Household Energy Disaggregation based on Difference Hidden Markov Model

Ali Hemmatifar (ahemmati@stanford.edu) Manohar Mogadali (manoharm@stanford.edu)

***Abstract*—We aim to address the problem of energy disaggregation into individual appliance contribution for a given household. We do this based on a Machine Learning approach that involves modeling each appliance's load contribution to be a Markov chain. Once the Hidden Markov Model (HMM) is trained, we can get the parameters such as the number of states and their prior probabilities, transition matrix and the emission characteristics for each appliance (each state is modeled as a Gaussian distribution). The disaggregation of the total load is modeled as a difference Hidden Markov Model (difference HMM), based on HMMs for each of the appliance learned previously. Finally, the most probable set of states for all the appliances is obtained using a custom Viterbi algorithm. This approach to energy disaggregation works well for fixed state appliances, as can be seen in the results.**

Key Words: Energy disaggregation, Markov Chain, HMM, Difference HMM, Viterbi algorithm

I. Introduction

Exponential increase in energy demands makes the energy conservation one of the biggest challenges of our time. Any effort in minimizing the energy wastage will then have direct impact on our life, both economically and environmentally. It is estimated that, on average, residential and commercial buildings consume as high as a third of total energy generation[7]. Recent studies[1], [7] suggest that household-specific energy monitoring and providing direct feedback (as opposed to indirect feedbacks such as monthly bills, etc.) can help reducing the energy consumption of residential buildings by about 10%. This could also be used to provide important information to utilities and lawmakers who shape energy policy. Energy disaggregation has been an active area of research in the recent past. [3]use sparse coding for even disaggregated load from monthly demands. Non-Intrusive Appliance Load Monitoring (NIALM) techniques are among the promising monitor-feedback systems. NIALM is a method of evaluating total current/voltage (or power) readings in order to deduce appliance-level energy consumption patterns. To this end, NIALM monitors the total load and identifies the certain appliance "signatures". These signatures correspond to certain (hidden) states of the appliances and can be used to disaggregate the total load to its individual components. Energy disaggregation, powered by various machine learning algorithms, has been an active area of research[2], [3], [4], [7]. Among all, supervised algorithms show better performance compared to unsupervised [3] algorithms, due to a prior knowledge on individual appliance energy states and signatures. In this project, we use a mixture of supervised learning algorithms for energy disaggregation. We first extract individual appliance energy states by approximating the states as a mixture of Gaussians. We then use Hidden Markov Model (HMM) with Gaussian outputs (or emissions) to learn energy signatures and transitions between states for each individual appliance. We then use Difference HMM (DHMM) algorithm developed by [6]along with a custom Viterbi algorithm to approximate the most probable sequence of hidden states for each appliance from total power consumption data. Using our model, we are equipped with a method to provide energy saving recommendations to

consumers. Energy consumption segmentation into 'classes' of appliances has been studied. These methods though being both stochastically advanced and accurate, do not help in providing valuable recommendations to a customer. We believe that by comparing with other households having similar energy patterns, one can provide more relevant recommendations and have a greater chance of behavioral change. This is a novel feature that has not been researched much on, according to our knowledge and is central to our recommendation scheme.

II. Appliance Modeling

In this section, we describe the procedure of extracting features of energy consumption of appliances. We use publicly-available REDD dataset which contains both aggregate and individual appliance power data of 6 households. We used the data from first home (measured at every 2 to 5 sec) for our appliance-level supervised learning. We identified the major energy consuming appliances (total of 7) and extracted their power curves. To this end, for each appliance, we "chopped" the power usage for period of times in which the appliance was on. This will provide a multi-observation sequence which can potentially result in more accurate training. We then used a histogram analysis to identify number of states of each appliance and provide reasonable initial guesses to the training algorithm. Figure 1(a) shows the actual power reading of dishwasher as well as its corresponding energy histogram. This analysis and all the following algorithms are implemented in MATLAB. As shown in Figure 1(a), we created an energy histogram of each appliance (normalized frequency of energy vs. energy level) to identify the most frequent energy levels and the standard deviation corresponding to those states. This is a crude yet essential step for us in appliance modeling, as it provides reasonable initial values for the next step in appliance modeling. These results are then fed into a HMM as initial values. Since the power readings are continuous, we approximate the output of HMM as multivariate Gaussian (with each state corresponding to a single Gaussian). For this version of HMM, we used the algorithm formulated by [6]. We briefly introduce this model here. For emission matrix ϕ, we have $\phi_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$. The estimation formulae for the parameters of this model can be shown to be of the form [6] $\mu_i = \frac{\sum_{t=1}^{T} \gamma_t(i) x_t}{\sum_{t=1}^{T} \gamma_t(i)}$ and $\sigma_i^2 = \frac{\sum_{t=1}^{T} \gamma_t(i)(x_t - \mu_i)^2}{\sum_{t=1}^{T} \gamma_t(i)}$ where x is the observation and μ_i and σ_i^2 are respectively mean and variance of i-th state. $\gamma_t(i)$ is the probability of being in state i at time t and is defined as

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{n} \alpha_t(i)\beta_t(i)}$$

where $\alpha_t(i)$ and $\beta_t(i)$ are backward and forward variables. The other parameters, such as prior probabilities and transmission matrix, are similar to the case of standard HMM. The result of Gaussian emission HMM in the case of dishwasher is shown in Figure 1(b). This appliance has four states and so the graph representation has four nodes. In Figure 1(c), we show the most probable state sequence (output of Viterbi algorithm) for the data shown in Fig. 1(a). Result show a reasonable agreement between predicted and measured power curve. We similarly train the rest of appliances use a difference Markov chain to disaggregation.

III. Disaggregation

Once all the appliances are modeled, the task of disaggregation is addressed. We use a difference Hidden Markov Model for this. For the disaggregation of total load, we have two input vectors- $\boldsymbol{x}$ (time series of the total load curve) and $\mathbf{y}$(time series of difference in $\mathbf{x}$ for consecutive time states), i.e., such that $y_t = x_t - x_{t-1}$ (hence this model is referred to as a difference HMM). This is done because $\mathbf{y}$ is a more important feature than the total load itself since any change of state is an appliance results in non-zero y_t values, and zero y_t values for steady operation. Figure 2 shows the schematic of a difference HMM where $\mathbf{x}$ and $\mathbf{y}$ are defined

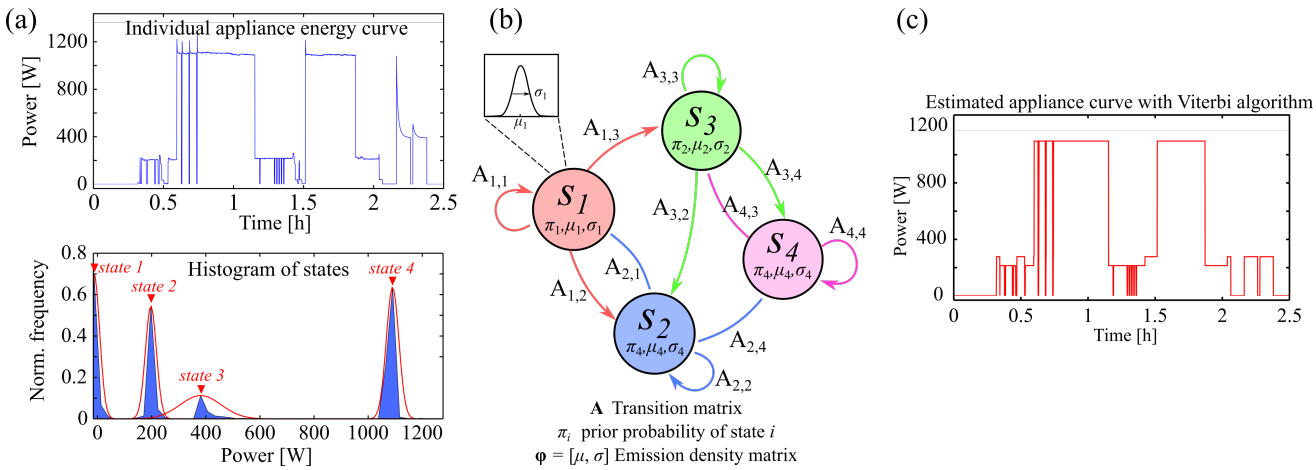

Figure 1. (a) Measured power draw and histogram of states for dishwasher, (b)State transition model, (c) Predicted state sequence via Viterbi algorithm

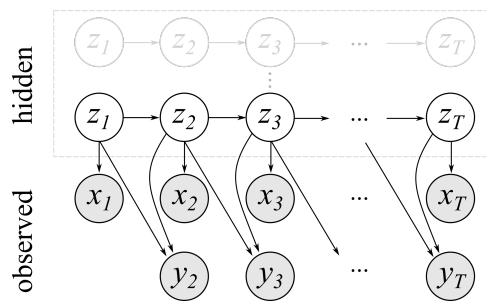

Figure 2. Schematic of difference HMM

above and $\mathbf{z}$ is the vector of appliance states at each time step.

For each appliance n, the model parameters can be defined as $\theta^{(\mathbf{n})} = \{\mathbf{\Pi}^{(n)}, \mathbf{A}^{(\mathbf{n})}, \phi^{(\mathbf{n})}\}$, respectively corresponding to the probability of an appliance's initial state, the transition probabilities between states and the probability that an observation was generated by an appliance state.[5] The probability of an appliance's starting state at $t = 1$ is represented by the prior distribution, hence:

$$p(z_1 = k) = \Pi_k \tag{1}$$

The transition probabilities from state i at $t-1$ to state j at t are represented by the transition matrix A such that:

$$p(z_t = j | z_{t-1} = i) = A_{i,j} \tag{2}$$

We assume that each appliance has a Gaussian distributed power demand:

$$p(w_{z_t} | z_t, \phi) = \mathcal{N}(\mu_{z_t}, \sigma^2_{z_t}) \tag{3}$$

where w_{z_t} is the actual power draw for the appliance. Then, y_t can be modeled as a Gaussian distribution which is a difference of 2 Gaussian distributions:

$$y_t | z_t, z_{t-1}, \phi \sim \mathcal{N}(\mu_{z_t} - \mu_{z_{t-1}}, \sigma^2_{z_t} + \sigma^2_{z_{t-1}}) \tag{4}$$

where $\phi_k = \{\mu_k, \sigma^2_k\}$, and μ_k and σ^2_k are the mean and variance of the Gaussian distribution describing this appliance's power draw in state k. So far, change in state of an appliance does not ensure that the appliance was capable of being in the initial state (or even if the appliance was ON). We can do this by imposing a constraint on the total power demand x_t, as a higher limit for the appliance power draw z_t:

$$P(w_{z_t} \leq x_t | z_t, \phi) = \int_{-\infty}^{x_t} \mathcal{N}(\mu_{z_t}, \sigma^2_{z_t}) dw$$
$$= \frac{1}{2}\left[1 + erf\left(\frac{x_t - \mu_{z_t}}{\sigma_{z_t}\sqrt{2}}\right)\right] \tag{5}$$

We further need another constraint which ensures a change of state is not predicted based

on noise in the observed total load x. In order to achieve this, we filter out the change of states when the joint probability is below a certain threshold(C) specific to each appliance: $t \in S$ if

$$max_{z_{t-1},z_t}(p(x_t, z_{t-1}, z_t|\theta)) \geq C \qquad (6)$$

Combining equations 1 through 6 mentioned above, we end with a joint probability:

$$p(\mathbf{x}, \mathbf{y}, \mathbf{z}|\mathbf{\Theta}) = p(z_1|\pi) \prod_{t=2}^{T} p(z_t|z_{t-1}, \mathbf{A})$$

$$\prod_{t=1}^{T} P(w_{z_t} \leq x_t|z_t, \phi) \prod_{t \in S} p(y_t|z_t, z_{t-1}, \phi) \qquad (7)$$

Using equation 7, we can figure out the most probable state for any appliance at a given time t, if we know its state at $t-1$. We still do not have a unique state change (a non zero y_t could predict change of state for more than one appliance). In order to overcome this, we employ the Viterbi algorithm, which is a dynamic programming algorithm that returns the most probable change of state across all appliances. For running the custom Viterbi algorithm, we make the assumption that there can be only one state change across all appliances. This is a fair assumption owing to high sample rates for the data used (2 - 5 seconds). Only the most likely change of state (as predicted by the Viterbi algorithm) is accepted and all others are discarded at each time step. The results of this algorithm are discussed in the next section.

4 Results and Discussion

In Figure 3, we show the total load curve of a household disaggregated into 4 appliance loads - an oven, a dryer, a microwve and a refrigerator. The total power demand curve is taken from one of the households in the REDD dataset. Table 1 shows the energy usage (which is the area under the Power-time curve) for each appliance. We observe that the L1 norm

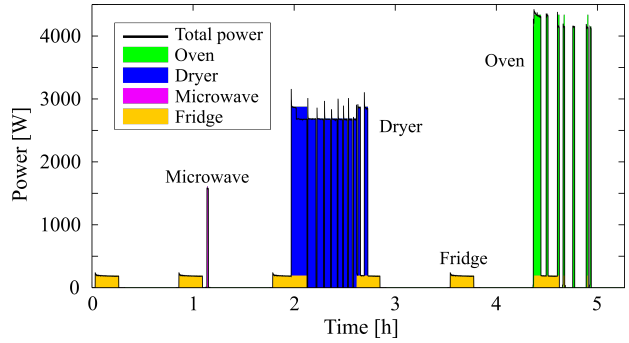

Figure 3. Energy disaggregation into four different appliances (oven, dryer, microwave and refrigerator)

	Actual Energy (%)	Estimated Energy(%)
Dryer	60.84	61.47
Oven	26.97	27.09
Refrigerator	11.54	10.58
Microwave	0.65	0.86

Table I
Actual and estimated total energy consumption for 4 appliances (oven, dryer, microwave and refrigerator)

of the error for this example is 2.58% of the total load curve. The results show that we have successfully disaggregated the total load for the building into its appliance components.

5 Applications and Scope

One application of the disaggregated data is to provide energy saving recommendations to individual households. This could be from identifying inefficient appliances, better usage recommendations such as using the washer-dryer during afternoon, when the electricity price is low, etc. One major feature required in a recommendation system is that it should be relevant to the consumer. In order to address this, we cluster the users into 5 different clusters based on their energy habits using k-means++ algorithm. We observe that k = 5 gives the best results. Individual household data is obtained from PecanStreet data, which provides 600 homes' load curves. The results are presented in Figure 4. The clustering of users makes comparison within the same energy cluster far more meaningful than when done in a

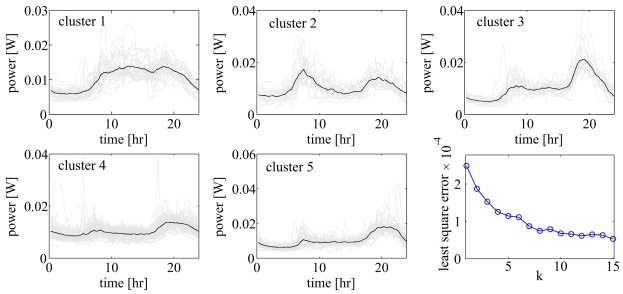

Figure 4. Clustered households via k-means ++ clustering(k = 5), Least Square error for various k values

general setting. Any inaccuracies that occur in the disaggregation algorithm can be treated as an online learning problem within each cluster to increase the accuracy.

6 Conclusion and Future Work

We notice that the difference HMM modeling approach to energy disaggregation based on HMM appliance models has a good prediction rate in the case of fixed state appliances. In our example, we have an L1 error norm of 2.58% of the total load. It should be noted the power surge that typically occurs when appliances start from the OFF state is not explained using the current model. Further, infinite state devices (such as dimmer lamps) cannot be modeled accurately using a finite state HMM model and hence, their contribution is not disaggregated accurately. Further, we state a feasible application for the work carried out, in the form of a novel energy saving recommendation provider by comparing households with similar energy patterns.

References

[1] K Carrie Armel, Abhay Gupta, Gireesh Shrimali, and Adrian Albert. Is disaggregation the holy grail of energy efficiency? the case of electricity. *Energy Policy*, 52:213–234, 2013.

[2] D Kelly and W Knottenbelt. Disaggregating smart meter readings using device signatures. *Imperial Computing Science MSc Individual Project*, 2011.

[3] J Zico Kolter, Siddharth Batra, and Andrew Y Ng. Energy disaggregation via discriminative sparse coding. In *Advances in Neural Information Processing Systems*, pages 1153–1161, 2010.

[4] J Zico Kolter and Tommi Jaakkola. Approximate inference in additive factorial hmms with application to energy disaggregation. In *International conference on artificial intelligence and statistics*, pages 1472–1482, 2012.

[5] Oliver Parson, Siddhartha Ghosh, Mark Weal, and Alex Rogers. Non-intrusive load monitoring using prior models of general appliance types. 2012.

[6] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[7] Ahmed Zoha, Alexander Gluhak, Muhammad Ali Imran, and Sutharshan Rajasegarar. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. *Sensors*, 12(12):16838–16866, 2012.