Stop! Exploring Bayesian Surprise to Better Train NILM

Richard Jones
School of Engineering Science
Simon Fraser University, Canada
rtj4@sfu.ca

Christoph Klemenjak
Institute of Networked and Embedded Systems
University of Klagenfurt, Austria
klemenjak@ieee.org

Stephen Makonin
School of Engineering Science
Simon Fraser University, Canada
smakonin@sfu.ca

Ivan V. Bajić
School of Engineering Science
Simon Fraser University, Canada
ibajic@ensc.sfu.ca

ABSTRACT

In Non-Intrusive Load Monitoring (NILM), as in many other machine learning problems, significant computational resources and time are spent training models using as much data as possible. This is perhaps driven by the preconception that more data leads to more accurate models and, eventually, better performing algorithms. When has enough prior training been done? When has a NILM algorithm encountered new, unseen data? This work applies the notion of Bayesian surprise to answer these important questions for both, supervised and unsupervised algorithms. We compare the performance of several NILM algorithms to establish a suggested threshold on two combined measures of surprise: postdictive surprise and transitional surprise. We validate the use of transitional surprise by exploring the performance of a particular Hidden Markov Model as a function of surprise threshold. Finally, we explore the use of a surprise threshold as a regularization technique to avoid overfitting in cross-house performance. We provide preliminary insights and clear evidence showing a point of diminishing returns for model performance with respect to dataset size, which can have implications for future model development, dataset acquisition, as well as aiding in model flexibility during deployment.

CCS CONCEPTS

• **Hardware → Smart grid**; • **Computing methodologies → Machine learning**.

KEYWORDS

NILM, Bayesian surprise, overfitting, training, datasets

ACM Reference Format:
Richard Jones, Christoph Klemenjak, Stephen Makonin, and Ivan V. Bajić. 2020. Stop! Exploring Bayesian Surprise to Better Train NILM. In *The 5th International Workshop on Non-Intrusive Load Monitoring (NILM '20), November 18, 2020, Virtual Event, Japan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3427771.3429388

NILM '20, November 18, 2020, Virtual Event, Japan
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8191-8/20/11…$15.00
https://doi.org/10.1145/3427771.3429388

1 INTRODUCTION

Non-Intrusive Load Monitoring (NILM), often referred to as load disaggregation, dates back to the seminal work presented in [11]. In a nutshell, NILM describes the problem of identifying present electrical appliances within a time series consisting of a sequence of (power) measurements taken at a central point in the distribution grid of a building. In recent years, more and more energy datasets have emerged (e.g., [16, 21, 23, 24] to name a few), which can vary considerably in terms of complexity, methodology, appliance characteristics and usage patterns, setting, etc. (e.g., see [17, 26]). With some datasets spanning several years of collection, considerable time and computational resources are spent in training new models. Newer approaches to NILM increasingly adopt deep learning methods (e.g., [10, 19]), which can involve millions of tunable parameters, not to forget the often arduous process of hyperparameter tuning. It stands to reason, then, that effectively isolating the most important segments of a dataset relative to a model could improve time-to-deployment as well potentially regularize against overfitting.

We relate the concept of Bayesian surprise [1, 13] to NILM by modeling appliance activations in a non-parametric Gaussian mixture model and introducing *postdictive surprise.* Additionally, we introduce the concept of *transitional surprise* by simply modeling the relationships between appliance states in a Markovian sense. Our preliminary results show that:

(1) the diminishing returns of increased amounts of similar data,
(2) the potential "model-agnostic" regularization effect of training data truncation, and
(3) the usefulness of transitional surprise to (crudely) approximate system dynamics.

2 RELATED WORK

Itti and Baldi [1, 13] define *Bayesian surprise* to be a measure of dissimilarity to assess the effect of data D on the belief distributions of an observer. More precisely, Bayesian surprise is defined as the dissimilarity (or divergence) between the prior and the posterior distributions over a set of possible models $\mathcal{M}$. Itti and Baldi's interpretation of Bayesian Surprise has found application in various forms: de-biasing of thematic maps in [6], automatic detection of landmarks in computer vision [27], detection of salient acoustic events [28], identification of calcifications in mammogram images [7], and to determine suitable thresholds for extreme value models [20].

In [18], Bayesian updating of an agent's beliefs was grouped into two general categories. First, Bayesian surprise is the term given to the change in beliefs over latent variables; the unobservable quantities inferred through observations. Second, *postdictive* surprise refers to the divergence between the prior and posterior predictive distributions, quantifying the surprise over observable quantities. In [8], the concept of confidence-corrected surprise is developed, in which the degree of commitment to a particular generative model influences the extent to which observations update an agent's beliefs. However, given that the intent of the present work is to develop a "model-agnostic" formulation for NILM datasets, surprise in the present work is restricted to a fixed model (i.e., $|\mathcal{M}| = 1$).

In [12], the authors propose a Bayesian surprise metric to differentiate between useful information and redundant observations during online learning of mixtures of Gaussians. The main motivation behind this measure is to prevent outliers from significantly changing the model parameters as well as restrict redundant samples from over-specifying component parameters, which would lead to overfitting. In the context of online learning, our work can be considered somewhat of an extension of [12] to non-parametric methods, rather than storing outliers and instantiating new components based on Gaussian Mean Shifting. However, the main focus of the present work is to use GMMs to explore the point at which the data is no longer surprising with respect to improving the performance of any model. By contrast, [12] uses the concept of Bayesian surprise within a GMM to optimize its own clustering performance.

3 SURPRISE METHODOLOGY FOR NILM

A natural approach to characterize the novelty of incoming data is to examine the change in the signal and compare it to the changes so far observed. In other words, clustering on the first-differences of the signal permits an intuitive notion of surprising data: appliance events not yet seen. Following the basic appliance characterizations in [11], simple ON-OFF or multi-state appliances can have their initial activations modelled as Gaussian around some mean value.

However, transient characteristics of appliances, such as the consumption spike at the start of a fridge's condenser cycle, can result in a highly varying activation value. Moreover, the consistency of these initial activations are dependent on sampling frequency. We consequently preprocess the data using a fast, steady-state block-filter developed in [14]. This filter imputes the mean value between change-points identified using an adaptive threshold on the raw power and first-differences in the signal. This steady-state power for individual appliance states is far more amenable to Gaussian modelling given its improved consistency. A Gaussian mixture model (GMM) can be learned using the discrete changes in the aggregate as in [12], but in contrast to Gausian Mean Shifting to detect new components, we extend GMMs to the Bayesian non-parametric regime, which relaxes the constraint of finite parameterizations.

In non-parametric GMMs, the unbounded number of mixture weights π_k are generated according to a Dirichlet Process prior, which ensures $\sum_{k=1}^{\infty} \pi_k = 1$. The stick-breaking representation introduced in [9] gives an intuitive generative process for these objects. Inference in these models can be stochastic, for example by using Markov Chain Monte Carlo methods such as Gibbs sampling. Alternatively, a variational approximation to the desired posterior can be posited, the parameters of which are by design amenable to optimization relative to the evidence lower bound (ELBO). The variational approximation first proposed in [4] for these models takes the form:

$$q(\nu, \theta, \mathbf{z}) = \prod_{k=1}^{K-1} q_{\gamma_k}(\nu_k) \prod_{k=1}^{K} q_{\tau_k}(\theta_k) \prod_{n=1}^{N} q_{\phi_n}(z_n), \tag{1}$$

where $\{\gamma, \tau, \phi\}$ are the variational parameters subject to coordinate ascent optimization. q_γ are beta distributions parameterized by the individual stick lengths, ν_k. q_τ are in our case Gaussians parameterized by $\theta_k = \{\mu_k, \Sigma_k\}$, although extension to general exponential families is possible. q_{ϕ_n} are multinomial, parameterized by indicator variables z_n, which denote the component to which the observation x_n is assigned. To speed up inference, a truncation on the maximum number of possible states is imposed on the variational approximation, similar to truncation in methods such as blocked Gibbs sampling [4]. This value, K, is itself a variational parameter which can be fixed or optimized with respect to the ELBO. K was fixed in our work to 30 unique components. Under this approximation, the resulting posterior predictive distribution needed for computing postdictive surprise can be neatly factored as expectations with respect to the variational distribution:

$$p(x_{N+1}|x_1, ..., x_N, \alpha, G_0) \approx \sum_{k=1}^{K} \mathbb{E}_q\big[\pi_k\big]\mathbb{E}_q\big[p(x_{N+1}|\theta_k)\big] \tag{2}$$

For many machine learning algorithms, decay in the postdictive surprise might be sufficient to demarcate useful data from superfluous data during training. However, it is often the case that temporal relationships between appliance states are learned and contribute to inference. Such methods would include Hidden Markov Models (HMMs) and their many extensions, more recent deep learning techniques such as those based on Recurrent Neural Networks, and many more. In the interest of simplicity, we restrict the notion of "transitional surprise" to the Markovian sense. That is, we treat the state sequence as a Markov chain, such that the current state of the system is determined only by the state before it. For a system of K appliance states, this transitional surprise constitutes comparing the rows of the $K \times K$ transition matrix. This approximation to the dynamics is clearly crude, but even weak convergence of the transition matrix to some stationary form can prove useful.

To summarize, for each sliding window of w events, preceded by N events, we compute the (approximate) postdictive surprise as:

$$S_o = d\big[p(x_{N+1}|x_{1:N}, \alpha, G_0) \;||\; p(x_{N+w+1}|x_{1:(N+w)}, \alpha^*, G_0)\big], \tag{3}$$

where d is some divergence metric (usually Kullback-Leibler divergence), and α^* is the posterior update for the concentration parameter if a prior was placed on it. Over the same window of w events, we compute the transitional surprise over the truncated maximum number of states K as:

$$S_t = \sum_{j=0}^{w} \sum_{k=1}^{K} d\big[T_k(z_{1:N+i})||T_k(z_{1:N+i+1})\big], \tag{4}$$

Table 1: REFIT House 3 Performance expressed as MAE for Decreasing Surprise Thresholds

(S_o, S_t)	% of Tr. Set	DAE	RNN	S2P	S2S	GRU
(0.05, 0.6)	9.2	40.7	36.1	37.9	30.7	32.9
(0.02, 0.5)	18.3	36.8	30.1	24.2	24.4	25.9
(0.01, 0.5)	26.1	31.9	30.0	25.2	25.1	27.7
(0.005, 0.3)	54.1	33.4	29.9	23.6	23.8	27.4

where at time t, $T_{j,k} = p(z_{t+1} = k | z_t = j)$. The notation $T_k(z_{1:N+i})$ denotes the transition row built using event indicators z for observations $1, 2, ..., N + i$.

In order to simplify the concept of a surprise threshold under which data is no longer considered surprising, S_o and S_t are normalized according to their maximum values. Since the initial value of the above divergences can certainly be exceeded as observations are made, the maxima were updated and preceding surprise values were re-normalized to the revised maxima. The postdictive and transitional surprise values can therefore be interpreted as the fraction of the maximum observed surprise, rather than the actual value of equations 3 and 4. Since in an online setting it would be unreasonable to wait indefinitely for surprising windows, we suggest a patience parameter, ρ. In the experiments that follow, we used $\rho = 100$; that is, 100 windows are observed beyond the most recent window exceeding the surprise threshold. If no other windows exceed the threshold, the previously surprising window is returned as the cutoff point.

4 EXPERIMENTS

To explore the usefulness of a surprise threshold, we made use of NILMTK, an open-source toolkit developed for NILM research that includes implementations of some benchmark algorithms [2, 3]. In this work, we explore the use of Denoising Autoencoders (DAE) [5], LSTM-based Recurrent Neural Networks (RNN) [15], Windowed Gated Recurrent Unit-based RNNs (WindowGRU) [19], Sequence-to-Sequence autoencoders (Seq2Seq) [15], and Sequence-to-Point convolutional networks (Seq2Point) [29].

To establish a relationship between algorithm performance and the proposed surprise metrics, three houses from the REFIT dataset [24] were selected for study using the above disaggregation methods. The included appliances in these experiments were the dishwasher, the washing machine, the refrigerator, the kettle, and the toaster. The Mean-Absolute Error (MAE) was used as a performance metric. For each house, the available data was split into a training set and test set by a 90%/10% split. 15% of the training set was reserved for validation. The surprise metric was computed on the remaining training data, such that each algorithm was training and validating on the same data. Each algorithm was trained over 15 epochs using Adam optimization with a batch size of 1024 samples. For a given house, each algorithm had its random seed fixed across surprise-based training set reductions, removing initialization variability from their appliance-averaged performance. We relied on the preprocessing functions, such as data normalization, part of NILMTK.

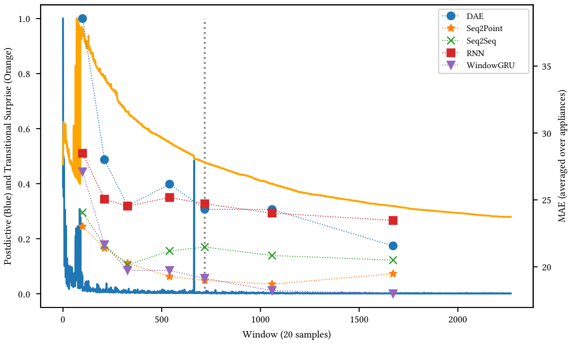

Figure 1: Appliance-averaged MAE performance, REFIT House 2

Table 2: REFIT House 5 Performance expressed as MAE for Decreasing Surprise Thresholds

(S_o, S_t)	% of Tr. Set	DAE	RNN	S2P	S2S	GRU
(0.05, 0.6)	12.8	34.9	32.7	27.8	26.3	31.1
(0.02, 0.5)	21.8	35.9	29.9	26.1	25.7	28.5
(0.01, 0.5)	26.4	32.2	29.9	25.0	25.1	29.0
(0.005, 0.3)	80.2	29.2	28.7	23.4	24.0	26.0

Figure 1, as well as Tables 1 and 2, show the behaviour of the MAE for the average appliance across the benchmark methods for houses 2, 3, and 5, respectively. The postdictive and transitional surprise was computed using Jensen-Shannon divergence, a symmetric extension of the KL-divergence.

Although no sharp transition exists between an optimally and sub-optimally sized training set, the behaviour of these algorithms' MAE in the three REFIT houses suggest that performance can indeed stagnate. Additional similar data, especially in houses 2 and 3, seem unlikely to appreciably improve performance. An example surprise threshold is shown in Figure 1 as a dotted grey line, indicating an approximate point where performance began to plateau. This cutoff was chosen as a joint threshold over postdictive and transitional surprise, defined by:

$$S_o(w : w + \rho) \le 0.01 \;\&\; S_t(w : w + \rho) \;\le 0.05, \tag{5}$$

where again, w is the window size and ρ is the patience parameter. We used this threshold to study further the potential regularizing effect of surprise-based training cutoff, although significant exploration with other available datasets is needed to further narrow down acceptable threshold values.

In [25], disaggregation performance on unseen homes in the same dataset as well as different datasets were examined. By their choice of architectures, the authors restricted the number of tunable parameters relative to the existing literature. They also made use of early stopping with an aggressive patience parameter to terminate training. With these complexity and temporal regularization methods, they showed intra- and inter-dataset transferability

Table 3: REFIT Cross-house ($3 \longrightarrow 5$) MAE for full and cutoff training in Watts

Benchmark Method	Full Training	Cutoff Training
WindowGRU	37.83	**33.03** ↓
DAE	34.78	**33.00** ↓
RNN	32.54	**30.62** ↓
Seq2Seq	**27.17**	29.43 ↑
Seq2Point	26.85	**26.74** ↓

with minimal performance losses relative to their chosen baseline. Nevertheless, these methods still make use of all available training data. Bayesian surprise metrics provide an attractive alternative/supplement to early stopping, which by contrast truncate the training set entirely. We examined the MAE performance of each algorithm when trained on all data of REFIT house 3 and the surprise-based subset determined by the joint threshold in equation 5. Table 3 shows the appliance-averaged MAE performance of each benchmark method when tested on REFIT house 5. All but one method showed improved cross-house transferability with a restricted training set, giving some substance to the claim that truncating the training set may provide regularization against over-fitting.

Finally, to illustrate the usefulness of including the concept of transitional surprise, we explored the performance of the super-state Hidden Markov Model introduced in [22]. Clearly, a Markovian model should suffice to show whether our Markovian notion of transitional surprise is useful. We used house 1 from the Rainforest Automation Energy (RAE) dataset [23], which consists of two blocks: a 9 day block beginning on February 7, 2016, and a 63 day block beginning March 6, 2016. Block 1 was used as the test set, and block 2 (and its surprise-based subset) was used for training the models. The seven appliances used for training were the clothes washer and dryer, refrigerator, dishwasher, furnace/hot water unit, and the heat pump.

Figure 2 shows the Van Rijsbergen's effectiveness measure (defined simply as 1− F1-score) as a function of cutoff point during training. This measure decays slightly faster than that of the transitional surprise, but significantly after the postdictive surprise had converged. This lends credence to the claim that postdictive surprise is an unreliable metric for terminating training in the general case. The difference in decay rate between transitional surprise and the effectiveness measure is understandable given that the SSHMM by definition encodes the Markovian dynamics between super-states of the user's home. The super-state of the home at a given instant in time can be thought of as the complete description of the home, denoting the operational mode of each appliance in the house. Each instant in time increments the underlying transition distributions between super-states of the home, rather than individual appliance states. This will in general encode the state dynamics more efficiently since there is more information used per time-step. Nevertheless, the basic notion of transitional surprise introduced here allows a useful overestimate of the learning rate of the system dynamics.

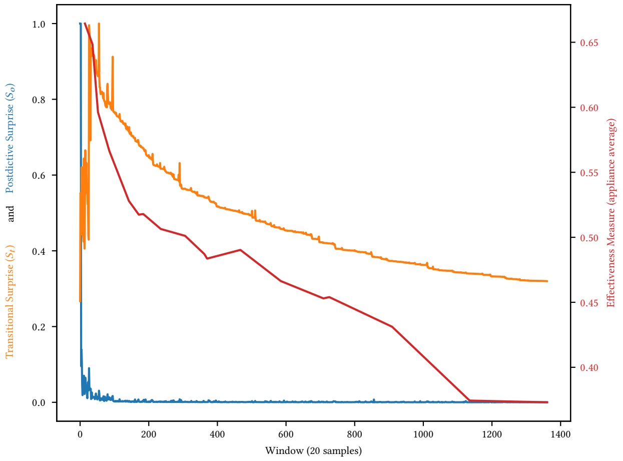

Figure 2: Effectiveness measure (1− F1-score) averaged over 7 appliances as a function of surprise-based training cutoff

5 CONCLUSIONS

Ultimately, the concept of surprise involves comparison over distributions as they are updated given new observations. The most useful such distributions are unavoidably model-specific. Nevertheless, there are features intrinsic to the data itself that could be used to predict the usefulness of more data in a model-agnostic way. This work explored a postdictive surprise defined over the likelihood of a non-parametric GMM. Furthermore, we explored a transitional surprise defined in a Markovian sense, which was described by the transitional relationships between latent states as determined by the state assignments of the GMM. This crude approximation to the system dynamics was shown to be useful relative to a strictly postdictive notion of surprise, at least in an HMM-based application. An approximate joint threshold was determined by examining the MAE performance of five benchmark methods supported by NILMTK over three REFIT homes. This threshold was used to explore the potential regularizing effect of a surprise-based training cutoff.

Although similar in motivation to early stopping, surprise-based truncation of data is fundamentally different in that the dataset itself is restricted, rather than the time spent learning the data. The two methods can thus be used together to protect against over-fitting and aid in the transferability of learned parameters. Truncating the training set using surprise-based methods allows a significant reduction in research costs, both in terms of computational time spent training and research time spent trying to optimize what may prove to be fruitless methods. Finally, postdictive surprise using non-parametric mixture models naturally extends to online settings, where deployed NILM algorithms quickly become obsolete without the flexibility to adapt to new appliances or appliance replacements.

REFERENCES

[1] Pierre Baldi and Laurent Itti. 2010. Of bits and wows: A Bayesian theory of surprise with applications to attention. *Neural Networks* 23, 5 (2010), 649–666.

[2] Nipun Batra, Jack Kelly, Oliver Parson, Haimonti Dutta, William Knottenbelt, Alex Rogers, Amarjeet Singh, and Mani Srivastava. 2014. NILMTK: An Open Source Toolkit for Non-Intrusive Load Monitoring. In *5th ACM International Conference on Future Energy Systems (e-Energy).*

[3] Nipun Batra, Rithwik Kukunuri, Ayush Pandey, Raktim Malakar, Rajat Kumar, Odysseas Krystalakos, Mingjun Zhong, Paulo Meira, and Oliver Parson. 2019. Towards Reproducible State-of-the-Art Energy Disaggregation. In *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys).*

[4] David M. Blei and Michael I. Jordan. 2005. Variational inference for Dirichlet process mixtures. *Bayesian Analysis* 1 (2005), 121–144.

[5] Roberto Bonfigli, Andrea Felicetti, Emanuele Principi, Marco Fagiani, Stefano Squartini, and Francesco Piazza. 2018. Denoising autoencoders for non-intrusive load monitoring: improvements and comparative evaluation. *Energy and Buildings* 158 (2018), 1461–1474.

[6] Michael Correll and Jeffrey Heer. 2016. Surprise! Bayesian weighting for de-biasing thematic maps. *IEEE transactions on visualization and computer graphics* 23, 1 (2016), 651–660.

[7] Inês Domingues and Jaime S Cardoso. 2014. Using Bayesian surprise to detect calcifications in mammogram images. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society.* IEEE, 1091–1094.

[8] Mohammadjavad Faraji, Kerstin Preuschoff, and Wulfram Gerstner. 2018. Balancing New against Old Information: The Role of Puzzlement Surprise in Learning. *Neural Comput.* 30, 1 (Jan. 2018), 34–83. https://doi.org/10.1162/neco_a_01025

[9] Thomas S. Ferguson. 1973. A Bayesian Analysis of Some Nonparametric Problems. *Ann. Statist.* 1, 2 (03 1973), 209–230. https://doi.org/10.1214/aos/1176342360

[10] A. Harell, S. Makonin, and I. V. Bajić. 2019. Wavenilm: A Causal Neural Network for Power Disaggregation from the Complex Power Signal. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 8335–8339.

[11] George W. Hart. 1985. *Prototype Nonintrusive Appliance Load Monitor.* Technical Report. MIT Energy Laboratory and Electric Power Research Institute.

[12] Erion Hasanbelliu, Kittipat Kampa, Jose C Principe, and James T Cobb. 2012. Online learning using a Bayesian surprise metric. In *The 2012 international joint conference on neural networks (IJCNN).* IEEE, 1–8.

[13] Laurent Itti and Pierre F Baldi. 2006. Bayesian surprise attracts human attention. In *Advances in neural information processing systems.* 547–554.

[14] R. Jones, A. Rodriguez-Silva, and S. Makonin. 2020. Increasing the Accuracy and Speed of Universal Non-Intrusive Load Monitoring (UNILM) Using a Novel Real- Time Steady-State Block Filter. In *2020 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT).* 1–5.

[15] Jack Kelly and William Knottenbelt. 2015. Neural NILM: Deep Neural Networks Applied to Energy Disaggregation. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments (BuildSys).*

[16] Christoph Klemenjak, Christoph Kovatsch, Manuel Herold, and Wilfried Elmenreich. 2020. A synthetic energy dataset for non-intrusive load monitoring in households. *Scientific Data* 7, 1 (2020), 1–17.

[17] Christoph Klemenjak, Stephen Makonin, and Wilfried Elmenreich. 2020. Towards comparability in non-intrusive load monitoring: on data and performance evaluation. In *2020 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT).* IEEE, 1–5.

[18] Antonio Kolossa, Bruno Kopp, and Tim Fingscheidt. 2015. A Computational Analysis of the Neural Bases of Bayesian Inference. *NeuroImage* 106 (02 2015), 222–337. https://doi.org/10.1016/j.neuroimage.2014.11.007

[19] Odysseas Krystalakos, Christoforos Nalmpantis, and Dimitris Vrakas. 2018. Sliding Window Approach for Online Energy Disaggregation Using Artificial Neural Networks. In *Proceedings of the 10th Hellenic Conference on Artificial Intelligence (SETN).*

[20] Jeong Lee, Yanan Fan, and Scott A Sisson. 2015. Bayesian threshold selection for extremal models using measures of surprise. *Computational Statistics & Data Analysis* 85 (2015), 84–99.

[21] Stephen Makonin, Bradley Ellert, Ivan V. Bajić, and Fred Popowich. 2016. Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014. *Scientific Data* 3, 1 (2016), 160037. https://doi.org/10.1038/sdata.2016.37

[22] S. Makonin, F. Popowich, I. V. Bajić, B. Gill, and L. Bartram. 2016. Exploiting HMM Sparsity to Perform Online Real-Time Nonintrusive Load Monitoring. *IEEE Transactions on Smart Grid* 7, 6 (2016), 2575–2585.

[23] Stephen Makonin, Z Jane Wang, and Chris Tumpach. 2018. RAE: The rainforest automation energy dataset for smart grid meter data analysis. *data* 3, 1 (2018), 8.

[24] David Murray, Lina Stankovic, and Vladimir Stankovic. 2017. An electrical load measurements dataset of United Kingdom households from a two-year longitudinal study. *Scientific Data* 4, 1 (2017), 160122. https://doi.org/10.1038/sdata.2016.122

[25] D. Murray, L. Stankovic, V. Stankovic, S. Lulic, and S. Sladojevic. 2019. Transferability of Neural Network Approaches for Low-rate Energy Disaggregation. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 8330–8334.

[26] Lucas Pereira and Nuno Nunes. 2018. Performance evaluation in non-intrusive load monitoring: Datasets, metrics, and tools—A review. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery* 8, 6 (2018), e1265.

[27] Ananth Ranganathan and Frank Dellaert. 2009. Bayesian surprise and landmark detection. In *2009 IEEE International Conference on Robotics and Automation.* IEEE, 2017–2023.

[28] Boris Schauerte and Rainer Stiefelhagen. 2013. "Wow!" Bayesian surprise for salient acoustic event detection. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing.* IEEE, 6402–6406.

[29] Chaoyun Zhang, Mingjun Zhong, Zongzuo Wang, Nigel Goddard, and Charles Sutton. 2018. Sequence-to-Point Learning with Neural Networks for Non-Intrusive Load Monitoring. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI).*