CS 458 Information Security Project

Smart Meter Data Disaggregation and Clustering

By-

Karthik Narisetti

Marcus Silva

Uzair Ahmed

Introduction

Advanced Metering infrastructure consists of a modern electric metering system technology that starts to replace the old one. Smart meters are an important part of the system and provide communications between the utility and the customers. They introduce the concept of demand response that enables the costumers to participate directly in energy markets and limit their usage during periods of peak demand, for example.

One of the approaches to implement demand response involves the sending of Load Shed Instructions (LSIs) by the ESP to specified appliances, and is called direct control. The appliances should accept Load Shed Verification in order to improve the reliability of this method. That means residential power meter is used to monitor electrical usage and analyzing load changes, and it sends a LSV response message when an appropriate change is verified. This leads to issues related to trust that must be addressed by the models proposed.

One of these models is based on the Non-Intrusive-Load Monitoring (NILM), which uses the aggregate reading of the meters over time to determine individual appliances that contribute to the load considering a previously knowledge of the appliance's power profile. The survey addresses this question, by the implementation of an algorithm based on knapsack, which takes aggregate power data and formulates the appliances' state in certain amounts of time.

Problem Description

In the context of smart meter technology, the customer's data is a valuable instrument to the ESPs, as they provide information for controlling energy generation costs and grid stability. The Companies offer customers discounts or other incentives if they agree to allow them to send Load Shed Instructions to specified appliances. An appropriated treatment and analysis of the sort of aggregate information gathered from the meters also enables deeper assumptions about customer's profiles, such as how frequently they utilize certain combinations of appliances.

Privacy concerns arise in this sense, if interested parties or even bad guys handle this kind of information. In fact, some researches show that it is possible to estimate personal information with a certain degree of accuracy, with relatively unsophisticated hardware and algorithms. In short, there is a need for discussion on the privacy aspect of data collection.

In the case of this survey, the problem is addressed as the deployment of an algorithm to deduce appliances' state from the aggregate meter reading of a household and some prior knowledge of the appliances' load profiles. There is also a clustering process based on power consumption and time of use of the appliances that enables interesting assumptions and visualization of the data.

Related Work

The Load Shed Verification technique requires some kind of confirmation message that the appliances should send in order to verify the compliance to the Load-Shed Instructions. The Non-Intrusive Load Monitoring technology plays an important role supporting prominent approaches in this area. The approaches are generally based on large quantities of detailed data and complex computations.

The question related to the privacy of the customer's information has led to solutions based on the use of home electrical power routing to moderate the home's load signature in order to hide appliance usage information, as well as data aggregation and cryptographic techniques.

Our Approach

In this project in order to achieve the goal we employ Knapsack algorithm to retrieve the individual appliance's power consumption signatures to indicate which appliance is on or off at a given instance of time and K means clustering to cluster the result of knapsack algorithm into group of appliances which are usually on at the same time interval.

Knapsack algorithm:

This is a combinatorial optimization algorithm which given a set of items, each with a mass and a value, determine the number of each item to include in a collection so that the total weight is less than or equal to a given limit and the total value is as large as possible. It derives its name from the problem faced by someone who is constrained by a fixed-size knapsack and must fill it with the most valuable items.

We use this algorithm to help us retrieve the appliances that are 'ON' at a given point of time from the input aggregate data of power consumption of a regular house hold. The input aggregate data has varied fields, which indicate different aspects of the power consumption of the household, we are only interested in the Day, time and power consumption in KW. For a particular day, and for a given instance of time in seconds, the amount of power consumption in KW is supplied to the knapsack algorithm as the input. The knapsack algorithm, using an already known data which has the individual appliance's signature. Using these power consumption signatures, the algorithm, tries to fit the maximum number of appliances within the Actual power consumption in KW from the aggregated input of the household for that given time instance. Employing this method we can derive which appliances are actually 'ON' at a given instance of time to use it for further processing.

K-means clustering:

It's a method of vector quantization, popular for cluster analysis in data mining. *K*-means clustering aims to partition *n* observations into *k* clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.

We use this algorithm to help us cluster the output of the knapsack algorithm, which is the appliances that are ON at a given instance of time. K means clusters the appliances to groups of clusters which show all the other appliances which are also ON along with it. This can be used for both good, an Electric service company attempting to save up energy during the peak hours and also save money for the consumers as well as nefarious purposes as a thief using this to evaluate the patterns and using them to his advantage, to maybe attack when the house owners are sleeping or are away.

Experimental results:

Code is divided into 2 scripts:

- First one derives each individual signature
- Second one will cluster the data found similar.

The below are the snapshots of results we performed over the give input.

Input:

1	date	time	kw	zero	rms
2	11/13/2009	12:00:00	4.9	0	123.6
3	11/13/2009	12:00:01	4.9	0	123.6
4	11/13/2009	12:00:02	4.9	0	123.6
5	11/13/2009	12:00:03	8.25	0	123.6
6	11/13/2009	12:00:04	8.25	0	123.6
7	11/13/2009	12:00:05	8.25	0	123.6
8	11/13/2009	12:00:06	9.65	0	123.7
9	11/13/2009	12:00:07	9.65	0	123.7
10	11/13/2009	12:00:08	9.65	0	123.7
11	11/13/2009	12:00:09	10	0	123.6
12	11/13/2009	12:00:10	10	0	123.6
13	11/13/2009	12:00:11	10	0	123.6
14	11/13/2009	12:00:12	10	0	123.6
15	11/13/2009	12:00:13	10	0	123.6
16	11/13/2009	12:00:14	13.35	0	123.6
17	11/13/2009	12:00:15	13.35	0	123.6
18	11/13/2009	12:00:16	13.35	0	123.6
19	11/13/2009	12:00:17	13.35	0	123.6
20	11/13/2009	12:00:18	13.35	0	123.6

In the above input we are using time and aggregate KW data to draw individual appliance signature. We consider to have six appliances and individual appliance consumption is as follows.

```
ind <<- c(2.15, 2.75, 3.35, 3.55, 4.20, 5.80)
```

Using knapsack as described in approach, we are deriving individual appliance signature as below. Output is a matrix as

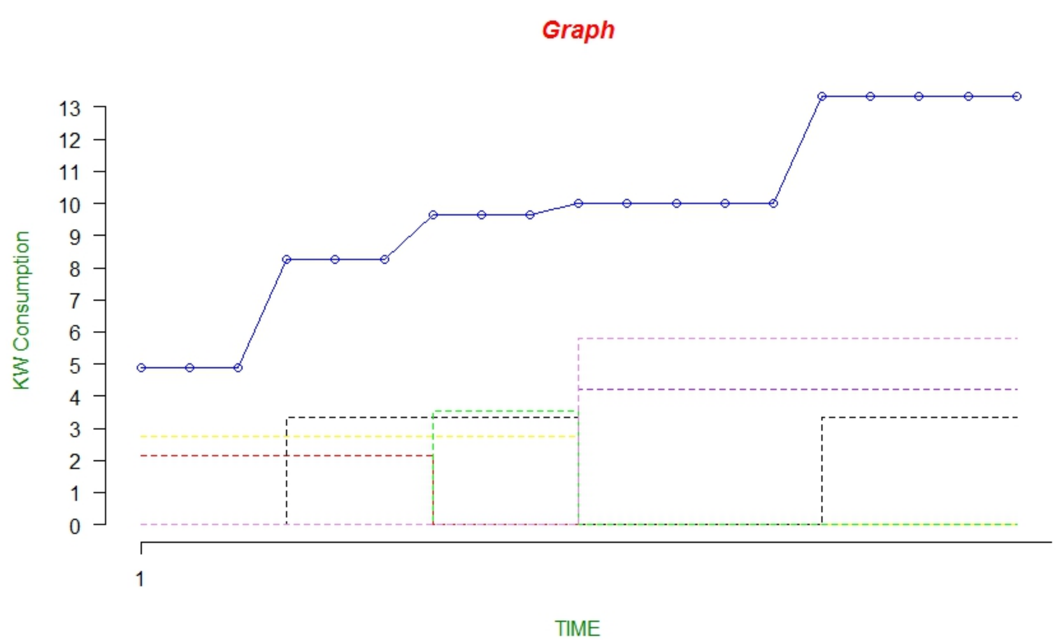
- **Row** – containing the data of all the appliances at a particular moment of time
  - **Value of one attribute is zero** – Then it is off then,
  - **If not zero** – Then it is the consumption of that appliance at that point of time.
- **Column** – Contains one appliance data over a period of time. So each column represents an appliance.
  - **Value of one attribute is zero** – Then it is off then,
  - **If not zero** – Then it is the consumption of that appliance at that point of time.

Output of the first script:

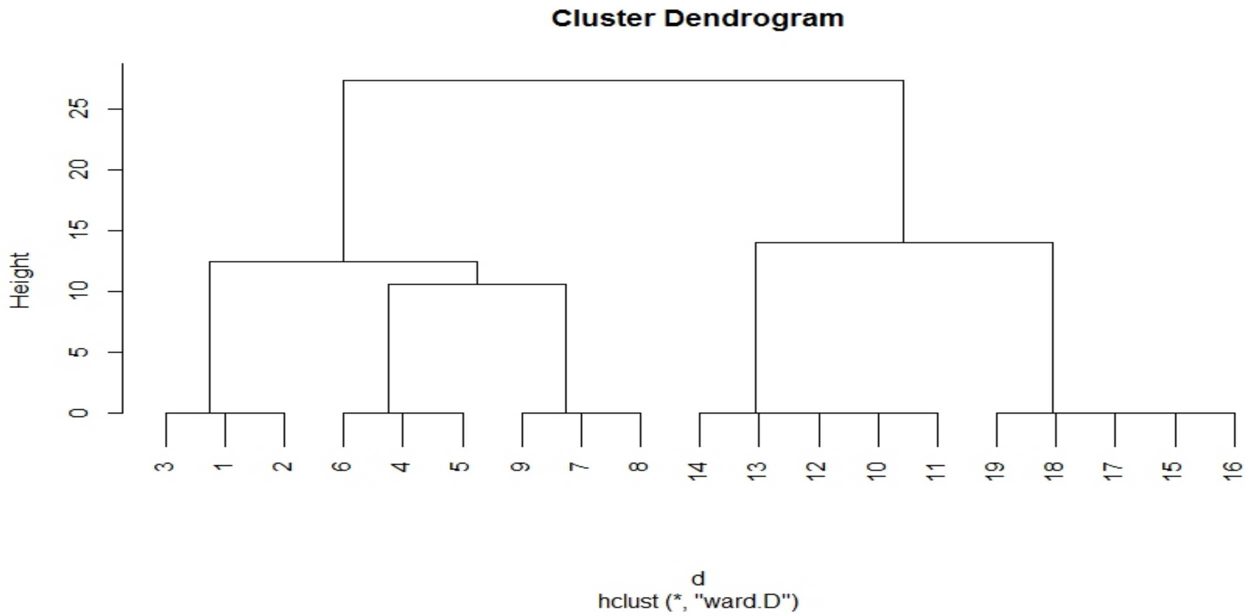
At end of first script, we get individual signatures are derived in a matrix as below.

```
File   Edit   Format   View   Help
2.15 2.75 0 0 0 0
2.15 2.75 0 0 0 0
2.15 2.75 0 0 0 0
2.15 2.75 3.35 0 0 0
2.15 2.75 3.35 0 0 0
2.15 2.75 3.35 0 0 0
0 2.75 3.35 3.55 0 0
0 2.75 3.35 3.55 0 0
0 2.75 3.35 3.55 0 0
0 0 0 0 4.2 5.8
0 0 0 0 4.2 5.8
0 0 0 0 4.2 5.8
0 0 0 0 4.2 5.8
0 0 0 0 4.2 5.8
0 0 3.35 0 4.2 5.8
0 0 3.35 0 4.2 5.8
0 0 3.35 0 4.2 5.8
0 0 3.35 0 4.2 5.8
0 0 3.35 0 4.2 5.8
```

And when we plot a graph of time against KW consumption (Both aggregate and individual appliance) it looks as below. Where aggregate is a bubbled line and each individual appliance consumption is a square wave drawn in dotted lines.



We are feeding the matrix data to a script that cluster it based on the commonly repeated rows and we are clustering all moments of time where the same appliances are on together. So it gives us a cluster dendogram, which explains hierarchically the data was clustered.



Assumptions and Limitations:

In this project we've made some assumptions, which when hold true ensures that the program runs perfectly and gives the results as expected. If the assumptions do not hold true, the experiments and results may not be accurate.

Assumptions:

We assume

- that the power consumption signatures of the appliances are known and are accurate
- the power consumption signatures of the appliances don't change, i.e, we assume that the power consumption of an appliance remains constant and doesn't vary
- that this household contains only one appliance of each kind, i.e no two appliances are of the same kind
- no two appliances have the same power consumption in KW. We assume that the appliances have different and distinct power consumption rate
- that the power consumption is uniform in most house holds

Limitations:

The limitations of this project are as follows,

- The output of the knapsack algorithm may not necessarily reflect the actual appliances which are ‘ON’
- If two appliance’s power consumption adds up to another appliance, then we can’t really infer which appliance is actually ‘ON’
- If a household has multiple appliances of the same kind which is very likely, then classifying them may become difficult.
- The power consumption of an appliance may change, sometime based on the mode of operation or just over time, in that case the results of this project may not be accurate
- The data set used was limited to one house hold, and is not a representative of most households to conclude that the results apply to more than one household

Future Work:

To improve the accuracy and efficiency of the results of this project, we could use more granular data, one which also gives the reactive power and so on, which would help further distinguish between two appliances, and even solve the limitation of having unique appliances and their power consumption signatures. And by using a larger data set, which comprises of more than one household, more accurate results can be obtained, which can be applied to multiple households.

Conclusion:

In conclusion, the smart meters have many aspects which can be improved, like better data disaggregation to better security measures. We demonstrate how using smart meter data power consumption of a house hold data can be disaggregated easily using knapsack algorithms and using this result it can further be clustered (using K-means) and used to generate patterns of a household. These found patterns can be used for both good and nefarious purposes. From an ESP wanting to save energy consumption during peak hours and saving money for their consumers to a mischief maker or a thief understanding the patterns to attack a household to pull off an organized robbery, is all possible and can be carried out easily.

References:

[1] “An Integrated Architecture for Demand Response Communications and Control,” by M. LeMay, R. Nelli, G. Gross, and C. A. Gunter.

[2] “Non- Intrusive Load Shed Verification”, by D. C Bergman, D. Jin, J. P Juen, N. Tanaka, C. A Gunter and A. K Wright.

[3] “Nonintrusive appliance load monitoring,” by G. W Hart.

[4] “Privacy- Aware sampling for residential demand response programs” by A. A Cardenas, S. Amin, G. Schwartz.

[5] “Power Signature Analysis,” C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, N. Les, and P. Armstrong

[6] “Collaborative Recommender Systems for Building Automation,” by M. LeMay, J. J Haas and C. A Gunter