENSOR

| Sensors id | Description | Code |
|---|---|---|
| M01.. M26 | motion sensors | 01..26 |
| I01.. I05 | item sensors for oatmeal, grapes, brown sugar, bowl, measuring spoon | 101..105 |
| I06 | medicine container sensor | 106 |
| I07 | pot sensor | 107 |
| I08 | phone book sensor | 108 |
| D01 | cabinet sensor | 113 |
| AD1-A | water sensor | 110 |
| AD1-B | water sensor | 111 |
| AD1-C | burner sensor | 112 |
| asterisk | phone usage | 0 |

The sensors were installed in a smart apartment on objects such as the phone book, the medicine container, a cooking pot, etc. to record the activation-deactivation events as the subject carrying out the five specific activities.

TABLE II. EXAMPLE OF DATA FORMAT

| Activity | Date | Time | Sensor id | Activation/ deactivation |
|---|---|---|---|---|
| Wash hands | 27/02/08 | 12:49:52 | M14 | ON |
| | 27/02/08 | 12:49:53 | M15 | ON |
| | 27/02/08 | 12:49:54 | M16 | ON |
| | ⋮ | ⋮ | ⋮ | ⋮ |
| | 27/02/08 | 12:50:40 | AD1-B | 0.467429 |
| | 27/02/08 | 12:50:42 | M17 | OFF |

This dataset included activities' names, dates and the list of the sensors activated during this activity with their type of activation and deactivation. Distribution of sensors is shown in Table I, Table II and an example of data format is shown.

### *A. Determining the Input and Output Layers*

After preprocessing of inputs and outputs, four features of {date, day, sensor id, activation/deactivation of the sensor} for input layers were defined in a 139x4 matrix. The first autoencoder has 400 hidden units and the second autoencoder has 200.

### *B. Training and Testing*

Activity recognitions have been varied out for 5 defined activities in a dataset. In this study, we have tried to find the best training parameters to obtain better results or higher accuracy. For this purpose, a total of 120 data, 80 data for a training set and 40 data for a test set were used. Table III presents the obtained accuracy for each activity.

The SEA algorithm showed better accuracy results compared to ANN in overall. However, the two algorithms have similar accuracy results for the tasks of phone calling and eating. The difference between SEA and ANN is mostly for the recognition of the cooking task. It may be interpreted that the longer the activity takes (cooking, with 7 steps, approximately 80 activations/deactivations), the more the SEA outperforms the ANN algorithm.

TABLE III. ACCURACY OF ACTIVITIES

| **Activity** | **Accuracy** | |
|---|---|---|
| | ANN | SEA |
| 1. Phone call | 77.8% | 77.8% |
| 2. Wash hands | 71.4% | 85.7% |
| 3. Cook | 75.0% | 100% |
| 4. Eat | 100% | 100% |
| 5. Clean | 71.4% | 77.8% |
| Total | 79.5% | 87.5% |

Table III shows that the proposed approach of activity recognition based on stacked autoencoders has given better results than that given by artificial neural networks. These results can be explained by the ability to learn by stacked autoencoders based on deep learning as well as in artificial neural networks.

A stacked autoencoder tends to learn features that form a good representation of its input. The first layer of a stacked autoencoder tends to learn first-order features in the raw input. The second layer of a stacked autoencoder tends to learn second-order features corresponding to patterns. Higher layers of the stacked autoencoder tend to learn even higher-order features.

## V. CONCLUSION

Deep Learning‘s Stacked Autoencoders have been used for human activity recognition according to a performance on WSU smart home dataset. The achieved results demonstrated that this algorithm has a considerable human activity recognition performance of 87.5% accuracy. It is noted that the dataset contains other parts in which activities are defined with specific errors. This part can be used to assess the consistency of activities of daily life. Furthermore, the given results are obtained for a particular environment (the smart apartment tested). In case of a different environment, it requires a new testing to create a suitable dataset.

## ACKNOWLEDGMENT

The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

## REFERENCES

[1] T. Chu Chong and T. Chong Eng, "A Neural Network Approach towards Reinforcing Smart Home Security" Asia-Pacific Symposium on Information and Telecommunication Technologies (APSITT), pp. 1-5, 2010.

[2] T. Yoon-Sik, K. Jongik, and H. Eenjun, "Hierarchical querying scheme of human motions for smart home environment" Engineering Applications of Artificial Intelligence, vol. 25, pp. 1301-1312, 2012.

[3] S. Victor R.L., Y. Cheng-Ying, and C. Chien Hung, "A smart home management system with hierarchical behavior suggestion and recovery mechanism" Computer Standards & Interfaces, vol. 41, pp. 98-111, 2015.

[4] A. Badlani and S. Bhanot, "Smart Home System Design based on Artificial Neural Networks" World Congress on Engineering and Computer Science (WCECS), vol. 1, pp. 19-21, 2011.

[5] H. Fang, L. He, H. Si , P. Liu, and X. Xie, "Human activity recognition based on feature selection in smart home using backpropagation algorithm" ISA Transactions, vol. 53, pp. 1629-1638, 2014.

[6] D. Cook and M. Schmitter-Edgecombe, "Assessing the quality of activities in a smart environment" Methods of Information in Medicine. vol. 48, no. 5, pp. 480-485, 2009.

[7] Y. Bengio, "Learning deep architectures for AI" Foundations and Trends R in Machine Learning, vol. 2, no. 1, pp. 1-127, 2009.

[8] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol, " Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion" Journal of Machine Learning Research 11, pp. 3371-3408, 2010.

[9] W. Li et al, "Stacked Autoencoder-based deep learning for remote-sensing image classification: a case study of African land-cover..." International Journal of Remote Sensing, vol. 37, no. 23, pp. 5632-5646, 2016.

[10] H. D. Mehr, H. Polat, and A. Cetin, "Resident Activity Recognition in Smart Homes by Using Artificial Neural Networks" International Istanbul Smart Grid Congress and Fair (ICSG), pp. 1-5, 2016.

[11] W.S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity", the bulletin of mathematical biophysics, pp. 115-133, 1943.

[12] S.Y. King and J.N. Hwang, “Neural network architectures for robotic applications” Robotics and Automation, IEEE Transactions, pp. 641-657, 1989.

[13] T. Yoshida and S. Omatu, “Pattern recognition with neural networks”, the IEEE 2000 International Geoscience and Remote Sensing Symposium. Taking the Pulse of the Planet: The Role of Remote Sensing in Managing the Environment, pp. 699-701, 2000.

[14] R. Ejbali, M. Zaied, and C. Ben Amar, “Wavelet network for recognition system of Arabic word”, International Journal of Speech Technology, pp. 163-174, 2010.

[15] G. E. Hinton, S. Osindero, and Y.-W. The, “A fast learning algorithm for deep belief nets”, Neural computation, pp. 1527-1554, 2006.

[16] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning”, Nature, pp. 436-444, 2015.

[17] A. Graves et al., “A Novel Connectionist System for Unconstrained Handwriting Recognition”, IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 855-868, 2009.

[18] V. Jain and S. Seung, “Natural image denoising with convolutional networks”, Advances in Neural Information Processing Systems (NIPS), pp. 769-776, 2009.

[19] C. Szegedy, A. Toshev, and D. Erhan, “Deep neural networks for object detection”, Neural Information Processing Systems (NIPS), pp. 2553-2561, 2013.

[20] D. C. Ciresan, U. Meier, J. Masci, L.M. Gambardella, and J. Schmidhuber, “Flexible, high performance convolutional neural networks for image classification”, Intl. Joint Conference on Artificial Intelligence (IJCAI), pp. 1237-1242, 2011.

[21] A. Karpathy et al., “Large-scale video classification with convolutional neural networks”, the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1725-1732, 2014.

[22] S. Vieira, W.H.L. Pinaya, and A. Mechelli, “Using deep learning to investigate the neuroimaging correlates of psychiatric and neurological disorders: Methods and applications”, Neuroscience & Biobehavioral Reviews, pp. 58-75, 2017.

[23] S. Hassairi, R. Ejbali, and M. Zaied, “Supervised Image Classification Using Deep Convolutional Wavelets Network”, the 15th International Conference on Intelligent Systems Design and Applications (ISDA), pp. 1-5, 2015.

[24] A. Eladel, R. Ejbali, and M. Zaied, “Dyadic Multi-resolution Analysis-Based Deep Learning for Arabic Handwritten Character Classification”, The 27th IEEE International Conference on Tools with Artificial Intelligence (ICTAI), pp. 1082-3409 , 2015.

[25] S. Haykin, "Neural networks: A comprehensive foundation", vol. 3, no. 5, 1998.

[26] G. E. Hinton and R. R. Salakhutdinov, “Reducing the Dimensionality of Data with Neural Networks” Science, vol. 504, no. July, pp. 504-507, 2006.

[27] J. Shore and R. Johnson, “Axiomatic Derivation of the Principle of Maximum Entropy and the Principle of Minimum Cross-Entropy” IEEE Transactions on Information Theory vol. 26, no. 1, pp. 26-37, 1980.

[28] L. Bottou, “Large-Scale Machine Learning with Stochastic Gradient Descent” International Conference on Computational Statistics, pp. 177-186, 2010.