Energy Aggregation using Product of Experts

Megha Gupta*, Haimonti Dutta†, Amarjeet Singh‡ and Ullas Nambiar§
*Dept. of Computer Science, IIIT Delhi
meghag@iiitd.ac.in
†Department of Management Science and Systems,
State University of New York, Buffalo
New York, 14260
haimonti@buffalo.edu
‡Dept. of Computer Science, IIIT Delhi
amarjeet@iiitd.ac.in
§EMC Corporation,
Bangalore, India
Ullas.Nambiar@emc.com

***Abstract*—The abstract goes here. DO NOT USE SPECIAL CHARACTERS, SYMBOLS, OR MATH IN YOUR TITLE OR ABSTRACT.**

***Keywords*-product of experts; energy aggregation; contrastive divergence;**

I. Introduction

Aggregation is a process in which data is gathered and used for statistical analysis. Energy aggregation is to collect data from multiple meters. The purpose of energy aggregation is to get some valuable information about single or multi-site units. In this paper, we are gathering energy consumption information of a building from two smart metres using product of HMMs. The concept is applied on two datasets, REDD dataset and the other was dataset was generated by us. The system architecture of our infrastructure consists of two smart meters S1 and S2 installed in a faculty housing building collecting data of twelve floors. S1 collects data from first six floors (0 to 5th) and the data from rest of the floors (6th to 11th) is collected by using S2. The data collected from two meters is aggregated using product of experts technique in a way that the contrastive divergence between the two probability distributions is minimized. The proof of concept of REDD dataset and faculty housing dataset is given in section IV and section V respectively.

II. Related Work

A. Automata and their products

Distributed networks can be modelled using interacting automata. Benveniste defines automaton as a quadruple, Á = (X,X_0,A,T) where X is a finite state of sets, X0 is the subset of initial states, A is a finite set of messages, T is a set of transitions of the form t = {x_,a,x} where x_ is the previous state, a is the message label on which the state transitions to the next state x. The figure 1 below explains the automata with an example.

For automaton R, X_R = {2; R1,R2}, X_{0R} = {R1}, A_R = {a,b}, T_R = {R1,a,R1; R1,b,R2; R2,a,R2; R2,b,R1}
For automaton S, X_S = {3; S1,S2,S3}, X_{0S} = {S1}, A_S = {a,b}, T_S = {S1,a,S1; S1,b,S2; S2,a,S2; S2,b,S3; S3,a,S3; S3,b,S1}
The product of two automata Á = R x S is defined as follows:
$X = X_R \text{ x } X_S$
$X_0 = X_{0R} \text{ x } X_{0S}$
$A = A_R \cup A_S$
Benveniste uses a notion of stuttering transition which helps to distinguish between local and global time by inserting dummy transitions between two transitions of a local automaton attached to a node. This stuttering transition does nothing but lets the rest of the world progress.

A	R1	R2
R1	0.6	0.4
R2	0.3	0.7

Table I
Transition probability, A

B	a	b
R1	0.2	0.8
R2	0.5	0.5

Table II
Observed probability, B

	R1	R2
π	0.4	0.6

Table III
Initial state probability, π

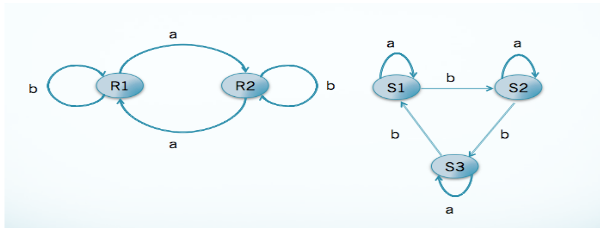

Figure 1. Automata R and S

Talking in terms of HMM, requires us to equip products of automata with probabilities. Benveniste defines HMM as a triple (Á, μ, π) where Á = (X,X_0,A,T) is an automaton, μ is the initial state probability, π is factored as state transition probability π_x and message transition probability π_A. He uses a random arbiter α, with values first, second, third to choose automaton to initiate transition. If α = first then first automaton chooses any transition having a private message whereas second automaton performs a stuttering transition, and vice versa for α = second. If α = both, then both automata agree on some shared message and move accordingly.

Using the traditional HMM notation of the parameters λ = {A, B, π } where A is the transition probability, B is the observed probability, π is the initial state probability. For automata R, we have the values of A, B, π as shown in table I, II, III respectively.

B. Product of HMM

Product of HMM is a way of combining HMM's to form distributed state time series model. The figure 2 is a product of two HMMs shown in 1. For P = R x S, the quadruple becomes

X = {6; R1S1, R1S2, R1S3, R2S1, R2S2, R2S3}
X_0 = {R1S1}
A = {a,b}
The rules for synchronised product construction are :
1. $<p,q>$ –a–> $<p',q>$ if a $\in A_R \cap A_S$ and p –a–> p' and q –a–> q'
2. $<p,q>$ –a–> $<p',q>$ if a $\in A_R$, a $\notin A_S$ and p –a–> p'
3. $<p,q>$ –a–> $<p,q'>$ if a $\notin A_R$, a $\in A_S$ and q –a–> q'

III. Methodology

A. Training product of experts by minimising contrastive divergence

High dimensional distributions are approximated as a product of one dimensional distributions. The product of individual distributions which are uniguassian or multivariate guassian will also be multivariate guassian. If the individual models are more complicated and contain one or more hidden variables, multiplying their distributions together and renormalizing them can be very powerful. These individual models are called "experts". The product of experts produce sharper distribution than the individual distributions [2].

IV. Proof of concept on REDD House 2

A. Aim

To represent streams of energy consumption data from n^1 appliances by product(s) of k HMMs.

B. Method

- **Data** The Reference Energy Disaggregation Data Set (REDD) is used in empirical analysis. The data contains power consumption from real homes, for the whole house as well as for each individual circuit in the house (labeled by the main type of appliance on that circuit). It is intended for use in developing disaggregation methods, which can predict, from only the whole-home signal, which devices are being used. The REDD data set contains two main types of home electricity data: high-frequency current/voltage waveform data of the two power mains (as well as the voltage signal for a single phase), and lower-frequency power data including the mains and individual, labeled circuits in the house. The main directory consists of several house directories, each of which contain all the power readings for a single house. Each house subdirectory consists of a labels and channels files. The labels file contains channel numbers and a text label indicating the general category of device on this channel. Each channel_i.dat file has two columns containing UTC timestamps (as integers) and power readings (recording the apparent power of the circuit) for the channel. Experiments reported here use the House 2 data from REDD. It has 11 channels where each channel corresponds to the following appliance:
  1) mains_1
  2) mains_2
  3) kitchen_1
  4) lighting
  5) stove
  6) microwave

[1] n=2

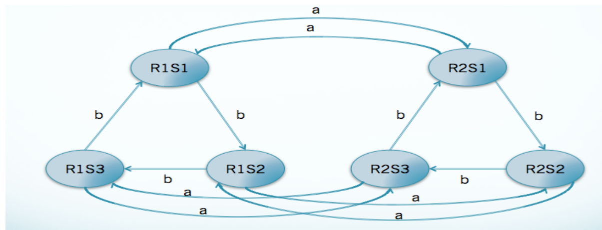

Figure 2. Product of HMMs, P = R x S

7) washer_dryer
8) kitchen_2
9) refrigerator
10) dishwaser
11) disposal

The dataset has 318759 records and 2 columns. We randomly sample 300 records for our initial experiment. Time series data from two appliances are represented as product of k HMMs.

- **Time Series :** The time series data of the microwave, dryer, kitchen_2 and refrigerator are plotted below in Figures 3, 4, 5, 6.
- **Code** The implementation of the product of experts model is obtained from Iain Murray's website [3]. It implements the technique described in Geoff Hinton's paper [2].
- **Additional details** Some additional details regarding experiments:
  1) The product of HMMs model (PoHMM) minimizes "contrastive divergence" as described in the paper [2].
  2) The number of experts, k used here is 15. This is set somewhat arbitrarily and needs to be experimented on.
  3) Learning rate is $\epsilon = \frac{1}{300}$.

C. Experimental Setup for REDD house 2

Experiments are performed on the REDD which contains 9 appliances each containing 318759 rows of energy consumption data. Experiments are subdivided into 4 parts, in the first part the number of data samples are varied corresponding to which the values of KL Divergence and convergence time are noted down. In the second part, the number of experts are varied keeping the best value of the sample from the first part constant. In the third part, number of iterations are varied keeping the best values from part 1 and 2 constant. In the fourth part, the no. of appliances to be aggregated are varied keeping the best values from above parts constant.

D. Results

The evaluation of how well the learning has taken place is done by using a Kullback-Leibler divergence metric which gives the difference between two probability distributions. The two probability distributions in the REDD example refer to the expert probabilities in real and fantasy data. The learned parameters from the training are fitted to the fantasy data to measure the information lost when fantasy data is used to approximate real data. This is denoted as D_{KL}(real data || fantasy).

Samples	$KLDiv$	$T(sec)$	$Iterations$
300	2.4864	186.212 $\pm$9.087	18600
500	0.6761	106.564 $\pm$10.046	10200
1000	1.1088	158.521 $\pm$1.97	11200
1500	3.8829	92.896 $\pm$8.075	5300
2000	1.8686	130.98 $\pm$1.932	6900
2500	0.4733	215.563 $\pm$ 2.471	9900
3000	2.8204	258.213 $\pm$1.918	11000
3500	1.2332	204.661 $\pm$1.713	7900
4000	0.8959	292.666 $\pm$0.619	10400
4500	1.1118	222.558 $\pm$1.967	7200
8000	6.392	381.635 $\pm$2.952	8100
10000	8.276	887.932 $\pm$13.824	10500
15000	0.7201	1368.514 $\pm$13.605	9400

Table IV
EFFECT OF VARYING SAMPLES ON KL DIV AND TIME

Experts	$KLDiv$	$T(sec)$	$Iterations$
5	0.774	72.968 $\pm$1.177	5200
10	1.424	117.482 $\pm$1.966	6700
15	0.473	210.249 $\pm$1.258	9900
20	1.56	217.739 $\pm$10.452	9000
25	7.469	347.019 $\pm$8.23	12100
30	2.4968	413.802 $\pm$7.304	12900
35	1.5012	348.906 $\pm$14.651	11300

Table V
EFFECT OF VARYING EXPERTS ON KL DIV AND TIME

Threshold	$KLDiv$	$T(sec)$	$Iterations$
.1	0.473	210.6 $\pm$1.493	9900
.05	0.443	240.607$\pm$2.436	10900
.01	0.454	431.536 $\pm$14.509	18000
.005	0.509	1167.243 $\pm$43.412	49800

Table VI
EFFECT OF VARYING MIN THRESHOLD ON KL DIV AND TIME

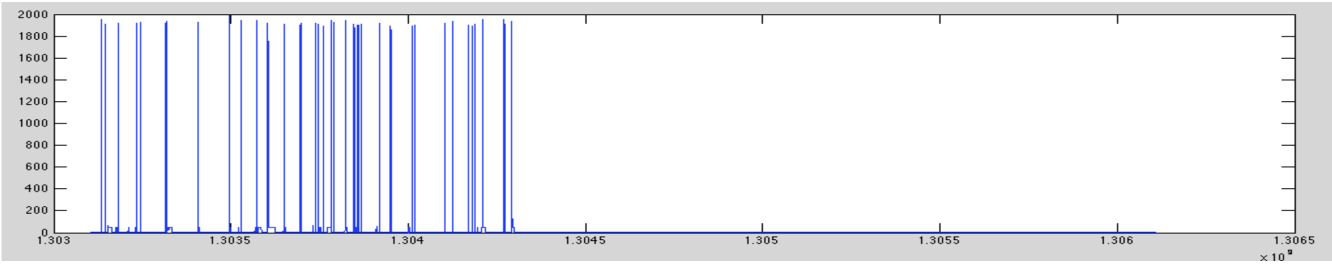

Figure 3. Microwave

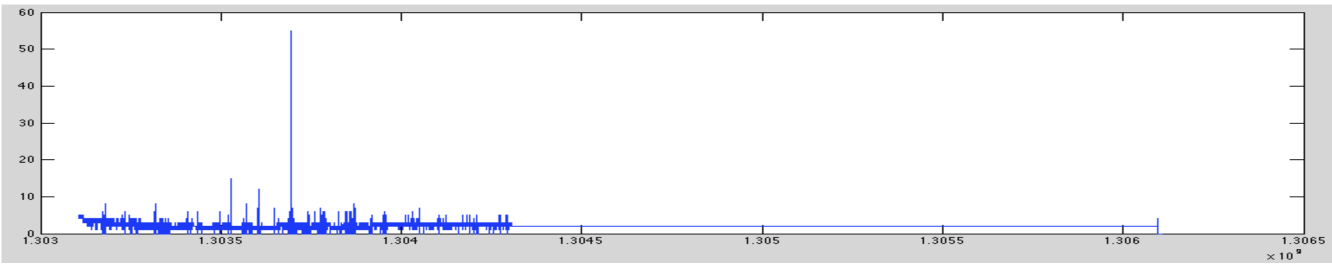

Figure 4. washer_dryer

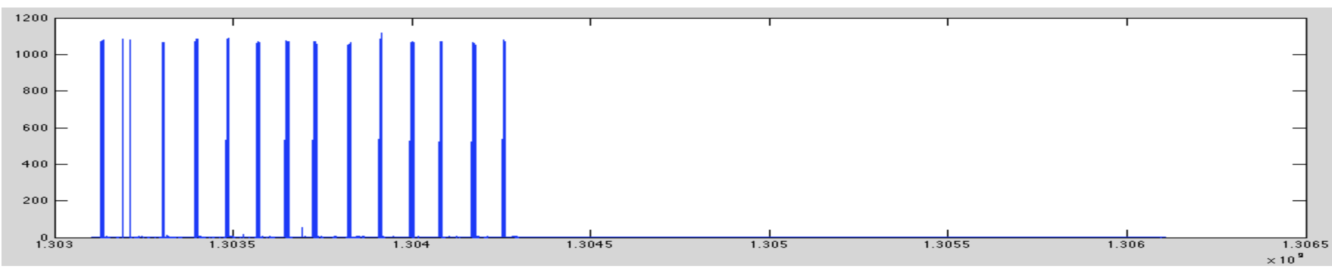

Figure 5. Kitchen_2

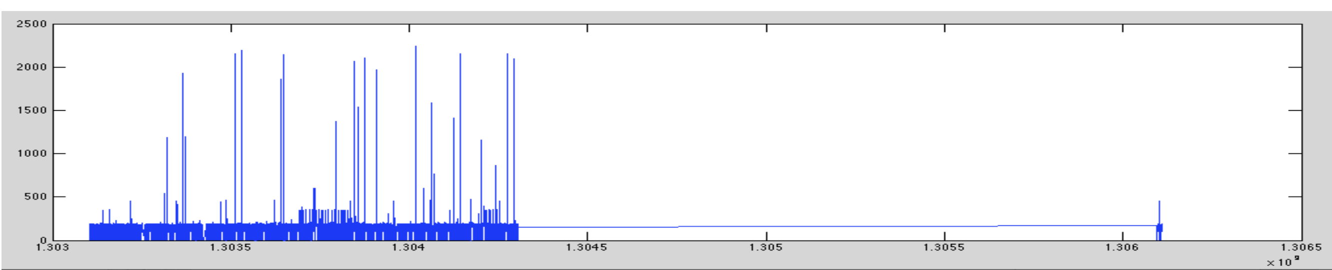

Figure 6. refrigerator

Appliances	$KLDiv$	$T(sec)$	$Iterations$
3	5.559	233.664 ±0.579	10700
4	0.188	465.634 ±5.275	19900
5	.432	338.416 ±3.988	13400
6	8.736	606.062 ±7.534	28100
7	5.054	411.457 ±10.051	17300
8	0.436	260.544 ±cc27.862	10700
9	0.15	474.579 ±14.619	20600

Table VII
EFFECT OF VARYING APPLIANCES ON KL DIV AND TIME

V. PROOF OF CONCEPT ON FACULTY HOUSING DATA

A. *Aim*

To represent streams of energy consumption data from all the floors of faculty housing as a product(s) of k HMMs.

B. *Method*

- **Data** This data represents the energy consumed by the IIIT Delhi faculty housing building. As a part of research, a team from IIIT Delhi has installed various temperature, light and motion sensors to perform real world studies and to analyse user preferences for energy conservation. For our analysis, we selected one month's historical data ranging from 01-01-2014, 00:01 hours to 31-01-2014, 23:59 hours. The two smart meters installed captures the data from all the floors. The first meter gives out readings from floors 0 to 5 and the second meter gives out readings from floors 6 to 11. The dataset includes timestamp and power consumed in watts and 84133 records. Time series data from two streams are modelled as a product of k HMMs. We also have the total power consumed by the faculty housing building which would serve as the ground truth to compare product of k HMMs with. The data is obtained from the website whose screenshot is shown in 7

- **Code** The implementation of the product of experts model is obtained from Iain Murray's website [3]. It implements the technique described in Geoff Hinton's paper [2].
- **Time Series :** The time series data of the energy consumption of floor 0 to 5, floor 6 to 11 and total power are plotted below in Figures 8, 9, 10.
- **Additional Details**

C. *Experimental Setup*

Each of the data stream is modelled as a HMM individually. There are three streams of data, the first stream D1 corresponds to the data from floor 0-5, D2 corresponds to floor 6-11 and D3 represents the total power from the faculty housing which is represented by a fixed test set, T. Firstly, the stream D1 is used to train the model by minimising contrastive divergence. The parameters learnt during training are provided to the test set T to obtain the conditional probability of the gaussians given the data D1 as pgauss1. Similarly, the second stream of data, D2 collected from floor 6-11, is used to learn the parameters of the model during the training phase which are then again provided to the test set T to obtain conditional probability of the gaussians given the data D2 as pgauss2. Finally the data D3 is used to learn the model and parameters which are then applied to the test set T to obtain the gaussian probability as pgauss3. Now, as we know that the total power consumption of the building should be approximately equal to the product of HMMs, hence the value of the product of pgauss1 and pgauss2 should be as close as possible to pgauss3.

D. *Results*

VI. Conclusion & Future Work

The conclusion goes here. this is more of the conclusion

Acknowledgment

The authors would like to thank... more thanks here

References

[1] Andrew Brown and Geoffrey Hinton. Proceedings of artificial intelligence and statistics 2001. In *Products of Hidden Markov Models*, number GCNU TR 2000-008, 2001.

[2] Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. Technical Report GCNU TR 2000-004, Gatsby Computational Neuroscience Unit, University College London, 2000.

[3] Iain Murray. Training products of experts by minimizing contrastive divergence, June 2012.

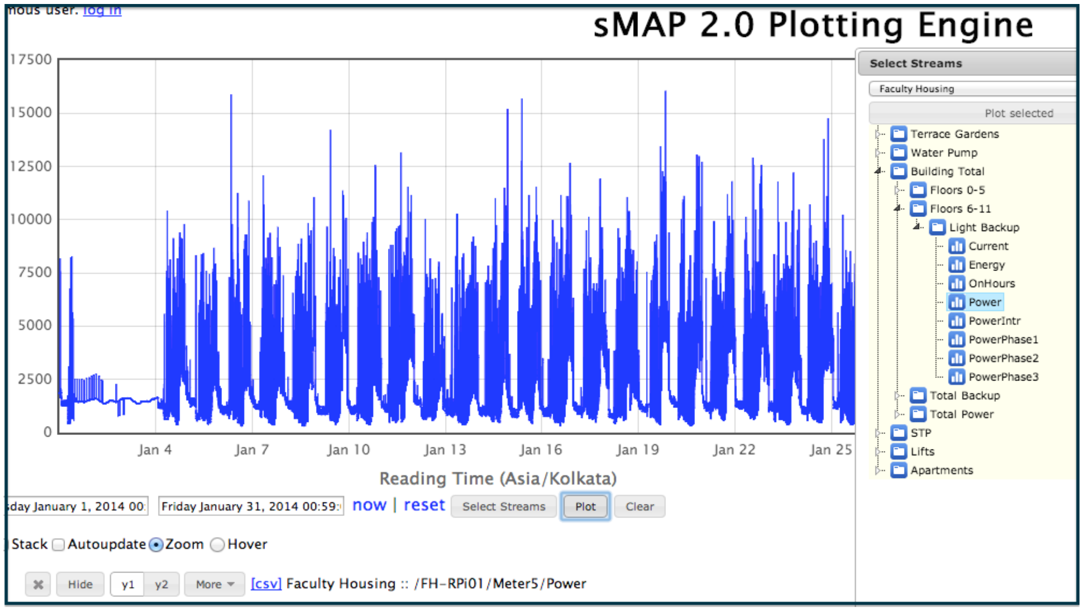

Figure 7. Screen shot of the webpage

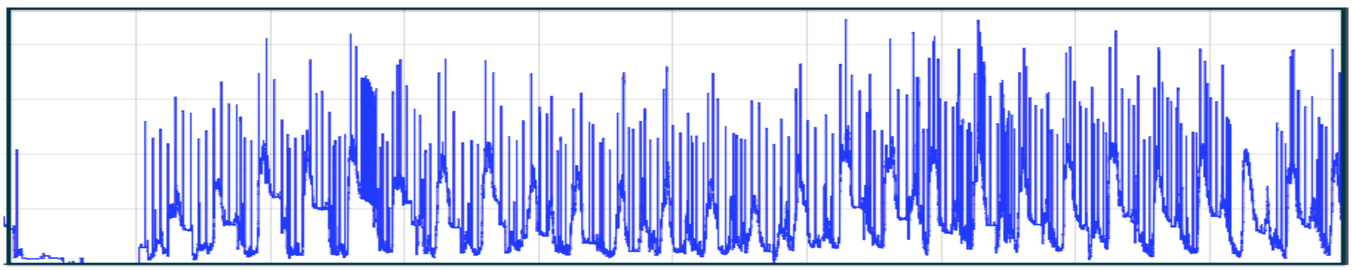

Figure 8. Stream 1: Power consumption of floors 0-5

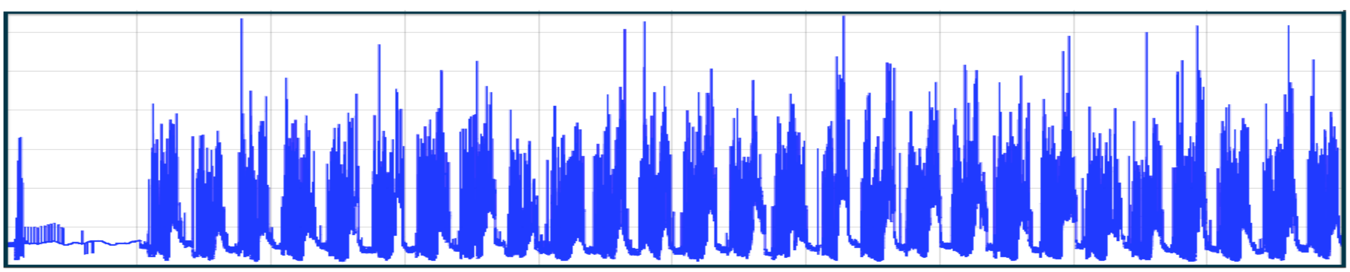

Figure 9. Stream 2: Power consumption of floors 6-11

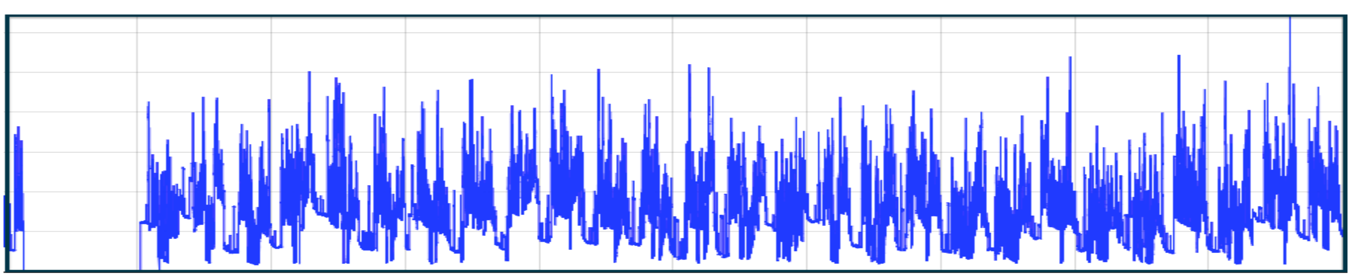

Figure 10. Total Power of the building