Energy Prediction Based on Resident's Activity

Chao Chen
Washington State University
Pullman, WA 99164
USA
cchen@eecs.wsu.edu

Barnan Das
Washington State University
Pullman, WA 99164
USA
barnandas@wsu.edu

Diane J. Cook
Washington State University
Pullman, WA 99164
USA
cook@eecs.wsu.edu

ABSTRACT

In smart home environment research, little attention has been given to monitoring, analyzing, and predicting energy usage, despite the fact that electricity consumption in homes has grown dramatically in the last few decades. We envision that a potential application of this smart environment technology is predicting the energy would be used to support specific daily activities. The purpose of this paper is thus to validate our hypothesis that energy usage can be predicted based on sensor data that can be collected and generated by the residents in a smart home environment, including recognized activities, resident movement in the space, and frequency of classes of sensor. In this paper, we extract useful features from sensor data collected in a smart home environment and utilize several machine learning algorithms to predict energy usage given these features. To validate these algorithms, we use real sensor data collected in our CASAS smart apartment testbed. We also compare the performance between different learning algorithms and analyze the prediction results for two different experiments performed in the smart home.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications – *data mining*; I.2.6 [**Artificial Intelligent**]: Learning – *knowledge acquisition*; H.4.m [**Information Systems**]: Information system Application – *Miscellaneous.*

General Terms

Algorithms, Performance, Experimentation, Human Factors.

Keywords

Energy Prediction, Smart Environments, Machine Learning.

1. INTRODUCTION

Recently, smart home environments have become a very popular topic, due to a convergence of technologies in machine learning and data mining as well as the development of robust sensors and actuators. In this research, attention has been directed toward the area of health monitoring and activity recognition. Georgia Tech Aware Home [2] identifies people based on the pressure sensors embedded into the smart floor in strategic locations. This sensor system can be used for tracking inhabitant and identifying user's location. The Neural Network House [3] designs an ACHE system, which provides an Adaptive Control of Home Environment, in which the home is proactive to program itself with the lifestyle and desires of the inhabitant. The smart hospital project [4] develops a robust approach for recognizing user's activities and estimating hospital-staff activities using a hidden Markov model with contextual information in the smart hospital environment. MIT researchers [5] recognize user' s activities by using a set of small and simple state-change sensors, which are easy and quick to install in the home environment. Unlike one resident system, this system is employed in multiple inhabitant environments and can be used to recognize Activities of Daily Living (ADL). CASAS Smart Home Project [6] builds probabilistic models of activities and used them to recognize activities in complex situations where multiple residents are performing activities in parallel in the same environment.

Based on a recent report [7], buildings are responsible for at least 40% of energy use in most countries. As an important part of buildings, household consumption of electricity has been growing dramatically. Thus, the need to develop technologies that improve energy efficiency and monitor the energy usage of the devices in household is emerging as a critical research area. The BeAware project [8] makes use of an iPhone application to give users alerts and to provide information on the energy consumption of the entire house. This mobile application can detect the electricity consumption of different devices and notify the user if the devices use more energy than expected. The PowerLine Positioning (PLP) indoor location system [9] is able to localize to sub-room level precision by using fingerprinting of the amplitude of tones produced by two modules installed in extreme locations of the home. Later work of this system [10] records and analyzes electrical noise on the power line caused by the switching of significant electrical loads by a single, plug-in module, which can connect to a personal computer, then uses machine learning techniques to identify unique occurrences of switching events by tracking the patterns of electrical noise. The MITes platform [11] monitors the changes of various appliances in current electricity flow for the appliance, such as a switch from on to off by installing the current sensors for each appliance. Other similar work [12] also proposes several approaches to recognize the energy usage of electrical devices by the analysis of power line current. It can detect whether the appliance is used and how it is used.

In our study, we extend smart home research to consider the resident's energy usage. We envision three applications of smart environments technologies for environmental energy efficiency:

SensorKDD'10, July 25, 2010, Washington, DC, USA.

1) analyzing electricity usage to identify trends and anomalies, 2) predicting the energy that will be used to support specific daily activities, and 3) automating activity support in a more energy-efficient manner. In this paper, we focus on the second task. The purpose of this paper is thus to validate our hypothesis that energy usage can be predicted based on sensor data that can be collected and generated by the residents in a smart home environment, including automatically-recognized activities, resident movement in the space, and frequency of classes of sensor events. The results of this work can be used to give residents feedback on energy consumption as it relates to various activities. Ultimately this information can also be used to suggest or automate activities in a more energy-efficient way.

In section 2, we introduce our CASAS smart environment architecture and describe our data collection and annotation modules. Section 3 presents the relationship between the energy data and the activities and describes machine learning methods to predict energy usage. Section 4 summarizes the results of our experiments and compares the performance between different learning methods and different experimental parameters.

2. CASAS SMART ENVIRONMENT

The smart home environment testbed that we are using to predict energy usage is a three bedroom apartment located on the Washington State University campus.

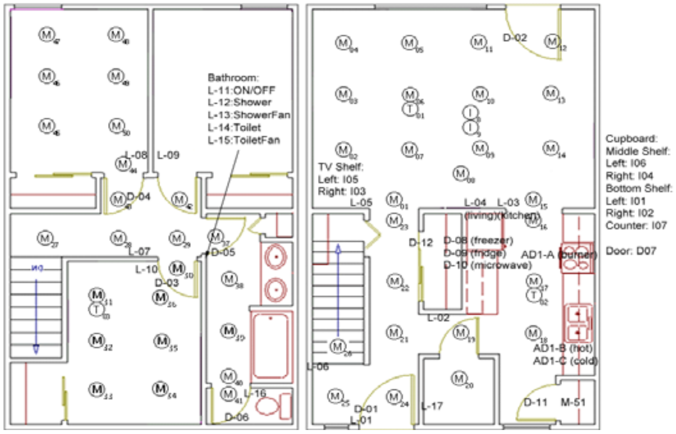


Figure 1. Three-bedroom smart apartment used for our data collection (motion (M), temperature (T), water (W), burner (B), telephone (P),and item (I)).

As shown in Figure 1, the smart home apartment testbed consists of three bedrooms, one bathroom, a kitchen, and a living/dining room. To track people's mobility, we use motion sensors placed on the ceilings. The circles in the figure stand for the positions of motion sensors. They facilitate tracking the residents who are moving through the space. In addition, the testbed also includes temperature sensors as well as custom-built analog sensors to provide temperature readings and hot water, cold water and stove burner use. A power meter records the amount of instantaneous power usage and the total amount of power which is used. An in-house sensor network captures all sensor events and stores them in a SQL database. The sensor data gathered for our study is expressed by several features, summarized in Table 1. These four fields (Date, Time, Sensor, ID and Message) are generated by the CASAS data collection system automatically.

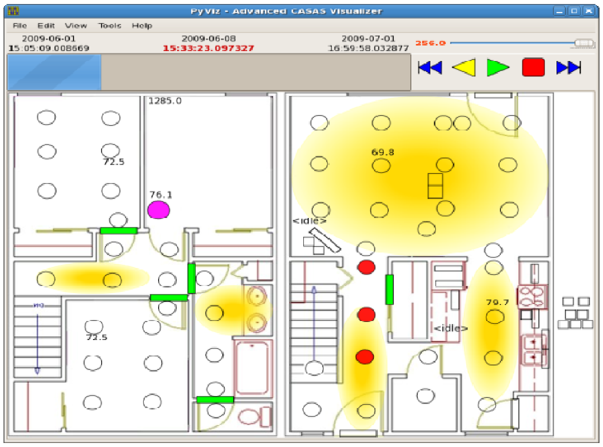
Table 1. Sample of sensor events used for our study. The first two events correspond to motion sensor ON and OFF messages. The third event is an ambient temperature reading, and the last two events represent current electricity usage.

Date	Time	Sensor ID	Message
2009-02-06	17:17:36	M45	ON
2009-02-06	17:17:40	M45	OFF
2009-02-06	11:13:26	T004	21.5
2009-02-05	11:18:37	P001	747W
2009-02-09	21:15:28	P001	1.929kWh

To provide real training data for our machine learning algorithms, we collect data while two students in good health were living in the smart apartment. Our training data was gathered over a period of several months and more than 100,000 sensor events were generated for our dataset. Each student had a separate bedroom and shared the downstairs living areas in the smart apartment. All of our experimental data are produced by these two students' normal lives, which guarantee that the results of this analysis are real and useful.

After collecting data from the CASAS smart apartment, we annotated the sensor events with the corresponding activities that were being performed while the sensor events were generated. Because the annotated data is used to train the machine learning algorithms, the quality of the annotated data is very important for the performance of the learning algorithms. As a large number of sensor data events are generated in a smart home environment, it becomes difficult for researchers and users to interpret raw data into residents' activities [13] without the use of visualization tools.

To improve the quality of the annotated data, we built an open source Python Visualizer, called PyViz, to visualize the sensor events. Figure 2 shows the user interface of PyViz for the CASAS project. PyViz can display events in real-time or in playback mode from a captured file of sensor event readings. Furthermore, we also built an Annotation Visualizer to visualize the resident's activities as shown in Figure 3.


PyViz - Advanced CASAS Visualizer

2009-06-01 15:05:09.008669

2009-06-08 15:33:23.097327

2009-07-01 16:59:58.032877

256.0

Figure 2. PyViz visualizer.

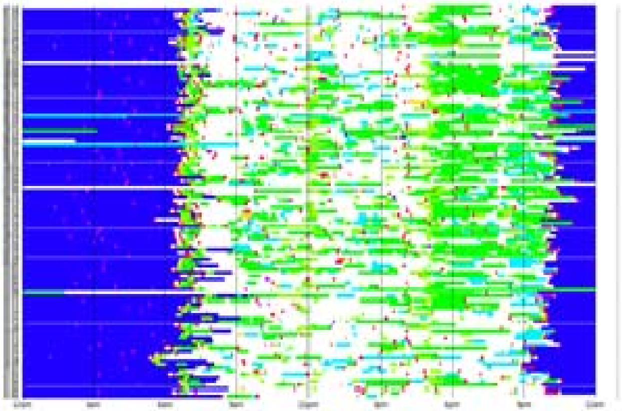


Figure 3. Visualizing activities in a smart home environment.

With the help of PyViz, activity labels are optionally added to each sensor event, providing a label for the current activity. For our experiment, we selected six activities that the two volunteer participants regularly perform in the smart apartment to predict energy use. These activities are as follows:

1. Work at computer
2. Sleep
3. Cook
4. Watch TV
5. Shower
6. Groom

All of the activities that the participants perform have some relationship with measurable features such as the time of day, the participants' movement patterns throughout the space, and the on/off status of various electrical appliances. These activities are either directly or indirectly associated with a number of electrical appliances and thus have a unique pattern of power consumption. Table 2 gives a list of appliances associated with each activity. It should be noted that, there are some appliances which are in "always on" mode, such as the heater (in winter), refrigerator, phone charger, etc. Thus, we postulate that the activities will have a measurable relationship with the energy usage of these appliances as well.

Table 2.Electricical appliances associated with each activity.

Activity	Appliances Directly Associated	Appliances Indirectly Associated
Work at computer	Computer, printer	Localized lights
Sleep	None	None
Cook	Microwave, oven, stove	Kitchen lights
Watch TV	TV, DVD player	Localized lights
Shower	Water heater	Localized lights
Groom	Blow drier	Localized lights

3. ENERGY ANALYSIS

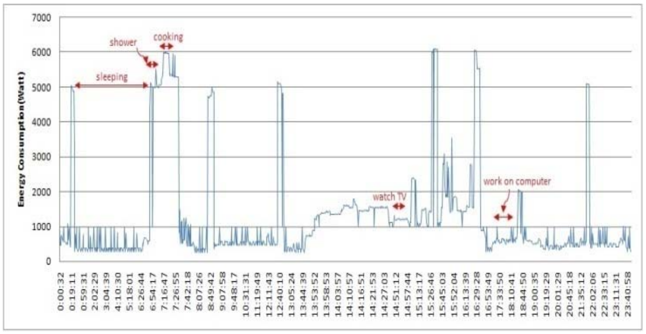


Figure 4. Energy usage for a single day.

Figure 4 shows the energy fluctuation that occurred during a single day on June 2nd, 2009. The activities have been represented by red arrows. The length of the arrows indicates the duration of time (not to scale) for different activities. Note that there are a number of peaks in the graph even though these peaks do not always directly correspond to a known activity. These peaks are due to the water heater, which has the highest energy consumption among all appliances, even though it was not used directly. The water heater starts heating by itself whenever the temperature of water falls below a certain threshold.

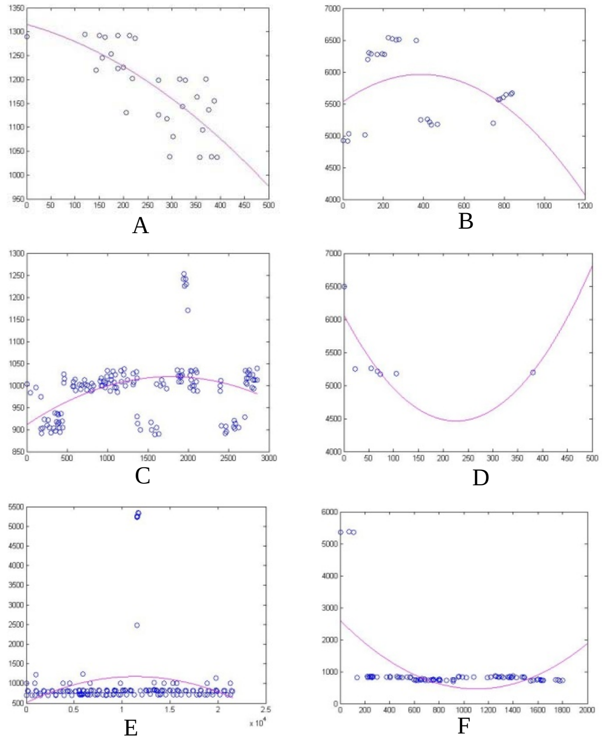


Figure 5. Energy data curve fitting for each activity.

(X-axis: wattage; Y-axis: second; A: Shower; B: Cook; C: Work on computer; D: Groom; E: Sleep; F: Watch TV)

Figure 5 plots typical energy data for each activity together with the result of applying curve fitting to the data. Curve fitting [14] is the process of building a mathematical function model that can best fit to a series of data points. It serves as an aid for data visualization, to approximate the values when no data are available, and to express the relationships between different data points. From the figure, we see that each resident's activity generates different energy patterns. The "cook" activity consumes the highest energy because the participants may open the refrigerator and use the stove or microwave oven, which need a relatively high power. Meantime, when the participants were sleeping, the energy consumption was the lowest because most appliances were idle.

3.1 Feature Extraction

Data mining and machine learning techniques use enormous volumes of data to make appropriate predictions. Before making use of these learning algorithms, another important step is to extract useful features or attributes from the raw annotated data. We have considered some features that would be helpful in energy prediction. These features have been generated from the raw sensor data by our feature extraction module. The following is a listing of the resulting features that we used in our energy prediction experiments.

1. Activity label
2. Activity length (in seconds)
3. Previous activity
4. Next activity
5. Number of kinds of motion sensors involved
6. Total number of times of motion sensor events triggered
7. Motion sensor M1…M51 (On/Off)

Target Feature: Total energy consumption range for an activity (in watts)

Activity label gives the name of the activity performed. Activity length is the duration of time a particular activity takes from beginning to the end. Features 3 and 4 represent the preceding and the succeeding activities to the current activity. Feature 5 takes into account the total number of different unique sensors used. Features 6 keeps a record of total number of sensor events associated with an activity. Feature 7 is not just one feature, but a collection of 51 features each representing a single motion sensor. Each of these sensor data records the total number of times a motion sensor was fired.

The input to the learning algorithm is a list of these seven features as computed for a particular activity that was performed. The output of the learning algorithm is the amount of electricity that is predicted to be consumed while performing the activity. To address the goal of predicting energy usage, we discretize the energy readings using equal width binning. Equal width binning [15] is also widely used in data exploration, preparation, and mining. Both of these binning techniques have been used to preprocess continuous-valued attributes by creating a specified number of bins, or numeric ranges. These benchmarks can be used to evaluate other machine learning classifiers we use in our experiments. In this paper, we discretize the target average energy data into several interval sizes (two classes, three classes, four classes and five classes, six classes) to assess the performance of our experiments.

3.2 Feature Selection

During feature extraction, our algorithm generates a large number of features to describe a particular situation. However, some of these features are redundant or irrelevant, resulting in a drastic raise of computational complexity and classification errors [16]. Features are selected by a method called attribute subset selection which finds a minimum set of attributes such that the resulting probability distribution of the data classes is as close as possible to the original distribution obtained using all attributes. In this paper, we have used information gain [17] to create a classification model, which can measure how well a given attribute separate the training examples according to their target classification. The performance of each attribute is measured in terms of a parameter known as information gain. It is a measure based on entropy, a parameter used in information theory to characterize the purity of an arbitrary collection of examples. It is measured as:

$$Entropy(S) \equiv -p_+ \log_2 p_+ - p_- \log_2 p_-$$

where, S is the set of data points, P^+ is number of data points that belong to one class (the positive class) and P^- is the number of data points that belong to the negative class. We adapt this measure to handle more than two classes for our experiments.

$$Gain(S, A) \equiv Entropy\,(S) - \sum_{v \in Values\,(A)} \frac{|S_v|}{|S|} Entropy\,(S_v)$$

where, *Values (A)* is the set of all possible values for attribute A. *Gain(S, A)* measures how well a given attribute separates the training examples according to their target classification. By using information gain, we can determine which features are comparatively more important than others for the task of target classification.

3.3 Energy Prediction

Machine learning [18] algorithms are capable to learn and recognize complex patterns and classify objects based on sensor data. In our study, we make use of four popular machine learning methods to represent and predict energy usage based on the features we selected: a Naïve Bayes Classifier, a Bayes Net Classifier, a Neural Network Classifier, and a Support Vector Machine. We test these four algorithms on the data collected in the CASAS smart home apartment testbed.

3.3.1 Naïve Bayes Classifier

A naïve Bayes Classifier [19] is a simple probabilistic classifier that assumes the presence of a particular feature of a class is unrelated to any other features. It applies Bayes' theorem to learn a mapping from the features to a classification label.

$$\mathrm{argmax}_{e_i \in E} P(e_i|F) = \frac{P(F|e_i)P(e_i)}{P(F)}$$

In this equation, E represents the energy class label and F stands for the features values we describe above. $P(e_i)$ is estimated by counting the frequency with which each target value e_i occurs in the training data. Based on the simplifying assumption that feature values are independent given the target values, the probabilities of observing the features is the product of the probabilities for the individual features:

$$P(F|e_j) = \prod_i P(f_i|e_j)$$

Despite its naïve design and over-simplified assumptions, the naïve Bayes classifier often works more effectively in many complex real world situations than other classifiers. It only requires a small amount of training data to estimate the parameters needed for classification.

3.3.2 Bayes Net

Bayes belief networks [20] belong to the family of probabilistic graphical models. They represent a set of conditional independence assumptions by a directed acyclic graph, whose nodes represent random variables and edges represent direct dependence among the variables and are drawn by arrows by the variable name. Unlike the naïve Bayes classifier, which assumes that the values of all the attributes are conditionally independent given the target value, Bayesian belief networks apply conditional independence assumptions only to the subset of the variables. They can be suitable for small and incomplete data sets and they incorporate knowledge from different sources. After the model is built, they can also provide fast responses to queries.

3.3.3 Artificial Neural Network

Artificial Neural Networks (ANNs) [21] are abstract computational models based on the organizational structure of the human brain. ANNs provide a general and robust method to learn a target function from input examples. The most common learning method for ANNs, called Backpropagation, which performs a gradient descent within the solution's vector space to attempt to minimize the squared error between the network output values and the target values for these outputs. Although there is no guarantee that an ANN will find the global minimum and the learning procedure may be quite slow, ANNs can be applied to problems where the relationships are dynamic or non- linear and capture many kinds of relationships that may be difficult to model by other machine learning methods. In our experiment, we choose the Multilayer-Perceptron algorithm with Backpropagation to predict electricity usage.

3.3.4 Support Vector Machine

Super Vector Machines (SVMs) were first introduced in 1992 [22]. This is a training algorithm for data classification, which maximizes the margin between the training examples and the class boundary. The SVM learns a hyperplane which separates instances from multiple activity classes with maximum margin. Each training data instance should contain one class label and several features. The goal of a SVM is to generate a hyperplane which provides a class label for each data point described by a set of feature values.

4. EXPERIMENT RESULTS

We performed two series of experiments. The first experiment uses the sensor data collected during two summer months in the testbed. In the second experiment, we collected data of three winter months in the testbed. The biggest difference between these two groups of data is that some high energy consuming devices like room heaters were only used during the winter, which are not directly controlled by the residents and are therefore difficult to monitor and predict. The test tool we use, called Weka [23], provides an implementation of learning algorithms that we can easily apply to our own dataset. Using Weka, we assessed the classification accuracy of our four selected machine learning algorithms using 3-fold cross validation.

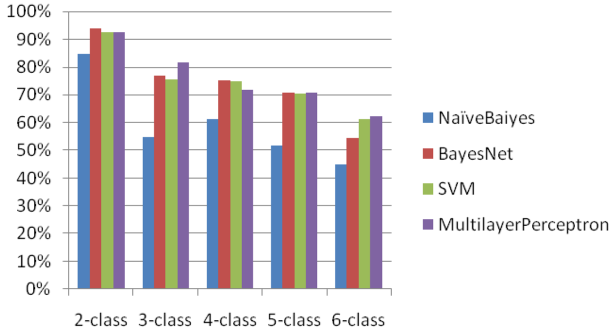


Figure 6. Comparison of the accuracy for summer dataset.

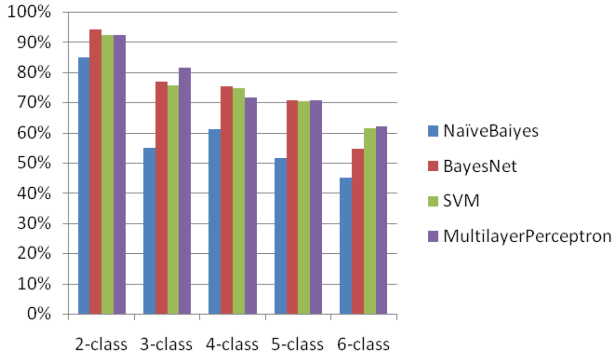


Figure 7. Comparison of the accuracy for winter dataset.

Figures 6 and 7 plot the accuracies of the two different group experiments, respectively. As shown in these two figures, the highest accuracy is around 90% for both datasets to predict the two-class energy usage and the lowest accuracy is around 60% for the six-class case in both datasets. These results also show that the higher accuracy will be found when the precision was lower because the accuracy of all four methods will drop from about 90% to around 60% with an increase in the number of energy class labels.

From the figures we see that the Naïve Bayes Classifier performs worse than the other three classifiers. This is because it is based on the simplified assumption that the feature values are conditionally independent given the target value. On the contrary, the features that we use, are not conditionally independent. For example, the motion sensors associated with an activity is used to find the total number of times motion sensor events were triggered and also the kinds of motion sensors involved.

To analyze the effectiveness of decision tree feature selection, we apply the ANN algorithm to both datasets with and without feature selection. From Figure 8, we can see the time efficiency has been improved greatly using feature selection. The time for building the training model drops from around 13 seconds to 4 seconds after selecting the features with high information gain. However, as seen in Figure 9, the classification accuracy is almost the same or a slight better than the performance without feature selection. The use of feature selection can improve the time performance without reducing the accuracy performance in the original data set.

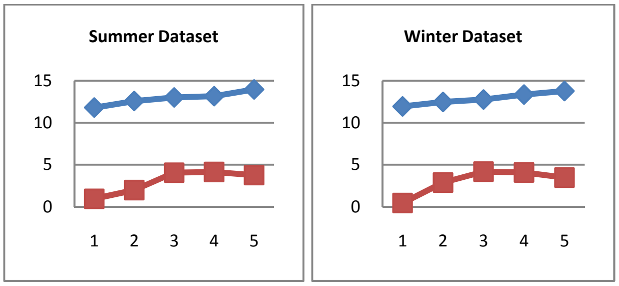


Figure 8. Comparison of time efficiency.

(1:2-class; 2:3-class; 3:4-class; 4:5-class; 5:6-class; Y-axis: second; Red: with feature selection; Blue: without feature selection). Time is plotted in seconds.

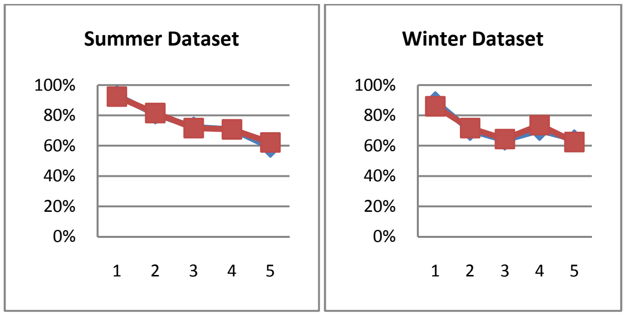


Figure 9. Comparison of prediction accuracy.

(1:2-class; 2:3-class; 3:4-class; 4:5-class; 5:6-class; Red: with feature selection; Blue: without feature selection).

Figure 10 compares the performance of the ANN applied to the winter and summer data sets. From the graph, we see that the performance for the summer data set is shade better than the performance for the winter dataset. This is likely due to the fact that the room and floor heater appliances are used during winter , which consumes a large amount of energy and are less predictable than the control of other electrical devices in the apartment.

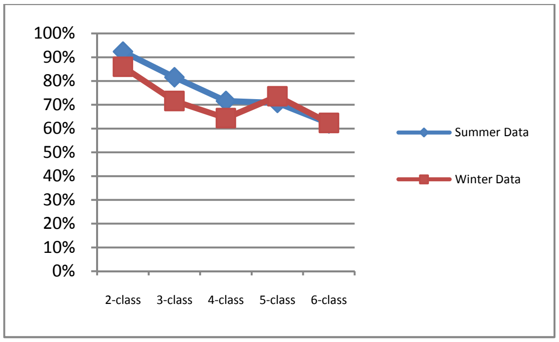


Figure 10 Comparison of the accuracy between two datasets.

5. DISCUSSIONS

Analyzing these results, we see that machine learning methods can be used as a tool to predict energy usage in smart home environments based on the human's activity and mobility. However, the accuracy of these methods is not as high as we anticipated when the energy data is divided into more than three classes. There are several reasons that lead to low performance of these algorithms. One reason is that some of the major devices are difficult to monitor and predict, such as the floor heater, which may rely on the outdoor temperature of the house. Another reason is that there is no obvious cycle of people’s activities. An additional factor we can't ignore is that there is some noise and perturbation motion when the sensors record data and transfer them into the database. Finally, the sensor data we collect is not enough to predict energy precisely. As a result, we intend to collect more kinds of sensor data to improve the prediction performance.

6. CONCLUSIONS

In this work, we introduced a method of predicting energy usage using an integrated system of collecting sensor data and applied machine learning in a smart home environment. To predict energy precisely, we extracted features from real sensor data in a smart home environment and selected the most important features based on information gain, then used an equal width binning method to discretize the value of the features. To assess the performance of the four machine learning methods, we performed two group experiments during two different periods, analyzed the results of the experiments and provided the explanation of those results.

In our ongoing work, we plan to further investigate new and pertinent features to predict the energy more accurately. To improve the accuracy of energy prediction, we intend to install more sensitive sensors to capture more useful information in the smart home environment. We are also planning to apply different machine learning methods to different environments in which different residents perform similar activities. This will allow us to analyze whether the same pattern exists across residents and environments. In our next step we will analyze the energy usage data itself to find trends and cycles in the data viewed as a time series. The results of our work can be used to give residents feedback on energy consumption as it relates to various activities and be also treated as a reference to research human's life style in their homes. In addition, predicted electricity use can form the basis for automating the activities in a manner that consumes fewer resources including electricity.

7. REFERENCES

[1] Brumitt, B., et al. 2000. Multi-Camera Multi-Person Tracking for EasyLiving. In *Conf. Proc. 3rd IEEE Intl. Workshop on Visual Surveillance.*

[2] Orr, R. J. and Abowd, G. D. 2000. The smart floor: A mechanism for natural user identification and tracking. In *Conference on Human Factors in Computing Systems*. 275–276.

[3] Mozer, M. C. 1998. The Neural Network House: An Environment hat Adapts to its Inhabitants. In *Proc. AAAI Spring Symp. Intelligent Environments*.

[4] Sánchez, D., Tentori, M. and Favela, J. 2008. Activity recognition for the smart hospital. *IEEE Intelligent Systems*. 23, 2 , 50–57.

[5] Tapia, E. M., Intille, S. S. and Larson, K. 2004. Activity recognition in the home using simple and ubiquitous sensors. *Pervasive Computing*. 158–175.

[6] Singla, G., Cook, D. J. and Schmitter-Edgecombe, M. 2010. Recognizing independent and joint activities among multiple residents in smart environments. *Journal of Ambient Intelligence and Humanized Computing*. 1–7.

[7] Energy Efficiency in Buildings. 2009. DOI= www.wbcsd.org.

[8] BeAware. 2010. DOI= www.energyawareness.eu/beaware.

[9] Patel, S.N., Truong, K.N. and Abowd, G. D. 2006. PowerLine Positioning: A Practical Sub-Room-Level Indoor Location System for Domestic Use. In *proceedings of UbiComp 2006: 8th international conference*. Springer Berlin / Heidelberg, 441-458.

[10] Patel, S.N., et al. 2007. At the flick of a switch: Detecting and classifying unique electrical events on the residential power line, In *proceedings of UbiComp 2007: 9th international conference*. Innsbruck, Austria, 271.

[11] Tapia, E., et al. 2006. The design of a portable kit of wireless sensors for naturalistic data collection. *Pervasive Computing*. 117–134.

[12] Bauer, G., Stockinger, K. and Lukowicz, P. 2009. Recognizing the Use-Mode of Kitchen Appliances from Their Current Consumption. *Smart Sensing and Context*. 163–176.

[13] Szewcyzk, S., et al. 2009. Annotating smart environment sensor data for activity learning. *Technol. Health Care*. 17, 3, 161-169.

[14] Coope, I. D. 1993. Circle fitting by linear and nonlinear least squares. *Journal of Optimization Theory and Applications.* 76, 2, 381–388.

[15] Liu, H., et al. 2002. Discretization: An enabling technique. *Data Mining and Knowledge Discovery*. 6, 4, 393–423.

[16] Bellman, R. E. 1961. *Adaptive control processes - A guided tour. Princeton*, New Jersey, U.S.A.: Princeton University Press. 255.

[17] Quinlan, J. R. 1986. Induction of Decision Trees. *Mach. Learn.*, 1, 1, 81-106.

[18] Mitchell, T. 1997. *Machine Learning*, New York: AMcGraw Hill.

[19] Rish, I. 2001. An empirical study of the naive Bayes classifier. In *IJCAI-01 workshop on "Empirical Methods in AI"*.

[20] Pearl, J. 1988. *Probabilistic reasoning in intelligent systems: networks of plausible inference*, Morgan Kaufmann.

[21] Zornetzer, S. F. 1955. *An introduction to neural and electronic networks*, Morgan Kaufmann.

[22] Boser, B. E., Guyon, I. M. and Vapnik, V. N. 1992. A training algorithm for optimal margin classifiers. In *COLT '92: Proceedings of the fifth annual workshop on Computational learning theory*, Pittsburgh, Pennsylvania, United States, 144-152.

[23] Witten, I. H. and Frank, E. 1999. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations* (The Morgan Kaufmann Series in Data Management Systems), Morgan Kaufmann.