Architectural Engineering -- Dissertations and Student Research

Architectural Engineering and Construction, Durham School of

ANN BASED LOAD IDENTIFICATION AND FORECASTING SYSTEM FOR THE BUILT ENVIRONMENT

Hosen Hasna
University of Nebraska-Lincoln, hhasna@unomaha.edu

ANN BASED LOAD IDENTIFICATION AND FORCASTING SYSTEM

FOR THE BUILT ENVIRONMENT

By

Hosen Hasna

A THESIS

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Master of Science

Major: Architectural Engineering

Under the Supervision of Professor Mahmoud Alahmad

Lincoln, Nebraska

December, 2011

ANN BASED LOAD IDENTIFICATION AND FORECASTING SYSTEM

FOR THE BUILT ENVIRONMENT

Hosen Hasna, M.S.

University of Nebraska, 2011

Advisor: Mahmoud Alahmad

"Phantom Loads" cause energy waste in homes and the built environment as a whole. Consumers spend more than $3 billion a year on "Phantom Load" in the United States alone. The goal of this work is to conserve energy by increasing consumer awareness on their energy usage and appropriate tools which in turn will be reducing and /or eliminating phantom loads and mismanagement of the load in the built environment. A further goal is to improve upon existing power distribution systems in the built environment with limited hardware additions to increase energy conservation. This work investigates remote identification of load types along the electrical circuitry where they (load) are being consumed. The load type and status (on, off, standby) are determined both remotely and in a non-intrusive manner using Non-Intrusive Load Monitoring Methods. Finally we added the short term for load forecasting were we will be able to predict the demand kW and leave the ability of using that for energy management strategies like load shifting and demand limiting.

Acknowledgements

I would like to thank my advisor, Professor Mahmoud Alahmad for his continuous helpful and inspiring feedback throughout the period of my MS studies.

Also I would like to show my deepest gratitude and respect to my family, especially my parents, Ahmad and Nehad Hasna, the ones to whom I owe all success in my life and my brother and sisters.

My appreciation is also extended to all colleagues in the Architectural Engineering especially Mohammad Amer Chaaban, Wisam Nader, Muhammad Farhan, and Emily Rieur for their help and their support.

Table of Contents

List of Figures

List of Tables

1.Chapter 1

Introduction

1.1 General

This chapter presents the motivation for the research addressed in this thesis. The

chapter also provides the main objectives of the research as well as the thesis chapters'

outlines and structure.

1.2 Motivation

The electric power sector is experiencing a major shift in its generation,

transmission and distribution system, in an effort to reduce CO_2 emissions, increase the

use of renewable sources for a sustainable energy future and conserve precious resources

by reducing energy consumption. The distribution system particularly plays a role in the

built environment as it is an enabling product, an intangible necessity used to power our

systems. The 2010 Annual Energy Outlook indicates a 14% increase in U.S. primary

energy consumption from 2008 to 2035, an average annual growth rate of 0.5%. Further,

CO_2 emissions are projected to increase by 0.3% per year for the same time frame. This

increase is mostly due to the electric power and transportation sectors. According to

United States Green Building Council (USGBC), the built environment in the United

States accounts for 72% of electricity consumption, 39% of energy use, 38% of all CO_2

emissions, 40% of raw materials use, 30% of waste output (136 million tons annually),

and 14% of potable water consumption[2]. Electricity consumed by an electronic device

while it is turned off or in standby mode is known as "Phantom Load". Focus on Energy with Wisconsin Utilities estimates more than $3 billion dollars spent annually on phantom loads [3].

Active research into energy conservation in the built environment is focusing on more efficient and sustainable systems and energy consumption awareness and behavior change. This behavior change is more prominent in the residential sector with Advanced Metering Initiatives and Infrastructure (AMI) being developed to empower the user and to facilitate the future of the smart grid [4]. Most of the research is addressing the energy consumption from a top approach, i.e. looking at total consumption to influence behavior. A 2010 study by American Council for an Energy-Efficient Economy (ACEEE) of 57 energy conservation projects from 1970-2010 found that feedback devices alone are unlikely to maximize energy savings[5]. Hence, there is a need to focus on creative and promising methods to encourage energy efficient lifestyles. For example, identification of specific load and location has recently emerged to develop smart home/building to conserve energy and provide for comfortable living and working environment.

1.3 Thesis Objective

In the process of creating a smart built environment, we are proposing to address energy needs using a bottom-up approach. The focus on this research will be on consumption and conservation at the electrical node, where an electrical node is a point on the electrical wiring system at which electrical current is taken to supply utilization equipment (load). Investigations are being done to develop non-intrusive and novel methods to identify the type and status of the connected loads. Information from these investigations will be used for modeling, simulation, and analysis to enable innovations

in building information modeling (BIM), energy consumption performance, and energy conservation and recovery using impulse saving behavior at the residential level and demand side energy management at the commercial/utility level. Some emerging technologies employ a lot of hardware additions to existing circuitry to achieve the intended goal of this research. The ultimate goal of this research is to be able to identify loads with minimal hardware additions to existing electrical circuitry in the built environment. Therefore, identification of loads and load types are being done remotely through Non-Intrusive Load Monitoring Methods (NILM).

1.4 Work Structure (thesis outline)

This thesis is structured as following:

1- Chapter 2 provides a general overview of the Energy structure and trends, evaluation and allocation of the energy waste and previous finding and potential impact in lowering the built environments.

2- Chapter 3 provides a general overview of the Artificial Neural Network, history of the Neural Network, Benefits of ANN, Architecture and learning method of the neural network.

3- Chapter 4 discusses the step by step building of the Neural Network, including all of the data preprocessing phase, validating and testing the criteria for the trained network, and finally general not about the training procedures.

4- Chapter 5 introduces the load identification and the designed neural network. A case study validates the proposed model.

5- Chapter 6 proposes an alternative approach for energy management or mid-term load forecasting. The forecasting model will be used to define the energy usage benchmark to be used as a reference for any demand limiting strategies for at least the following 24 hours.

6- Chapter 7 introduces future work and a summary of what we have achieved in this research. Future work also discussed with different scenarios and approaches.

2.Chapter 2

Evaluate Energy trends and locate waste

2.1 Building and Energy usage in U.S

United States is one of the most advanced countries in the world. Enormous amount of energy is being used every day in houses, business locations, industrial facilities, community service and transportation. The major consumers for energy in U.S could be divided into four main divisions consume [6]. Figure 1 below shows the energy consumed by each sector.

Share of Energy Consumed by Major Sectors of the Economy, 2010

Commercial 19%
Industrial 31%
Residential 23%
Transportation 28%
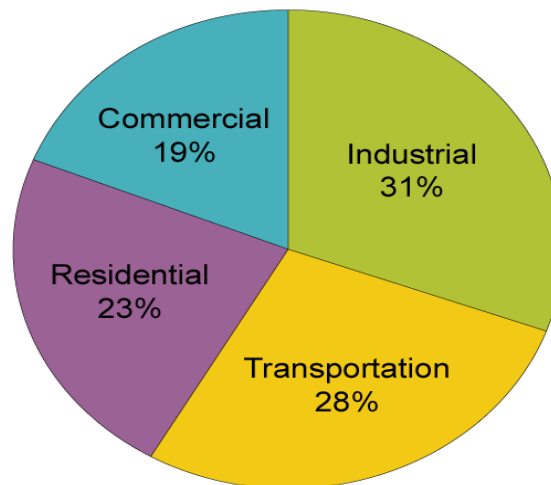
Source: U.S. Energy Information Administration, *Annual Energy Review 2009*, and *Monthly Energy Review (June 2011)*, preliminary 2010 data.

Figure 1 Share of Energy Consumed by Major sector of economy [6]

- Industrial part contains services and equipment that would be used for industrial, agriculture, excavating, and building.

- Transportation part involves automobiles that conveyance folks or materials, for example: wagons, vans, trucks, bikes, trains, underground, airplane, yachts and even barges,

- Residential segment consists of houses and flats.

- Commercial area includes different type of structures such as workplaces, shopping mall, stores, hospitals, schools, etc.

2.2 U.S. Energy Consumption by application between, 1949-2010

In general, in the last 60 years, energy usage on U.S. increased three times. The demand for the energy also increased in the previous decade. But 2009 was the year of depression for the U.S economy, lots of jobs were vanished and the previous steady increase in the energy demand starts declining.

In 2010 there was real gross domestic product (GDP) cut down by 2% compared to year of 2008, statistics showed that the energy consumption was decreased in each sectors as the following: industrial (-9%), commercial (-1%), transportation (-4%) and residential by (-2%). Figure 2 shows the changes in the energy consumption over the last 60 years [6].

Energy Consumption by Sector, 1949-2010

Quadrillion Btu
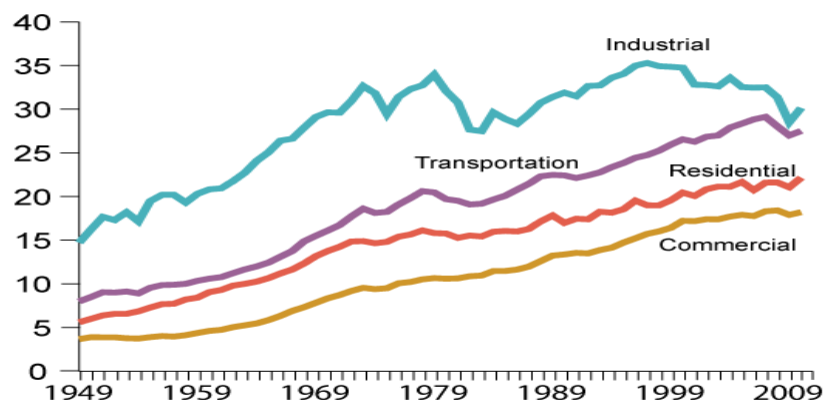
Source: U.S. Energy Information Administration, *Annual Energy Review 2009*, Table 2.1a, and Monthly Energy Review (June 2011), preliminary 2010 data.

Figure 2 changes in energy consumption from 1949-2010 [6]

Industrial area had significant decrease in the energy consumption. This cut was affected by the recession. Transportation section had for the second time where energy usage decreased for two consecutive years.

The year of 2010 was the comeback for the energy usage in U.S. the economy start to recover from the previous depression. The improved economy reflected on the energy consumption in each sector from the previous year: commercial 2%, residential 5%, industrial 6% and transportation 2%. [6]

2.3 Changes of energy used by application in 1978 and 2005:

Between 1978 and 2005, there were significant changes over the portions of the energy usage. Space heating became more efficient after enacting the federal energy efficiency standard. In numbers in 1987 energy consumption for the spaces was 6.96

quadrillion Btu where in 2005 it was 4.3 quadrillion Btu. Appliances and electronic was doubled over the last 25 years in spite of the Federal standard for energy efficiency. In numbers in 1978 the energy usage was 1.77 quadrillion Btu where in 2005 it was 3.25 quadrillion Btu. All in all energy usage in the residential sector was decreased from 10.58 in 1978 to 10.55 in 2005. The obvious decrease in energy consumption from the HVAC system was gained by electronic appliances. Figure 3 represent the changes on energy consumption on the residential area between 1978 and 2005.

Source: U.S. Energy Information Administration, 1978 and 2005 Residential Energy Consumption Survey
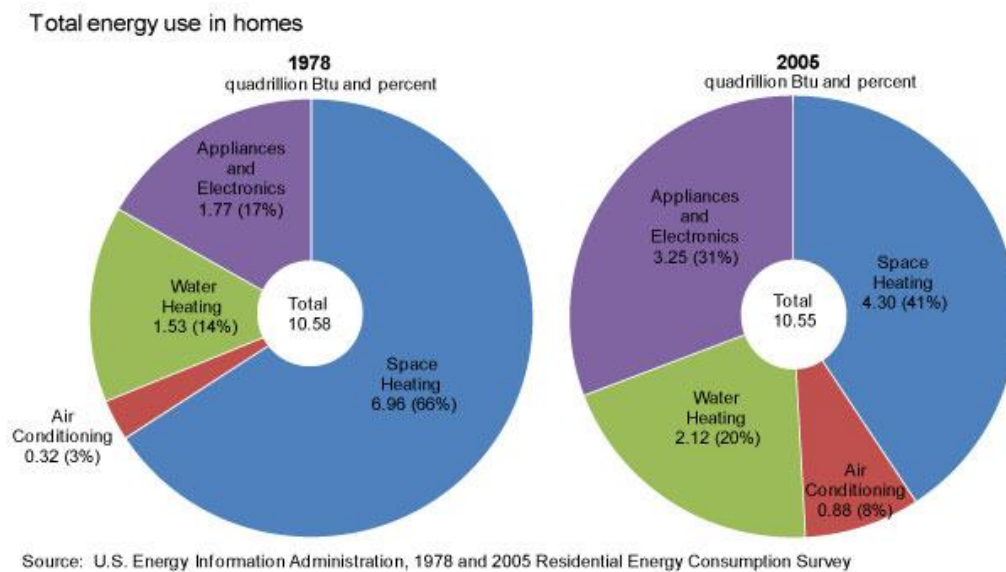
Figure 3 Changes of energy used by application in 1978 and 2005 [6]

The statistics was collected throw survey was made on the energy consumption from the residential sector. The energy usage for heating spaces was decreased 31 percent per house but because of the dramatic increase in the number of electronic appliances the total energy consumption virtually stayed the same.

Some decline that the decrease with the energy consumption not only due to the improvement of the equipment or better insulation but also to the population movement toward warmer areas [6-7].

2.3.1   Dramatic increase in household appliances use

The federal efficiency standard were passed to be applied in between 1988and 1994for the main major appliances. Energy which consumed by the appliances almost doubled between 1978 and 2005[6]. In 2009 Residential and Energy Consumption Survey (RECS) showed that the gains in efficiency for these appliances were offset by other factor like:

- U.S. households number raised by 34.5 million from 1978 to 2009

- Improved living standards became better and the increase in the income caused more electronic appliances in the houses

- Households more and more using the central air conditions which caused significant increase from 23 percent in 1978 to 61% in 2009

- Saturation, where houses start to have more and more appliances like dish washer or laundry machine. That's in total increased the number from 74 percent to 82 percent[6]

2.3.2 Contribution of Increased use of electronics on the efficiency gains

In spite of the increase in appliances efficiency over the last 30 years, the house holders in U.S have more and more electronic appliances. The main three devices that became more popular and most of us can see it in every single house owners, personal computer, television and rechargeable devices.

In the late of 1970s, personal computer was very luxurious and only families with high status would afford to buy one. In 2009, over 75 percent of U.S. homes were able to own at least one computer. Figure 4 shows the number of the electronic devices per household.
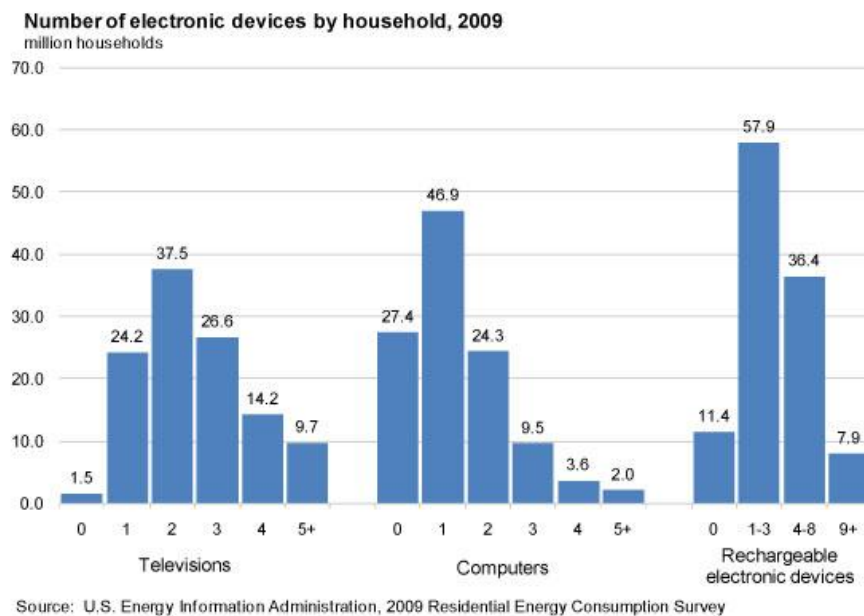
Figure 4 Number of electronic devices by household, 2009 [6]

Also in the late of 1970s, a television was available almost in every single house. In 2009, the average number of televisions in the household was 2.5 televisions.

Almost 15 years ago DVD players and Digital Video Recorders (DVR) did not exist, while now they are common. In average 79 percent of homes had a DVD player and 43 percent had a DVR by 2009. Finally almost a third of all households had at least four electronic devices, such as radio, game box [6-7].

2.4 Electricity use is expected to keep growing

Quick comparison between1950 and 2010 of the electrical consumption will show grow for more than 13 times. This Massive growth occurred because of the electricity usage in main three sectors, residential, commercial, and industrial. Industrial segment of energy consumption showed the greatest instability. Energy that has been sold to the residences and commercial segments surpassed the industrial sector early 1990s. Figure 5 below shows the changer of the energy sale for the four sectors, industrial, commercial, residential and transportation
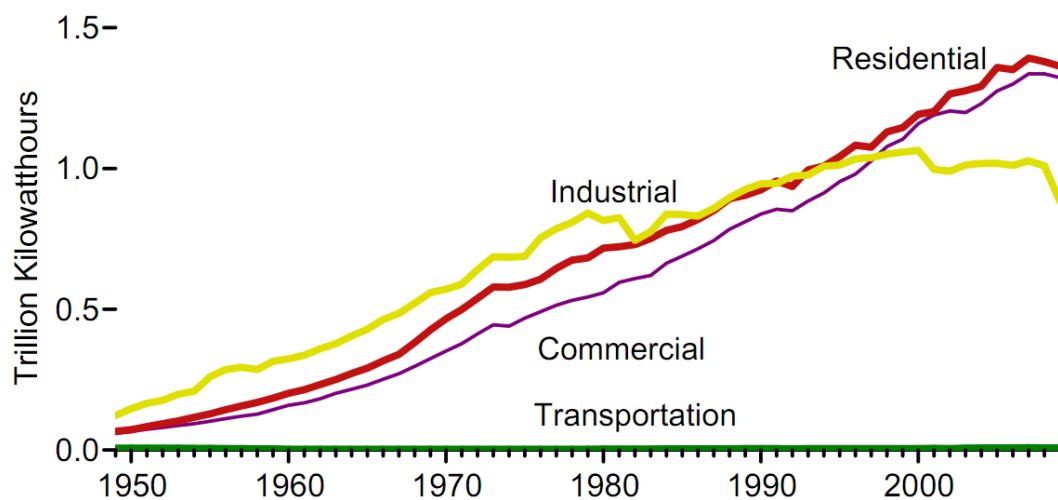
Figure 5 Electricity Retail Sales by Sector [6]

Demand on electricity change according to weather condition economy and oil prices. Over the long term the demand on the electricity keep increasing. Never the less demand growth is anticipated to keep growing about 1% until 2035. Any way the previous percentage is really low when we compare it to the 1950s where grew on demand was 9%. After 1950 demand for electricity started to decreased until it reached 2.5% in the 1990s and 0.5% on from 2000 to 2009 [6].

Over all Electricity is anticipated to continue growing fast even for the worldwide concluded 2030. The highest demand growth will be the highest in the developing country like China, Brazil and India.

2.5 U.S. Homes Electrical usage and trends

In United States building segment considered as the largest consumer for the energy. All over U.S Households number are 106, commercial buildings count around 4.6 million and industrial area around 15.5 trillion square feet [8].

Energy consumption is straight tight to (GHG) Green House Gas emissions— every quad of energy consumed in the building sector results in approximately 40 Million metric tons of carbon (MMTC) emissions (and costs almost $8 billion in 2001$) [9]. Figure 6 below shows the electricity use by different application in the homes.

How Electricity Is Used in Homes, 2010

All Other Appliances and Lighting 56%

Air Conditioning 22%

Water Heating 9%

Refrigerators 7%

Space Heating 6%
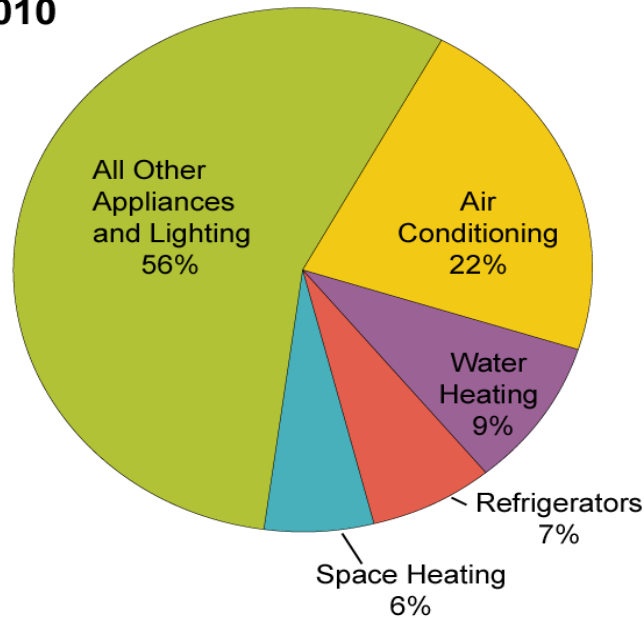
Source: U.S. Energy Information Administration, *Annual Energy Outlook 2011,* Table 4, Reference Case. Projections based on the Residential Energy Consumption Survey 2005.

Figure 6 Electricity Used in Homes, 2010 [6]

In conclusion air conditioning and space heating still consume the biggest portion of the energy in the homes, but if we added all the light and the electrical appliances that would reaches in average 56% of the total energy use of the houses [6].

2.6 Evaluation and location of Energy waste

Recent research in the residential consumption showed that there is around 41% energy waste out of the total power supply [10]. This amount of waste represents great opportunities for saving. One of the ways was educating the consumer and offering information on the electricity usage and how much they can save out of that. Consumer

might not know how much is some of the appliances still drawing energy while it's on standby mode.

Phantom loads refer to the electric energy being consumed by electronic appliances while they are on a standby mode, these kind of loads will keep adding up to the electrical utility bill each time unless the phantom load not being shut off completely[10].

2.6.1   Phantom loads

Most of the plug load could be considered as a main source for the phantom loads. According to the Lawrence Berkley National Laboratory, phantom loads, also called standby power or vampire loads refer to electricity used by appliances and equipment while they are or not accomplishment their primary function [11].Almost any application with an external power supply or distant control, will keep drawing power continuously.

With the increase demand and usage in the building and the trend of the steady growth in the demand on energy, there is a need to have more practical and affective way to tackle these losses. One of the main straight forward approaches to cut down Phantom loads is by physically unplug loads.   The study looked at units at Hilyard House apartments in Eugene, Oregon.
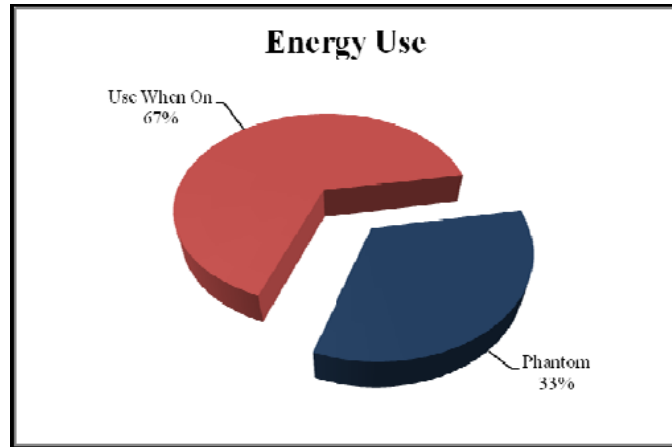
Figure 7 In-Use and Phantom Loads Energy use Distribution [11]

The actual phantom loads measured during this study made up 33% of the total metered energy use [12].

Laboratory evaluations that phantom loads represent around 5-10% of residential electricity use in most advanced countries and this percent will rise in developing countries and mainly in cities. Standby power use is roughly responsible for 1% of global CO_2 emissions [11]. Finally cutting down the phantom load will reduce the energy usage without affecting the consumer lifestyle.

2.6.2   Phantom loads distribution by appliances

Phantom load break down into different categories. For example home devices will be like stove, microwave, coffee pot and dish washer. As for the offices equipment has more dense energy usage like Television, cable set, laptop, and varied list of rechargeable devices like cell phone,MP3 player, plasma LCD and all entertainment devices.. Figure 8 below shows the energy use breakdown in homes.
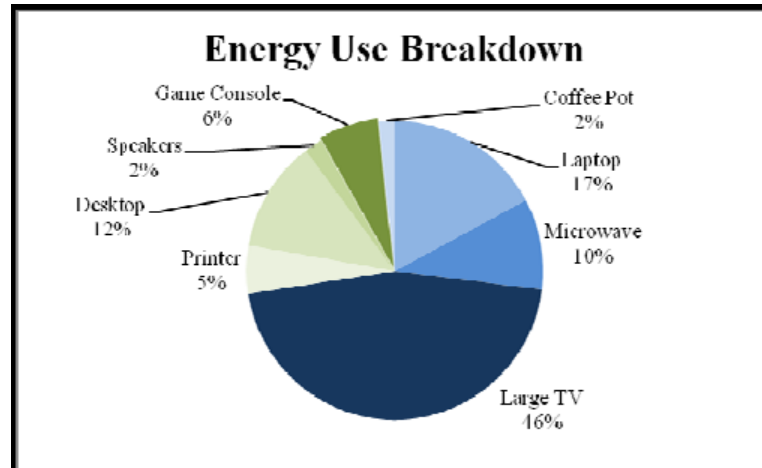
Figure 8 Energy use Breakdown by Appliances [12]

To aid understanding, a pie chart can be prepared showing the percentage of the electrical demand of each major and minor appliance as assessed in the home. Although the previous pie chart doesn't show the frequency of use, still that would point out the main load that we would like to monitor and control [12]. Figure 9 below shows the phantom load distribution by appliances.
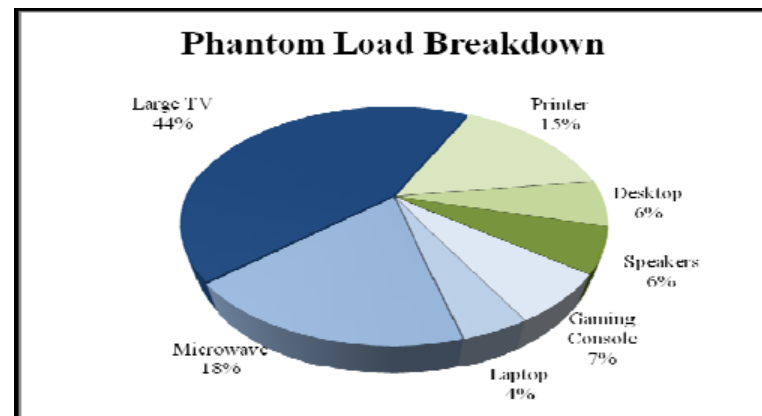
Figure 9 Phantom Load Distributions by Appliances [12]

2.6.3 Previous finding and potential impact in lowering house energy

Florida solar Energy Centre introduced and applied case study where they use the feedback in the most practical way in order to educate the home owners. The idea was installing energy monitoring or meters; record the feedback, read the data, turns off each appliances if that's possible and later repeat for each circuit until the whole home was covered. Based in the outcomes, data can guide drive such positive power off switches for different circuit in the whole system. Figure 10 below shows how standby loads were reduced in the household [13].
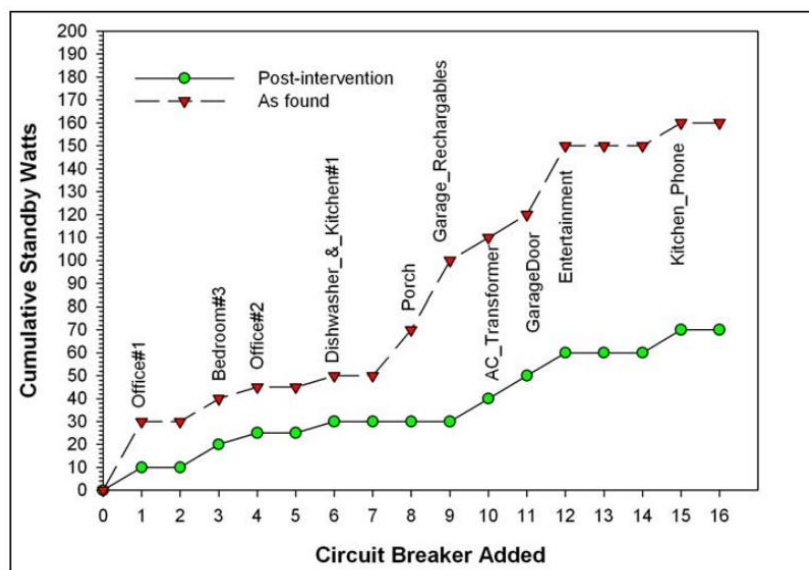
Figure 10 Evaluation of Standby Electric reductions, pre and post intervention [13]

Early research suggests that the effective feedback about the energy consumption to the consumers can be powerful resources of changing behavior and consumption. Nevertheless, behavior influences by the feedback wasn't researched well in the recent years but the also the feedback wasn't in real time manner at all. Also the impact of the

consumption over the day demand wasn't introduced in digestible way. The amount of reduction would be used as incentive and motivation to change the consumer behavior.

Another possibly interesting topic is integrating the different pricing rate of the utility. Recent study showed that introducing the real time date with the proper pricing rate would add significant value to the consumer behavior [13].

3. Chapter 3

Artificial Neural Networks

3.1 What are ANNs

It is well-known that the brain does not function like a computer. For one, the brain is quite accomplished at organizing its structural constituents, or neurons, making it faster than the world's top computers in performing calculations. The brain can easily perform perceptual recognition based task like identifying a face or an unfamiliar environment. The concept of artificial neural networks, commonly referred to as neural networks, was developed based on the idea that the brain computes in an entirely different manner from a conventional digital computer. When the neural network carries out a task, it tries to emulate how the brain performs tasks. A neural network is a massively parallel distributed processor made up of simple processing units that can store experimental knowledge and make it available for use. It can be implemented using electronic components or simulated in software on a digital computer. The network resembles the human brain in the following two ways:

1. The ANN obtains the information from the environment through a learning process.

2. Interneuron connection strengths usually called the Synaptic weights which will be used to store knowledge.

The procedure used to perform the learning process is aptly called a learning algorithm. The function of this algorithm is to modify the synaptic weights of the network in an orderly fashion to attain a desired design objective [14].

3.2 History of ANN

Neural network simulations appear to be a recent development. However, in reality, this field was established even before the advent of computers. Many of most important advances are the result of inexpensive computer emulations. Following an initial period of intense interest in neural networking among the scholarly community, problems with coding led to a period in which it was hard to secure funding for the research, and as a result little was published. This period of declining interest was led by scholars Minsky and Papert, who in 1969 published a widely accepted book that challenged the use of neural networks. Currently, thanks to recent efforts of scholars and researchers, neural network field enjoys a resurgence of interest and a corresponding increase in funding. The first artificial neuron was produced in 1943 by the neurophysiologist Warren McCulloch and the logician Walter Pits. But the technology available at that time did not allow them to do too much.

The neural network research declined throughout the 1970 and until mid-80's because the perception couldn't learn certain important functions. Neural Network regained importance in 1985-86. The researchers, Parker and leCun discovered a learning algorithm for multi-layer network called back propagation that could solve problems that were not linearly separable [14-15-16].

3.3 Why do we use ANN

Since neural networks work in nonlinear way, it can extract pattern and detect trend that are difficult to pick up by humans and other computational approaches. Neural networks need to be trained to decipher complex data or imprecise data. A trained neural network can be describe as the learning process of humans in this case it can be thought of as "expert" in the category of information it has been given to analyze (e.g. in school environment, there is a teacher for history and another one for physics, so an "expert in this particular information"). This expert (input training data) should be able to provide projections given new situations of interest and answer the "What if" questions. Other advantages of neural network are described in (KUMAR, 2009)[14-17]:

1. Adaptive Learning: To know how to perform a task based on training data or initial experience.

2. Self-Organization: While it learns, an ANN organizes or represents the information it receives by itself.

3. Real Time Operation: ANN computations may be carried out in parallel. Currently hardware devices are being designed and manufactured to take advantage of this capability.

4. Fault Tolerance Via Redundant Information Coding: Partial network destruction will cause a degredation in performance, though some network capabilities can be retained after major network damage.

5. Implementation Ability: Since the neural network is massively parallel it is very efficient at accomplishing certain tasks. As such, a neural network is best suited for implementation using very-large-scale-integrated (VLS1) technology.

3.4 Benefits of ANN

Neural networks provide significant benefits in many applications. Below we have listed some of the benefits of neural networks.

1. They are extremely powerful computational devices.

2. Massive parallelism makes them very efficient.

3. They can learn and generalize from training data – so there is no need for enormous feats of programming.

4. They are particularly fault tolerant – this is equivalent to the "graceful degradation" found in biological systems.

5. They are very noise tolerant – so they can cope with situations where normal symbolic systems would have difficulty.

6. In principle, they can do anything a symbolic/logic system can do, and more

3.5 Biological Model:

The human nervous system can be broken down into three stages that may figure 11 represent the human nervous system.
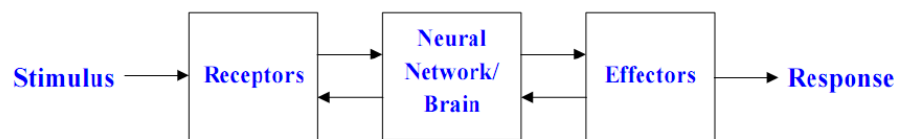
Figure 11 Block Diagram of a Human Nervous System [20].

The receptors collect information from the environment. The effectors generate interactions with the environment e.g. activate muscles. The flow of

information/activation is represented by arrows. There is a hierarchy of interwoven levels of organization:

1. Molecules and Ions

2. Synapses

3. Neuronal microcircuits

4. Dendritic trees

5. Neurons

6. Local circuits

7. Inter-regional circuits

8. Central nervous system

There are approximately 10 billion neurons in the human cortex. Each biological neuron is connected to several thousands of other neurons. The typical operating speed of biological neurons is measured in milliseconds.

The majority of neurons encode their activations or outputs as a series of brief electrical pulses. The neuron's cell body processes the incoming activations and converts the into output activations. The neurons nucleus contains the genetic material in the form of DNA. This exists in most types of cells. Dendrites are fibers which emanate from the cell body and provide the receptive zones that receive activation from other neurons. Axons are fibers acting as transmission lines that send activation to other neurons. The junctions that allow signal transmission between axons and dendrites are called synapses. The process of transmission is by diffusion of chemicals called neurotransmitters across the synaptic cleft [18-19-20]. Figure 12 shows structure of neural cell in the human brain.
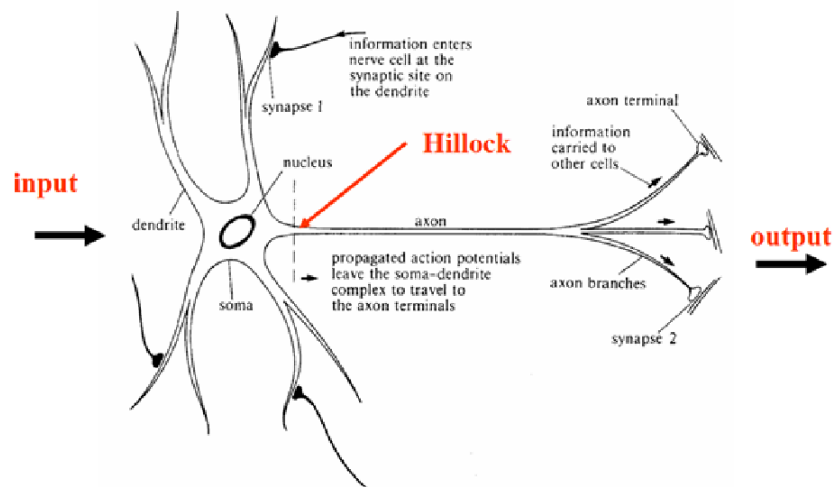
Figure 12 Structure of a neural Cell in the Human Brain [15]

3.6 Model of Artificial Neural Network:

An artificial Neural network (ANN) is a data processing system, consisting large number of simple highly interconnected processing elements as artificial neuron in a network structure. A neuron is an information processing unit that is fundamental to the operation of a neural network [14]. Figure 13 shows the Nonlinear model of a neuron. The three basic elements of the neuron model are:
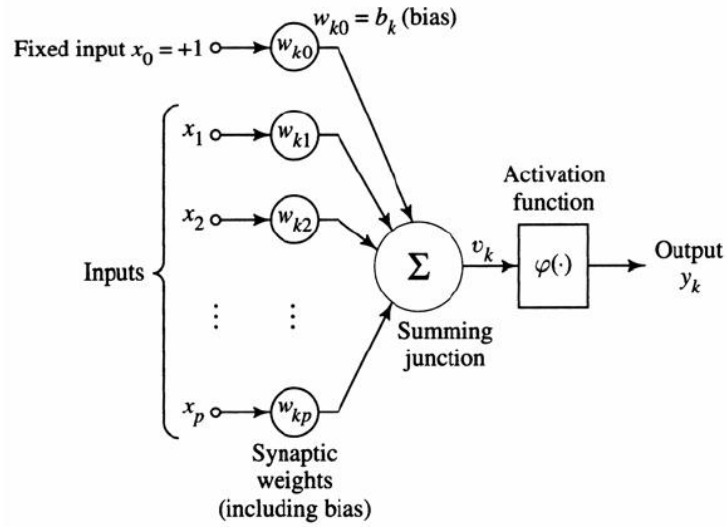
Figure 13 Nonlinear model of a neuron[14]

1. A set of weights, each of which is characterized by a strength of its own. A signal "xj" connected to neuron "k" is multiplied by the weight "wkj". The weight of an artificial neuron may lie in a range that includes negative as well as positive values.

2. An adder for summing the input signals, weighted by the respective weights of the neuron.

3. An activation function for limiting the amplitude of the output of a neuron. It is also referred to as squashing function which squashes the amplitude range of the output signal to some finite value.

$$v_k = \sum_{j=1}^{p} w_{kj} \times x_j \qquad \text{Equation -1}$$

And

$$y_k = \varphi(v_k + \theta_k) \qquad \text{Equation -2}$$

3.7 Network Architecture:

Previously we focused on the definition of a neural network, but in this one we will focus on classes of ANN. There are three fundamental different classes of network architectures [14-20, 24].

3.7.1   Single-layer Feed forward Networks:

A layered neural network is a network in which the neurons are organized in the form of layers. In a feed forward type network, the simplest form of a layered network, an input layer of source nodes projects onto an output layer of neurons, but not vice versa. A single-layer network is comprised of one input and one output layer, however, since no calculations take place at the input layer it is not considered an actual layer. In other words, in a feed forward network data can flow only in one direction, that is, from the input layer through the hidden layers and finally on to the output layer. The sum of the products of the weights and the inputs can be calculated for each neuron node, and if the value is above the threshold (typically 0) the neuron fires and takes the activated value (typically 1), if not it takes the deactivated value (typically -1)[14-20,24]. Figure 14 shows Single layer feed forward network.
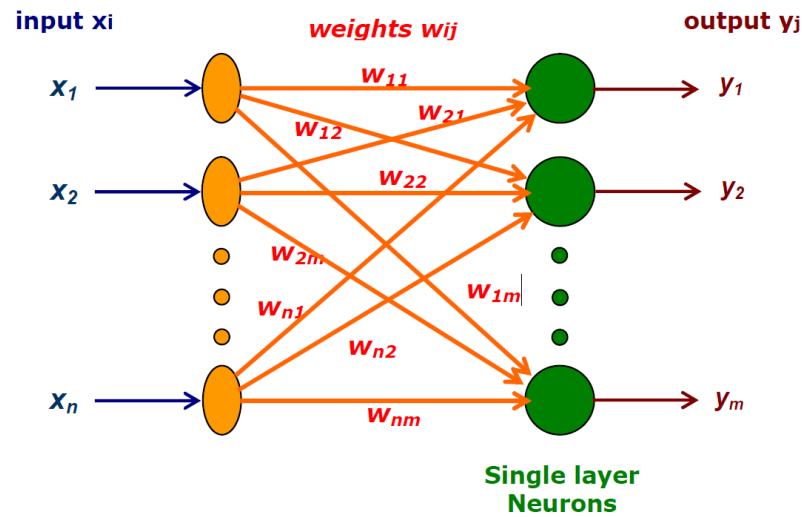
Figure 14 Single layer feed forward network [16]

3.7.2 Multilayer Feed Forward Networks:

The second type of a feed forward neural network has one or more hidden layers. The computational nodes of this network are called hidden neurons. The purpose of hidden neuron is to increase the computation between the input layer and the output one in some useful manner (e.g. to improve the accuracy of the used network so that it can do more difficult operations etc...). The network can comprehend more input data and extract higher order statistics when additional hidden layers are added. The input signal is applied to the neurons in the second layer and then the output signal of second layer becomes the input for third layer, and this pattern continues for the rest of the network [14-20]. Figure 15 shows Multilayer feed-forward network configuration
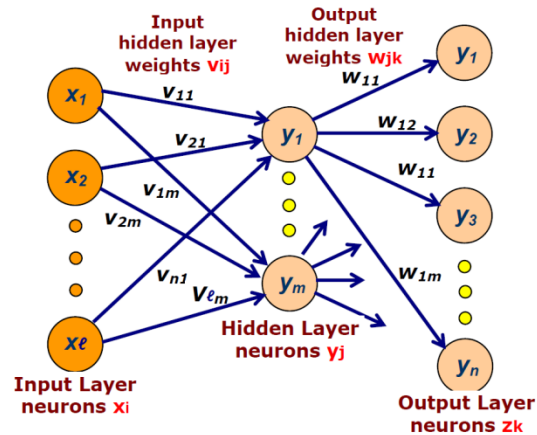
Figure 15 Multilayer feed-forward network configuration [16]

3.7.3 Recurrent Networks:

A recurrent neural network has at least one feedback loop. Recurrent networks are comprised of a single layer of neurons. Each of these neurons sends its output signal to the input of another neuron. During self-feedback, the output of a neuron is fed back into its own input. The learning capability of the network and its performance is influenced by the presence of feedback loops. In other words, activation values of the units undergo a relaxation process such that the network will evolve to a stable state in which these activations no longer change [16]. Figure 16 shows Recurrent neural networks

Fig Recurrent Neural Network
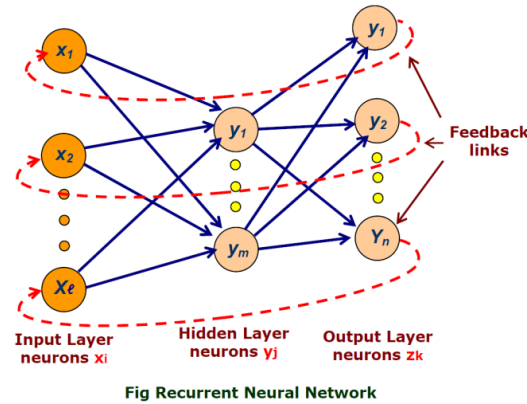
Figure 16 Recurrent neural networks [16]

3.8 Learning Method in the Neural Networks:

A neural network has to be configured in order to be effective. One way to do this is to make the application of a set of inputs produce the desired set of outputs. One approach by which the strengths of the connections can be set is by using a priori knowledge. The other method is to "train" the neural network. This means modifying the weights and bias by feeding it teaching patterns and letting it change its weights according to a learning rule. By a learning rule we mean a procedure for modifying the weights and biases of a network. The purpose of learning rule is to train the network to perform some task (e.g. Load prediction). Figure 17 Learning methods in the neural network. They fall into three broad categories [14-21, 24]:
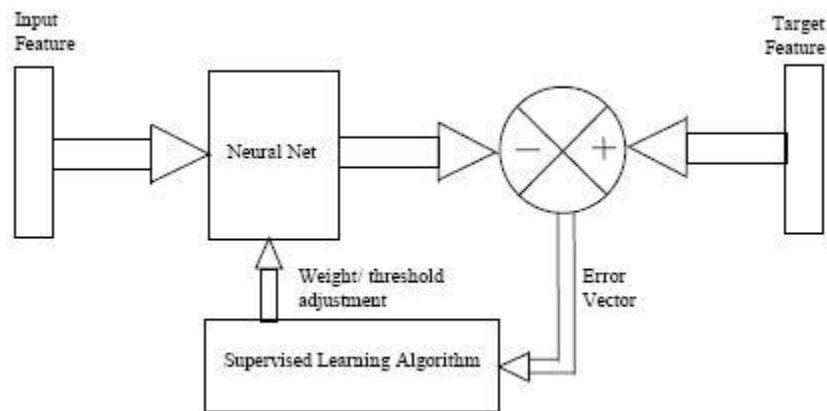
Figure 17 Learning methods in the neural network [21]

3.8.1 Supervised Learning:

In order for supervised learning to occur a teacher is need to give the neural network a target response. Using the training vector and the error signal, the network parameters adjust under the combined influence of the training vector and the error signal. Figure 18 shows supervised Learning. Error correction learning (a step-by-step adjustment process) is used to get the neural network to copy the teacher [14-20].
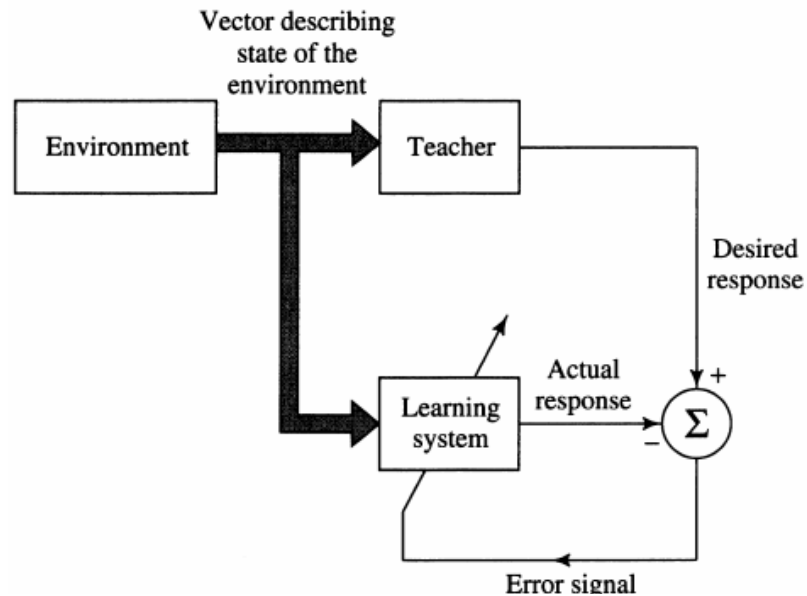
Figure 18 supervised Learning [14]

3.8.2 Reinforcement Learning:

In reinforcement learning (as opposed to supervised learning), the algorithm is only given a grade, or a measure of the network performance over an input sequence, instead of being provided with the correct output for each network input. If the output grade is a high number, such as 10, then the output is considered the same as the measured value. On the other hand, the prediction improves if the output is a lower number such as 0 [14-20]. In other words, in reinforcement learning, a teacher is presented but doesn't present the expected or desired output but only indicated if the computed output is correct or incorrect

3.8.3 Unsupervised Learning:

Unsupervised or self-organized learning is learning that occurs without the use of a teacher. In other words, there are no specific samples of the function that the network

needs to learn.   Once the network understands the statistical regularities of the input data,

it can automatically create internal representations for encoding features of the input data

[14-20]. Figure 19 shows Unsupervised learning.
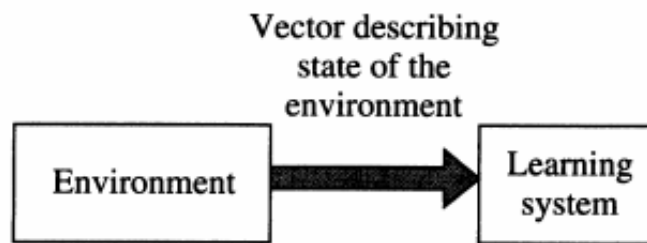
Figure 19 Unsupervised learning [14]

3.9 Back Propagation Algorithm

3.9.1   Introduction:

The error back-propagation algorithm is an algorithm that relies on the error-correction learning rule to train neural networks. It is considered a generalization of the commonly used adaptive filtering algorithm or least mean square (LMS) algorithm. Error back-propagation learning involves a forward pass and backward pass through the different layers of the network (Turetsky, 2000). An input vector is applied to the nodes of the network in a forward pass which then propagates through the network layer by layer. The response of the network is to create a set of outputs. While the network weights are fixed during the forward pass, during the backward pass they are adjusted in tandem with the error correction rule. An error signal is produced when the actual response of the network is subtracted from a desired response. It is then propagated

backward through the network in the opposite direction of the synaptic connections. By adjusting the weights, the actual network response will approach the desired response. It should be noted that a multilayer perceptron has three distinctive characteristics:

A nonlinear activation function is necessary for each neuron in the network. Generally this means using a sigmoid function commonly defined by the logistic function:

$$y = \frac{1}{1+e^{-x}}$$
  Equation -1

Another commonly used function is hyperbolic tangent.

$$y = \frac{1-e^{-x}}{1+e^{-x}}$$
  Equation -2

1 Without nonlinearities, the input-output relation of the network will only be a single layer perceptron.

2 Difficult tasks are accomplished through the use of hidden neurons. The network contains one or more layers of these neurons that are not part of the input or output of the network. The network exhibits a high degree of connectivity. A change in the connectivity of the network requires a change in the population of their weights [14-24, 25].

3.9.2 Learning Process:

To illustrate the process, a three layer neural network with two inputs and one output, as shown in the picture below, is used. Figure 20 shows Three layer neural network with two Inputs and a single output.
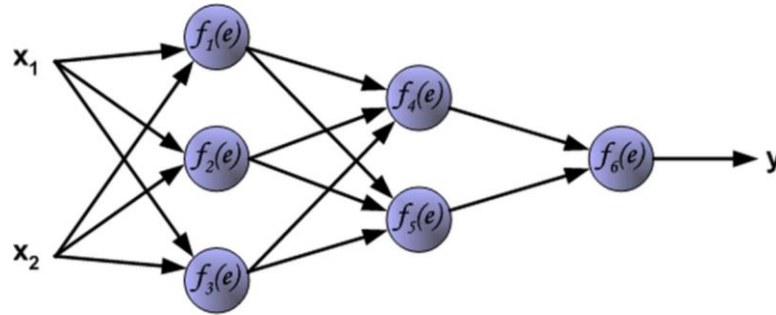
Figure 20 Three layer neural network with two Inputs and a single output [24]

Signal "z" is adder output signal, and "y = f(z)" is output signal of nonlinear element. Signal "y" is also output signal of neuron. The training data set consists of input signals (x1 and x2) assigned with corresponding target (desired output) y'. The network training is an iterative process. In each iteration weights coefficients of nodes are modified using new data from training data set. Symbols "wmn" represent weights of connections between output of neuron "m" and input of neuron "n" in the next layer. Symbols "yn" represents output signal of neuron n.

$$y_1 = f_1(w_{11}x_1 + w_{21}x_2) \qquad\qquad \text{Equation -3}$$

$$y_2 = f_2(w_{12}x_1 + w_{22}x_2) \qquad\qquad \text{Equation -4}$$

$$y_3 = f_3(w_{13}x_1 + w_{23}x_2)$$ Equation -5

$$y_4 = f_4(w_{14}y_1 + w_{24}y_2 + w_{34}y_3)$$ Equation -6

$$y_5 = f_5(w_{15}y_1 + w_{25}y_2 + w_{35}y_3)$$ Equation -7

$$y_6 = f_6(w_{46}y_4 + w_{56}y_5)$$ Equation -8
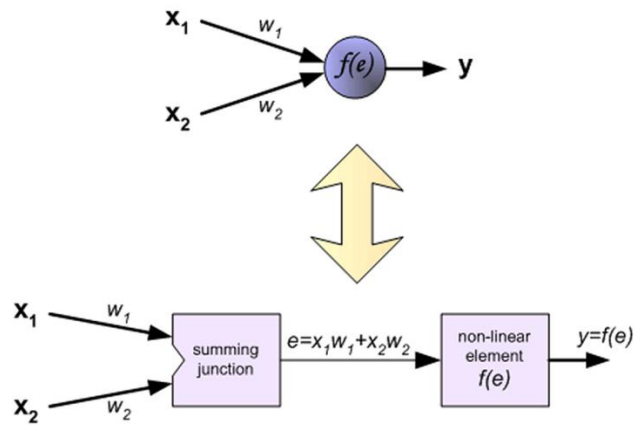
Figure 21 Learning process [24]

The desired output value (the target), which is found in training data set. The difference is called error signal δ of output layer neuron.

$$\delta = y' - y$$ Equation -9

$$\delta_4 = w_{46}\delta$$ Equation -10

$$\delta_5 = w_{56}\delta$$ Equation -11

$$\delta_3 = w_{34}\delta_4 + w_{35}\delta_5$$ Equation -12

$$\delta_2 = w_{24}\delta_4 + w_{25}\delta_5 \hspace{4cm} \text{Equation -13}$$

$$\delta_1 = w_{14}\delta_4 + w_{15}\delta_5 \hspace{4cm} \text{Equation -14}$$

When the error signal for each neuron is computed, the weights coefficients of each neuron input node may be modified. In formulas below df(z)/dz represents derivative of neuron activation function. The correction wij(n) applied to the weight connecting neuron j to neuron i is defined by the delta rule: Weight correction = learning rate parameter*local gradient*i/p signal of neuron i

$$\Delta w_{ij}(n) = \eta.\delta_i(n).y_j(n) \hspace{4cm} \text{Equation 3.9-15}$$

The local gradient δi(n) depends on whether neuron i is an output node or a hidden node:

If neuron i is an output node, δi(n) equals the product of the derivative dfi(z)/dz and the error signal ei(n), both of which are associated with neuron i.

If neuron j is a hidden node, δi(n) equals the product of the associated derivative dfi(z)/dz and the weighted sum of the δs computed for the neurons in the next hidden or output layer that are connected to neuron j [25-26].

3.9.3   Flow Chart:

As long as the desired response can be calculated, the BP algorithm should be reformed so that each hidden neuron is treated like an independent neuron. Usually a matrix (or array) is required to implement the BP algorithm and to store weights, the

input vector, and output vector. Each entry in the matrix corresponds to a neuron's relative position in the network. A pointer is maintained for each neuron that connects the synapses to all the linked neurons. It is important that the desired response for each neuron be given in advance, and not only for the neuron in the output layer [27]. Figure 22 shows Flowchart showing working of BPA
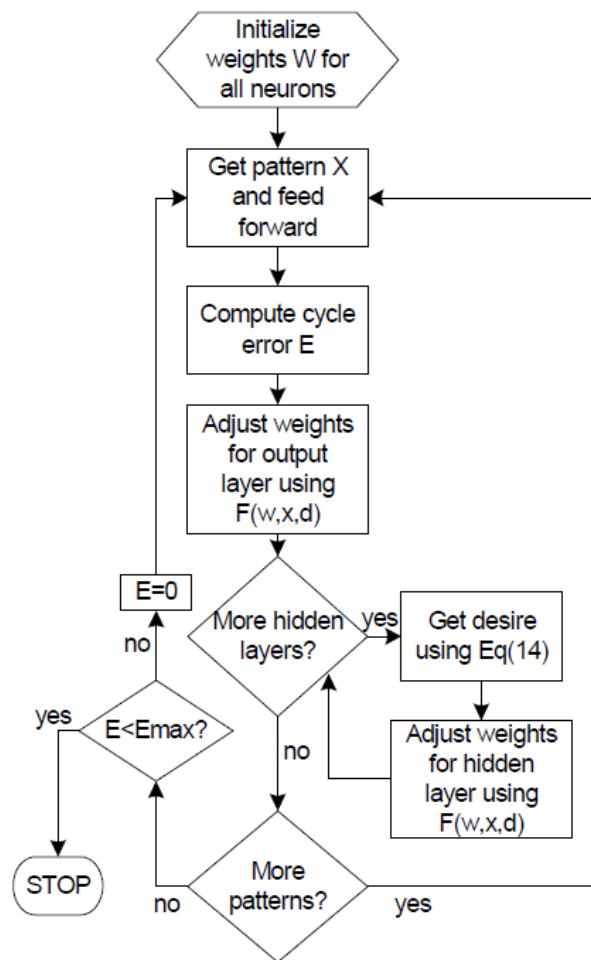
Figure 22 Flowchart showing working of BPA [27]

3.9.4 Stopping Criteria:

Reasonable criteria based on the unique properties of a local or global minimum of the error surface is used to terminate the weight adjustments since there are no well-defined criteria for stopping the operation of the back-propagation algorithm [14-24].

1. When the Euclidean norm of the gradient vector approaches a sufficiently small gradient threshold, the back-propagation algorithm converges. Unfortunately using this convergence criterion often begets longer learning times.

2. Typically, successful convergence of the back-propagation algorithm occurs when the absolute rate of change of the average squared error per epoch is in the range of 0.1 to 1 percent per epoch.

3. Termination of the back-propagation algorithm take place when the weight updates are sufficiently small.

4. Listed below is another useful criterion for convergence: Essentially, the network is tested for its generalization performance following each learning iteration. When the generalization performance peak or the generalization performance is considered adequate, the learning process stops. Figure 23 shows validation of early stopping
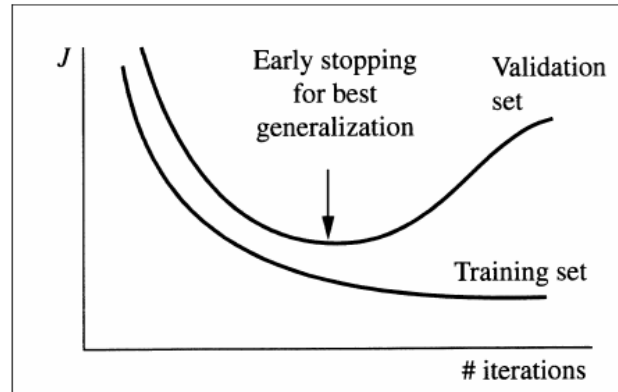
Figure 23 Validation of early stopping [14]

5. When the shape and properties of the error surface are not well defined, a stopping criterion for ending the training is implemented. This can be accomplished by limiting the iterations to a specific predefined value.

3.9.5   Pattern and Batch Modes of Training:

In practical applications of the back-propagation algorithm, learning results from the many presentations of training set to the network. An epoch represents one complete presentation of the entire training set during the learning process. This process is maintained on an epoch-by-epoch basis until the synaptic weights stabilize and the average squared error over the entire training set converges to attain a minimum value. By randomizing the order of the training examples from epoch to epoch, the search in weight space becomes stochastic over the learning cycles so that there will be no limit cycles in the evolution of the synaptic weight vectors [14-24].

Back-propagation learning may proceed in one of two ways for a given training set:

1. Weight updating is performed immediately after the presentation of each training example in the pattern mode of back propagation learning or in the operating mode for derivation of back-propagation method.

2. After the presentation of all training examples for a particular epoch, weight updating is performed in the batch mode.

In general, it is better to use the pattern mode of training instead of the batch mode. The likelihood that the back-propagation algorithm will be trapped at a local minimum is low since the patterns are presented to the network randomly, and the use of sample-by-sample updating of weights makes the search in weight space stochastic in nature. The positive of using the batch mode of training is that a more accurate estimate of the gradient vector is maintained [14].

3.9.6   Generalization:

A network is said to generalize well when the input-output relationship computed by the network is correct (or nearly so) for input/output patterns (test data) never used in creating or training the network. Here, of course, it is assumed that test data are drawn from the same population used to generate the training data.

Figure 24 illustrates how generalization might occur in a hypothetical network. The nonlinear input-output mapping represented by the curve depicted in this figure is computed by the network as a result of learning the points labeled as training data. The point marked on the curve as generalization is thus seen to be simply as the result of interpolation performed by the network[14]. Figure 24 and 25 shows example of generalization.
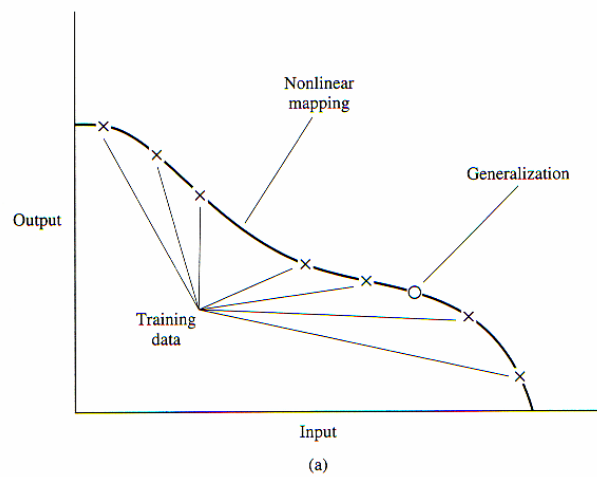
Figure 24 (a) properly fitted data (good generalization). [14]
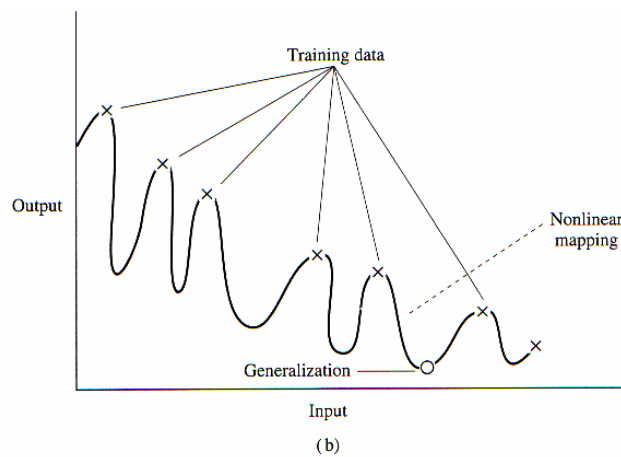
Figure 25 over fitted data (poor generalization). [14]

When, however, a neural network teaches too many specific input-output relations (i.e. it is over-trained), it may memorize the training data and, as such, be less able to generalize between similar input-output patterns. Such a network typically uses more hidden neurons than is actually necessary, with the result that its plasticity is too high and unintended curves in the problem space are stored in the synaptic weights. An example of

how poor generalization due to memorization in a neural network may occur is illustrated

in Figure 4-7b for the same data depicted in Figure 4-7a. The smoothness of input-output

mapping can be linked to model-selection criteria such as the Occam razor, in which the

simplest function is often selected in the absence of any prior knowledge that might

contradict it. In this context, the simplest network refers to the smoothest function that

approximates the mapping for a given error criterion [14-24].

4.Chapter 4

Design of Artificial Neural

4.1 Introduction:

In this chapter will present important fundamental steps in creating artificial neural network in order to identify the load type and the load forecasting. Steps here will be discussed for the general Architecture of the neural network, single layered and multi layered.

A method of designing a neural network model into distinct steps is used here. The steps are:

Step 1: Variable selection

Step 2: Data acquisition

Step 3: Data Conversion & Normalization.

Step4: Data portioning.

Step 5: Neural Network Paradigms.

Number of hidden layers

Number of hidden neurons

Number of output neurons

Transfers function

Step 6: Training.

Step 7: Testing and Evaluation criteria.

Figure 26 show the main work parts that are to be done in this chapter:
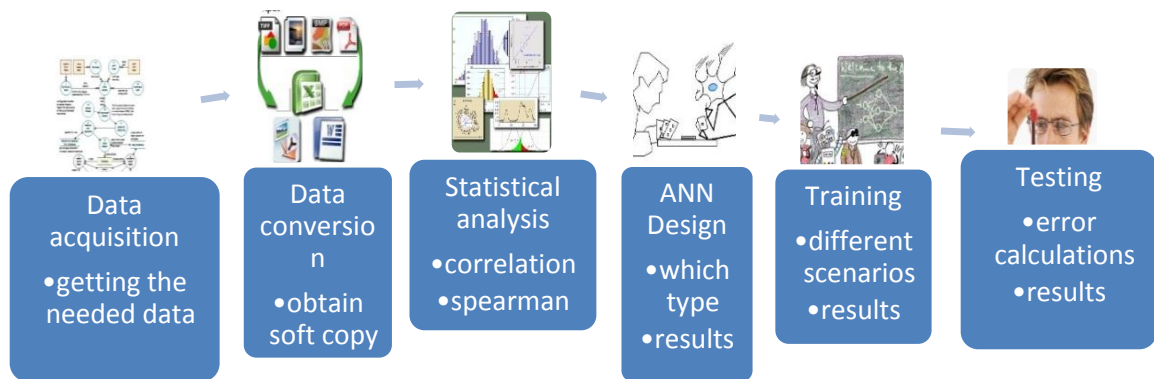
Figure 26 the steps used for load prediction

The procedure is usually not a single-pass one, but may require visiting previous steps especially between training and variable selection.

4.2 Step1 : Variable selection:

Achieve good design for the neural network comes after clear understanding for the problem, the correlation between the input and outputs and the quality of data [28]. Knowing which input variables are vital in the load being identified. This is easier said

than done because the very reason for relying on a neural network is for its powerful ability to detect complex nonlinear relationships between numbers of altered variables. Nevertheless at this point in the design process, the concern is about the raw data from which a variety of indicators will be developed. These pointers will form the actual inputs to the neural network. Finally the frequency and range of data will depend upon the research objectives of the researcher.

4.3 Step2 : Data Acquisition:

Before starting any research with neural network we data availability should be considered, Also the number of the variables and correlation between the input and the output. Time spend collecting data cannot be used for preprocessing, training and evaluating network performance. though, all data should still be checked for errors by examine day to day changes, ranges, logical consistency and missing observations[29]. In this research data collection has been done to the location of interest. The available data about the selected appliances and location are the active, reactive power harmonics, currents, date of the week outside air temperature vacations humidity for the two different approaches of this research. Later I will sort out the specifics character for each approach. Unfortunately the available data didn't cover large range of appliances when it comes to load identification approach.

4.4 Step3 : Data Conversion & Normalization:

In all of the neural network application, data processing is critical for reaching a good prediction performance. When applying neural networks for load time series prediction and pattern recognition. The input and output variables for which the data was collected are rarely fed into thee network in raw form. The raw data must be scaled between the upper and lower promises of the transfer functions most of the time zero is the minus and one is the highest. Now here I give a hint a head for the main two approaches for the this research

- First for load forecasting using characteristics of the proposed load prediction tool it was important to analysis the available data (Input, Target data of the ANN tool), as the input, Target data is a measured data of Minute of the day, Weekend and holidays, Temperature and Real power with reading every 15 minute there it is a huge amount of input data that need to be calcified as importance (in this step the data can be used as in the row data or after conversion)

- Second for the Pattern recognition, the characters that we are going to use over here are active power reactive power, current, apparent power, total power factor and. The target in this approach is determining the type of load based the previous Characters.

Usage row data right away without normalization will provide us with better results but will not overwhelmed the problems of poor measured data measurement. In other cases where the row data was used for two different years, this will not be an issue because there are different heights.

The strength of the relation between the different input data and specially if its related to the kW demand should be highly considered. This approach can be done by the statistical analysis. So in general correlation analysis is a statistical tool which studies the relationship between two variables.

4.4.1   Correlation analysis:
4.4.1.1 Introduction:

Correlation analysis includes approaches and practices used for studying and measuring the extent of the relationship between two variables. So correlation analysis is statistical process by which we can determine the degree of association or relationship between two or more variables. The amount of correlation in a sample of data is measured by the sample coefficient of correlation, which is, generally, denoted by 'r' or by 'ρ'.

4.4.1.2 Properties of coefficient of correlation:

- Measure of the intimacy of a fitting in a relative sense.

- Correlation coefficient lies between '-1' and '+1', i.e., $-1 \leq r \leq +1$.

- The correlation is prefect and positive if 'r=1' and is perfect and negative if 'r=-1'.

- If 'r=0', then there is no correlation between the two variables and thus the variables are said to be independent.

- The correlation coefficient is a pure number and is not affected by a change if origin and scale in magnitude.

4.4.1.3 Methods to study correlation:

- Scatter diagram method or Dot diagram method.

- Karl Perason's coefficient of correlation method

- Spearman's rank correlation method.

- Concurrent deviation.

- Two-way frequency table method.

4.4.1.4 Spearman's rank correlation coefficient:

The coefficient of rank correlation is based on the various values of the varieties and is denoted by 'R'. It is applied to the problems in which data cannot be measured quantitatively but qualitative assessment is possible. In this case the vest individual is given rank number 1, next rank 2 and so on; this coefficient of rank correlation is given by the formula:

$$R = 1 - \frac{6 \times \Sigma D^2}{n(n^2 - 1)} \qquad \qquad \text{Equation-1}$$

Where D^2 is the square of the difference of corresponding ranks, and 'n' is the number if pairs of observations.

4.4.1.5 Our Case:

In our case we don't have the ranking given with the data of interest (Current, Active Power Reactive power, and Harmonics) and duplication is available in the data that is why using Spearman's when equal ranks are given to more than two attributes will be needed.

If two or more individuals are placed together in any classification with respect to an attribute, i.e., if in case of variable data, that are more than one item with the same rank in either or both the series then the Spearman's Rank correlation coefficient formula given above does not give the correlation coefficient for tie rank. The problem is solved by assigning average rank to each of these individuals who are put in tie. In order to find the rank correlation coefficient of repeated ranks or tie ranks, an adjustment or correction factor is added to the Spearman's rank correlation formula. This factor is given as:

$$correction\ factor = \frac{m(m^2-1)}{12}$$
Equation 4.4-2

Where 'm' is the number of times an item is repeated. This correction factor is to be added for each repeated value in both the series.

$$R = 1 - \frac{6\left[\Sigma D^2 + \frac{m_1(m_1^2-1)}{12} + \frac{m_2(m_2^2-1)}{12} + \frac{m_3(m_3^2-1)}{12} + \cdots\right]}{n(n^2-1)}$$
Equation 4.4-3

When applying the last equation (3) on our data it is important to keep in your mind the following notes:

You need to keep the same order of the data as it is given (whatever you do with the data you have to make sure that the same value of first variable 'X' is still corresponding to the same first value of the second variable 'Y' and so on). Make sure that repeated data will take the same rank value and will be averaged afterword.   Make

sure that the value directly coming after the repeated one is taking the next value of ranking without repeating taken into consideration.

4.5 Step4 : Data Portioning:

Common practice is to divide the time series into three distinct sets called the training, testing and validation sets. The training set is the largest set and is used by neural network to learn the patterns present in data. The testing set, ranging in size from 10% to 30% of the training set, is used to evaluate the generalization ability of a supposedly trained network. A final check on the validation set chosen must strike a balance between obtaining a sufficient sample size to evaluate a trained network and having enough remaining observations for both training and testing. The validation set should consist of the most recent contiguous observations. In this work the approach in evaluation neural networks used as follows:

- Training Set: 70%

- Testing Set: 15%

- Validation Set: 15%

4.6 Step6 : Neural Network Paradigms:

There are an infinite number of ways to construct a neural network. Neurodynamics and architecture are two terms used to describe the way in which a neural network is organized. The number of input neurons is one of the easiest parameters to select once the independent variables have been reprocessed because each independent variable is represented by its own input neuron [30]. The tasks of selection of the number of hidden

layers, the number of the neurons in the hidden layers, the number of input neurons as well as the transfer functions are much more difficult.

4.6.1   Number of Hidden Layers:

The hidden layer(s) provide the network with its ability to generalize. In practice, neural networks with one and occasionally two hidden layers are widely used and have performed very well. Increasing the number of hidden layers also increases computation time and the danger of over fitting which leads to poor out-of-sample forecasting performance. In the case of neural networks, the number of weights, which is inexorably linked to the number of hidden layers and neurons, and the size of the training set (number of observations), determine the likelihood of over fitting (Baum and Haussler,1989)[29-31]. It was applied neural networks structure with one and two hidden layers to a comparison.

4.6.2   Number of Hidden Neurons:

Despite its importance, there is no "magic" formula for selecting the optimum number of hidden neurons. Therefore researches fall back on experimentations. However, some rules of thumb have been advanced. A rough approximation can be obtained by the geometric pyramid rule proposed by Masters (1993). For a three-layer network with an input neurons and m output neurons, the hidden layer would have n · m neurons. Baily and Thompson (1990) suggest that the number of hidden layer neurons is a three-layer neural network should be 75% of the number of input neurons. Katz (1992) indicates that the optimal number of hidden neurons will generally be found between one-half to three times the number of input neurons. Ersoy (1990) proposes doubling the number of hidden

neurons until the network´s performance on the testing set deteriorates. Klimasauskas[32] (1993) suggests that there should be at least five times as many training facts as weights, which sets an upper limit on the number of input and neurons. Because of these features, this work applied different structures to all data, chosen randomly, with 22, 34, 40 and 50 neurons in the hidden layer as to describe the best structure according to the index.

4.6.3   Number of Output Neurons:

Deciding on the number of neurons is somewhat more straightforward since there are compelling reasons to always use only one output neuron. Neural network with multiple outputs, especially if these outputs are widely spaced, will produce inferior results as compared to a network with a single output [29]. The modeling applied in this work aims to one a half month past closing value in the future forecasting, and as cited above, was utilized one output layer structure.

4.6.4   Transfer Function:

The majority of current neural network models use the sigmoid transfer function, but others such as the tangens hyperbolicus, arcus tangens and linear transfer functions have also been proposed (Haykin, 2001). Linear transfer functions are not useful for nonlinear mapping and classification. Levich and Thomas (1993) and Kao and Ma (1992) found that financial markets are nonlinear and have memory suggesting that nonlinear transfer functions are more appropriate. Transfer functions such as the sigmoid are commonly used for time series data because they are nonlinear and continuously differentiable which are desirable properties for network learning. In this study, sigmoid transfer function was applied in the network proposed.

4.7 Step6 : Training:

Training a neural network to learn patterns in the data involves iteratively presenting it with examples to the correct known answers. The objective of training is to find the set of weights between the neurons that determine the global minimum of the error function. Unless the model is over fitted, this set of weights should provide good generalization. The back propagation network, applied in this work, uses the gradient descent training algorithm which adjusts the weights to move down the steepest slope of the error surface. Finding the global minimum is not guaranteed since the error surface can include many local minima in which the algorithm can become "struck". This section will discuss when to stop training a neural network and the selection of learning rate and momentum values.

4.8 Step7 : Testing and Evaluation Criteria:

In this part we will try to determine the accuracy of the proposed ANN tool for load prediction and pattern recognition. After giving a good idea about the expected results of the used ANN tool for load prediction (refer to spearman's rank correlation & Regression analysis), a method for accuracy is needed. By accuracy we mean the ability to calculate the error of the prediction and be able to compare results of the different proposed prediction tools. The most used way of error calculation is the Root mean square deviation.

4.8.1   Root Mean Square Deviation:

The root mean square deviation (RMSD) or root mean square error (RMSE) is a frequently-used measure of the differences between values predicted by a model or an estimator and the values actually observed from the thing being modeled or estimated. RMSD is a good measure of precision. These individual differences are also called residuals, and the RMSD serves to aggregate them into a single measure of predictive power. (Wikipedia®, http://en.wikipedia.org/wiki/Root_mean_square_deviation, 2010)

$$\boldsymbol{RMSD} = \sqrt{\boldsymbol{MSE(x^{'})}} = \sqrt{\frac{\sum_{i=1}^{n}\left(x_i^{'}-x_i\right)^2}{n}} \qquad\qquad \text{Equation -1}$$

Where:

- "n" is the total number of reading (in our case is the total number of measurement available used as target data of the ANN tool for prediction).

- "$x\_i^{\wedge}$"" is the "I" value of the predicted parameter.

- "x_i" is the "I" value of the measured parameter.

With this step done we can now have more accurate understanding of the output results not only a general over view like before. The result of this analysis will help in differentiating between the different results and is a very important parameter for the next step of energy prediction (where the error of this step should be as small as possible so that it will not have that negative effect on the next step of energy prediction tool).

4.8.2   Mean Absolute Error:

MAE: Mean absolute error performance function is a network performance function. It measures network performance as the mean of absolute errors. The calculation were done like following,

If the network is given a batch of inputs P, The error is calculated by subtracting the output A from target T. Then the mean absolute error is calculated.

$$p = [\text{-10 -5 0 5 10}]; t = [0\ 0\ 1\ 1\ 1];$$

$$y = sim(net, p)$$

$$e = t - y$$

$$perf = mae(e)$$

4.8.3   Mean Squared Error (MSE):

MSE: is a measure of how close a fitted line is to data points. For every data point, you take the distance vertically from the point to the corresponding y value on the curve fit (the error), and square the value. The smaller the Mean Squared Error, the closer the fit is to the data. The MSE has the units squared of whatever is plotted on the vertical axis. Calculation were done like the fooling,

The network is given a batch of inputs P. The error is calculated by subtracting the output A from target T. Then the mean squared error is calculated.

$$p = [\text{-10 -5 0 5 10}]; t = [0\ 0\ 1\ 1\ 1];$$

$$y = sim(net, p)$$

$$e = t - y$$

$$perf = mse(e)$$

$$mse = mean(e(:).^2)$$

4.8.4 Normalized-Mean-Square Error (NMSE):

NMSE is just another way of evaluating the ANN performance. Here are the steps to calculate NMSE: If y is the matrix of N p-dimensional column estimates of t, the most naive model is the constant model:

$$y00 = repmat(mean(t, 2), 1, N)$$

With error and mean square error

$$e00 = t - y00$$

$$MSE00 = mean(e00(:)^2)$$

The corresponding quantities for the more complex model are

$$e = t - y$$

$$MSE = mean(e(:).\ 2)$$

Resulting in: $NMSE = MSE/MSE00$

4.8.5 Mean Absolute Percent Error:

MAPE: mean absolute percent error is the last method I used to measure the model Performance. The results we got using this method are pretty good and acceptable especially for the excluded gap data. Here how we calculate the performance using this method.

$$p = [-10 \ -5 \ 0 \ 5 \ 10]; \ t = [0 \ 0 \ 1 \ 1 \ 1];$$

$$y = sim(net, p)$$

$$e = t - y$$

$$mape = nanmean(abs(r.(\frac{1}{t} * 100)$$

$$mse = mean(e(:).^2)$$

In conclusion we can use one of the previous error calculation to define how accurate is our model.

5 Chapter 5

Load Identification using ANN

5.1 Introduction

Different application-specific methods have been investigated to understand how the load affects the power consumption, power quality, efficiency, reactive power compensation, system planning, and so forth. At the core of these approaches are the needs to understand the characteristic–load signature. Non-intrusive load monitoring (NILM) is one of the earliest approaches to nonintrusive monitoring and can determine the operating schedule of electrical loads in the target system from measurements made at a centralized location by identifying times in which the electrical power measurement changed from one nearly constant (steady- state) value to another, normally corresponding to loads either turning on or off and characterized by their magnitude and sign in real and reactive power. Reactive and active power is not unique and may become crowded and indistinguishable as the load number increases. Research similar to this includes echo-resonance load monitoring (ERLM), sign-up load registration (SULR), and developing a library and taxonomy of load signatures [34-39]. The following steps are used to define the state and type of Loads. Fig. 27 shows a block diagram representing the load identification process.
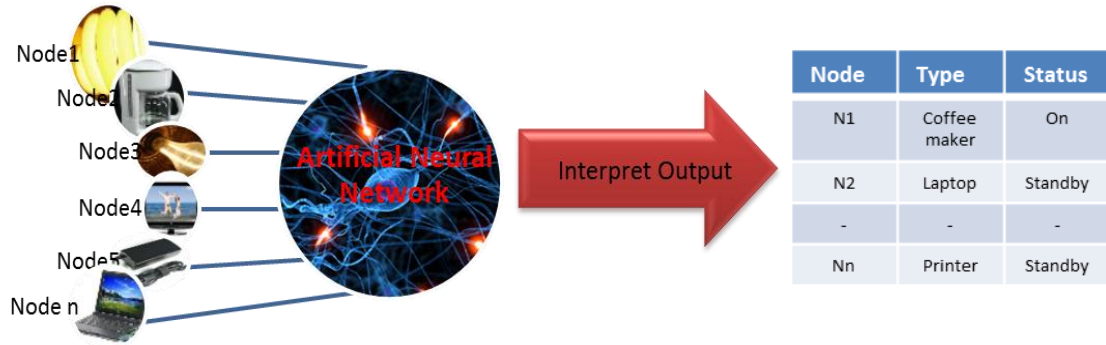
Figure 27 step used in load identification

5.2 Approach

5.2.1 Defining the type of load (Load Recognition)

In this section, a real-time power monitoring system is developed for data acquisition. A Current and Voltage transducer integrated with op amps and microcontroller will be installed in the loads end. Using these power sensor current and voltage waveforms are sampled every 100 micro seconds, these samples (digital values) are sent to the computer via a serial port using the RS232 protocol. By using this process load characteristics will be measured and represented. A signal processing technique and estimation algorithm will be created for signal filtering, signal disaggregation, and load recognition.   Previously, different methods have been used to understand the different load features (power quality, efficiency, reactive power compensation system planning, and load shedding). These approaches analyze the load signature. The concept of load recognition for the purposes of this research is based on the special load signature in which every load has a unique feature in its consumption behavior that is very close to a

human signature. Load features include the active power, RMS current, apparent power wave, power factor and THD current. The load disaggregation process is as follows:

5.2.1.1 Data acquisition

First, the steady state and transient state signal will be captured. In the steady state, the fixed samples rate is set. The transient signal is used to capture the waveform during switching.

Active and reactive power of a number of appliances was measured using data analysis Power Quality Logger and model "PQL120". The Appliances are measured individually. The data obtained in the measurement is stored into a computer for further study; the appliances include

- Fan

- Light bulb(candescent/Incandescent Light)

- Laptop

- Printer

- Coffee maker

- Charger

The active and reactive powers in their steady states are observed on the power quality logger (oscilloscope). Tables (1-6) below represent samples of the captured data. Each appliance was represented with the following signature in each table.

Table 1 captured characters for Fan load

	Current [A]	0.342	0.34	0.34	0.34	0.34	0.34	0.34
Fan	Power [W]	20	20.02	20.12	20.32	20.16	20.28	20.3
	Apparent Power [VA]	41.08	40.38	40.4	40.62	40.14	40.08	40.02
	Power Factor	0.482	0.492	0.494	0.496	0.496	0.502	0.502
	Current THD	8.06	7.88	7.98	8	7.34	7.14	7.02

Table 2 captured characters Fluorescent Light Load

	Current [A]	0.29	0.292	0.284	0.268	0.28	0.264	0.264
Flour-Light	Power [W]	10.26	10.32	10.24	10.04	10.36	9.84	9.68
	Apparent Power [VA]	34.64	34.96	33.9	32.08	33.68	31.74	31.72
	Power Factor	0.29	0.292	0.304	0.318	0.31	0.316	0.308
	Current THD	1.9934	1.9862	1.9596	1.933	1.9594	1.9494	1.9568

Table 3 captured characters for Laptop Load

	Current [A]	0.386	0.36	0.366	0.37	0.356	0.352	0.38
Laptop	Power [W]	41.1	38.42	38.98	39.66	38	37.56	40.54
	Apparent Power [VA]	46.92	44.02	44.68	45.5	43.66	43.12	46.42
	Power Factor	0.872	0.87	0.87	0.87	0.868	0.87	0.87
	Current THD	0.4692	0.4402	0.4468	0.455	0.4366	0.4312	0.4642

Table 4 captured characters for Printer Load

	Current [A]	0	0	2.93	0	0	0	0
Standby Printer	Power [W]	0	0	341.86	0	0	0	0
	Apparent Power [VA]	0.72	0.76	342.52	0.7	0.66	0.7	0.66
	Power Factor	0	0	0.408	0.002	0	0.004	0
	Current THD	0.0072	0.0076	3.4252	0.007	0.0066	0.007	0.0066

Table 5 captured characters for Charger Load

	Current [A]	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Charger	Power [W]	-0.42	-0.4	-0.4	-0.4	-0.38	-0.4	-0.38
	Apparent Power [VA]	1.44	1.4	1.38	1.4	1.44	1.44	1.44
	Power Factor	-0.354	-0.356	-0.358	-0.352	-0.318	-0.336	-0.322
	Current THD	0.7282	0.7814	0.744	0.7392	0.8998	0.8174	0.8852

Table 6 captured characters for Incandescent light Load

	Current [A]	**0.63**	**0.624**	**0.62**	**0.622**	**0.624**	**0.624**	**0.62**
	Power [W]	74.1	74	73.94	73.98	73.98	74	73.94
Incandesce-	Apparent Power [VA]	74.1	74	73.94	73.98	73.98	74	73.96
	Power Factor	1	1	1	1	1	1	1
	Current THD	0.01362	0.06646	0	0	0	0	0

The power analyzer will allow us to form reports for each measured load. The report will be multi-function where we can have most of the well-known characters for the loads. Each report shows the changes on the characters over time which in turn will point put the standby and on mode for each load.

Fig 28-31 shows some example of waveform of the tested appliances
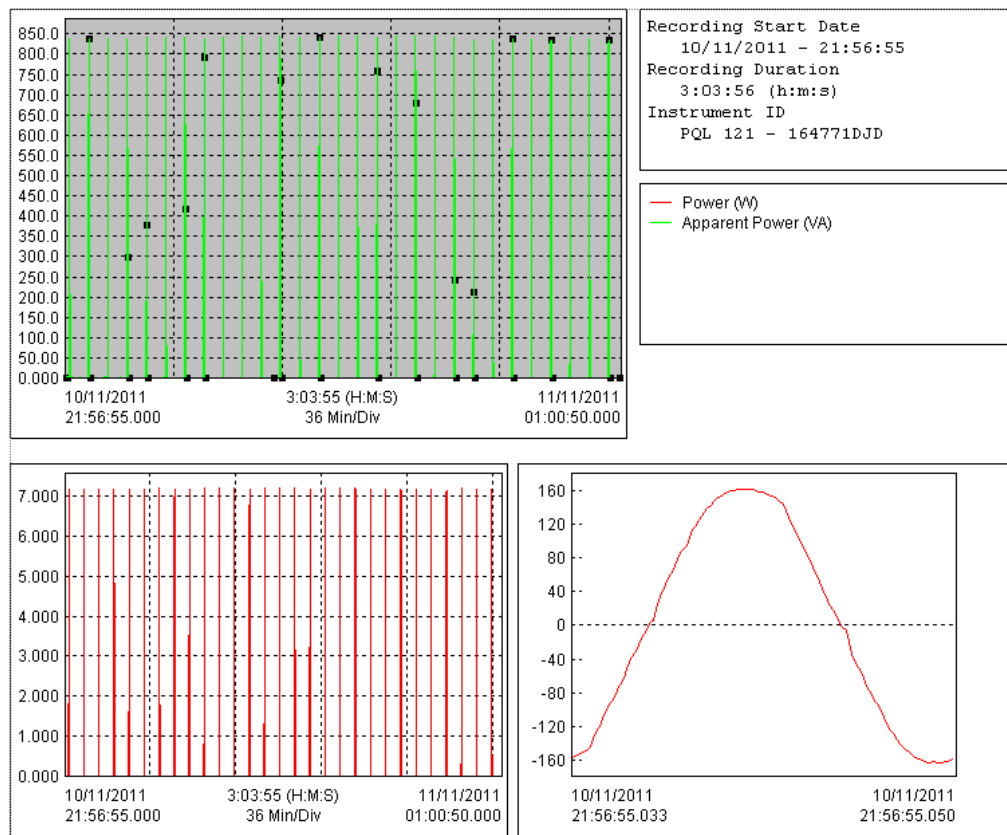
Figure 28 Active and apparent power of Coffee Maker
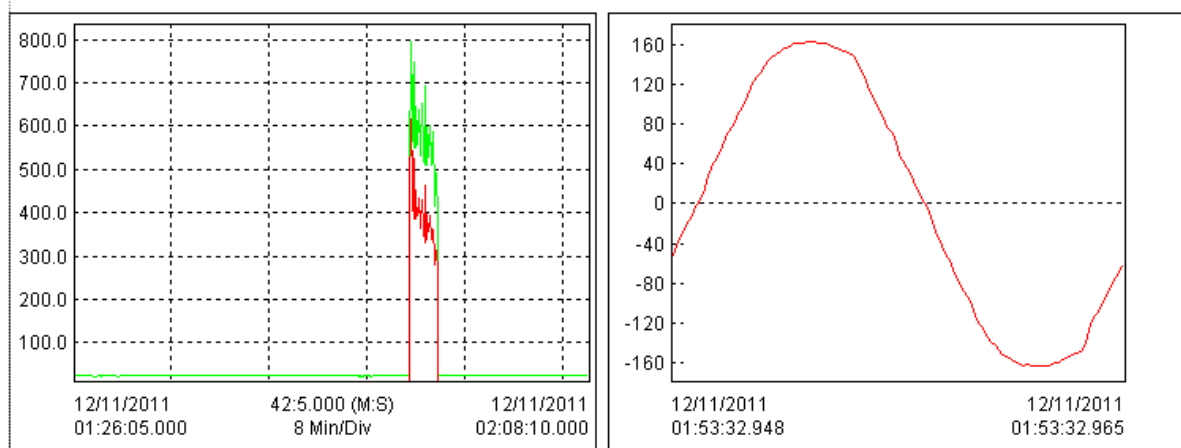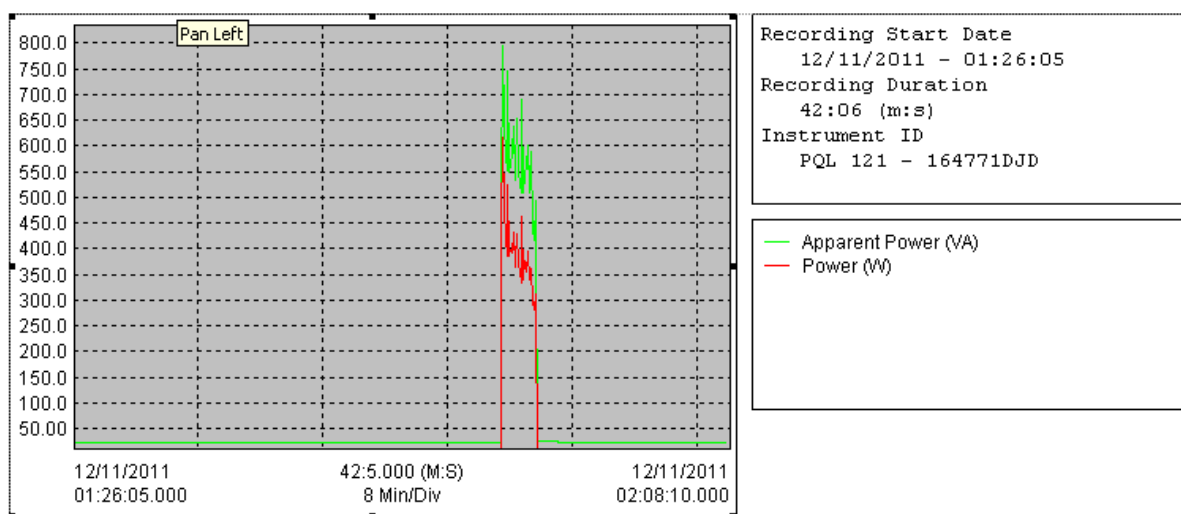
Figure 29 Active power, apparent power, voltage and current RMS of Fan
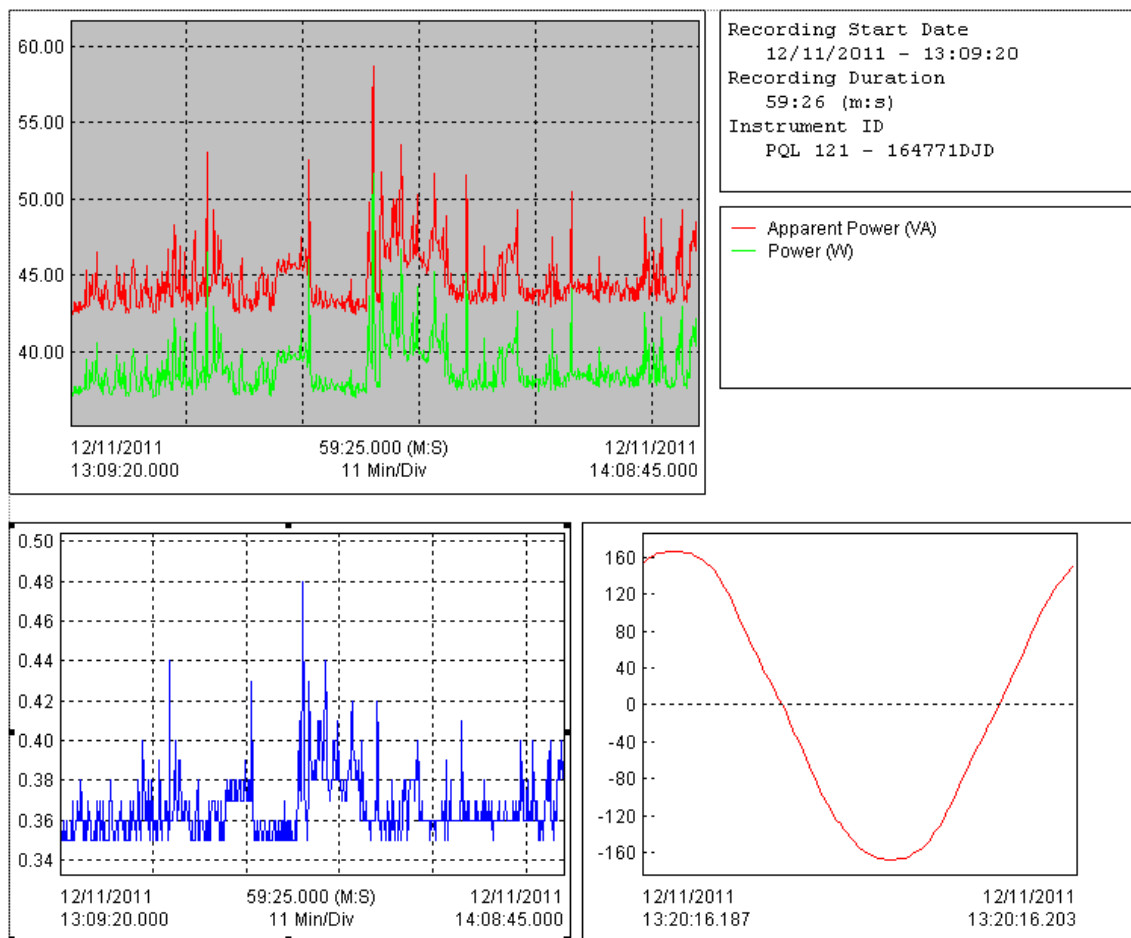
Figure 30 Active power, apparent power, voltage and current RMS of laptop
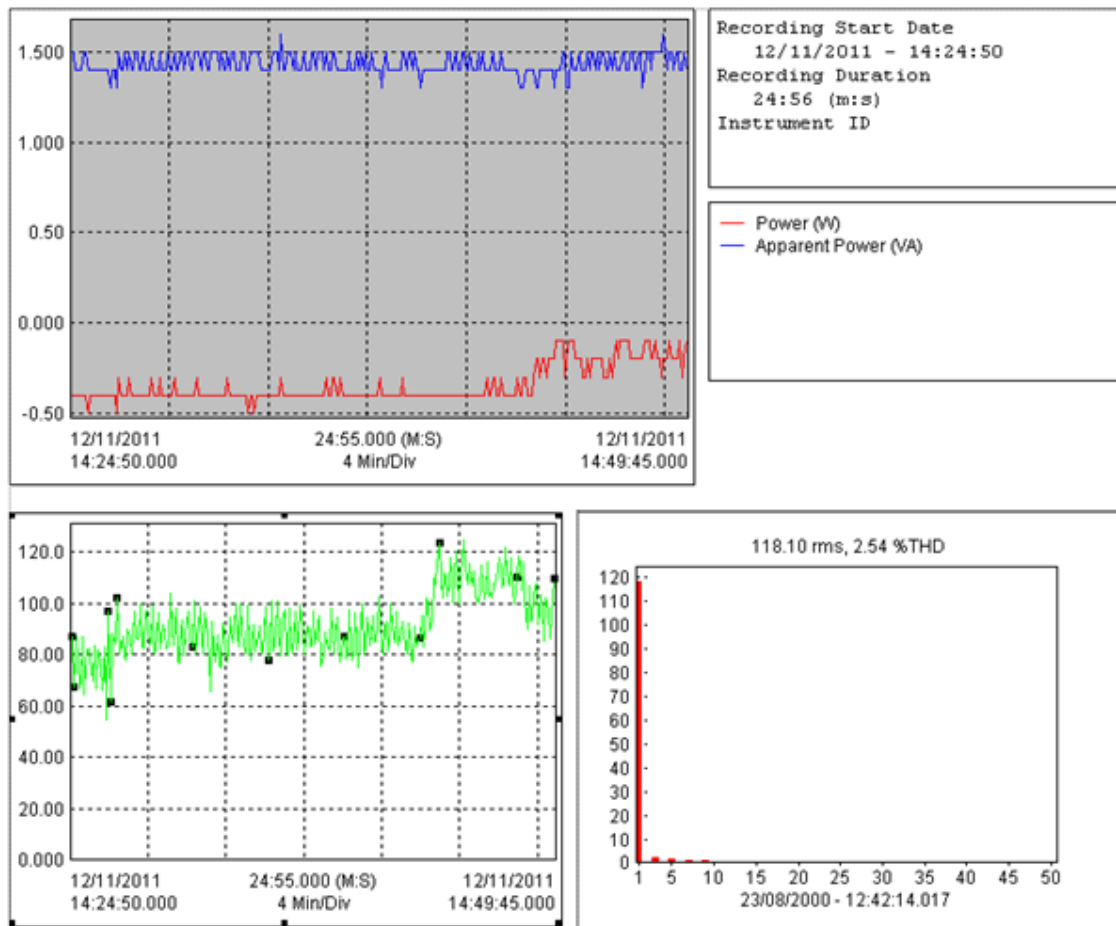
Figure 31 Active power, apparent power, voltage and current RMS of charger

The power consumption was recorded for the ON states and steady states, the difference between the two consecutive powers will determine the waste of energy represented by phantom loads. Other characteristic like Current, total harmonic distortion, reactive power, and RMS for the current and the voltage where obtained and stored for further study. More appliances were measured to obtain a full database of load signature.

5.2.1.2 Data Processing Model

The data from the signals will be conditioned, and re-sampled to align the current respective to the voltage and filtering. The data which was showed in the table before was the ones we indicate to in this part. The over fitted data was deleted with all the noises and nonsense.

5.2.1.3 Event Detection

The data from the preceding steps will be used to detect when the actual appliances are being switched on/off. This step result will be added up in the result part of this chapter

5.2.1.4 Pattern Recognition

The artificial neural network (ANN) will be taught to learn the specific features of the different appliances and identify them. In this study, Multi-layer-Feed forward neural network is selected. The method used to train the system is based on the neural network approach [41].

5.2.1.5 Committee Decision Mechanism

This mechanism will be used to extract different features such as the PQ, current wave, harmonics, and the instantaneous power wave. This will be called the candidate pool. The committee Decision mechanism evaluates the potential solutions and renders the best final solution from a group of potential solutions [42]. Below I have included the program that will take the decision after each class gave a threshold of certainty. So the program will scan each raw and pick up the class with the highest rate.

```
function [output]=decide(simout)

input=simout.signals.values';

for i=1:size(input,2)
    seeker=find(input(:,i)==max(input(:,i)));
    switch seeker
        case 1
            output{i}='Fan';
        case 2
            output{i}='Flour Light';
        case 3
            output{i}='Laptop';
        case 4
            output{i}='printer';
        case 5
            output{i}='charger';
        case 6
            output{i}='Incandescent Light';
        case 7
            output{i}='Coffee Maker';
    end
end

end
```

5.2.2   Detecting Load States

The steady state, or the condition in which the phantom load exists and potential energy is wasted, will be defined. The transient state and steady state determine what load type is turned "on". Therefore, the steady state plays a vital rule in terms of creating energy savings. The cluster analysis algorithm is used to locate these changes in two dimensional signatures of Δp and ΔQ. The total load P[t] is sampled at a fast sampling rate, and the difference in ΔP[t]between the two consecutive readings is calculated [43].

The appliance switches on/off as a result of the increase or decrease in the power level. In the first part of the study a bottom-up approach is used in which the majority of loads will be connected to sensors. As the study progresses, the configuration will be changed so that there will be one sensor in centralized location rather than on each node.   The

difference in the power level of the switching characteristics of the appliances is detected and calculated using the change in delta P, given below:

$$\Delta P[t] = P[t] - P[t-1]$$

When it is difficult to determine the difference between ΔP and ΔQ, another signature (like harmonics) can be used to define the loads [43].

5.3 ANN Model

The neural network being employed for this research is the Multi-Layer Feed Forward. It is also known as the multi-layer perception (MLP) network. This neural network consists of one or more hidden layers, whose computational nodes are correspondingly called hidden neurons. The purpose of the hidden neuron is to add up more computation between the input and output layer of cores in some useful manner. The input signal is applied to the neurons in the second layer. The output signal of second layer is used as inputs to the third layer, and so on for the rest of the network. Figure 32 represent the topology of an MLP.
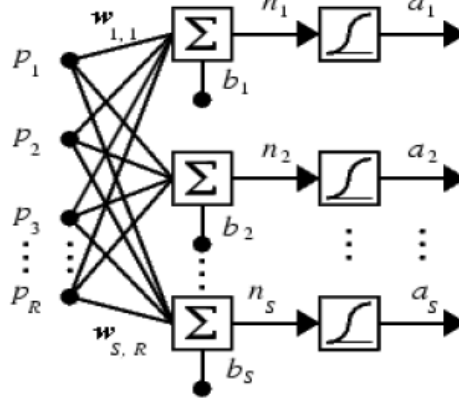
Figure 32 topology of an MLP
Topology of an MLP where R is the Number of Elements in the Input Vector, S is the Number of Neurons in the Layers, and Wij are the Weighting Factors between the Different Layers.

An MLP operates in two steps, training and testing. Training functions are used to minimize the difference between outputs of MLP and the desired values of the network. When the training state is done, neural weights will be frozen and new input data will be used to conclude the suitable output [44].

5.3.1 Training and Testing pattern of Neural Network

The training process uses a trial and error method to achieve the optimal behavior of the network. The fastest training function is trainlm, which is also the default training function for feed forward net. After achieving, the optimal training, the network tested for load prediction accuracy. The current method under investigation is the ANN tool using Neural Network. Figure 33 show the structure of the ANN that we adopted after trying different structures to all data, chosen randomly, with 22, 30, 40 and 50 neurons in the hidden layer as to describe the best structure according to the index
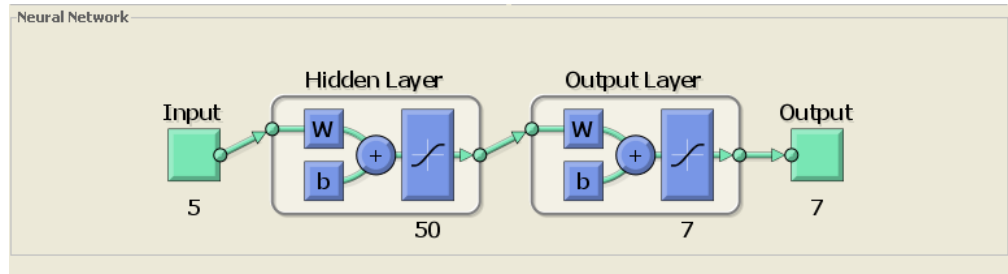
Figure 33 simulated artificial neural network

Figure 34 is a measure of how close a fitted line is to data points. The smaller the Mean Squared Error is the better and the closer to the fit is to the data. The MSE has the units squared of whatever is plotted on the vertical axis. Calculation result below showed for the selected Neural Network was the minimum and MSE was one of the criteria to choose 50 neurons for ANN.

Figure 34 result of the MSE %E

5.4 Program

Below is the program that we can generate from the selected neural network. This program can be used as programing file and deployed for different application. The program shows the steps of training, command, transfer function, number of layer and neuron, portioning of the data sets and the validating parameters.

val =

Neural Network object:

architecture:

   numInputs: 1

   numLayers: 2

  biasConnect: [1; 1]

  inputConnect: [1; 0]

  layerConnect: [0 0; 1 0]

 outputConnect: [0 1]

  numOutputs: 1 (read-only)

numInputDelays: 0 (read-only)

numLayerDelays: 0 (read-only)

subobject structures:

   inputs: {1x1 cell} of inputs

   layers: {2x1 cell} of layers

   outputs: {1x2 cell} containing 1 output

   biases: {2x1 cell} containing 2 biases

 inputWeights: {2x1 cell} containing 1 input weight

 layerWeights: {2x2 cell} containing 1 layer weight

functions:

   adaptFcn: 'trains'

  divideFcn: 'dividerand'

 gradientFcn: 'calcgrad'

   initFcn: 'initlay'

 performFcn: 'mse'

  plotFcns: {'plotperform','plottrainstate','plotconfusion','plotroc'}

  trainFcn: 'trainscg'

parameters:

adaptParam: .passes

divideParam: .trainRatio, .valRatio, .testRatio

gradientParam: (none)

initParam: (none)

performParam: (none)

trainParam: .show, .showWindow, .showCommandLine, .epochs,

.time, .goal, .max_fail, .min_grad,

.sigma, .lambda

weight and bias values:

IW: {2x1 cell} containing 1 input weight matrix

LW: {2x2 cell} containing 1 layer weight matrix

b: {2x1 cell} containing 2 bias vectors

other:

name: ''

userdata: (user information)

```matlab
1  function [output]=runsimneu
2
3  paramNameValStruct.SimulationMode = 'rapid';
4  paramNameValStruct.AbsTol         = '1e-5';
5  paramNameValStruct.SaveState      = 'on';
6  paramNameValStruct.StateSaveName  = 'xoutNew';
7  paramNameValStruct.SaveOutput     = 'on';
8  paramNameValStruct.OutputSaveName = 'youtNew';
9
10 sim('neutest',paramNameValStruct);
11 [output]=decide(simout);
12 end
13
14 function [output]=decide(simout)
15
16 input=simout.signals.values';
17
18 for i=1:size(input,2)
19     seeker=find(input(:,i)==max(input(:,i)));
20     switch seeker
21         case 1
```

```
22              output{i}='coffee maker';
23          case 2
24              output{i}='toaster';
25          case 3
26              output{i}='printer';
27      end
28 end
29
30 end
```

5.5 Result

After finishing the training, further paces considered to validate the network performance and define if any changes need to be made to the training process, the network architecture or the data sets or even the type of the neural network itself. After that we can start with training record to check the validation of the ANN.

The training record showed different variables during the training, like the value of the performance function, the magnitude of the gradient, etc.

Figure 35, as indicated by training best epoch, shows the iteration at which the validation performance reached a minimum was 66. The training continued for 6 more iteration before the training stopped. Figure 35 below shows the changes between the validation, Training and testing where the Main square Error was minimum.
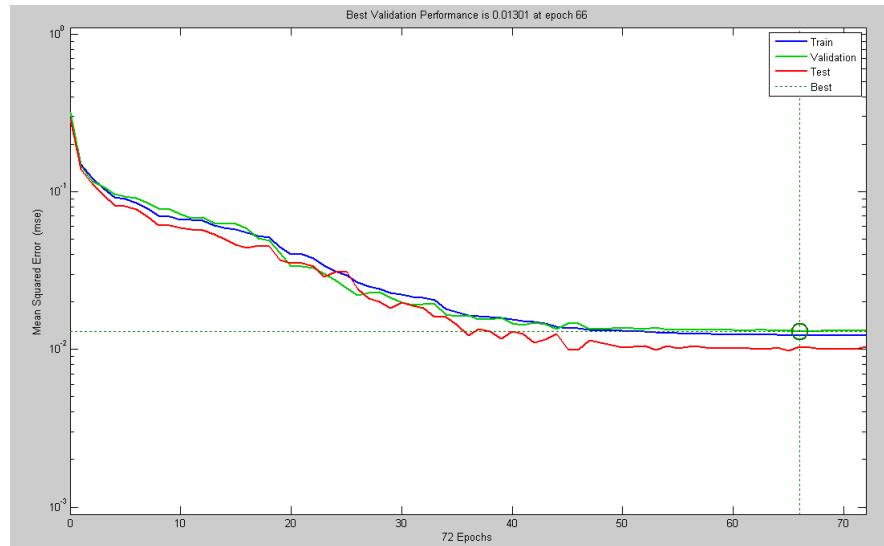
Figure 35 the Best Performance Validation

Figure 35 show no major problems with the training. The test and validation figures are very similar. The over fitting might happened If the test curve had increased significantly before the validation curve increased.

The next phase in validating the network is a regression plot, which shows the relationship between the outputs of the network and the targets. The best scenario for the training validation were the target would be exactly equal to the output but this is would be the optimal case which is not in our case here or in any practical practice..

The result is shown in figure 36. The main three axes signify the training, validation and testing data. The dashed line in each axis represents the faultless result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets. R value is an sign of the relationship between the targets and outputs. If R = 1, this mean that there is an exact linear relationship between outputs and targets. If R is around zero, then there is no linear relationship between targets and outputs.

For this research, the training data indicates a very good fit. The validation and test results show R values that greater than 0.92. The confusion plot is helpful in showing that certain data points have poor fits. Since we have very few poor fitting points there will be no need to investigate if it represents extrapolation. In other cases where many poor data point exist in the scattered plot, at that time other data should be composed to be used in the validation set. Figure 36 shows the regression analysis for the chosen network.
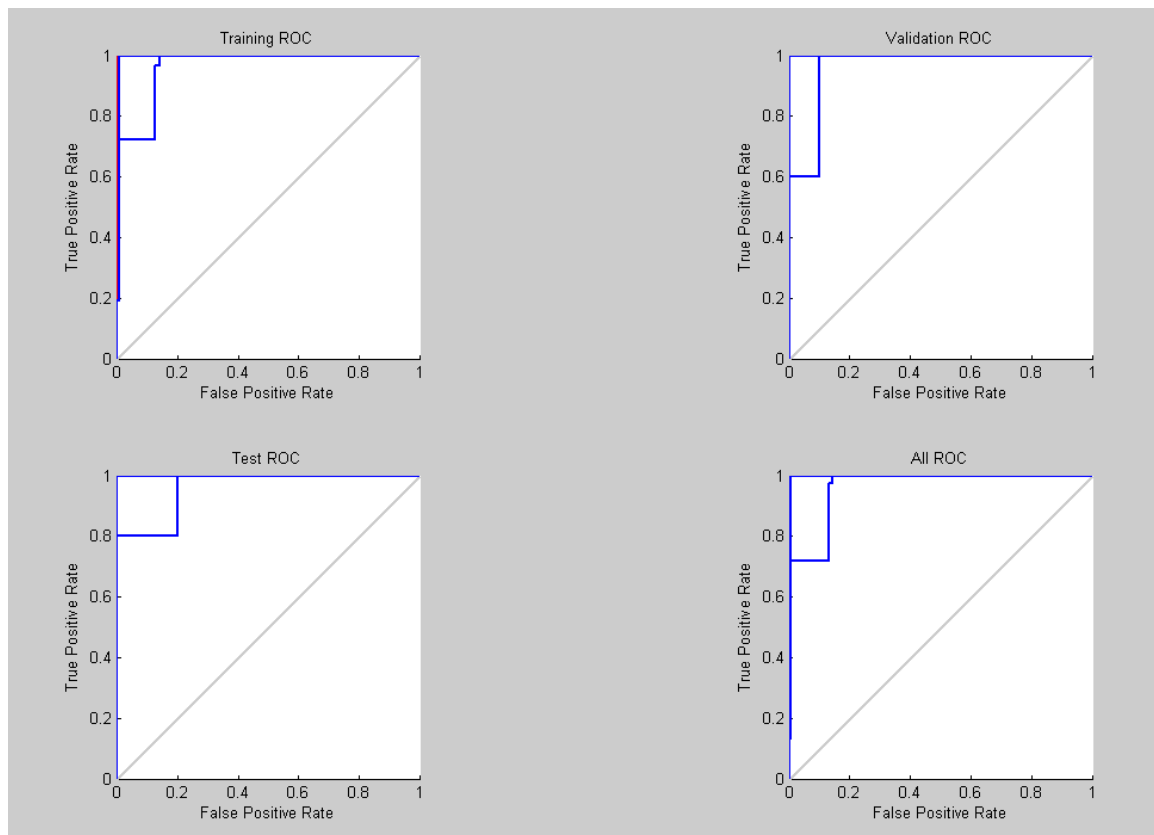
Figure 36 validating the network is a regression

Figure 37 shows the confusion that happened during the training.
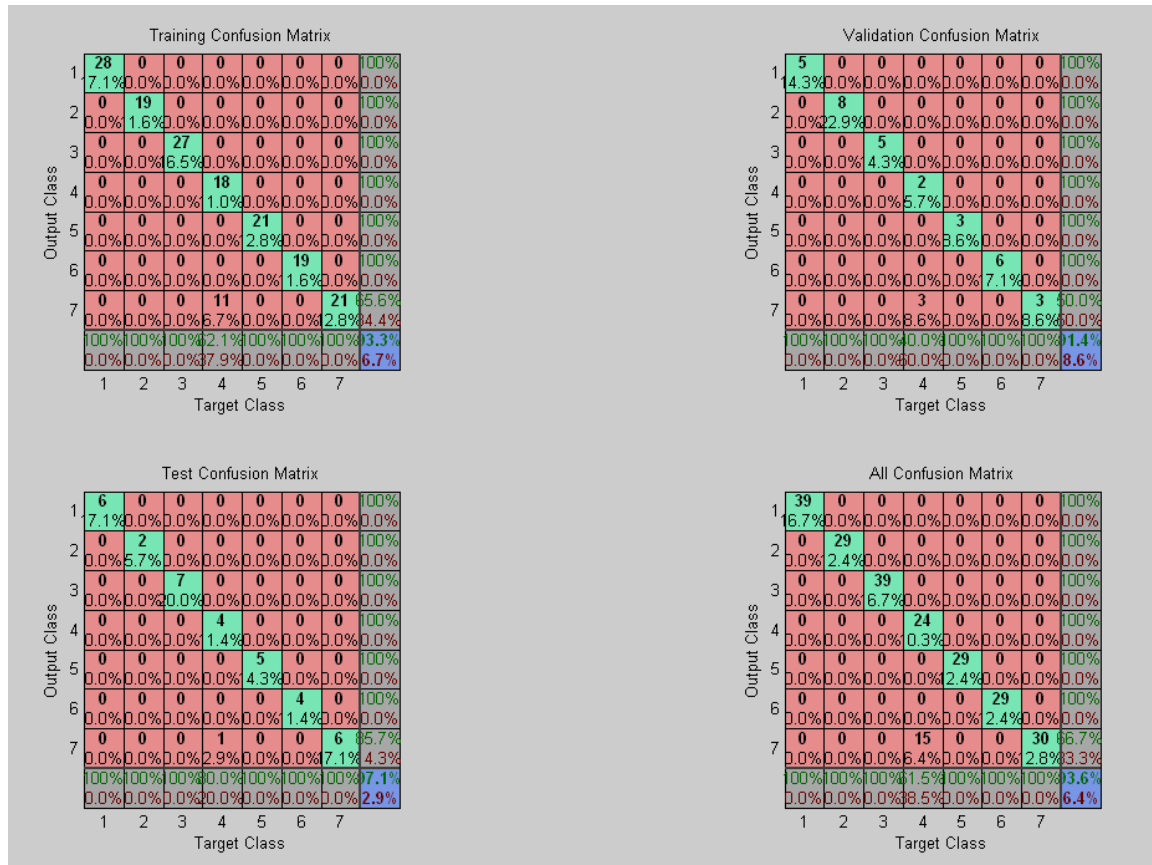
Figure 37 confusion Matrix

The result is rational because of the following considerations:

- Mean-square error is significantly small.

- Test set error and the validation has minimum error

- No significant over fitting has occurred by iteration 66 where the best validation performance happens.

Table 7 below shows the neural network structure

Table 7 MLP structure

MLP STRUCTURE	
Neural Network	MLP
Number of hidden layer	2
Number of hidden neuron	50
Activation function used in hidden layer	tan-sigmoid
Activation function used in output	lay

The results obtained from the approved trained neural network on new data for the pattern/ load identification are presented below in graphical form. Figure 37 Shows pattern recognition for a sample load using the Neural Network shown

Defining phantom loads is the main purpose of the load type identification. The difference in power consumption for specific loads will indicate whether the load is in the ON/OFF or Standby mode. With load location identified, the NILM will have additional information to accurately identify the type of load. Figure 38 shows how the ANN respond when certain class is being detected.
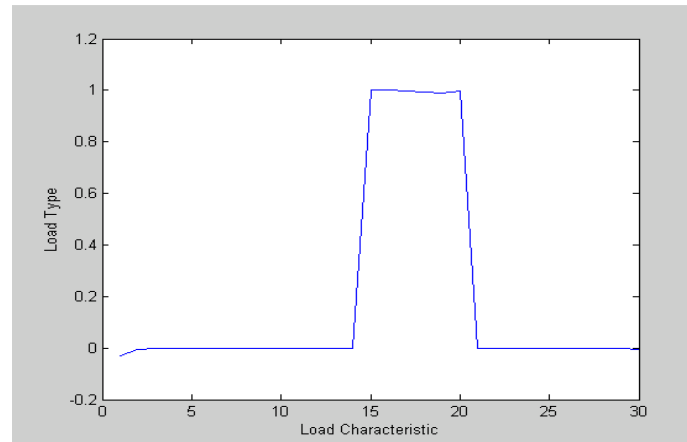
Figure 38 Load Type (Printer, Laptop...Etc.) vs. Load Characteristic (watt)

Overall, the above error values translate to an absolute mean error of 1.8% for the network. This represents a high degree of accuracy in the ability of neural networks to forecast electric load.

After validating the neural network, generate a Simulink for the neural network in order to apply new inputs for the model. The output of the neural network will be presented as a threshold of certainty where each class is represented by a value between 0 and 1. The decision maker will scan the output to select the highest combined rate. In other words, each case (1-7) ,which represent the loads, of the outputs (fan, fluorescent light, laptop, printer etc..) give a specific input between the value of 0 and 1. The generated test results given in Table 8 show that the neural network is effective. Input block was indicated to with name Testfinal where the sets of data applied to ANN. Pattern2 represent the out file for the ANN where all the result will be stored.
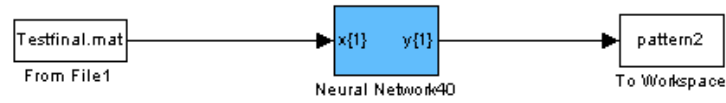
Figure 39 Simulink of the ANN

Table 8 Output (pattern2) of the ANN

index	Case1	Case2	Case3	Case4	Case5	Case6	Case7
1	0.999984	0.024717	0.02735	4.90E-08	0.009376	2.67E-07	2.07E-06
2	0.999927	0.049235	0.027691	5.70E-08	0.00798	2.77E-07	1.73E-06
3	1.70E-05	0.965924	0.000727	0.019681	0.001789	2.04E-08	0.001382
4	1.71E-05	0.966181	0.00073	0.019586	0.00179	2.04E-08	0.001373
-	-	-	-	-	-	-	-
n	1.70E-05	0.965981	0.000726	0.01967	0.001784	2.05E-08	0.001381

Table 9 represents the measured character of the differing loads. These can in turn be applied as inputs in the previously designed model to get the classes or load represented by the previous features.

Table 9 Prospective Input and Output for each load

Input	**Current [A]**	**0.34**	**0.34**	**0.3**	**0.3**	**0.35**	**0.356**	**0.19**	**0.19**
	Power [W]	20.88	20.8	10.3	10.3	36.84	37.3	11.78	11.76
	Apparent Power [VA]	40.06	40.1	35.48	35.76	42.4	42.92	23.6	23.58
	Power Factor	0.52	0.513333	0.288	0.286	0.86	0.868	0.498	0.496
	Current THD	7.28	6.633333	1.999	1.9974	0.5	0.5014	0.236	0.2358
Result/Output		fan		Fluorescent Light		laptop		Printer	

5.6 Conclusion

In this chapter, load identification model was built and tested. We were able to simulate the model with real measured data and the output of this model represents the type and the state of the load. So in general smart home technologies could be introduced to conserve energy model in the built environment. The load identification methods using NILM technologies can detect and evaluate energy waste whether that waste is the result of phantom loads by detecting the state of that load. The simulation results indicate that a possible sensor technology can be developed that can upgrade the existing electrical systems. This will reduce cost of upgrading and increase energy efficiency in the built environment.

6 Chapter 6

Load forecasting using ANN

6.1 Introduction

Electric load forecasting is the procedure used to forecast future electric load, given past load and weather information and the electric load forecasting will be the output.[In conventional methods, the models are designed based on the relationship between load power and plenty of factors that could influence the output which is the power load. The previous method has the benefit that we can forecast load power with a modest, simple not complicated prediction model. Nevertheless, since load power and the factors that influencing him is nonlinear, it is difficult to identify it's the relation which is nonlinearity by using conventional methods. Lately, quite a lot of methods based on similarity have been reported for load forecasting [45-46].

These methods are based on similarity forecast future power load curve by using data of the day being alike to weather condition of the targeted day or forecast day. These kind of methods have the benefit of dealing with not only the nonlinear part of load power, but also with other factors like weekend and special days, outdoor air temperature, relative humidity etc. In prediction methods grounded on similarity, the power load on several selected similar days is averaged to increase the accuracy of load forecasting. Still, if there is a considerable error between similar days on a forecast day and that on the load power, we are not able to assume that the good prediction accuracy for averaging

the load power on similar days. Numerous approaches have been studied using neural networks for load forecasting [47].

Artificial Neural network are appropriate for load forecasting because of their approximation ability for nonlinear mapping and generalization. Most of Artificial neural network based methods, reported so far uses weather information, load power. In general Load forecasting can be divided into three major categories:

1. Long-term forecasting for electric loads, used to predict the future supply for the electrical companies and utilities so they can plan for maintenance expanding and applying strategies for managements.

2. Medium-term forecasting for electric loads, used for the determination of setting up fuel supplies and maintenance schedules.

3. Short-term forecasting for electric loads used to supply necessary information for the system management of day- to-day operations and unit commitment.

Short-term electric load forecasting is used to deliver utility company administration with future data that shows the electric load demand in order to assist them in running more economical and trustworthy day-to-day actions. The power load during the year shadowed the same the daily and weekly periods of electric load, which signifies the daily and weekly cycles of human activities and behavior patterns, with some recurring and random changes.

The significance of space cooling and HVAC system on the electric load is very obvious during the summertime and very cold wintertime. When the temperature

increases, obviously the demand for electricity increases. On the other hand, during the wintertime the reverse relationship between temperature and electric demand exists because of the need for space heating. When the temperature reductions, for sure the load demand for electricity rises. [49]

Lately, industry combined with academic research more and more has been focused on regulatory energy spending and refining the energy efficiency of the built environment. In the past, different application-specific methods have been investigated to understand, among other things, how loads would affect power consumption, efficiency, power quality, reactive power compensation, and system preparation. The purpose of this chapter is to understand the characteristic–load consumption. In this research, a neural network method will be employed to predict electrical loads for typical commercial areas. The neural network will provide a Kw Model. This model employs past trained data and serves as a baseline using past trained data to reflect an increase or decrease in energy use. The real-time monitored Kw could be compared to the model Kw (baseline).

The difference between the two should reflect the potential energy savings. In other words, the Model Kw will make it easier for engineers to define acceptable performance thresholds. The following section will introduce NN for Electrical loads in order to enable researchers to estimate the energy consumption for a specific locale. Non-intrusive load monitoring (NILM) is considered to be one of the earliest approaches to nonintrusive monitoring. Engineers could determine the operating schedule of electrical loads in the target system from measurements made at a centralized location [34-39]. The following describes the different steps needed in the load prediction process:

- Data acquisition or preprocessing data

- Data processing

- Statistical analysis

- Neural Network Design (Model Kw Design)

- Addressing energy exploitation using load location Identification and the created model.

6.2 Approach

A broad range of factors that affect the system's load level such as trend effects, cyclic-time effects, and weather effects, random effects like human activities, load management, comfort issue and storms. Thus the load demand is lively in nature with temporal, seasonal and annual variations. In my project I have developed a system that predicted load demand for the one and a half months.

As inputs the past year reading of a building. Seattle was chosen and used the daily temperature, humidity. The energy demand (in kilowatts) was sampled and stored for every fifteen minutes. The weather parameters were provided by the local weather channel. Readings were collected every minute. The outside air temperature (OAT) and outside air relative humidity (ORAH) are two parameters employed in training our model. Hours and holidays were included as well as input parameters. Normalization was applied for all of the previous parameters the inputs were fed into our Artificial Neural Network (ANN) and after sufficient training were used to predict the load demand for the next year. The inputs given are:

2. Hourly load demand.

3. Day of the week.

4. Outside air temperature (OAT).

5. Outside relative Humidity (ORAH).

6. Holidays.

The target for this model was the demand Kw and the how that is correlated to the previous inputs.
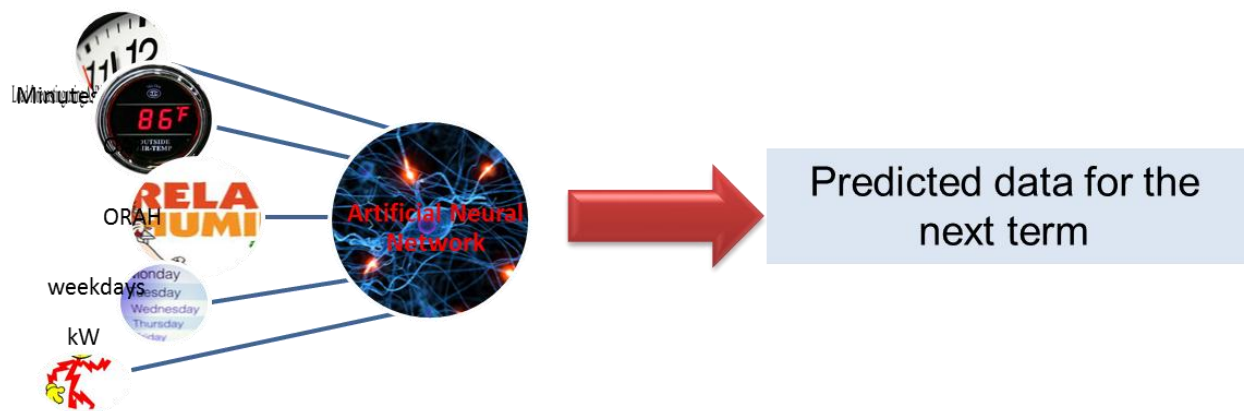
Figure 40 general overview for the load forecasting model using ANN

And the output obtained was the predicted load demand for the 5 weeks.

6.3 Preprocessing

The data that we used for training and testing the neural network were obtained from the client website for the period of one year January 2010 to January 2011. Due to

wrong measurements and other human errors, some out-of- range and out fitted values were observed in the historical load data as obtained from the building web page. Corrections were made to such outlier values by replacing them with the average of both the preceding and succeeding values in the series. After that we used the program MATLAB where we implemented the previous formatted data and training, testing and validating data sets. After the sequences had been corrected, the data were normalized so that their values would be between the values -1 and +1. This was achieved by using the „prestd‟ function in MATLAB. Table 10 below shows sample of data were is data is represented in the polished well correlated form.

Table 10 sample of the stored data

Minutes	Holidays-Weekend	OAT	ORAH	Load(KW)
1395	1	37.6	72	4176
1410	1	37.5	73	4185.6
1425	1	37.8	73	4195.2
0	0	37.6	73	4185.6
----	----	----	----	----
----	----	----	----	----
1410	0	44.3	98	3993.6
1425	0	44.5	98	3993.6

6.4 ANN Model

The neural network employed in this research is the Multi-Layer Feed Forward Network, also commonly known as the Multi-Layer Perception (MLP) Network. It consists of one or more hidden layers with computational nodes that are correspondingly called hidden neurons. The purpose of the hidden neuron is to increase the computation between the input and output core layers in some useful manner. The input signal is applied to the neurons in the second layer while the output signal of the second layer is used as an input for the third layer, and so on for the rest of the network. Figure 40 represents the MLP network topology.
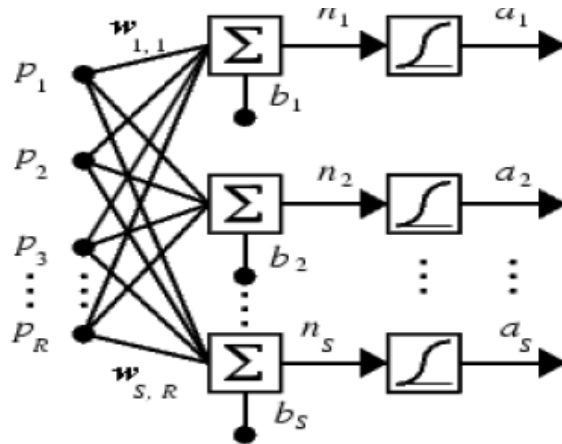
Figure 41 Topology of an MLP
Topology of an MLP where R is the Number of Elements in the Input Vector, S is the Number of Neurons in the Layers, and Wij are the Weighting Factors between the Different Layers
.

An MLP operates in two steps: Training and testing. Training functions minimize the difference between the MLP outputs and the desired values of the network. At the

completion of the training state the neural weights will be frozen and new input data used to create a suitable output [16].

Using the trial and error method, the training process achieved the optimal behavior of the network. The fastest training function is trainlm, which is also the default training function of the feed forward network. In future research, the investigators plan to test other methods such as the quasi-Newton method and trainbfg. Finally, 2 hidden layers with feed forward activity and 20 neurns were chosen. The basic traning data included a sample size of 3100. After achieving the optimal training the network will be tested for load prediction accuracy. For the purpose of testing, a 15% or 20% deviation out of the load samples was applied to the data content table for a typical residence.

6.5 Program

Below is the program that we can generate from the selected neural network. This program can be used as programing file and deployed for different application. The program shows the steps of training, command, transfer function, number of layer and neuron, portioning of the data sets and the validating parameters.

val =

Neural Network object:

architecture:
numInputs: 1
numLayers: 2

biasConnect: [1; 1]

inputConnect: [1; 0]

layerConnect: [0 0; 1 0]

outputConnect: [0 1]

numOutputs: 1 (read-only)

numInputDelays: 0 (read-only)

numLayerDelays: 0 (read-only)

subobject structures:

inputs: {1x1 cell} of inputs

layers: {2x1 cell} of layers

outputs: {1x2 cell} containing 1 output

biases: {2x1 cell} containing 2 biases

inputWeights: {2x1 cell} containing 1 input weight

layerWeights: {2x2 cell} containing 1 layer weight

functions:

adaptFcn: 'trains'

divideFcn: 'dividerand'

gradientFcn: 'gdefaults'

initFcn: 'initlay'

performFcn: 'mse'

plotFcns: {'plotperform','plottrainstate','plotregression'}

trainFcn: 'trainlm'

parameters:

adaptParam: .passes

divideParam: .trainRatio, .valRatio, .testRatio

gradientParam: (none)

initParam: (none)

performParam: (none)

trainParam: .show, .showWindow, .showCommandLine, .epochs,

.time, .goal, .max_fail, .mem_reduc,

.min_grad, .mu, .mu_dec, .mu_inc,

.mu_max

weight and bias values:

IW: {2x1 cell} containing 1 input weight matrix

LW: {2x2 cell} containing 1 layer weight matrix

b: {2x1 cell} containing 2 bias vectors

other:

name: ''

userdata: (user information)

```matlab
function net = create_fit_net(inputs,targets)
%CREATE_FIT_NET Creates and trains a fitting neural network.
%
%  NET = CREATE_FIT_NET(INPUTS,TARGETS) takes these arguments:
%    INPUTS - RxQ matrix of Q R-element input samples
%    TARGETS - SxQ matrix of Q S-element associated target samples
%  arranged as columns, and returns these results:
%    NET - The trained neural network
%
%  For example, to solve the Simple Fit dataset problem with this
function:
%
%    load simplefit_dataset
%    net = create_fit_net(simplefitInputs,simplefitTargets);
%    simplefitOutputs = sim(net,simplefitInputs);
%
%  To reproduce the results you obtained in NFTOOL:
%
%    net = create_fit_net(Input',Target');

% Create Network
numHiddenNeurons = 20;  % Adjust as desired
net = newfit(inputs,targets,numHiddenNeurons);
```

```
net.divideParam.trainRatio = 70/100;  % Adjust as desired
net.divideParam.valRatio = 15/100;  % Adjust as desired
net.divideParam.testRatio = 15/100;  % Adjust as desired

% Train and Apply Network
[net,tr] = train(net,inputs,targets);
outputs = sim(net,inputs);

% Plot
plotperf(tr)
plotfit(net,inputs,targets)
plotregression(targets,outputs)
```

6.6 Result

After finishing the training, further paces considered to validate the network performance and define if any changes need to be made to the training process, the network architecture or the data sets or even the type of the neural network itself. After that we can start with training record to check the validation of the ANN.

The training record showed different variables during the training, like the value of the performance function, the magnitude of the gradient, etc.

Figure 44, as indicated by training best epoch, shows the iteration at which the validation performance reached a minimum was 126. The training continued for 6 more iteration before the training stopped. Figure 35 below shows the changes between the validation, Training and testing where the Main square Error was minimum.
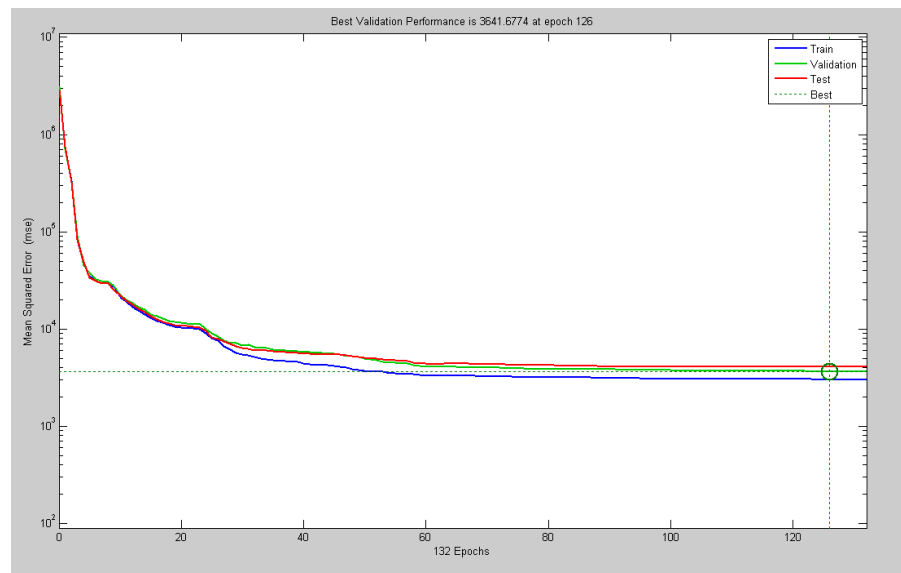
Figure 42 Best Performance Validation

Figure 44 show no major problems with the training. The test and validation figures are very similar. The over fitting might happened If the test curve had increased significantly before the validation curve increased.

The next phase in validating the network is a regression plot, which shows the relationship between the outputs of the network and the targets. The best scenario for the training validation were the target would be exactly equal to the output but this is would be the optimal case which is not in our case here or in any practical practice..

The result is shown in figure 36. The main three axes signify the training, validation and testing data. The dashed line in each axis represents the faultless result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets. R value is an sign of the relationship between the targets and outputs.

If R = 1, this mean that there is an exact linear relationship between outputs and targets. If R is around zero, then there is no linear relationship between targets and outputs.

For this research, the training data indicates a very good fit. The validation and test results show R values that greater than 0.99. The confusion plot is helpful in showing that certain data points have poor fits. For example, there is a data point in the validation set whose network output is close to 4200, while the corresponding target value is about 4600. Since we have very few poor fitting points there will be no need to investigate if it represents extrapolation. In other cases where many poor data point exist in the scattered plot, at that time other data should be composed to be used in the validation set. Figure 45 shows the regression analysis for the chosen network.

.
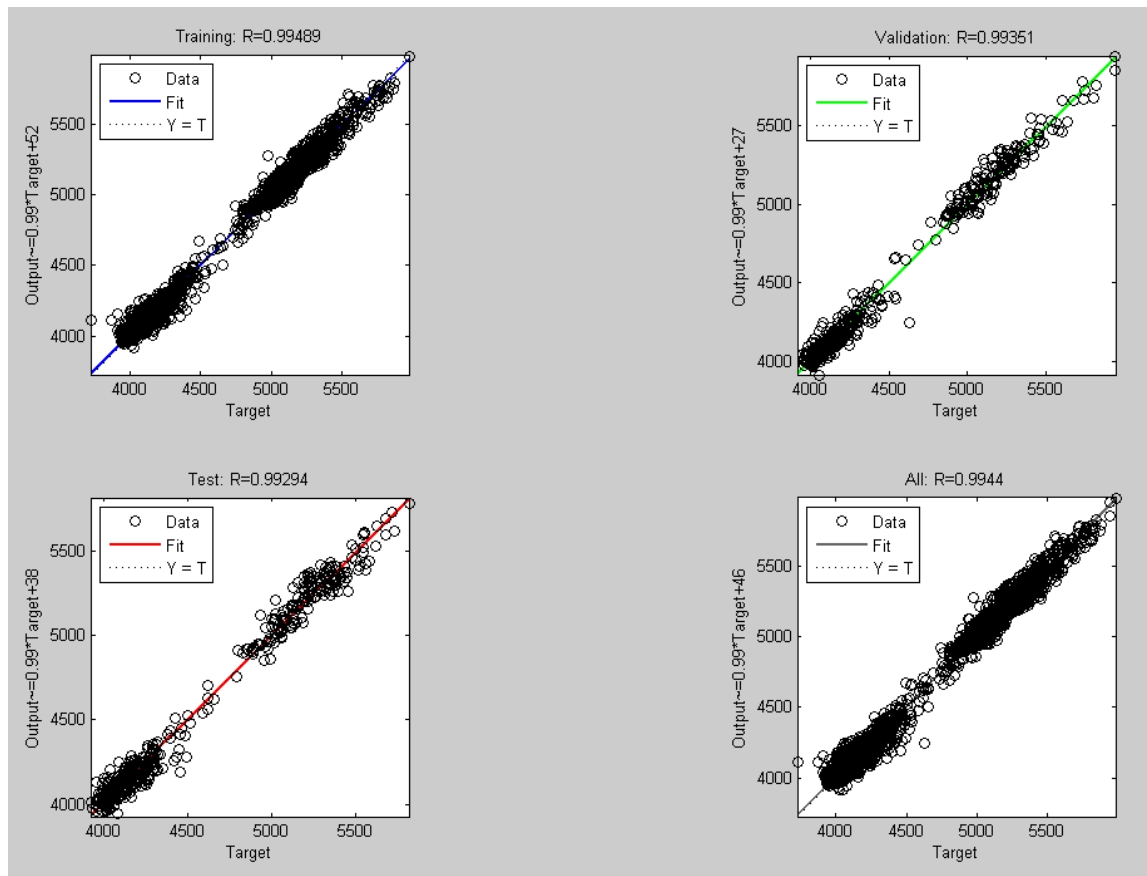
Figure 43 validating the network is a regression for load forecasting NN

The result is rational because of the following considerations:

- Mean-square error is significantly small.

- Test set error and the validation has minimum error

- No significant over fitting has occurred by iteration 17 where the best validation performance happens.

Table 11 shows the neural network structure

Table 11 MLP structure for load forecasting

MLP STRUCTURE	
Neural Network	MLP
Number of hidden layer	1
Number of hidden neuron	17
Activation function used in hidden layer	tan-sigmoid
Activation function used in output	lay

The results obtained from the approved trained neural network on new data for the next 6 are presented below in graphical form. The graph shows a plot of both the predicted and actual load in kW values against the hour of the day. The absolute mean error AME (%) between the „predicted and „actual loads for the predicted period has been calculated and presented in the table. Figure 46 shows a comparison between the measured and the predicted data.
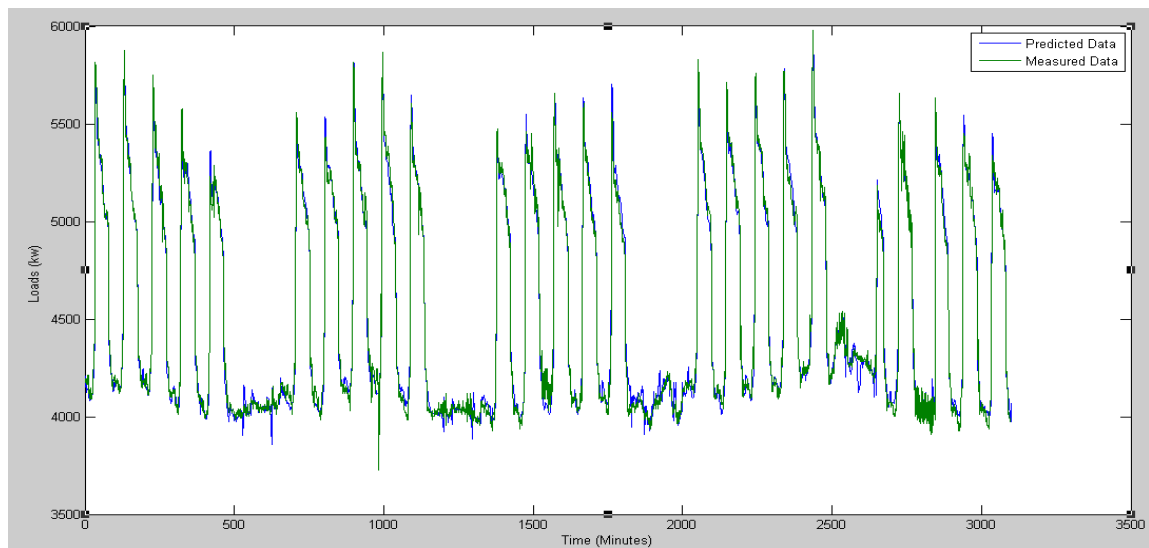

Figure 44 Deviation of Predicted and Measured Energy data

Overall, the above error values translate to an absolute mean error of 2.8% for the network. This represents a high degree of accuracy in the ability of neural networks to forecast electric load.

In future studies, it is hoped that the proposed methods will lower energy spending by 15-25% spanning the lifecycle of most home electronics. The load location identification method can be combined with the Model kw to both detect the energy saving potential and locate the source of the savings. This can be accomplished using only minimal additional hardware and software that can be used in many applications. Further, this saved energy could be used to charge electric vehicles since they could be connected to the house load without the risk of overload.

6.7 Conclusion

In this chapter, load forecasting model was built and tested. We were able to simulate the model with real measured data and the output of this model represents the prediction for the load for the next five months. So in general smart home technologies could be introduced to conserve energy model in the built environment. The load forecasting method can detect and evaluate energy waste whether that waste is the result of mismanagement of the load by setting benchmark for the energy usage. This bench

mark can be implemented in control strategies where it becomes the reference for all the comparison that we might need to do in order to take control action.

The simulation results indicate that it's possible to implement this model in real time manner process. This will give pretty instantaneous feedback for the control system or for the facilities owners. This in turn will reduce cost of upgrading and increase energy efficiency in the built environment

7 Chapter 7

Summary and Future Work

7.1 Summary

The main objective of this thesis is to save energy in the built environment. To achieve this objective, the thesis proposed new methods that can identify the load type and state and short term load for casting. The proposed systems has been validated and Simulink the models using real captured data. The results of this study indicate that the proposed systems improve the efficiency and could increase the life span of the different monitored applications. The two main approaches will contend each other to achieve the goal of saving energy, where load identification would tackle the phantom loads, load forecasting can be used to do load shifting and save the wasted energy on the mismanagement side of loads.

Load identification model will give enormous contribution in terms of feedback (where/when the energy being used or wasted). The output of this model could be implemented on any control system or used as feedback for the consumer "change the consumer behavior".

Load forecasting model can be used as bench mark for any possible energy management strategies.

Previous work can be introduced to smart home technologies to conserve energy in the built environment. And derive the potential energy savings that can be for example used to charge electric vehicles.

7.2 Future work

This research had two main approaches, each one of them can be taken into another level and be part of big picture which is energy saving and management in the smart built environment.

Figure represent the future work that might apply in general on both approaches. The two main additions to current research is the real time streaming data and implement control strategies using the output of the model.
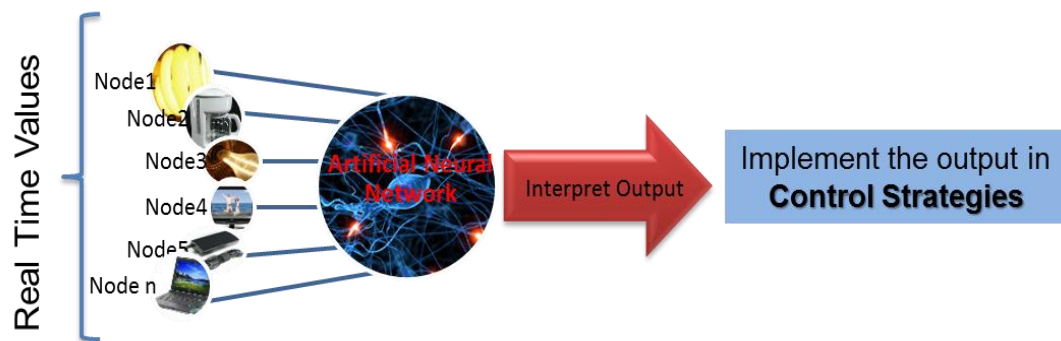
Figure 45 future works on the models

Starting with load identification approach:

This research didn't achieve online model of neural network for Load identification. In fact online neural network will improve ANN result and make it more accurate which

in total will reflect better recognition for the load patterns. Another potential advantage for the online pattern recognition is the real time monitoring for the load. This feedback could be a powerful tool in case we know how to use it. For example previous research showed that the householders were welling to save energy if they knew how and when. So real time load classification will show the consumers what load is now on, off or standby, after this they can physically switch on or off the load. This is educational approach where the main goal is improve consumer behavior. Now for more sophisticated approach. Load identification model might be implemented the output, of our pattern recognition, into control strategies. For example in the new smart home environment automatic and controlled switches is available so whenever we detect unoccupied space and phantom load, the circuit that supply that space could be switched off and reset again to the ON mode when the space became occupied again.

As for Load forecasting model also could take similar future work in terms of online ANN implementation, where online mode will enhance and improve the accuracy of the predicted data. An energy prediction tool can be improved, when there is more energy data is available which in turn drive us to look around for more correlated value for the model target. But we should keep in mind the availability of each variable we use to train the model because need the whole set of the variable when we put the model in action. Another sophisticated future work is to use this model as a part of web control. In other word to use predicted data as an input for a control strategies. The short term predicted kW and other factor like comfort threshold, the outside temperature, and occupant status would be the factors to turn on the control strategies or not. The control strategies is like demand limiting where we would like to cut the kW peak in order to avoid the high rate

price on the high demand period. Another control strategies might put on is applying the predicted weather to get the predicted values for the next day. Predicted weather value could be obtained through website or weather stations. The all over all idea that the model will give us insight glance for the future demand so we can be anticipate the suitable management protocols

References

[1] U.S. Green Building Council, "Green Building Facts." U.S. Green Building Council. www.usgbc.org/ShowFile.aspx?DocumentID=5961, Accessed 7-21-2010.

[2] Chevron's energy efficiency and conservation, http://www.chevron.com/globalissues/energyefficiencyconservation/, Accessed 7-21-2010.

[3] Wisconsin Focus on Energy,"what is "phantom load" and how does it cost you money?" 2011.

[4] Ed Carroll, Eric Hatton, Mark Brown, "Residential Energy Use Behavior change Pilot," CMFS   project code B21383, April 20, 2009 Franklin Energy.

[5] Karen Ehrhardt-Martinez, Kat A. Donnelly, & John A. "Skip" Laitner, "Advanced metering Initiatives and Residential Feedback Programs: A Meta-Review of Household Electricity-Saving Opportunities," American Council for an Energy-Efficient Economy, report Number E105, June.

[6] Independent Statistic & Analysis, U.S. Energy Information Adminstration(eia), http://www.eia.gov/electricity/.

[7] RESIDENTIAL ENERGY CONSUMPTION SURVEY (RECS),eia, http://205.254.135.24/consumption/residential/index.cfm .

[8] Pacific Northwest National Laboratory. 1997. An Analysis of Buildings-Related Energy Use in Manufacturing,

PNNL-11499, p. 4.2, table 4.1 and p. 4.4, table 4.3. Pacific Northwest National Laboratory, Richland, WA.

[9] Office of Energy Efficiency and Renewable Energy, 2002, op cit, p. 1-4, 1-7, and 4-1, tables 1.2.1,

1.3.1, and 4.1.2.

[10] Williams, Eric D., & Matthews, H. Scott. (2007). Scoping the Potential of Monitoring and Control .Technologies to Reduce Energy use in Homes. IEEE.

[11] What exactly is standby power and why is it necessary?. Lawrence Berkley National Laboratory. 2009 http://standby.lbl.gov/faq.html#what.

[12] Are Phantom Loads Haunting Your Energy Bill, http://pages.uoregon.edu/hof/W09HOF/19Phantom_ppr.pdf

[13] How much Energy are we using potenial of residential.

[14] H.Hajek, Neural Networks, http://www.cs.unp.ac.za/notes/NeuralNetworks2005.pdf.

[15] Freeman J.A., Skapura D.M.: Neural networks - Algorithms, applications, and programming *techniques*, Addison-Wesley, Reading, MA 1991

[16] Hertz J., Krogh A., Palmer R.G.: Introduction to the theory of neural computation Addison-*Wesley*, Redwood City, CA 1991

[17] Zurada J.M.: *Introduction to artificial neural systems*, West Publishing Company, St. Paul 1992

[18] Haykin S.: Neural networks – A comprehensive Foundation (2nd ed.), Prentice Hall, Upper Saddle River, 1999

[19] Principe J.C., Euliano N.R., Lefebvre W.C.: *Neural and adaptive systems – Fundamentals through simulations*, Wiley, New York, 2000

[20] Duda R.O., Hart P.E., Stork D.G.: Pattern classification, Wiley, New York, 2001.

[21] "Soft Computing and Intelligent Systems - Theory and Application", by Naresh K. Sinha and Madan M. Gupta (2000), Academic Press, Chapter 1-25, page 1-625.

[22] "Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence" by J. S. R. Jang, C. T. Sun, and E. Mizutani, (1996), Prentice Hall, Chapter 1-15, page 1-607.

[23] "Soft Computing : Integrating Evolutionary, Neural, and Fuzzy Systems", by Tettamanzi, Andrea, Tomassini, and Marco. (2001), Springer, Chapter 1-9, page 1-323.

[24] "Neural Network Design", by Martin T. Hagan, Howard B. Demuth and Mark Hudson Beale, (1996) , PWS Publ. Company, Chapter 1-19, page 1-1 to 19-14.

[25] "Neural Network Toolbox", Mathworks, Matlab, http://www.mathworks.com/help/toolbox/nnet/.

[26] "Sieci neuronowe", Ryszard Tadeusiewcz, Karakow 1992.

[27] Data Flow Design for the Backpropagation Algorithm Cheng- Yuan Liou∗ and Yen-Ting Kuo Dept. of Computer Science and Information Engineering, National Taiwan University Supported by National Science Council under Project NSC 90-2213-E-002-092.

[28] Nelson, M. M. and Illingworth. (1991), "A Practical Guide to Neural Nets". Addison Wesley, Reading, MA

[29] Masters, T. (1993), "Practical Neural Network Recipes in C++". Academic Press, New York.

[30] R.P. Lippman, An introduction to computing with neural nets, IEEE ASSP Mag. (April 1987) 4-22

[31] Baum, E. B. and Haussler, D. (1989), "What size net gives valid generalization?". Neural Computation, 6, pp. 151-160.

[32] Klimasauskas, C. C. (1993), "Applying Neural Networks". In: R. R. Trippi and E. Turban, editors, "Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real World Performance", pp. 64-65. Probus, Chicago.

[33] Ersoy, O. (1990), "Tutorial at Hawaii International Conference on Systems Sciences". January 1990, Hawaii.

[34] Shaw, S., S. Leeb, L. Norford and R. Cox, " Nonintrusive load monitoring and diagnostics in power systems," IEEE Trans. On Instrumentation and Measurement, Vol 57, issue 7, 2008, pp 1445-1454

[35] G. w. Hart, "Nonintrusive appliance load monitoring," Proc. Of the IEEE, Vol. 80, pp. 1870-1891. Dec, 1992.

[36] F. Sultanem, "Using appliance signatures for monitoring residential loads at meter panel level," IEEE Transactions on Power Delivery, Vol. 6, pp. 1380-1385, 1991.

[37] C. Laughman, K Lee, R. Cox, S. shaw, S. Leeb, L. Norford and P. Armstrong, " Power Signature analysis, "Power and Energy Magazine, IEEE, Vol. 1, pp. 56-63, 2003

[38] W K Lee, G S K Fung, H Y Lam, F H Y Chan, M. Lucente, " Exploration on Load signatures", International conference on electrical Engineering (ICEE), July 2004, Japan.

[39] K. H. Ting, Mark Lucente, George S. K. Fung, W. K.. Lee and S. Y. R. Hui, "A taxonomy of Load Signatures for Single-Phase electric Appliances," IEEE PESC, 12-18 June 2005, Brazil .

[40] Load signature study-part 1: basic concept, structure, and Methodology   Jian Liang, Simon K. K, Member, IEEE, Gail Kendall, and John W. M. cheng, Member, IEEE.

[41] J. Duan et al. "Neural Network Approach for Estimating of Load comostion", 2004.

[42] Load signature Study-Part I: Basic Concept, Structure, and Methodology " Jian Liang, Simon K.k. Ng, Member, IEEE, Gail Kendall, and John W.M.Cheng", Member,IEEE

[43] Active power residential non-intrusive appliance load monitoring system "Arend j. bijker, xiaohua xia", senior member, IEEE, and jiangfeng zhang

[44] L.Fausett, Fundamentals of Neural Networks, prentice-Hall Inc.,1994.

[45] T. Senjyu, S. Higa, T. Yue-Jin, and K. Uezato, "Future load curve shaping based on similarity using Fuzzy logic approach," in Proc. Int. Power Eng. Conf (IPEC), vol. II, 1997, pp. 483–488.

[46] S. Rahman and G. Shrestha, "A priory vector based technique for load forecasting," IEEE Trans. Power Syst., vol. 6, pp. 1459–1464, Nov. 1993.

[47] A. S. AlFuhaid, M. A. El-Sayed, and M. S. Mahmoud, "Cascaded artificial neural networks for short-term load forecasting," IEEE Trans. Power Syst., vol. 12, pp. 1524–1529, Nov. 199

[48] Dr. John A. Bullinaria, "Introduction to Neural Networks - 2nd Year UG, MSc in Computer Science: Lecture Series".
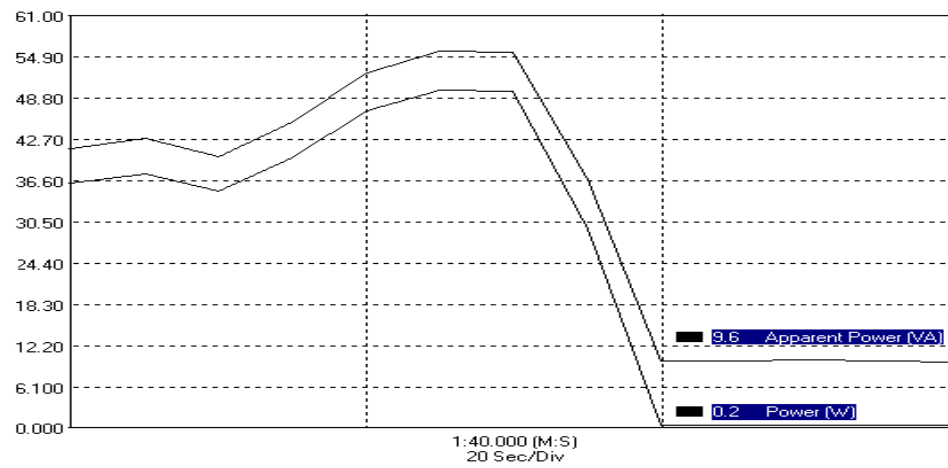
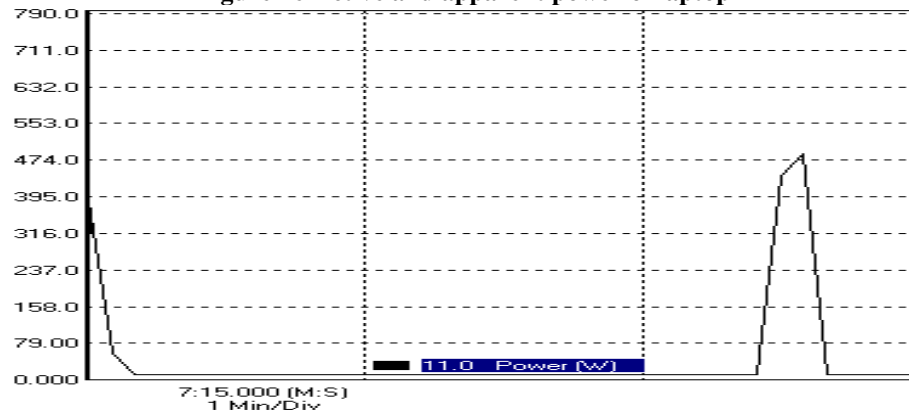Appendix

Figure 46 Active and apparent power of laptop

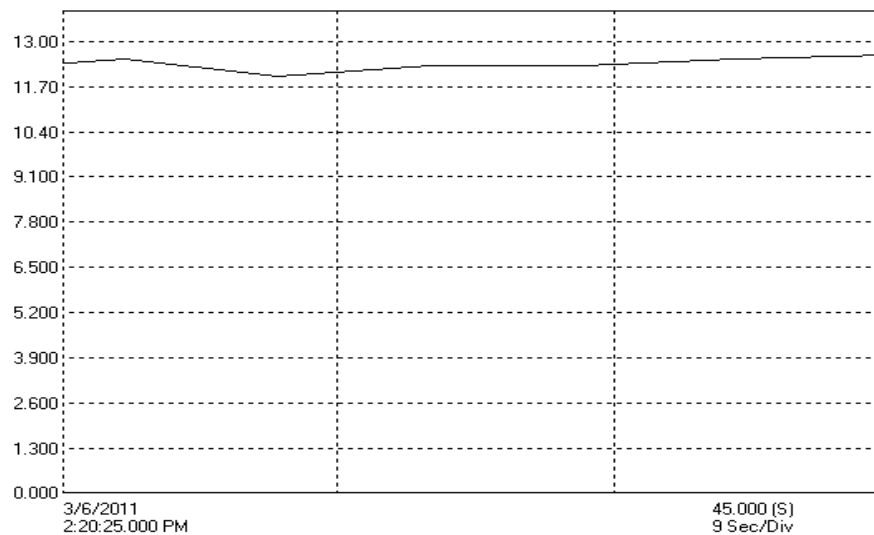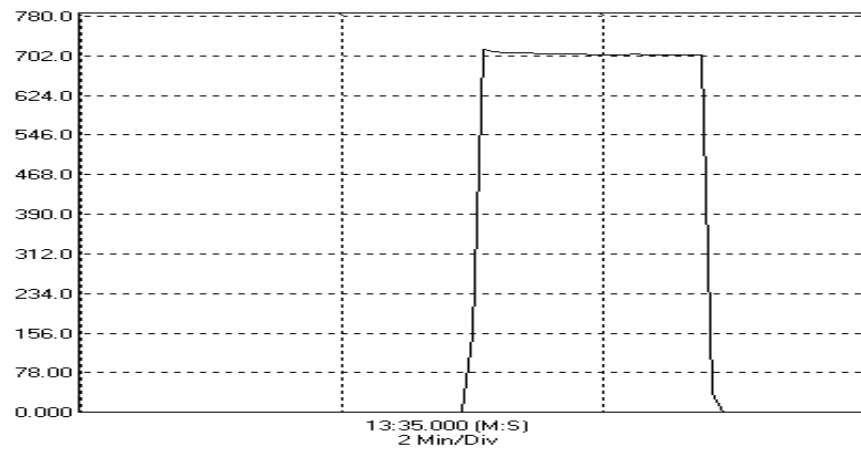**Figure 47 Active power of Painter**

Figure 48 Active power of Lighting Bulb

Figure 49 active power for Toaster