 doi:10.20944/preprints201808.0551.v1

Peer-reviewed version available at *Electronics* **2018**, *7*, 235; doi:10.3390/electronics7100235

Article

An Extreme Learning Machine Approach to Effective Energy Disaggregation

Valerio M. Salerno and Graziella Rabbeni

Kore University of Enna; valerio.salerno@unikore.it, graziella.rabbeni@unikore.it

Abstract: Power disaggregation aims at determining the appliance-by-appliance electricity consumption leveraging upon a single meter only, which measures the entire power demand. Data-driven procedures based on Factorial Hidden Markov Models have been proven remarkable results on energy disaggregation. Nevertheless, those procedures have various weaknesses: there is a scalability problem as the number of devices to observe raises and the algorithmic complexity of the inference step is severe. DNN architectures, such as Convolutional Neural Networks, have demonstrated to be a viable solution to deal with FHMMs shortcomings. Nonetheless, there are two significant limitations: a complicated and time-consuming training system based on back-propagation has to be employed to estimates the neural architecture parameters, and large amounts of training data covering as many operation conditions as possible need to be collected to attain top performances. In this work, we aim to overcome those limitations by leveraging upon the unique and useful characteristics of the extreme learning machine technique, which is based on a collection of randomly chosen hidden units and analytically defined output weights. Experiment evaluation has been conducted using the UK-DALE corpus. We find that the suggested approach achieves similar performances to recently proposed ANN-based methods and outperforms FHMMs. Besides, our solution generalises well to unseen houses.

Keywords: Non-intrusive Load Monitoring; Machine Learning; Deep Modeling; Extreme Learning Machine; Data Driven Approach.

1. Introduction

Energy consumption in household spaces increases every year [1], becoming a serious concern of politics across Europe and U.S. due to limited energy resources, and the negative implications on the environment (e.g. CO2 emissions). Indeed, the residential area in both Europe and the U.S. represents the third highest in power consumption, providing almost a quarter of the whole energy used [2, 3]. Moreover, devices, electronics, and lighting are ranked as the second largest group, contributing 34.6% of the residential electricity usage [4]. Unfortunately, up to 39% of the energy used by the domestic sector can be wasted [5]. Many studies have affirmed energy savings if feedback on power consumption information is provided to energy users [6, 7]. Governments and public utilities have reacted to the energy efficiency efforts by supporting smart grids and deploying smart meters. Private corporations perceive chances in the market potentials of power efficiency, and we have thus observed the increase of commercial home power monitoring products available in the market. Stakeholders from public, and private sectors envision smart energy management including the transmission as well as the distribution of power between energy producers, and energy consumers in addition to seamless data exchange on electricity usage and pricing information between generators, and users. That requires connection among devices and servers. Such a requirement belongs to the broader technology realm called the Internet of Things (IoT) in the smart home [8].

Peer-reviewed version available at *Electronics* **2018**, *7*, 235; doi:10.3390/electronics7100235

In the energy consumption literature, the classification of the electrical loads in a house is usually divided into two groups - Intrusive Appliance Load Monitoring and Non-Intrusive Appliance Load Monitoring - relating how much engrave to the expenses and to the simplicity to install the system, or fix it when required. Intrusive Appliance Load Monitoring (IALM, or ILM) is usually classified as the collection of current monitoring techniques where a power meter is connected to each appliance in the household. Therefore, those methods need entering the house, which is rated as intrusive; furthermore, IALM systems are usually costly - all of the home devices need, at least, one power meter each, and challenging to install and configure. Notwithstanding those adverse reasons, ILM techniques provide highly trustable and authentic results. Non-Intrusive Appliance Load Monitoring (NIALM, or NILM), on the contrary, is a way of specifying both the household power usage, and the state of operation of every connected appliance, on the evaluation of the entire load measured by the main power meter in the house. Consequently, NILM leads to a lower cost, since the number of the energy meters can be reduced to just one per house. Nonetheless, NILM 's burden stays in finding effective and efficient solutions to disaggregate the total of power consumption measured at a single point into individual electrical devices power consumption [9-11]. The energy or power consumption for individual electrical appliances can be determined from disaggregated data [12]. That is in contrast to the deployment of one sensor per device in standard intrusive-type of energy monitoring. NILM is worthwhile because it reduces costs, since multiple sensors configurations and installation complexity linked with intrusive load monitoring are avoided [10]. Researchers and engineers prefer the NILM method because of both economic and practical reasons. That means that NILM research is not only focused on theoretical approaches, but also on the deployment of these systems in the real cases. A large-scale deployment favors NILM over ILM because of the lower costs. The only benefit of intrusive methods is the high accuracy in measuring the power consumption of specific appliances, but in many cases, it can be approximated without experiencing any dangerous consequences.

In this work, we are concerned with NILM, also known as energy disaggregation, which can be broadly defined as a set of techniques used to obtain estimates of the electrical power consumption of individual appliances from analyses of voltage and/or current taken at a limited number of positions in the power distribution system of a building. The original concept traces its roots back to George Hart's seminal work at the Massachusetts Institute of Technology (MIT)[9]. In that work, Hart suggested the use of variations in real and reactive power consumption, as measured at the utility meter, to automatically trace the operation of individual appliances in a home. A significant number of studies has then put forth to resolve this problem under the NILM framework, and the interested reader is referred to [10, 11, 14, 15]. NILM approaches initially focused on feature selection and extraction with light emphasis on learning and inference techniques [9, 16]. Progresses in computer science and machine learning methods have driven to innovations in data prediction and disaggregation techniques.

The most popular approach to power disaggregation is based on Factorial Hidden Markov Models (FHMMs) [17]. FHMMs have proven to appropriately perform for the task of load disaggregation due to their ability to include in their learning temporal as well as appliance state transition data. Nonetheless, the complexity of the HMM exponentially grows as the number of target appliances increases, which limits the applicability of this learning approach. Moreover, if any a new device class has to be added, the entire model needs to be retrained from scratch. Alternative existing machine learning algorithms, such as the Support Vector Machine (SVM) [18], k-Nearest Neighbour (k-NN) [19], and Artificial Neural Network (ANN) [20–22], can also have a significant impact on the development of NILM. The principal advantage of using machine learning is that those approached can efficiently and effectively solve very complex classification and regression problems [23, 24] . However, much effort is still required to reduce the error caused by different prediction and disaggregation algorithms to within a satisfactory range, as explained in [13].

Deep neural networks fulfill complicated learning tasks by forming learning machines with deep architectures, which has been proven to have a stronger recognition ability for highly nonlinear patterns compared to shallow networks [25]. Deep learning has been widely studied and applied in various frontier fields and has become the state-of-the-art in speech recognition [26, 27], handwriting

Peer-reviewed version available at Electronics 2018, 7, 235; doi:10.3390/electronics7100235

recognition [28] and image classification [29]. Those recent advancements and applications of deep learning have provided new ideas for energy disaggregation [24]. For example, Kelly and Knottenbelt [21] suggested three different approaches: (i) a solution employing a specific type of recurrent neural network [30], (ii) a solution reducing noise with de-noising auto-encoders and (iii) a regression algorithm predicting the start time, end time and average power demand for each device. Mauch and Yang [31] presented a different architecture of deep recurrent LSTM network in order to test whether this kind of network could overcome the known problems of the previous NILM approaches. Such problems involve (i) disaggregation of various appliance types, (ii) automatic feature extraction from low-frequency data, (iii) generalization of a solution to other buildings and unseen devices, and (iv) estensibility of the method to continuous time and computational tractability. Zhang et al. [32] proposed a deep learning solution for the problem of single-channel blind source separation with application in NILM. The method is called sequence-to-point learning because it uses a window as the input and a single point as the target. The proposed solution is a deep convolutional neural network (CNN) [28], which also learns the signature of the devices in a house.

Although deep neural architectures can achieve excellent results, those models have two main weaknesses: (1) deep models considers the multilayer architectures as a whole that is fine-tuned by several passes of back-propagation (BP) based fine-tuning in order to obtain reasonable learning capabilities – such a training scheme is cumbersome and time-consuming, and (2) huge volumes of training data are needed to achieve top performances, which may limit the deployment of DNN-based solutions in several real-world applications. In this work, we, therefore, aim to overcome those issues by introducing an alternative machine learning NILM framework based on the unique and effective characteristics of the extreme learning machine (ELM) algorithm [33], namely: high-speed training, good generalization, and universal approximation/classification capability. ELMs can play a pivotal role in many machine learning applications, e.g. traffic sign recognition [34], gesture recognition [35], video tracking [35], object classification [36], data representation in big data [37], water distribution and wastewater collection [38], opal grading [39], and adaptive dynamic programming [40]. In [41], the authors have shown that ELMs are also suitable for a wide range of feature mappings, rather than the classical ones. Moreover, to take advantage of multi-layer models, we deploy an energy disaggregation algorithm with hierarchical ELMs (H-ELMs). To test the capability of ELM and H-ELM, we conducted a series of experiments on the standardized UK-DALE dataset [43]. Notably, the amount of training data in this dataset is limited in comparison to that used in speech and image recognition, for example. That permits us to prove that ELMs are indeed a viable solution to energy disaggregation when the amount of training data is limited, which hinders training and generalisation capabilities of state-of-the-art deep models, such as CNN, and LSTM. Indeed, the proposed ELM based algorithms outperform the algorithms based on more traditional back-propagation based artificial neural networks, as demonstrated in the related experimental investigation.

The rest of this paper is organized as follows: Section II introduces the energy disaggregation problem. Section III gives a brief survey of related works. Section IV presents the ELM/H-ELM based speech enhancement algorithms. Section V gives our experimental setup and results. The conclusions from this study are drawn in Section VI.

2. Energy Disaggregation Problem

The purpose of power disaggregation is to break the total power drawn down into its components. In a domestic building, the resultant power is the outcome of the energy consumption of each electrical appliance. Thus, the difficulty consists of identifying how much power each appliance consumes. The superimposition of the power of L appliances in a time period T, can be defined as:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2018 doi:10.20944/preprints201808.0551.v1

Peer-reviewed version available at *Electronics* **2018**, *7*, 235; doi:10.3390/electronics7100235

$$y[n] = \varepsilon[n] + \sum_{i=1}^{L} x_i[n], \qquad n \in \{t_0, t_1, \dots, T\} \tag{1}$$

where $y(n)$ is the aggregate (total) power at time n, $x_i(n)$ is the power of the i^{th} appliance at time n, and $\varepsilon(n)$ is some unwanted noise at time n. Let *Y = (540, 540, 600, 500, 800, 800, 750, 830, 850, 750, 570, 570, 570, 590)* be the sequence of power readings in Watt taken every 20 minutes, that have to be disaggregated. A feasible solution to the energy disaggregation problem is given in Table 1. Interestingly, solution is not unique.

Table 1. A simulated example of the energy disaggregation problem.

Timestamp (20 minutes)	Total consumption	Oven	TV	Dishwasher	Laptop	Others
t_0	540	0	130	390	20	0
t_1	540	0	130	390	20	0
t_2	600	80	0	390	50	80
t_3	500	80	120	300	50	50
t_4	800	450	0	300	50	0
t_5	800	450	0	300	50	0
t_6	750	450	0	300	0	0
t_7	830	450	80	300	0	0
t_8	750	450	0	300	0	0
…	…	…	…	…	…	…

A non-intrusive load monitoring (NILM) system collects energy consumption data from the central meter of a house. It after can assume the consumption of each appliance, present in residence. The NILM framework is made of three necessary steps: (i) data acquisition step relates to how the energy data is collected, which is mostly based on hardware solutions; (ii) the device feature extraction step, and (iii) inference and learning, such as the mathematical model that disaggregates the total power signal into device level signals. In this paper, we focus on the third step only.

In machine learning approaches to power disaggregation, the problem can be addressed as an estimation, a classification, or regression problem. In the next section, a brief overview of related work, which addresses the energy disaggregation task under the machine learning framework, will be given.

3. Related Work

Energy disaggregation is time-dependent by nature. NILM researchers have always imagined models of sequential data and time as potential solutions. Hidden Markov Models (HMMs) have therefore received increased attention from the research community. The first work that we are aware of that uses HMMs is [43], where the authors apply a Factorial HMM (FHMM) to the energy disaggregation task. FHMMs were also explored in [17] and [44]. In [45], the authors apply Additive Factorial HMMs using an unsupervised technique for selecting signal pieces where individual devices are isolated. Parson et al. have also employed HMM-based methods in [46], and [47] where they have fused prior models of general device types (e.g., refrigerators, clothes dryers, etc.) with HMMs. More recently, [48] introduced a method, referred to as Particle Filter-Based Load Disaggregation, where appliances' load signatures and superimposition of them were modeled with HMMs and factorial HMMs, respectively. Inference was carried out by Particle Filtering (PF).

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2018 doi:10.20944/preprints201808.0551.v1

Peer-reviewed version available at *Electronics* **2018**, *7*, 235; doi:10.3390/electronics7100235

Paradiso et al. [49] proposed a new electrical load disaggregation system, which utilizes FHMMs and exploits context-based features. The context data consists of the user presence and the power using patterns of devices. Aiad and Lee [50] suggested an unsupervised disaggregation model, taking into account the interactions among devices. The device interactions were shaped using FHMMs and inference was implemented using the Viterbi algorithm. Stephen Makonin et al. [51] proposed a new algorithm, tackling the efficiency problem of the Viterbi algorithm. This proposal was based on super-state HMM and a modified version of the Viterbi algorithm. A super-state was determined as an HMM that defines the overall power state of a set of devices. Every appliance could be ON or OFF, and when operating could have a distinct state. Each combination of the devices' states expressed a single state of the home. The central advantage was that exact inference was feasible in computationally efficient time, by calculating sparse matrices with a large number of super-states. Disaggregation could also run in real time.

Although HMM-based strategies have attracted much consideration, as the brief literature review examined above demonstrates the main weaknesses of those models have not been overwhelmed. In fact, those models are restricted in relatively small discrete state space, the algorithmic complexity for inference is intractable, and the state space can quickly grow exponentially. This exponential space complexity is unfavorable when enlarging the model in context window [17]. In more recent time, deep neural architectures have demonstrated exceptional results in sequential models, and some NILM researchers have shown a keen interest in those solutions. In the following, the most critical efforts using deep neural architectures are briefly described.

Kelly and Knottenbelt [21] concentrated on answering the problem of power disaggregation employing deep networks. The authors submitted three different approaches: a) a solution applying a particular type of recurrent neural network using LSTM hidden nodes, b) a noise reduction solution employing denoising autoencoders, and c) a regression algorithm forecasting the start time, the end time and the average power request for each appliance connected to the electrical network. Mauch and Yang [31] presented another architecture of deep recurrent LSTM network, in order to test if this kind of network is possible to overwhelm the known problems of the previous NILM approaches. The proposed solution required one network for each device in a home. As a result, in these experiments three networks were used, one for each of the three target appliances. Zhang et al. [32] proposed a deep learning solution for the problem of single-channel blind source separation with application in NILM. That method is named sequence-to-point (seq2point) learning because it uses a window as input and a single point as a target. The suggested solution is a deep CNNs, which also learns the signature of the appliances in a house.

4. Extreme Learning Machine in a nutshell

In the previous section, we have mentioned that artificial neural network based approaches to NILM are a viable solution. Nevertheless, those models require back-propagation (BP) based fine-tuning in order to obtain suitably learning capabilities – but this is a time-consuming job; moreover, vast amounts of training data are required to achieve top performances. The latter may restrict the deployment of DNN-based solutions to a meagre collection of real-world applications. Deep models in [21], for instance, require many training data in order to achieve good performance, since those models have a massive quantity of trainable parameters (the network weights and biases). The neural networks employed in the recent NILM litterature have between 1 million to 150 million trainable parameters. To overcome those issues, we suggest an Extreme Learning Machine (ELM) approach to energy disaggregation. In fact, ELMs have only one layer, the last one, made of trainable parameters even in their deep configuration, referred to as Hierarchical-ELMs (H-ELMs). ELMs were introduced by Huang et al. for single layer feed-forward networks (SLFNs) to overcome problems with the BP algorithm. ELM gives an adequate and agile learning process that does not need the heavy fine-tuning of parameters [53].

In this paper, we also frame energy disaggregation as a regression task, which is called as "denoising" in [21]. The aggregate power requirement $y[n]$ hence is composed of the clean target power demand signal of the target device $x[n]$ and of the background additive noise signal $v[n]$ provided by the other appliances:

$$y[n] = x[n] + v[n] \qquad (2)$$

The goal is to recover an estimate of *x[n]* from the noisy signal *y[n]*. We propose to use ELM-based models to peform the denoising task.

4.1. The ELM model

In the following sections, we introduce the ELM model in its general form. In the experimental sections, we present further details concerning the neural architectures.

4.2 Shallow ELM

The ELM model was proposed by Huang et al. [33] to train single-layer feedforward networks (SLFNs) at extremely fast speeds. In the ELM, the hidden layer parameters are randomly initiated and do not require fine tuning compared to conventional SLFNs. The only parameters that require training are the weights between the last hidden layer and the output layer. Experimental results from previous studies have verified the effectiveness of the ELM algorithm by accommodating extremely fast training with good generalization performance compared to traditional SLFNs [33].

We present the ELM in its generic form following [33]. The function of the ELM can be written as

$$f(x_i) = \sum_{l=1}^{L} \beta_l \, \sigma\,(w_l \, . x_i \; + \; b_l) \qquad (3)$$

where $x_i = [x_{i1}\,,\; x_{i2}\,, \dots, x_{iN}]^T \in R^N$ is the input vector, $w_l = [w_{l1}\,,\; w_{l2}\,, \dots, w_{lN}]^T \in R^N$ is the weight vector connecting the l-th hidden node and the input vector, b_l is the bias of the l-th hidden node, $\beta_l = [\beta_{l1}\,,\; \beta_{l2}\,, \dots, \beta_{lM}]^T \in R^M$ is the weight vector from the l-th hidden node to the output nodes, L is the total number of neurons in the ELM hidden layer, and $\sigma(\cdot)$ is the nonlinear activation function to approximate the target function to a compact subset. The output function can be formulated as

$$f(x_i) = \sum_{l=1}^{L} \beta_l \; h_l\,(x) = \; h(x)B \qquad (4)$$

where B is the output weight matrix and $h(x) = [h_1(x), \dots, h_L(x)]$ is the nonlinear feature mapping. The relationship above can compactly be described as

$$\mathbf{HB} = \mathbf{Y} \qquad (5)$$

where H is the hidden layer output matrix, and Y is the target data matrix.

Peer-reviewed version available at Electronics 2018, 7, 235; doi:10.3390/electronics7100235

$$H = \begin{bmatrix} \sigma(w_1 \cdot x_1 + b_1) & \ldots & \sigma(w_L \cdot x_1 + b_L) \\ \vdots & & \vdots \\ \sigma(w_1 \cdot x_N + b_1) & \ldots & \sigma(w_L \cdot x_N + b_L) \end{bmatrix}_{N\times L}$$

$$B = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L\times M}, \text{ and } Y = \begin{bmatrix} y_1^T \\ \vdots \\ y_N^T \end{bmatrix}_{N\times M} \quad (5a)$$

The output weight matrix B is computed as

$$B = H^{+} Y \quad (6)$$

where H^{+} is the Moore–Penrose (MP) pseudoinverse of H that can be calculated using different methods such as orthogonal projection methods, Gaussian elimination, and single-value decomposition (SVD).

In order to solve the linear inverse problem arising at the ELM output, we adopted in this study a fast-iterative shrinkage-threshold algorithm (FISTA) [55], which is an extension of the gradient algorithm and offers better convergence properties for problems involving large amounts of data. It should be clarified that that Eq. (3) gives an estimate of the energy consumption for the targeted appliance.

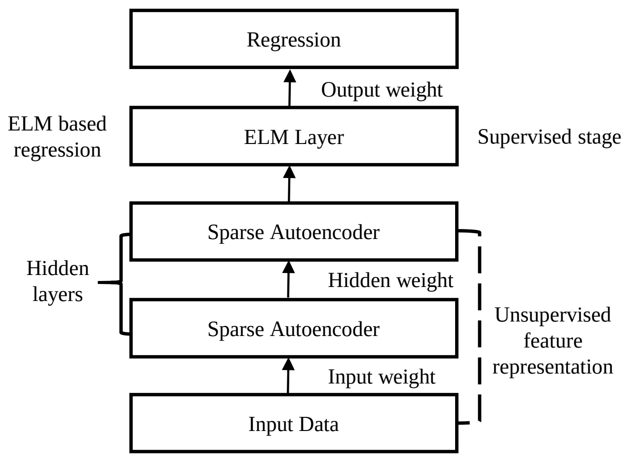

Figure 1. H-ELM architecture.

4.3 Hierarchical ELM

Spurred by DNNs, where the features are extracted using the multilayer framework with unsupervised initialization, Tang et al. [56] extended ELM and introduced H-ELM for multilayer perceptron's (MLPs). The entire structure of the H-ELM model is presented in Fig. 1. The H-ELM framework has two steps, i.e., unsupervised feature extraction, and supervised feature regression. In unsupervised feature extraction, high-level features are extracted using ELM based autoencoder by analyzing every layer as an independent layer. The input data is introduced into the ELM feature space before feature extraction, to make use of information among training data. The output of the unsupervised feature extraction stage can then be used as the input to the supervised ELM regression stage [56] for the final result, based on the learning from the two stages.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2018 doi:10.20944/preprints201808.0551.v1

Peer-reviewed version available at Electronics 2018, 7, 235; doi:10.3390/electronics7100235

5. Experimental Setup & Results

In this chapter 5, the experimental setup will be explained first. Next, we will present the performance evaluation criteria. Finally, the experimental results will be exhibited, accompanied by a short dissertation on the main findings.

Table 2. Houses used for training and testing.

Appliance	Training	Testing
Kettle	1, 2, 3, 4	5
Fridge	1, 2, 4	5
Washing Machine	1, 5	2
Microwave	1, 2	5
Dish washer	1, 2	5

5.1. Datasets

Our goal is to prove that the proposed solution can achieve comparable, if not better, performance than recently proposed deep learning methods. Hence, we need to deploy an experimental setup that allows a comparison with what available in the literature. To this end, we follow [21] and use UK-DALE [23] as our source dataset, so that we can perform sound quantitative comparison between the proposed approach and what available in the literature. Each submeter in UK-DALE samples once every 6 seconds. All houses record aggregate apparent mains power once every 6 seconds. Houses number 1, 2 and 5 also record active and reactive mains power once a second. In these houses, we downsampled 1 second active mains power to 6 seconds to align with the submetered data and used this as the real aggregate data from these houses. Any gaps in appliance data shorter than 3 minutes are assumed to be due to RF issues and so are filled by forward-filling. Any gaps longer than 3 minutes are assumed to be due to the appliance and meter being switched off and so are filled with zeros.

We used the five target appliances selected in [21] in all our experiments, namely: the fridge, washing machine, dish washer, kettle and microwave. Those appliances were chosen because each is present in at least three houses in UK-DALE. This means that, for each appliance, we can train our nets on at least two houses and test on a different house. These five appliances consume a significant proportion of energy, and the five appliances represent a range of different power 'signatures' from the simple on/off of the kettle to the intricate pattern shown by the washing machine, as shown in Figure 2 adapted from [21]. In [21], the authors define an "appliance activation" to be the power drawn by a single appliance over one complete cycle of that appliance. We adopt here the same terminology.

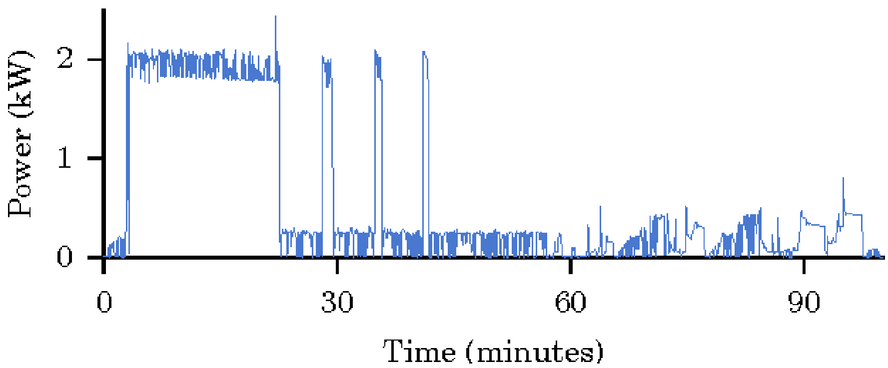

Figure 2. Appliance activation for the washing machine adapted from [21].

Peer-reviewed version available at Electronics 2018, 7, 235; doi:10.3390/electronics7100235

In [21], the authors pointed out that artificial data had to be generated in order to regularize the training of the parameters (biases and weights) of their deep neural networks, having between 1 million to 150 millions of trainable parameters. Subsequently, large training datasets play a crucial role. Our approach is back-propagation free; therefore, there are fewer parameters to learn, which allow us to use only real data during the ELM learning phase. That is a key feature of the proposed approach. For each house, we reserved the last week of data for testing and used the rest of the data for training, as in [21]. The specific houses used for training and testing is shown in Table 4.

5.2. Performance evaluation criteria

The most common performance evaluation criteria among NILM researchers, as described by [58] are split into two categories. The first category is based on the comparison between the observed aggregate power signal and the reconstructed signal after disaggregation. These metrics include: mean average error (MAE) in watts, and relative error in total energy (RET), which equations are:

$$MAE = \frac{1}{T}\sum_{t=1}^{T}|f(y_t) - x_t| \tag{12}$$

$$RET = \frac{|\hat{E}-E|}{\max(E,\hat{E})} \tag{13}$$

where $x(t)$ is the actual power of the appliance, $f(y_t)$ is the estimated power after disaggregation, and T is the number of examples.

The second category describes how effectively the disaggregated signal signatures are assigned to appliance signatures and include: precision (P), recall (R), accuracy (Acc), F-measure (F1), total energy correctly assigned (TECA), Accuracy (A). These metrics are defined as follows:

$$P = \frac{TP}{TP+FP} \tag{14}$$

$$R = \frac{TP}{TP+FN} \tag{15}$$

$$F1 = \frac{2\times P\times R}{P+R} \tag{16}$$

$$TECA = 1 - \frac{\sum_{t=1}^{T}\sum_{m=1}^{M}|f(y_t^m)-x_t^m|}{2\sum_{t=1}^{T}\overline{y_t}} \tag{18}$$

$$A = \frac{TP+TN}{P+N} \tag{20}$$

where TP is the true positive that the appliances was working, FP is the false positive that the appliance was working, TN is the true negative and FN is the false negative. P is the number of positive cases in ground truth, and N is the number of negative cases in ground truth. Moreover, x_t^m is the actual power for the mth appliance at time t, $f(y_t^m)$ is the estimated power for mth appliance at time t, and T is the number of samples.

Table 3. Disaggregation performance on seen houses during training.

Technique	Appliance	F1	P	R	A	RET	TECA	MAE

Peer-reviewed version available at *Electronics* **2018**, *7*, 235; doi:10.3390/electronics7100235

CO	*Kettle*	0.31	0.45	0.25	0.99	0.43	0.93	65
	Dish Washer	0.11	0.07	0.50	0.69	0.28	0.90	75
	Fridge	0.52	0.50	0.54	0.61	0.26	0.94	50
	Microwave	0.33	0.24	0.70	0.98	0.85	0.92	68
	Washing Machine	0.13	0.08	0.56	0.69	0.65	0.92	88
FHMM	*Kettle*	0.28	0.30	0.28	0.99	0.57	0.91	82
	Dish Washer	0.08	0.04	0.78	0.37	0.66	0.85	111
	Fridge	0.47	0.39	0.63	0.46	0.50	0.91	69
	Microwave	0.43	0.35	0.69	0.99	0.80	0.93	54
	Washing Machine	0.11	0.06	0.87	0.39	0.76	0.88	138
Autoencoder	*Kettle*	0.48	1.00	0.39	0.99	0.02	0.98	16
	Dish Washer	0.66	0.45	0.99	0.95	-0.34	0.97	21
	Fridge	0.81	0.83	0.79	0.85	-0.35	0.97	25
	Microwave	0.62	0.50	0.86	0.99	-0.06	0.99	13
	Washing Machine	0.25	0.15	0.99	0.76	0.18	0.96	44
LSTM	*Kettle*	0.71	0.91	0.63	1.00	0.36	0.98	23
	Dish Washer	0.06	0.03	0.63	0.35	0.76	0.83	130
	Fridge	0.69	0.71	0.67	0.76	-0.22	0.96	34
	Microwave	0.42	0.28	0.92	0.98	0.50	0.97	22
	Washing Machine	0.09	0.05	0.62	0.31	0.73	0.88	133
Shallow ELM	*Kettle*	0.27	0.23	0.24	0.99	0.58	0.91	81
	Dish Washer	0.09	0.06	0.81	0.33	0.62	0.85	112

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2018 doi:10.20944/preprints201808.0551.v1

Peer-reviewed version available at Electronics 2018, 7, 235; doi:10.3390/electronics7100235

	Fridge	0.42	0.23	0.67	0.47	0.48	0.92	68
	Microwave	0.41	0.45	0.71	0.98	0.81	0.90	52
	Washing Machine	0.15	0.12	0.82	0.42	0.72	0.82	137
H-ELM	*Kettle*	**0.72**	**1.00**	**0.70**	**1.00**	**0.01**	**0.98**	**15**
	Dish Washer	**0.75**	**0.89**	**0.99**	**0.98**	**-0.54**	**0.98**	**19**
	Fridge	**0.89**	**0.88**	**0.80**	**0.88**	**-0.38**	**0.98**	**20**
	Microwave	**0.66**	**0.52**	**0.87**	**0.99**	**-0.05**	**0.99**	**12**
	Washing Machine	**0.50**	**0.73**	**0.99**	**0.76**	**0.09**	**0.97**	**27**

5.3 ELM Architectural Details

We implemented our neural nets in Python. We trained our ELMs on an NVIDIA GeForce GT 750M GPU with 2 GB of GDDR5. On this GPU, our nets typically took 1 and 3 hours to train per appliance for the shallow and hierarchical ELM architecture, respectively. We train one ELM per given appliance. The output of the ELM is a window of the power demand of the target appliance. The input to every ELM is a window of aggregate power demand. The input window width is decided on an appliance-by-appliance basis. For example, a window of 128 samples (13 minutes) for the kettle; whereas, 1536 samples (2.5 hours) are used for the dishwasher. On the one hand, the window width has to be selected to ensure that the majority of the appliance activations are captured. However, a drop in performance can be observed if the width is too large, as reported in [21], where the autoencoder for the fridge failed to learn anything useful with a window size of 1024 samples, for instance. The shallow ELM has the following structure: Input dimension determined by the appliance duration, fully connected layer with 4096 hidden nodes having Sigmoid activation function, and output dimension determined by the appliance duration. The H-ELM has the following structure: Input dimension determined by the appliance, four hidden non-linear layers with 2048 nodes having Sigmoid activation function, and output dimension determined by the appliance duration.

Table 4. Disaggregation performance on unseen houses during training. This evaluation is essential to assess the generalization capability of the pattern recognition technique.

Technique	Appliance	F1	P	R	A	RET	TECA	MAE
CO	*Kettle*	0.31	0.23	0.46	0.99	0.85	0.94	73
	Dish Washer	0.11	0.06	0.67	0.64	0.62	0.94	74

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2018 doi:10.20944/preprints201808.0551.v1

Peer-reviewed version available at *Electronics* **2018**, *7*, 235; doi:10.3390/electronics7100235

	Fridge	0.35	0.30	0.41	0.45	0.37	0.94	73
	Microwave	0.05	0.03	0.35	0.98	0.97	0.93	89
	Washing Machine	0.10	0.06	0.48	0.88	0.73	0.93	39
FHMM	*Kettle*	0.19	0.14	0.29	0.99	0.88	0.92	98
	Dish Washer	0.05	0.03	0.49	0.33	0.75	0.91	110
	Fridge	0.55	0.40	0.86	0.50	0.57	0.94	67
	Microwave	0.01	0.01	0.34	0.91	0.99	0.84	195
	Washing Machine	0.08	0.04	0.64	0.79	0.86	0.88	67
Autoencoder	*Kettle*	0.93	1.00	0.87	1.00	0.13	1.00	6
	Dish Washer	0.44	0.29	0.99	0.92	-0.33	0.98	24
	Fridge	0.87	0.85	0.88	0.90	-0.38	0.98	26
	Microwave	0.26	0.15	0.94	0.99	0.73	0.99	9
	Washing Machine	0.13	0.07	1.00	0.82	0.48	0.96	24
LSTM	*Kettle*	0.93	0.96	0.91	1.00	0.57	0.99	16
	Dish Washer	0.08	0.04	0.87	0.30	0.87	0.86	168
	Fridge	0.74	0.71	0.77	0.81	-0.25	0.97	36
	Microwave	0.13	0.07	0.99	0.98	0.88	0.98	20
	Washing Machine	0.03	0.01	0.73	0.23	0.91	0.81	109
H-ELM	*Kettle*	**0.95**	**1.00**	**0.92**	**1.00**	**0.10**	**1.00**	**4**
	Dish Washer	**0.55**	**0.35**	**1.00**	**1.00**	**-0.28**	**0.98**	**22**
	Fridge	**0.89**	**0.90**	**0.92**	**0.94**	**-0.22**	**0.98**	**23**

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 August 2018 doi:10.20944/preprints201808.0551.v1

Peer-reviewed version available at Electronics 2018, 7, 235; doi:10.3390/electronics7100235

	Microwave	**0.36**	**0.32**	**0.98**	**0.99**	**0.65**	**0.99**	**7**
	Washing Machine	**0.43**	**0.10**	**1.00**	**0.84**	**0.51**	**0.97**	**21**

5.4. Experimental Results and Discussion

For the sake of comparison, we report results with combinatorial optimisation (CO), factorial hidden Markov model (FHMM), long short-term memory (LSTM) recurrent neural networks, and autoencoder algorithms. The performance of those solutions is given as reported in [21]. It should be pointed out that there is a single LSTM per appliance in [21], and LSTMs have a number of trainable parameters that is around 1M. There is also an auto-encoder neural architecture per appliance, with a number of trainable parameters that range from 1M to 150M depending on the input size. The interested reader is referred to [21] for more details on both the LSTM and autoencoder training phase. The autoencoder acts as a denoiser, and it is therefore similar to our ELM-based solutions.

The disaggregation results on seen houses are shown in Table 3. The results on houses unseen during training are shown in Table 4. From the seen houses in Table 3, we observe that LSTM outperforms more conventional CO and FHMM on two-state appliances (kettle, fridge and microwave), it falls behind CO and FHMM on multi-state appliances (dishwasher and washing machine). The shallow ELM solution attains comparable performance with CO and FHMH, yet it is worse than the LSTM and autoencoder solutions. The proposed H-ELM instead outperforms CO, FHMM, LSTM, and the autoencoder on every appliance on F1 score, P score, accuracy, the proportion of total energy correctly assigned and MAE, on both two-state, and multi-state appliances.

Finally, it is significant to verify the generalisation capabilities on houses unseen during training. From the results reported in Table 4, we see that autoencoder outperforms CO and FHMM on every appliance on every metric except the relative error in total energy. Instead, the H-ELM outperform all of the other solutions, as expected. We have already demonstrated that H-ELMs are superior in performance than shallow ELMs, so we do not report ELM results on unseen houses.

6. Conclusions

ELMs are a promising artificial neural network method introduced in [50] that has a high-speed learning capability and has been employed in many tasks successfully. For instance, ELMs for wind speed forecasting were used in [59, 60]. In [61], the authors apply ELSs to a state-of-charge estimation of battery with success. ELMs were also suggested as an effective solution to guarantee the continuous flow of current supply in smart grids. In this paper, we have proposed to apply ELMs to NILM. ELMs, both in their shallow and hierarchical configurations, act as a non-linear signal enhancement system allowing us to recover the power load of the target appliance from the aggregated load. In this work, we have proposed to extend ELMs to the energy disaggregation problem, and we have reported top performance on the UK-DALE dataset. ELMs are much more straightforward to train than deep models reported in [21] while attaining superior performance in both seen and unseen house and across appliances. The main advantage of the ELMs over LSTMs and denoise autoencoder is that there is no need to fine-tune of the whole network by the iterative

 doi:10.20944/preprints201808.0551.v1

Peer-reviewed version available at *Electronics* **2018**, *7*, 235; doi:10.3390/electronics7100235

back-propagation algorithm, and that can quicken the learning speed and strengthen the generalisation performance.

Finally, it is worth noting that the comparison of various NILM approach is still cumbersome. However, there has been much improvement in the latest years, and there are novel approaches and mathematical tools that haven't been used extensively yet. We believe that ELM shows great potential for NILM, and further improvement could be achieved through multi-task learning. We also believe that Extreme Learning Machines could play a key role for management of Power Consumption in Industrial Wireless Sensor Networks [63].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. J. Uteley and L. Shorrock. Domestic energy fact file 2008. Available: http://www.bre.co.uk/page.jsp?id=879 (July 8th ,2018)

2. EIA energy consumption report. Available: https://www.eia.gov/totalenergy/data/annual/pdf/sec2.pdf (July 8th ,2018)

3. EU energy consumption. Available: http://www.eea.europa.eu/data-and-maps/figures/final-energy-consumption-by-sector-eu-27 (July 8th ,2018)

4. EIA residential report. https://www.eia.gov/consumption/residential/index.php (July 8th, 2018)

5. Meyers, R. J.; Williams, E. D.; Matthews, H. S. Scoping the potential of monitoring and control technologies to reduce energy use in homes. Energy and Buildings 2010, 42, 563-569, https://doi.org/10.1109/ISEE.2007.369401.

6. Darby S., The effectiveness of feedback on energy consumption: a review for DEFRA of the literature on metering 486 2006.

7. Abrahamse, W; Steng, L.; Vlek, C; Rothengatter, T. A review of intervention studies aimed at household energu conservation, J. Environ. Psycho. 2005, 273-291.

8. Yu, L.; Li, H.; Feng, X; Duan, J Nonintrusive appliance load monitoring for smart homes: recent advances and future issues, IEEE Instrum. Meas. Mag. 2016, 19 (3), 56–62, https://doi.org/10.1109/MIM.2016.7477956.

9. Hart, G.W. Nonintrusive appliance load monitoring. Proc. IEEE 1992, 80, 1870–1891.

10. Zeifman, M.; Roth, K. Nonintrusive appliance load monitoring: Review and outlook. IEEE Trans. Consum. Electron. 2011, 57, 76-86.

11. Zoha, A.; Gluhak, A.; Imran, M.A.; Rajasegarar, S. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. Sensors 2012, 12, 16838–16866

12. Chang. H. H.; Y. Lian; Su, Y. C.; Lee, W. J. Power spectrum-based wavelet transform for non-intrusive demand monitoring and load identification. IEEE Trans. Ind. Appl. 2014, 50, 2081-2089.

Peer-reviewed version available at Electronics 2018, 7, 235; doi:10.3390/electronics7100235

13. Haq, A.U.; Jacobsen, H.-A. Prospects of Appliance-Level Load Monitoring in Off-the-Shelf Energy Monitors: A Technical Review. Energies 2018, 11, 189.

14. Meehan, P.; McArdle, C.; Daniels, S. An Efficient, Scalable Time-Frequency Method for Tracking Energy Usage of Domestic Appliances Using a Two-Step Classification Algorithm. Energies 2014, 7, 7041-7066.

15. Kwak, Y.; Hwang, J.; Lee, T. Load Disaggregation via Pattern Recognition: A Feasibility Study of a Novel Method in Residential Building. Energies 2018, 11, 1008.

16. Norford, L.K.; Mabey, N. Non-Intrusive Electric Load Monitoring in Commercial Buildings; Technical Report; Energy Systems Laboratory, TAMU: College Station, TX, USA, 1992. Available online: http://hdl.handle. net/1969.1/92921 (accessed on 9 January 2018).

17. Zia, T.; Bruckner, D.; Zaidi, A. A Hidden Markov Model Based Procedure for Identifying Household Electric Loads. Proceedings of the 37th Annual Conference on IEEE Industrial Electronics Society, Melbourne, Australia, 7–10 November 2011, 3218–322.

18. Onoda, T.; Rätsch, G.; Müller, K.R. Applying support vector machines and boosting to a non-intrusive monitoring system for household electric appliances with inverters, Proceedings of the Second ICSC Symposium on Neural Computation, Berlin, Germany, 23–26 May 2000.

19. Fomby, T.B.; Barber, T. K-Nearest Neighbors Algorithm: Prediction and Classification; Southern Methodist University: Dallas, TX, USA, 2008; pp. 1–5.

20. Gomes, L.; Fernandes, F.; Sousa, T.; Silva, M.; Morais, H.; Vale, Z.; Ramos, C. Contextual intelligent load management with ANN adaptive learning module, Proceedings of the 2011 16th International Conference on Intelligent System Application to Power Systems, Hersonissos, Greece, 25–28 September 2011.

21. Kelly, J.; Knottenbelt, W. Neural NILM: Deep neural networks applied to energy disaggregation. Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments, Seoul, Korea, 4–5 November 2015; ACM: New York, NY, USA, 2015; pp. 55–64.

22. Ruzzelli, A.G.; Nicolas, C.; Schoofs, A.; O'Hare, G.M. Real-time recognition and profiling of appliances through a single electricity sensor, Proceedings of the 7th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, Boston, MA, USA, 21–25 June 2010.

23. Nalmpantis, C.; Vrakas, D. Machine learning approaches for non-intrusive load monitoring: from qualitative to quantitative comparison. Artif. Intell. Rev. 2018, https://doi.org/10.1007/s10462-018-9613-7.

24. Zeifman M. Disaggregation of home energy display data using probabilistic approach. IEEE Trans Consum Electron 2012, 58, 23–31, https://doi.org/10.1109/TCE.2012.6170051.

25. Bengio, Y. Learning deep architectures for AI. Found Trends Mach. Learn. 2009, 2, 1–127.

26. G. Hinton et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups", IEEE Signal Process. Mag. 2012, 29, 82-97.

27. Siniscalchi, S. M.; Salerno, V. M. Adaptation to new microphones using artificial neural networks with trainable activation functions. IEEE Trans. Neural Netw. Learn. Syst. 2017,28, 1959–1965.

28. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. Proc. IEEE 1998, 86, 2278–2324.

29. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. 2012, 1097–1105.

30. Hochreiter, S.; Schmidhuber J. Long short-term memory, Neural Comput. 1997, 9, 1735–1780.

31. Mauch L.; Yang, B.; A new approach for supervised power disaggregation by using a deep recurrent LSTM network. Proceedings of the 3rd IEEE Global conference on signal and information processing, 2015 63–67

32. Zhang, C.; Zhong, M; Wang, Z.; Goddard, N.; Sutton, C. Sequence-to-point learning with neural networks for nonintrusive load monitoring, arXiv, 2016.

33. G.-B. Huang, An insight into extreme learning machines: random neurons, random features and kernels, Cognitive Computation 2014, 6, 376-390.

34. Huang, Z.; Yu, Y; Gu, J.; Liu, H. An efficient method for traffic sign recognition based on extreme learning machine, IEEE T. Syst. Man Cy. C 2017, 47, 920-933.

35. Tang, J.; Deng, C.; Huang, G.-B. Extreme learning machine for multilayer perceptron, IEEE Trans. Neural Netw. Learn. Syst. 2016, 27, 809-821.

36. Sun, F.; Liu, C.; Huang, W.; Zhang, J. Object classification and grasp planning using visual and tactile sensing, IEEE T. Syst. Man Cy. C 2016, 46, 969-979.

37. Kasun, L. L. C.; Zhou, H.; Huang, G.-B.; Vong, C. M. Representational learning with ELMs for big data, IEEE Intel. Syst. 2013, 28, 31-34.

38. Zhao, W.; Beach, T. H.; Rezgui, Y. Optimization of potable water distribution and wastewater collection networks: A systematic review and future research directions, IEEE T. Syst. Man Cy. C 2016, 46, 659-681.

39. Wang, D.; Bischof, L.; Lagerstrom, R.; Hilsenstein, V.; Hornabrook, A.; Hornabrook, G. Automated Opal Grading by Imaging and Statistical Learning, IEEE T. Syst. Man Cy. C 2016, 46, 185-201.

40. Liu, D.; Wei, Q.; Yan, P. Generalized policy iteration adaptive dynamic programming for discrete time nonlinear systems, IEEE T. Syst. Man Cy. C 2015, 45, 1577-1591.

41. Huang, G.-B.; Zhou, H.; Ding, X.; Zhang, R. Extreme learning machine for regression and multiclass classification, IEEE T. Syst. Man Cy. B 2012, 42, 513-529.

42. Kelly, J.; Knottenbelt, W. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes 2015, Sci Data 2:150007, Available: https://doi.org/10.1038/sdata.2015.7 (July 8th ,2018)

43. Kim, H.; Marwah, M.; Arlitt, M.; Lyon, G.; Han, J. Unsupervised disaggregation of low frequency power measurements, Proceedings of the 2011 SIAM International Conference on Data Mining, 2011, 747–758.

44. Kolter J. Z. ; Johnson, M. J. Redd: A public data set for energy disaggregation research, Proceedings of Workshop on Data Mining Applications in Sustainability, San Diego, CA, 2011.

45. Kolter, J. Z.; Jaakkola, T. Approximate inference in additive factorial HMMs with application to energy disaggregation, Proceedings of International Conference on Artificial Intelligence and Statistics, 2012, 1472–1482.

46. Parson, O.; Ghosh, S.; Weal, M.; Rogers, A. Using hidden Markov models for iterative non-intrusive appliance monitoring, Proceedings of Neural Information Processing Systems workshop on Machine Learning for Sustainability, December 2011

47. Parson, O.; Ghosh, S.; Weal, M.; Rogers, Non-intrusive load monitoring using prior models of general appliance types, Proceedings of the Twenty-Sixth Conference on Artificial Intelligence, July 2012, 356–362.

48. Egarter D.; Bhuvana V.P.; Elmenreich W. PALDi: online load disaggregation via particle filtering, IEEE Trans Instrum Meas 2015, 64, 467–477. https://doi.org/10.1109/TIM.2014.2344373

49. Paradiso, F.; Paganelli, F.; Giuli, D.; Capobianco, S. Context-based energy disaggregation in smart homes, Future Internet 2016, 8.

50. Aiad, M.; Lee, P.H. Unsupervised approach for load disaggregation with devices interactions, Energy Build 2016, 116, 96–103.

51. Makonin, S.; Popowich, F.; Bajic, IV; Gill, B. Bartram, L. Exploiting HMM sparsity to perform online real- time nonintrusive load monitoring, IEEE Trans Smart Grid 2015, 7, 2575-2585.

52. Lange, H.; Bergés, M. The neural energy decoder: energy disaggregation by combining binary subcomponents, Proceedings of the NILM2016 3rd international workshop on non-intrusive load monitoring, 2016

53. Huang, G.-B.; Zhou, H.; Ding, X.; Zhang, R. Extreme learning machine for regression and multiclass classification, IEEE Trans Syst Man Cybern B Cybern 2012, 42, 513-529.

54. Huang, G.-B.; Chen, L.; Siew, C. K. Universal approximation using incremental constructive feedforward networks with random hidden nodes, IEEE Trans. on Neural Netw. 2006, 17, 879-892.

55. Beck, A; Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM Jour. on Imaging Sci. 2009, 2, 183-202.

56. Tang, C. Deng, and G.-B. Huang, "Extreme learning machine for multilayer perceptron," IEEE trans. on neural networks and learning systems, vol. 27, pp. 809-821, 2016.

57. Huang, G.-B. An insight into extreme learning machines: random neurons, random features and kernels, Cognitive Computation 2014, 6, 376-390.

58. Bonfigli, R.; Squartini, S.; Fagiani, M; Piazza, F. Unsupervised algorithms for non-intrusive load monitoring: An up-to-date overview, Proceedings of the IEEE 15th international conference on environment and electrical engineering, 2015, 1175–1180.

59. Zhou, J.; Sun, N.; Jia, B.; Peng, T. A Novel Decomposition-Optimization Model for Short-Term Wind Speed Forecasting. Energies 2018, 11, 1752.

60. Wang, R.; Li, J.; Wang, J.; Gao, C. Research and Application of a Hybrid Wind Energy Forecasting System Based on Data Processing and an Optimized Extreme Learning Machine. Energies 2018, 11, 1712.

61. Chin, C.S.; Gao, Z. State-of-Charge Estimation of Battery Pack under Varying Ambient Temperature Using an Adaptive Sequential Extreme Learning Machine. Energies 2018, 11, 711.

62. Sun, W.; Zhang, C. A Hybrid BA-ELM Model Based on Factor Analysis and Similar-Day Approach for Short-Term Load Forecasting. Energies 2018, 11, 1282.

63. M. Collotta, V.M. Salerno, G. Pau, G. Scatà — A fuzzy based algorithm to Manage Power Consumption in Industrial Wireless Sensor Networks — INDIN 2011 — IEEE 9th International Conference on Industrial Informatics — July 2011 Lisbon, Portugal.