International Journal of Computer Science
and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 4, Issue 2, Mar 2015, 39-44

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

A NOVEL AND EFFICIENT TECHNIQUE OF COMMUNICATION IN IOTS USING AUTOENCODERS

SANGEETA RAO[1] & SAURABH YADAV[2]

[1]Department of Computer Science, Central University of Haryana, Mahendergarh, Haryana, India

[2]Electronics & Communication Engineering Dept. Delhi Technological University, New Delhi, India

ABSTRACT

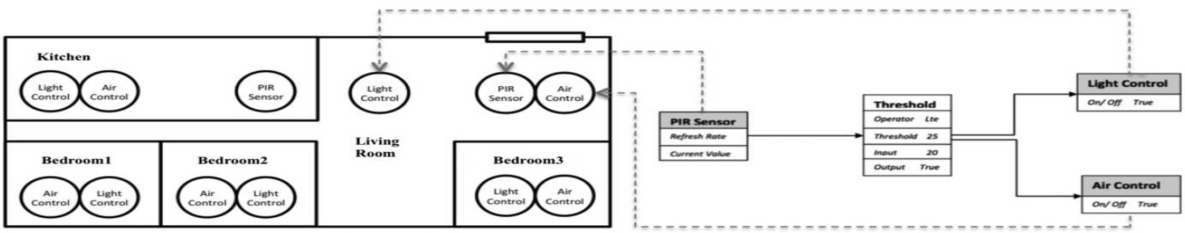
In the today's fast growing scenario, sensors and communication capabilities have been added into many traditional devices, controllers, and infrastructures so that systems can make informed and smart decisions. It involves flow of lot of personal and sensitive data through the network.

The Internet of Things (IoT) is the network of physical objects embedded with electronics, software, sensors, and network connectivity, which enables these objects to collect and exchange data that will help in taking smart decisions. Issue with these many IoT devices is network bandwidth utilization and energy cost. One strategy is to provide key based encryption for transmitted data and then increase communication efficiency using compression techniques in order reduce both network and bandwidth utilization. Common techniques for both approaches are compute intensive and not much suited for low power IoT devices. We propose use of deep learning network consisting of stacked autoencoders for increasing communication efficiency. Our method provides unified approach for both compression and encryption for IoT devices with the simplicity suitable for low power devices.

KEYWORDS: IoT, Autoencoder, Encoder, Decoder, Configuration File

INTRODUCTION

The vision of Internet of Things (IoT) has recently brought many new and game-changing products to the market. Sensors and communication capabilities have been added into many traditional devices, controllers, and infrastructures so that systems can make informed and smart decisions can be taken by themselves. Many of the market researchers have predicted that by 2020, more than 50 billion smart devices will be deployed and connected to Internet that will serve people more timely and properly. New applications have been developed using various IoT platforms, for sensing and collecting information to identify our needs, then composing and deploying smart services to make our lives simple and safe. The below diagram shows that an FBP can be deployed in the living room for automatically controlling light and room temperature according to the needs of the people there.[1]



Flow Based Program for a Smart Home

One of the primary issues for perpetually running IoT services on distributed located devices is the energy cost. Running billions of devices and communicating among them will use a lot of energy. Researchers have proposed various device sleep scheduling algorithms [4] to keep some devices power off or running at a low-power mode. Another approach is to reduce network communication traffic to conserve energy. These different devices generally produce highly correlated data, generated from different type of sensors. In this research, we use stacked autoencoder based deep neural network to reduce communication, in order to minimize the total energy cost for an application. Stacked autoencoders can identify pattern embedded in multisensory data and tune themselves for most efficient way of data compression and dimensionality reduction [2, 3].

Another issue with these IoT devices is the huge data which is transferred on network. Like I smart houses, data produced by living rooms cameras or mic based devices might be sensitive and requires secure data transmission techniques for data transfer. Generally key based encryption techniques are used that compute intensive and bandwidth inefficient data. Autoencoder based approach is one of the promising solution for this issue. As Deep Neural Networks' structure will contain pattern information of the communication through that channel, it is impossible to reproduce the original data from DNN output data without knowing the DNN size and bias configuration. So, DNN structure configuration which is only known to transmitting IoT device and endpoint Server, will act as Key for transmitted data and this Key will help in reproducing the data.

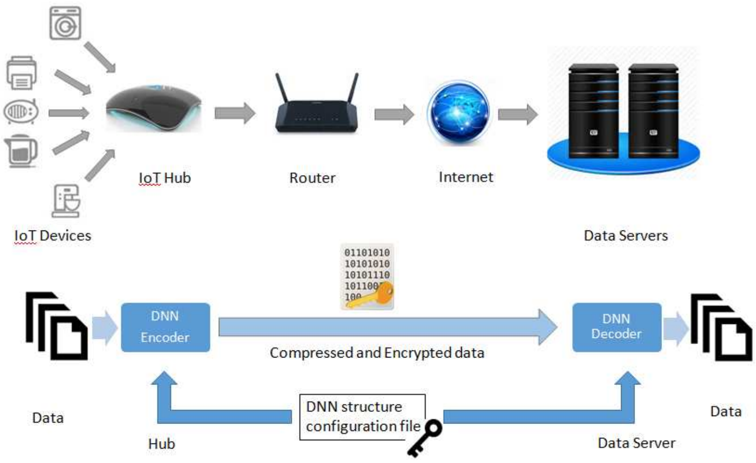
DEEP LEARNING

Deep-belief networks are a relatively new approach to neural networks that are unique in having more than one hidden layer embedded inside (stacked autoencoders) [2, 5]. These networks are typically trained one layer at a time, with a set of sparsity constraints to minimize the number of connections between nodes. Training networks in this way allows one to develop hierarchical representations while ensuring that individual layers represent meaningful components of the information. For example, with images of human faces, a deep-belief network may contain a first layer composed of various edges, a second layer that uses edges to form face parts, and a third layer that uses face parts to form entire face representations. The components generated at each layer are referred to as basis functions which are to be used in the next layer. Due to the sparsity constraints, these basis functions are developed such that they maximize the amount of information that can be represented through a minimal combination of bases. The training method for these networks is designed to encode data using very little information.

Autoencoder is a multi-layer perceptron (MLP) that has symmetric structure and designed to learn an approximation to the identity function, so as to the output is as similar to the input as possible. An autoencoder is composed of two networks, the “encoder” network which is used to transform the input from high-dimensional space into features or codes in low-dimensional space and a symmetric “decoder” network which is to used reconstruct the input from corresponding codes [5]. Two networks are trained jointly by tuning the weights of decoder network first and encoder network next as the output must be close to the input. The goal is to minimize the difference between the reconstruction/output and origin/input. Usually, the weights are initialized randomly through sampling a given distribution and backpropagation is chosen as learning algorithm.

PROPOSED APPROACH

The proposed method for IoT communication used the stacked autoencoder based DNN on IoT hub before transmitting data to network. DNN configuration knowledge base will contain the information about DNN structure size and biases and will be shared between IoT hub and Data Server. The IoT hub aggregates data from surrounding devices and sends to internet network. DNN will run on hub and will compress and encrypt the sensor generated data. The output packet from DNN will be transmitted to server which will decode the original data by using DNN configuration knowledge.



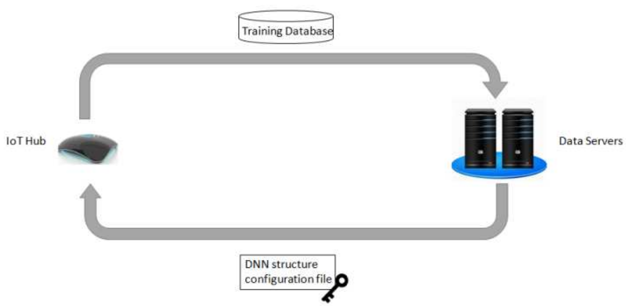
The Process Can be Divided Into 3 Phases:

- Adaptation Phase
- Encoding Phase
- Decoding Phase

Adaption phase is training Phase of DNN. Encoding and Decoding phases are using the DNN for Hub side encoding and Data server side decoding.

THE ADAPTATION PHASE

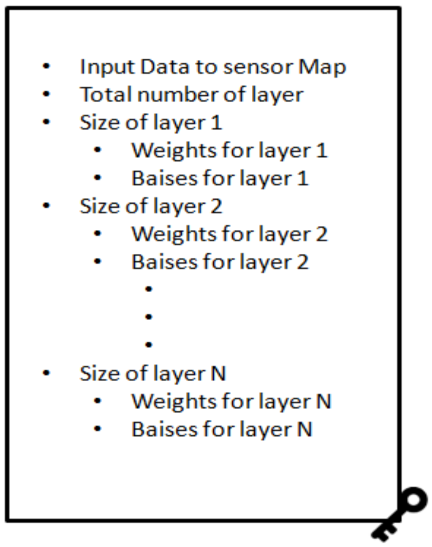
In this Phase, the DNN will be trained for data reduction based specific to hub. In this period, hub will assemble data from all other IoT devices for a week period. Data accumulation can be handled by either of following scenarios:



- Hub transfers this data to Data Server using some other encryption techniques. Data Server will store data and train the DNN. It will send DNN configuration knowledge file to Hub, which can be used in subsequent phases.

- Hub stores this data locally. Complete data set can be transmitted to Data Server after end of this phase period and resultant DNN configuration file can be obtained via PC based network.

The below diagram shows the structure of configuration file that will act as Key between the IoT hub and the data server.



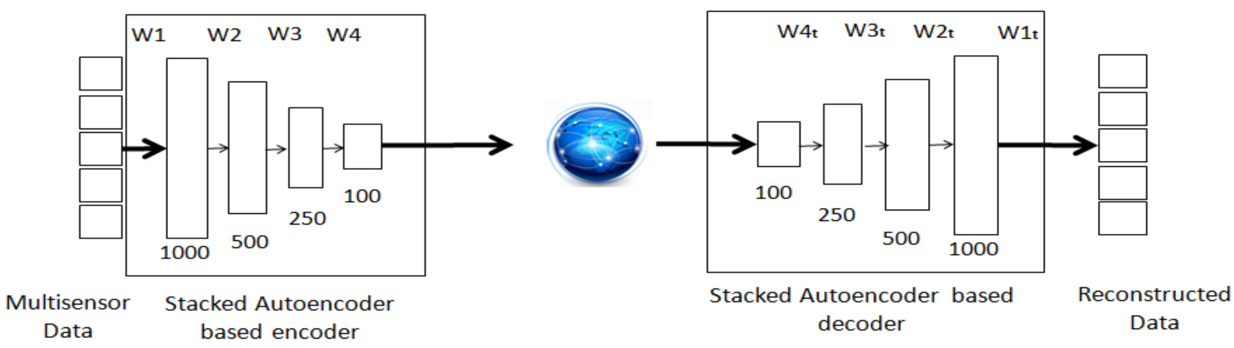
DNN structure configuration file

THE ENCODING PHASE

Once DNN has been trained and its configuration file is available to Hub, it can use DNN based channel for communication. Hub will gather the data from all IoT devices, and encode it using the DNN and transmit the data. Optionally, it can calculate the error by reconstructing the input from output packet. The difference in generally very less and can be easily corrected by using simple error correcting techniques. It can even be dropped as in most of sensory data like images/audio, this difference would not matter.

Using notation from the autoencoder section, let W(k,1), W(k,2), b(k,1), b(k,2) denote the parameters W(1),W(2),b(1),b(2) for kth autoencoder. Then the encoding step for the stacked autoencoder is given by running the encoding step of each layer in forward order:

$$a^{(l)} = f(z^{(l)})$$
$$z^{(l+1)} = W^{(l,1)} a^{(l)} + b^{(l,1)}$$



THE DECODING PHASE

Data Server will reconstructing the original sensor data from transmit packet using configuration file as key. If Error correcting techniques has been deployed from hub, then it can use these techniques to remove any critical errors.

The decoding step is given by running the decoding stack of each autoencoder in reverse order:

$$a^{(n+l)} = f(z^{(n+l)})$$
$$z^{(n+l+1)} = W^{(n-l,2)} a^{(n+l)} + b^{(n-l,2)}$$

Trigger to Adaptation Phase

Adaptation Phase can be triggered again in the case of following events to adapt new environment changes:

- New device has been installed.
- Significant increase in Frequency of DNN Reconstruction Error being greater than threshold.

ANALYSIS

As after the exclusive experiments, it has already been proved that the autoencoder works significantly in the compression of files and dimensionality reduction i.e. conversion of high dimensional data into low dimensional data. So, its use in communication in IoT devices will surely reduce the data traffic and provide a data and energy efficient way of communication.

CONCLUSIONS AND FUTURE SCOPE

The autoencoder based communication in IoT is one of the promising ways of data and energy efficient technique of communication as it will provide compression of data in an efficient way. For further improvements, error function can be biased more towards some critical sensor data so that error in these sensor data can be minimized. Also Error correcting techniques can be included for error correction in reconstruction to provide lossless communication. And even further work can be done these error correction techniques can biased towards critical sensor data.

REFERENCES

1. Zhenqiu Huanga, Kwei-Jay Lin, Shih-Yuan Yu, Jane Yung-jen Hsu, Co-locating services in IoT systems to minimize the communication energy cost, Journal of Innovation in Digital Ecosystems 1 (2014) 47–57.
2. Anand Atreya and Daniel O'Shea, Novel Lossy Compression Algorithms with Stacked Autoencoders.
3. Jing Wang, Haibo Hea, Danil V. Prokhorov, A Folded Neural Network Autoencoder for Dimensionality Reduction, Proceedings of the International Neural Network Society Winter Conference (INNS-WC 2012).
4. Z. Huang, K.-J. Lin, A. Han, An energy sentient methodology for sensor mapping and selection in IoT systems, IEEE International Symposium on Industrial Eletronics, 2014
5. G. E. Hinton, R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.