

## SAED: Self-Attentive Energy Disaggregation

Nikolaos Virtsionis Gkalinikis ·  
Christoforos Nalmpantis · Dimitris  
Vrakas

Received: date / Accepted: date

**Abstract** The field of energy disaggregation deals with the approximation of appliance electric consumption using only the aggregate consumption measurement of a mains meter. Recent research developments have used deep neural networks and outperformed previous methods based on Hidden Markov Models. On the other hand deep learning models are computationally heavy and require huge amounts of data. The main objective of the current paper is to incorporate the attention mechanism into neural networks in order to reduce their computational complexity. For the attention mechanism two different versions are utilized, named Additive and Dot Attention. The experiments show that they perform on par, while the Dot mechanism is slightly faster. The two versions of self-attentive neural networks are compared against two state-of-the-art energy disaggregation deep learning models. The experimental results show that the proposed architecture achieves faster or equal training and inference time and with minor performance drop depending on the device or the dataset.

**Keywords** energy disaggregation · non-intrusive load monitoring · artificial neural networks · self attention

---

Nikolaos Virtsionis Gkalinikis  
School of Informatics, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece  
E-mail: nvirtsion@csd.auth.gr

Christoforos Nalmpantis  
School of Informatics, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece  
E-mail: christofn@csd.auth.gr

Dimitris Vrakas  
School of Informatics, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece  
E-mail: dvrakas@csd.auth.gr

## 1 Introduction

Energy disaggregation, also known as non-intrusive load monitoring (NILM), aims to separate appliance-level electric data from the total power consumption of a electrical installation. The main benefit of NILM is the possible improvement of electrical energy management. In the long term, the unnecessary waste of energy will be avoided, positively affecting the global warming and the climate change problems. Further analysis on the targets' consumption may identify functionality inefficiencies.

Due to the rise of Internet of Things (IoT), the usage of smart meters in residential buildings increases [24]. As a result, NILM is becoming popular among the energy data analytics techniques of the residential and small commercial sector [1]. Home energy management systems (HEMS) are capable of monitoring and management of electrical appliances in many smart houses [32]. There are two ways the appliance load monitoring (ALM) can be developed; either with intrusive or non-intrusive methods [26, 40]. Against intrusive-loading monitoring (ILM), NILM is cheaper and more straightforward because it depends only on measurements from a single mains meter, without the use of extra equipment. On the contrary, ILM provides better accuracy than NILM, being more expensive and demanding in terms of installation.

The contribution of the current paper to the research field of energy disaggregation could be summarized as follows. A novel lightweight recurrent neural network architecture is designed. The attention mechanism, a technique borrowed from Natural Language Processing sector, is inserted in a typical NILM architecture, by significantly reducing its complexity. A set of baseline results and a meticulous analysis are presented, emphasizing not only the performance but also the efficiency of the models. It should be noted that the current research extends the work published by Virtsionis et al. [37]. An ablation study is performed to highlight the fact that the attention mechanism boosts the network in *on/off* events detection. Extended experiments on more devices of a different data set, alongside comparisons with one more popular state-of-the-art architecture, lead to significant insights on which components are more suitable for designing lightweight and efficient NILM architectures. A knowledge transfer scenario is demonstrated, using the extracted features of complicate devices to simpler appliance disaggregation. Overall, the current study reveals that the proposed model can perform on par with the state-of-the-art models on most occasions, achieving stronger generalization properties in scalable training and inference times.

The anatomy of this article is as follows. As a starting point, there is a brief presentation of the related work on NILM and energy disaggregation. Secondly, the attention mechanism is analyzed. The third section presents a concise explanation of Attention operators. Section 4, includes a thorough explanation of the purpose and operation of each individual part of the proposed architecture. Section 5 includes a description of the methodology of experiments. The most important of the results are presented in section 6. In section 7,

an ablation study is demonstrated. Last but not least, conclusions and future work proposals are introduced.

## 2 Related Work

Energy disaggregation was firstly introduced by Hart in mid 80s as Non Intrusive Load Monitoring. Later, Hart proposed a combinatorial technique to tackle the problem of NILM [10]. This method extracts the optimal states of the target devices in such a way that the sum of appliance power consumptions would be the same as the meter reading of total power. The drawback of the combinatorial method is that it can be applied only on simple devices with finite number of operation states.

Before the rise of deep learning, the most popular techniques for NILM included Factorial Hidden Markov Models (FHMM) [15, 19, 30]. A FHMM architecture is essentially a set of multiple independent Hidden Markov Models. A combination of all the individual hidden states constitutes the observed output of the model. Kolter and Jaakkola [19] proposed an additive novel FHMMs, where as output the sum of the individual HMMs is calculated.

Recent developments on hardware engineering opened the door for the rapid evolution of machine and deep learning. New approaches and algorithms thrive in complex tasks from the sectors of Natural language processing (NLP), Computer Vision and Time Series Analysis. Hence, NILM research started to focus on adjusting many of these techniques for the problem of energy disaggregation alongside developing new ones. Motivated by the current trends, Kelly and Knottenbelt [14] designed three deep neural networks; a recurrent architecture, a denoising autoencoder (DAE) and a ANN model to regress start/end time and power. The results on the UK-DALE [12] data set were more than promising, with the novel models outperforming both Hart’s method and FHMM. A similar architecture with LSTM recurrent neurons is in [25]. This method was tested on real data from REDD [20] alongside with synthetic data, achieving good results for appliances with cycling motives in power consumption.

A state-of-the-art architecture called Sequence-to-Point was implemented by Zhang et al. [39] only with the use of convolutional neural network (CNN) and dense layers. The name *Sequence-to-Point* comes from the fact that this technique uses a sliding window of aggregate data measurements to disaggregate the appliance consumption on a single midpoint time step. The latter constitutes a core difference versus the other methods presented by Kelly and Knottenbelt [14] and Mauch et al. [25]. Krystalakos et al. [21] used a different sliding window technique, utilizing Gated Recurrent Units (GRUs), a variation of LSTMs, in combination with dropout layers to improve previous RNN architectures in terms of performance and efficiency. As the popularity of RNN architectures grew, authors propose more variants of these methods [13, 8].

Recently, the attention mechanism was introduced in the NILM sector. A variant of Google’s Transformer [36], Bert4NILM [38] was adjusted to the problem of disaggregation. The model achieves great results, but it has a

large number of parameters, which affects training and inference. An encoder-decoder type of model with temporal attention is proposed by Piccialli et al. [31]. The results show that the attention helps in event detection, resulting to good generalization in unseen data.

To successfully compare various methods and models Symeonidis et al. [35], synthesized a benchmark methodology. Also, an exploration of the Stacking method of five well known models is conducted, providing good results on simple 2-state appliances. Nevertheless, regarding the reproducibility and comparability of energy disaggregation frameworks, a standardization of the assessment procedures is recommended [18,27].

Despite the breakthroughs that Deep Learning brought in NILM research, deployment issues remain. The main reason is the training and inference duration, due to the massive number of parameters of the state-of-the-art architectures. Moreover, for years the centre of attention of the NILM field of study was the development of one model per device. Hence, a complete energy disaggregation system should be consisted of a number of models equal to the number of devices the target electrical installation contains. In real time cases, energy measurements output massive quantities of data even at low sampling rates, which makes deploying NILM on embedded devices a challenging task. In order to do so, a number of steps should be taken. The development of lightweight architectures is the first one. Next, it is suggested that multi-label machine learning models should be designed. Multi-label models are trained in order to estimate the electrical power consumption of more than one appliances, making the relation of "models per device" in "1-to-many" situation.

Basu et al. [3,4] introduced the multi-label classification in energy disaggregation using algorithms such as decision trees and boosting. Recently an article on multi-label disaggregation was published by Nalmpantis and Vrakas [29], proposing a novel framework called multi-NILM. This approach combines a dimensionality reduction technique called Signal2vec [28] with a lightweight disaggregation model, showcasing promising results. A different approach on reducing computational resources constitutes a family of methods known as transfer learning. Transfer learning is used in NILM research with some success in [7,11]. Kukururi et al. [22] proposed various compression methods in order to make deep neural networks suitable for deployment *on the edge*, alongside a multi-task method based on a hard parameter sharing approach, in a similar approach as transfer learning methods.

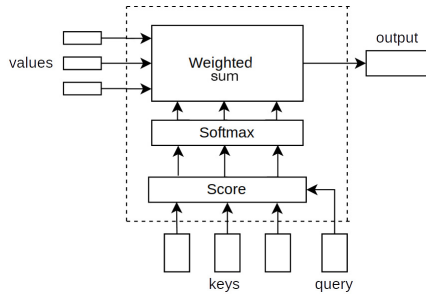
### 3 Attention Mechanism

The extraction of input-output relations is a common task in machine learning and pattern recognition, with uses in image captioning, machine translation etc. Sequence to sequence models (seq2seq) consist a go-to approach regarding the Deep Learning techniques. In [34] the original seq2seq 3 model, as proposed by Sutskever et al., contains two major components; the encoder and the decoder. Essentially these components are two RNNs. The role of the encoder is the compression of the sequence input into a vector of fixed length, known

as the *context*. The intuition is that this vector suppresses the most important information of the source input. Given the same context vector, the decoder is capable to re-construct the source sequence. The drawback of this architecture is that it fails to process very long sequences, due to the fixed length of the context vector.

To improve the efficiency of seq2seq architecture, Bahdanau et.al [2] proposed Attention. This mechanism gives the decoder the power to concentrate on the parts of the input that matter the most, in relation to the corresponding output. At each time step of the decoder, Attention calculates the relations between the entire input sequence and the decoder output. These calculations create an alignment vector, that contains the score between the input's sequence and the decoder's output at the corresponding step. The resulting context vector is a combination of both the alignment vector and the encoder's output.

Considering how the scores and alignments are calculated, the most popular types of attention are the Additive [2] and the Multiplicative/Dot [23]. In a different setting called Self-Attention [6], the attention mechanism is applied on the same sequence, in order to relate different parts of it. Self-Attention can integrate either Bahdanau's or Luong's scoring methods. The proposed architecture in the current research uses Additive and Dot attention mechanisms. As inputs to an Attention layer three kinds of vectors are given; a query, a key and a value. The layer output is calculated as described bellow. A summary of the steps is shown in Fig.3.



**Fig. 1** Inside Attention mechanism.

To begin with, the similarity between a query ( $q$ ) and a key ( $k_i$ ) is calculated, estimating for each query-key pair, a score ( $a_i$ ).

$$a_i = \text{score}(q, k_i) \quad (1)$$

Next, a softmax function is used to normalize the scores in order to sum up to one. The attention weights are obtained as follows.

$$b_i = \frac{\exp(a_i)}{\sum_j \exp(a_j)} \quad (2)$$

The final output is the weighted sum of the values ( $v$ ):

$$output = \sum_{i=1}^n b_i v_i \quad (3)$$

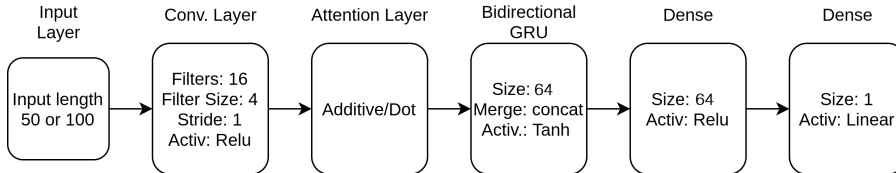
Between Additive and Dot attention mechanisms the scoring function differs. Specifically, the dot scores are given by the dot product of keys and queries. On the other hand, the additive scoring function is a non-linear sum.

#### 4 Self Attentive Network Topology

The development of a computationally light neural network is the main objective of the current research. To design a lightweight model, inspiration was draught by the architecture known as Window GRU (WGRU) [21].

WGRU [21] is composed of the following ANN layers: a convolutional layer, two Bidirectional GRU layers and one Dense layer before the output. Dropout technique [33] is used between layers against overfitting problems. In order to approximate the appliance power consumption at a single time step, a sliding window of past aggregate data points is used. The core element of the WGRU architecture, is the GRU layer, a variation of the LSTM recurrent layer.

Instead of using two GRU layers back to back, the novel network contains an Attention layer before one GRU layer. The proposed model is called *Self-Attentive-Energy-Disaggregation* (SAED) and, comparing to the WGRU, achieves up to 7.5 times faster training and up to 6.5 times faster inference. In terms of performance, there is a trivial trade-off which is explored thoroughly in upcoming sections.



**Fig. 2** Architecture of the Attention model.

SAED architecture is a synthesis of four different types of ANNs. In order to extract new features from the input a 1D convolution layer is used. This type of layer is time invariant; it can learn local patterns found at certain positions of the sequence, which is able to identify at different spots of other sequences. Using Attention, the network learns to focus on the most crucial of those features. Next the output of the attention layer is provided to a GRU layer, to recognize possible sequential patterns. The final result is given after passing through a dense layer, functioning as a regressor. A graphical representation of the architecture is depicted in Fig.2. Two important notes must be highlighted. Firstly, the Attention layer operates as a Self-Attention

**Table 1** Learnable Parameters and Size of the models.

Model	Parameters	Size(MB)
WGRU	270k	3.3
Seq2Point	2600k	31.3
SAED-dot	40k	0.54
SAED-add	42k	0.54

mechanism given as input the output of the CNN layer. Secondly, SAED comes in two variations concerning the attention mechanism; with either Additive or Dot attention, mentioned as SAED-add and SAED-dot correspondingly.

The proposed network was developed with Python 3.8 and Tensorflow 2.2.0. Tensorflow provides two Attention layers; Attention and AdditiveAttention which corresponds to Dot Attention and Additive Attention. Adam optimization algorithm [16] and MSE loss were used for the training. The experiments were executed on a Nvidia GPU GTX-1060 6Gb. NILMTK framework [5] was used for data loading.

## 5 Structure of Experiments

For the experiments only real measurements were considered. The sampling period was 6 seconds and the batch size 1024. Seven electric devices were chosen for the experiments; dish washer (DW), fridge (FZ), kettle (KT), microwave (MW), washing machine (WM), television (TV) and computer (PC). The optimal size of the sliding windows depends on the device and on the algorithm [21]. At the current research, time window was 50 samples for all the target appliances and models with the exception of washing machine, where a window of 100 samples was used. The experiments were conducted comparing three different architectures; the proposed SAED architecture, the WGRU [21] and the Seq2Point [39] as implemented by Krystalakos et al. [21]. The number of learnable parameters of all the models are presented in Table 5. SAED models have 65 and 6.5 times less parameters than the Seq2Point and WGRU accordingly, resulting to considerably smaller space storage requirements on deployment. It should be noted that for the devices television and computer, the dropout ratio for the Seq2Point model was 25%. For all the models the training duration was 5 epochs, while the benchmark basis described in [35] was followed. In this methodology the experiments are divided in four distinct categories; Single Building NILM, Single Building learning and generalization on same dataset, Multi building learning and generalization on same dataset and Generalization to different dataset.

The first category of experiments is about training and inference on the same house at different time periods. Therefore, the models are evaluated in the same environment where training was executed. Models with low performance in these experiments are probably weak [35]. In the second category of experiments training and inference happen on different houses of the same dataset. These experiments serve the purpose of measuring the generalization potential of the model on different buildings. Briefly, different buildings lead

to divergent energy patterns that derive from multiple factors such as the different habits of the residents, the use of other electric devices. Nevertheless, similarities between measurements of the same data set are also expected. Properties like electricity grid, weather conditions and regionality are some of the possible reasons for this fact. Hence, in addition to this category more experiments are needed in order to explore the generalization ability of the models in a greater detail.

The third category is about experiments where training data is collected from different buildings and testing is executed on an unseen building. All the measurements for this category belong in the same dataset. On the other hand, even though in the fourth category of experiments training data is also collected from different houses, the inference is executed on houses of a different dataset. The purpose of the last two categories of experiments is to evaluate the sufficiency of an algorithm in learning from a variety of sources. Naturally, the challenge for the model rises even more in the fourth category, because the inference is done on unknown data from a completely different dataset. Due to these challenges, models that succeed in the last two categories of experiments could be considered strong [35].

Due to the number of devices that were used, we divided them in two groups based on the data sets used for training and inference. In the first group of devices the experiments operated on UK-DALE [12] and REDD [20] data sets, whereas in the second group REFIT [9] and UK-DALE data sets were used. It should be noted that UK-DALE and REDD contain power consumption measurements of households in UK and USA correspondingly, while REFIT contains power readings of 20 residential houses in UK, with a wider range of devices than the UK-DALE.

For the devices of the first group (dish washer, fridge, kettle, microwave and washing machine), categories of experiments 1-3 were executed on the UK-DALE data set, while for category 4 inference was evaluated on the REDD data set. Due to the lack of kettle device data in the REDD data set, the fourth category of experiments on kettle was not conducted. Training for categories 1 and 2 of experiments was conducted on house 1 of UK-DALE during the first 9 months of 2013 while the last 3 months of the same year were used for testing. Regarding the experiments of categories 3 and 4, the ratio of training versus inference data depends on the device.

For the remaining electric devices (television, computer) REFIT [9] was used for training, while data from UK-DALE were used for inference. Concerning the experiments on this device group, three months of data from REFIT was used, while the inference was executed on measurements of length 1 month. These experiments may highlight how the models perform in the case of limited data. Thus, the models do not perform great in some categories of experiments. All the experiments are summed up in Table 2.

For the evaluation and comparison of the models, the following metrics are calculated; F1 score, Relative Error in Total Energy (RETE) and Mean Absolute Error (MAE). The ability of model to detect on/off energy states is evaluated with F1 score. As seen in eq. 4, F1 score is computed as the harmonic



**Table 2** Buildings used for train and test. For the first 4 devices: In Categories 1-3, UK-DALE was used for training and testing. In Category 4, UK-DALE was used for training and REDD was used for testing. For the last 3 devices: In Categories 1-3, REFIT was used for training and testing. In Category 4, REFIT was used for training and UK-DALE was used for testing.

Device	Category1		Category2		Category3		Category4	
	Train	Test	Train	Test	Train	Test	Train	Test
DW	1	1	1	2,5	1,2	5	1,2	1,2,3,4,6
FZ	1	1	1	2,4,5	1,2,4	5	1,2,4	1,2,3,5,6
KT	1	1	1	2,3,4,5	1,2,3,4	5	-	-
MW	1	1	1	2,3,5	1,2	5	1,2	1,2,3,5
WM	1	1	1	2,4,5	1,5	2	1,5	1,2,3,4,5,6
TV	6	6	6	14,17,19	6,17	14,19	6,17	1,5
PC	6	6	6	16,17,19	6,17	16,19	6,17	5

mean of Precision and Recall, presented in eq. 5 and 6. Precision measures the ratio of the actual true positives (TP) versus the total predicted positives. In addition, Recall is the percentage of TP versus the actual positives.

On the other hand, MAE (measured in Watts) and RETE (dimensionless) evaluate the capability of the models to estimate the actual electric power consumption of the device. MAE and RETE are given in equations 7 and 8, where  $E'$  is the predicted total energy,  $E$  is the true value of total energy,  $T$  is the length of the predicted sequence,  $y_t'$  the estimated electrical power consumption and  $y_t$  the true value of active power consumption at moment  $t$ .

$$F1 = 2 \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

$$RETE = \frac{|E' - E|}{\max(E', E)} \quad (5)$$

$$MAE = \frac{1}{T} \sum |y_t' - y_t| \quad (6)$$

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

In an effort to investigate the generalization properties of the proposed SAED model even further, the use of more metrics is inevitable. As proposed by Klemenjak et al. [17], the amount of seen and unseen installations where a model is evaluated should be taken upon consideration. Thus, the idea of generalization loss (G-loss) was proposed. The intuition is that between seen and unseen installations there may be a change in the value of a metric. This indicates a change in the performance of the model when tested on unseen data. Whether the metric is used to evaluate event detection or power approximation, the G-loss is calculated as described in eq. 9 or eq. 10, where  $u$  stands for unseen and  $s$  for seen installations.

For example, a calculated G-loss of 15% on F1 score means that the measured F1 score on the unseen house data is 15% lower than on the seen data, where the training took place. On the contrary, 10% G-loss on MAE denotes that the error measured on the unseen data is 10% higher than the error measured on the seen building measurements.

The mean of all the G-losses calculated for the unseen houses resembles the mean generalization loss (MGL), which represents the overall performance loss. In order to evaluate the the generalization properties of an architecture, accuracy on unseen houses (AUH) and error on unseen houses (EUH) can be calculated also. The above metrics are given by eq. 11-13, where N is the number of the unseen building.

$$G - loss = 100(1 - \frac{F1_u}{F1_s}) \quad (9)$$

$$G - loss = 100(\frac{MAE_u}{MAE_s} - 1) \quad (10)$$

$$MGL = \frac{1}{N} \sum_i^N G - loss_i \quad (11)$$

$$AUH = \frac{1}{N} \sum_i^N F1_{ui} \quad (12)$$

$$EUH = \frac{1}{N} \sum_i^N MAE_{ui} \quad (13)$$

A different aspect of generalization was suggested by D’Incecco et al. [7]. The intuition is that extracted features learned on training ”complex” devices could be used to disaggregate appliances with ”simpler” electric signatures. The main benefit of this idea offers is the speed up of training, thus the need of less computational resources. The authors proposed two scenarios of model knowledge transfer; appliance (ATL) and cross domain transfer learning (CTL). In CTL schema, a model is trained and tested on different data sets, in similar way as in the benchmark method described in [35]. On the other hand, in ATL scenario, the model is trained and fine-tuned before the final inference. In the current article we compare the models on the ATL scenario.

## 6 Results and Comparisons

The models are compared on 3 levels. At first, in terms of performance on the four categories of experiments. Next, on generalization by computing the generalization loss on unseen data. Furthermore, on the possible knowledge transfer of latent learned features. Finally, inference speed on different sizes of data is computed in order to compare the scalability of the models.

## 6.1 Performance Comparison

The most important results are summarized in Tables 3 – 5, where the mean training epoch time in seconds is noted as *time/ep* and the best values are highlighted. The complete python code and the produced results are contained in: <https://github.com/Virtisionis/SelfAttentiveEnergyDisaggregator>.

As shown in Table 3, in Category 1 of Dish Washer the SAED models perform on par with WGRU and Seq2Point. Concerning training time per epoch, SAED is up to 7.1 times faster than WGRU, but with almost same speed as Seq2Point. In Category 2, SAED-dot is the clear winner with similar metric values as the SAED-add model, but with almost half the training time per epoch versus the WGRU. In Category 3 of the same device, the SAED models show similar performance with the Seq2Point but the WGRU is better in the error metrics. In terms of speed, SAED-dot is the fastest. In Category 4 the SAED-add achieves better F1 score and MAE, while SAED-add has the lowest RETE. The general conclusion is that SAED shows promising results on Dish Washer in comparison to the WGRU and the Seq2point, with faster training and better performance in Categories 2 and 4.

In Category 1 of Washing M., SAED-dot is 7.5 times faster than WGRU trading of maximum 10% performance regarding the metrics F1 and MAE. In Category 2, SAED-dot performs on par with WGRU but with 7.5 times faster training time per epoch. The SAED-add has best F1 score in Category 3, while Seq2Point achieves the lowest RETE. In terms of MAE in this category of experiments, the SAED models are better. In the fourth category, the SAED models are trained faster with lower RETE and MAE values.

It is notable that disaggregating Dish Washer and Washing Machine, the SAED models have comparable or better performance with the state-of-the-art models while training time per epoch was up to 7.5 times faster than the WGRU. Also, Seq2Point shows lower performance when disaggregating the Washing Machine.

Results for the Fridge are summed also in Table 3. In Categories 1 and 2, the state-of-the-art models achieve greater F1 score, while SAED-add shows promising results with the smallest RETE and MAE, reaching up to 4 times faster training times versus the WGRU. On the other hand, in Categories 3 and 4 all the models perform the same, indicating the good generalization capabilities of the SAED method.

In Categories 1 and 2 of the Kettle, shown in Table 4, the models have comparable RETE and MAE values, but the WGRU achieves the best F1 score in 7.7 slower training time than the SAED-dot. In the third category of experiments, the WGRU is the winner in terms of F1 and RETE, whereas in MAE all the models perform the same. These results reveal that, comparing to the WGRU, the SAED models show difficulties in disaggregating devices with simple behavior, such as the Fridge and the Kettle. The Kettle is a two-state device, while the Fridge has a finite number of states and repetitive time series. Especially, in Categories 1-2 of the Fridge and the Kettle the SAED has low values on F1 score, but it achieves good results in Categories 3-4 of

**Table 3** Performance Comparison for Dish Washer(DW),Washing M.(WM),Fridge(FZ).

Device	Cat.	Train	Test	Model	F1s	RETE	MAE	time/ep
DW	1	1	1	WGRU	<b>0.33</b>	<b>0.17</b>	13.22	550
				Seq2Point	0.31	0.35	15.44	79
				SAED-dot	0.28	0.31	13.03	77
				SAED-add	0.25	<b>0.17</b>	<b>12.03</b>	141
DW	2	1	2	WGRU	0.26	0.77	37.47	550
				Seq2Point	0.35	0.83	41.77	79
				SAED-dot	<b>0.63</b>	<b>0.62</b>	<b>33.48</b>	77
				SAED-add	0.6	0.63	34.31	141
DW	3	1,2	5	WGRU	0.33	<b>0.34</b>	<b>20.75</b>	575
				Seq2Point	<b>0.35</b>	0.7	40.52	108
				SAED-dot	0.33	0.57	26.45	74
				SAED-add	0.25	0.62	31.01	138
DW	4	1,2	4	WGRU	0.3	0.65	<b>8.6</b>	575
				Seq2Point	0.31	0.2	13.1	101
				SAED-dot	0.45	<b>0.1</b>	12.71	74
				SAED-add	<b>0.53</b>	0.77	<b>8.6</b>	138
WM	1	1	1	WGRU	<b>0.54</b>	<b>0.12</b>	<b>16.55</b>	1097
				Seq2Point	0.25	0.15	18.5	150
				SAED-dot	0.51	0.26	18.51	147
				SAED-add	0.45	0.29	28.55	416
WM	2	1	2	WGRU	<b>0.34</b>	0.43	<b>10.45</b>	1097
				Seq2Point	0.1	0.66	20.57	150
				SAED-dot	0.3	<b>0.34</b>	13.1	147
				SAED-add	0.3	0.53	22.01	416
WM	3	1,5	2	WGRU	0.12	0.36	22.74	585
				Seq2Point	0.14	<b>0.16</b>	17.2	147
				SAED-dot	0.19	0.36	<b>14.66</b>	81
				SAED-add	<b>0.2</b>	0.21	15.18	81
WM	4	1,5	1	WGRU	<b>0.26</b>	0.66	43.65	585
				Seq2Point	0.22	0.54	42.22	147
				SAED-dot	0.18	<b>0.39</b>	50.65	84
				SAED-add	0.18	0.7	<b>41.93</b>	81
FZ	1	1	1	WGRU	<b>0.63</b>	0.27	33.29	562
				Seq2Point	<b>0.63</b>	0.3	33.2	78
				SAED-dot	0.59	0.17	32.78	73
				SAED-add	0.59	<b>0.13</b>	<b>30.56</b>	145
FZ	2	1	2	WGRU	0.82	<b>0.13</b>	28.46	562
				Seq2Point	<b>0.91</b>	<b>0.13</b>	33.43	78
				SAED-dot	0.82	0.21	<b>26.86</b>	73
				SAED-add	0.84	0.23	27.33	145
FZ	3	1,2,4	2	WGRU	<b>0.52</b>	0.18	51.18	519
				Seq2Point	<b>0.52</b>	<b>0.03</b>	<b>49.52</b>	74
				SAED-dot	<b>0.52</b>	0.29	51.35	69
				SAED-add	<b>0.52</b>	0.22	50.52	70
FZ	4	1,2,4	1	WGRU	<b>0.53</b>	0.32	<b>52.57</b>	519
				Seq2Point	0.42	<b>0.27</b>	60.06	72
				SAED-dot	0.49	0.29	50.89	69
				SAED-add	0.5	0.33	51.39	70

the Fridge. The low values of F1 score indicate the difficulty of the models to identify the On/Off states of the test devices. Also, it is notable that SAED performs better than the Seq2Point on the Kettle Categories 1 and 2 and in a tad bit faster training times.

**Table 4** Performance Comparison for Kettle (KT), Microwave(MW), Television(TV).

Device	Cat.	Train	Test	Model	F1s	RETE	MAE	time/ep
KT	1	1	1	WGRU	<b>0.65</b>	<b>0.09</b>	<b>7.35</b>	563
				Seq2Point	0.28	0.24	17.6	79
				SAED-dot	0.44	0.14	8.57	73
				SAED-add	0.34	0.26	9.46	143
KT	2	1	2	WGRU	<b>0.9</b>	0.31	<b>14.04</b>	563
				Seq2Point	0.39	0.36	29.8	79
				SAED-dot	0.62	0.3	19.03	73
				SAED-add	0.49	<b>0.28</b>	17.35	143
KT	3	1,2,3,4	5	WGRU	<b>0.41</b>	<b>0.05</b>	<b>9.92</b>	1096
				Seq2Point	<b>0.41</b>	0.56	10.44	150
				SAED-dot	0.27	0.27	12.24	141
				SAED-add	0.31	0.18	10.95	271
MW	1	1	1	WGRU	<b>0.32</b>	<b>0.09</b>	6.29	560
				Seq2Point	0.22	0.35	<b>6.01</b>	79
				SAED-dot	0.16	0.14	7.51	74
				SAED-add	0.18	0.16	7.61	144
MW	2	1	2	WGRU	<b>0.44</b>	0.25	<b>4.36</b>	560
				Seq2Point	0.37	0.54	5.29	79
				SAED-dot	0.25	0.19	5.97	74
				SAED-add	0.26	<b>0.17</b>	5.98	144
MW	3	1,2	5	WGRU	0.08	0.59	60.53	440
				Seq2Point	0.1	0.555	59.61	41
				SAED-dot	0.21	0.58	56.93	41
				SAED-add	<b>0.22</b>	<b>0.51</b>	<b>59.36</b>	41
MW	4	1,2	1	WGRU	<b>0.41</b>	0.19	23.53	440
				Seq2Point	0.36	<b>0.08</b>	<b>22.68</b>	74
				SAED-dot	0.34	0.2	25.67	41
				SAED-add	0.34	0.15	25.13	41

The results of the experiments on the Microwave are also displayed in Table 4. In Categories 1-2 the WGRU is the clear winner. In the third category of experiments SAED models outperform both the WGRU and Seq2Point, where in Category 4 the WGRU achieves 17% better F1 score than the SAED in 10 times slower training time. Considering that the Microwave is a multi-state device with variable power consumption and on-state duration, the SAED models show descent performance comparing with the state-of-the-art.

Overall, the SAED models achieve good performance in disaggregating multi-state devices instead of simpler devices. Furthermore, the SAED performs good in experiments of Categories 3-4, a fact that reveals the great generalization capability of the proposed models. In addition, Seq2Point seems to perform on par with the WGRU showing faster training times.

As shown in Table 4, in Category 1 of Television the SAED models perform better than the state-of-the-art models showing identical F1 scores alongside with lower MAE errors and faster training times. Furthermore, in Category 2 of experiments, all the models perform on par, with Seq2Point scoring 5% higher F1 measure, while SAED-add achieved lower RETE and MAE alongside with faster training per epoch. It is notable that all the models show better performance in this category of experiments than in the situation where the training and testing was on the same house (Category 1). In Category 3 SAED-dot achieves 34% higher F1 score than the state-of-the-art models. In terms of

RETE all the models perform on par, although concerning the MAE, SAED models score up to 60% lower errors. In the fourth category of experiments, the SAED models show better performance in overall.

Concerning the experiments on the Computer, the results are displayed in Table 5. In Category 1 the SAED-add and Seq2Point perform on par in terms of identifying the On/Off events with up to 35% better F1 score than the WGRU. Regarding the RETE values, WGRU performs better, whereas the MAE scores of all models are nearly the same. Comparing the SAED models results that the SAED-add performs 10% better than the SAED-dot, but with a slower training time per epoch. In Category 2, WGRU shows slightly better performance in comparison to the SAED models. Specifically, in terms of F1 score WGRU achieves 14% better values with maximum value of 66%. In addition, the WGRU achieves lower RETE and MAE values. In Categories 3 and 4 of the same target device, SAED-dot model is a clear winner achieving better F1 and MAE scores, while WGRU shows lower RETE value. In terms of training time per epoch, the SAED-dot is almost two times faster than the SAED-add and up to three times faster than the WGRU in all the categories of experiments. On the contrary, Seq2Point achieves almost equal training times with the SAED-dot, even though it consists of a huge number of parameters.

## 6.2 Generalization Evaluation

To explore on a deeper level the generalization ability of the SAED, in comparison to the WGRU and Seq2Point, a computation of more metrics took place. Table 6 presents the values of AUH, EUH alongside with the corresponding MGL calculations. These metrics are calculated using the F1 scores and MAE measured in the Category 1 of experiments. Because of the size of experiments only some of the measurements are used. To compare the models the interest concentrates on MGL values, where lower means better.

In terms of MGL and Classification Accuracy, the SAED models achieve the lowest values on all the test devices, except the Computer. Thus, SAED shows great generalization ability when detecting on/off events. Also, the negative values of MGL indicate that the SAED models perform better on the unseen houses than on the seen house. Regarding the MGL and Estimation Accuracy, mixed results are observed with the SAED showing finer values than the state-of-the-art models on Washing Machine and Television. As a result, on these test devices, SAED seems to generalize better than the WGRU in terms of power estimation levels. On the Dish Washer and Kettle, Seq2Point shows lower values. The above results strongly highlight the generalization power of the SAED approach in the task of NILM.

## 6.3 Knowledge Transferability Comparison

To explore the knowledge transfer capacity of the SAED method, a transfer learning schema was executed. At first the models are trained on the WM

**Table 5** Performance Comparison for Television(TV), Computer(PC).

Device	Cat.	Train	Test	Model	F1s	RETE	MAE	time/ep
TV	1	6	6	WGRU	<b>0.68</b>	0.49	40.38	145
				Seq2Point	<b>0.68</b>	0.52	42.63	54
				SAED-dot	0.67	0.5	35.68	54
				SAED-add	0.65	<b>0.41</b>	31.67	102
TV	2	6	17	WGRU	0.79	0.36	32.06	144
				Seq2Point	<b>0.8</b>	0.4	32.94	57
				SAED-dot	0.75	0.34	32.41	53
				SAED-add	0.72	<b>0.24</b>	<b>30.15</b>	101
TV	3	6,17	14	WGRU	0.31	0.65	36.5	164
				Seq2Point	0.31	0.65	36.3	102
				SAED-dot	<b>0.47</b>	<b>0.6</b>	<b>14.37</b>	62
				SAED-add	0.39	0.67	15.08	114
TV	4	6, 17	1	WGRU	0.14	0.79	42.21	165
				Seq2Point	0.14	0.72	36.4	102
				SAED-dot	<b>0.56</b>	0.52	<b>9.02</b>	62
				SAED-add	0.49	<b>0.35</b>	9.66	112
PC	1	6	6	WGRU	0.34	<b>0.33</b>	45.97	148
				Seq2Point	<b>0.54</b>	0.46	40.44	53
				SAED-dot	0.43	0.5	44.2	52
				SAED-add	0.51	0.44	<b>40.1</b>	101
PC	2	6	17	WGRU	<b>0.78</b>	<b>0.54</b>	<b>36.52</b>	145
				Seq2Point	0.62	0.62	46.72	50
				SAED-dot	0.67	0.65	52	51
				SAED-add	0.62	0.62	48	100
PC	3	6,17	16	WGRU	0.27	0.7	40.2	169
				Seq2Point	0.27	0.62	30.66	105
				SAED-dot	<b>0.37</b>	<b>0.37</b>	<b>17.05</b>	63
				SAED-add	0.34	0.51	20.52	112
PC	4	6,17	5	WGRU	0.54	<b>0.1</b>	45.36	167
				Seq2Point	0.54	0.19	55.95	105
				SAED-dot	<b>0.71</b>	0.2	<b>32.41</b>	62
				SAED-add	0.62	0.2	35,84	112

appliance. Then, fine-tuning of the network was applied on the target device. Finally, inference was performed on unseen data of the target device. The data for all the stages of this experiment was the same as the Category 1 described in Table 2. The results on KT are presented on Table 6.3. In comparison to the results of Table 5, WGRU shows similar performance, whereas SAED-add and Seq2Point achieved better results. It should be noted that fine-tuning and testing on other devices showed unsatisfying results, indicating that this method should involve devices with similar electric signatures.

#### 6.4 Scalability Comparison

An important and frequently neglected parameter when comparing models is the inference time. Large scale applications involve feeding a disaggregation models with batches of data from many houses. The cost of this application is critical and depends heavily on the scalability of inference time of the model. The scalability is simulated by increasing the time period of disaggregation from one day to 3 months and measuring the inference time for the various sizes of test data. From observing the results in Fig. 3, it is obvious that the

**Table 6** Classification and Estimation Accuracy of the SAED in comparison to the WGRU and the Seq2Point. Seen and Unseen houses are noted as  $S$  and  $U$  correspondingly.

Device	S U	Model	Classification Accuracy			Estimation Accuracy		
			F1s	AUH	MGL[%]	MAEs	EUH[W]	MGL[%]
DW	1 2,5	WGRU	0.33	0.26	19.3	13.22	31.29	136.7
		Seq2Point	0.32	0.32	-4.85	15.44	31.4	<b>103.35</b>
		SAED-dot	0.28	0.48	-72.5	13.03	30.42	133.6
		SAED-add	0.25	0.45	<b>-82</b>	12.03	31.72	163.6
WM	1 2,5	WGRU	0.54	0.29	46.9	16.55	25.02	51.2
		Seq2Point	0.25	0.12	54	18.55	29.14	57.5
		SAED-dot	0.51	0.27	48.1	18.51	23.56	27.3
		SAED-add	0.45	0.26	<b>43.4</b>	28.55	35.1	<b>22.9</b>
FZ	1 2,5	WGRU	0.63	0.69	-9.8	33.3	34.08	<b>2.3</b>
		Seq2Point	0.63	0.67	-5.55	33.2	37.46	12.85
		SAED-dot	0.59	0.69	-18.4	32.78	32.85	3.25
		SAED-add	0.59	0.7	<b>-19.65</b>	30.56	32.68	6.89
KT	1 2,5	WGRU	0.66	0.59	9.9	7.35	24.44	232.5
		Seq2Point	0.24	0.3	-7.15	17.6	27.85	<b>58.25</b>
		SAED-dot	0.44	0.45	-2	8.57	23.49	174.1
		SAED-add	0.33	0.37	<b>-10.4</b>	9.46	21.05	122.5
MW	1 2,5	WGRU	0.32	0.33	-1.7	6.29	12.79	<b>103.5</b>
		Seq2Point	0.22	0.28	-25	6.01	13.34	125.05
		SAED-dot	0.16	0.26	<b>-68.6</b>	7.5	18.07	140.9
		SAED-add	0.18	0.28	-53.9	7.61	17.59	131.2
TV	6 1,17	WGRU	0.68	0.52	24.25	40.38	24.27	-39.05
		Seq2Point	0.68	0.49	28.7	42.63	32.67	-23.35
		SAED-dot	0.67	0.55	18.7	35.68	21.75	<b>-39.7</b>
		SAED-add	0.65	0.53	<b>18.45</b>	31.67	20.67	-34.7
PC	6 16,17	WGRU	0.34	0.54	<b>-58.8</b>	45.97	30.74	<b>-33.15</b>
		Seq2Point	0.54	0.51	5.55	40.44	31.92	-21.1
		SAED-dot	0.43	0.51	-18.6	44.2	34.75	-21.4
		SAED-add	0.51	0.49	3.9	40.1	31.81	-20.65

**Table 7** Knowledge Transferability Comparison for Kettle(KT), on UK-DALE House 1 data.

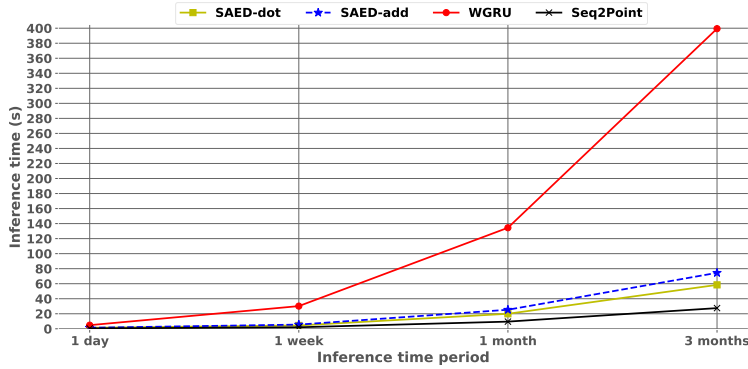
Device	Model	F1	RETE	MAE
KT	WGRU	0.66	0.07	9.04
	Seq2Point	0.54	0.1	8.14
	SAED-dot	0.33	0.18	7.25
	SAED-add	0.56	0.19	7.55

SAED models achieve similar inference times as the Seq2Point, whereas the WGRU is a lot slower.

## 7 Ablation Study

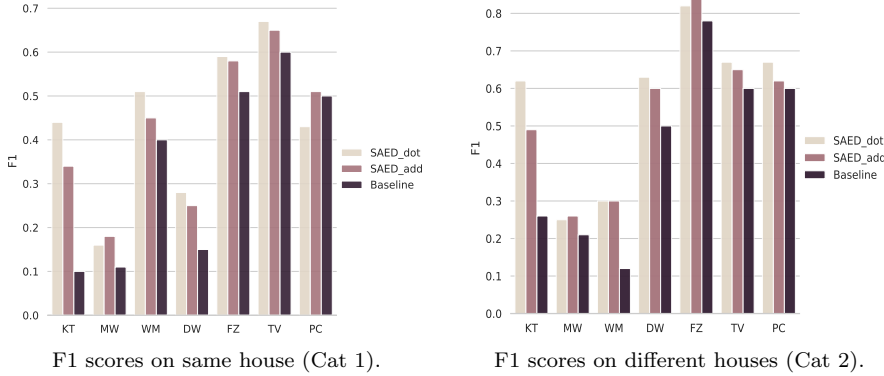
As demonstrated in the previous sections of this article, the attention mechanism provides great generalization capabilities and performance gain to a lightweight neural network. To quantify these enhancements, an ablation experiment was conducted, where the same network (Baseline) is tested on some situations without the attention mechanism. Specifically, the models were compared side by side on experiments of Categories 1 and 2, as described in Table





**Fig. 3** Inference time versus inference time period for fridge, where 1 day of data is equal to 14400 samples.

2. The results are shown in Fig. 4. In overall, the SAED models achieved better F1 scores than the Baseline model on both categories of experiments. In addition, the SAED-dot model achieves the best F1 scores on Category 1, whereas on Category 2 SAED-add achieves best scores on MW and FZ appliances. Regarding the MAE errors, there are mixed results, with the Baseline showing similar performance to the SAED models. The differences between SAED and Baseline models on F1 scores, highlight the fact that the attention mechanism assists the network in the energy changes detection task. Thus the SAED method are more capable on detecting on/off events than the Baseline.



**Fig. 4** Performance comparison of SAED models to a Baseline model.

## 8 Conclusions and Proposals for Future Work

Comparing the proposed SAED models to the lightweight state-of-the art WGRU leads to promising conclusions. In general, the results of the SAED models were comparable and, in some cases, better than the WGRU. Concern-

ing the disaggregation on different devices, SAED achieved better performance on complex devices than on devices with simple time series, although this is not the case when disaggregating Television. Experiments on a wider range of target devices may provide more insights on this topic. In addition, the proposed architecture possesses good generalization capabilities, as pointed out by the results on the Categories 3 and 4 of experiments. Interestingly, in cases of limited data, SAED models show encouraging results, performing better than the WGRU in the majority of cases. The fact that the proposed architecture is faster in training and inference than the WGRU, causes the deployment of SAED models on embedded systems to be more feasible.

After inspecting the performance between SAED method and Seq2Point, interesting conclusions occurred. SAED models perform on par or even better than the Seq2Point in many cases of the experiments. One of the strengths of SAED is the ability to generalize on out-of-distribution data. In terms of speed, Seq2Point achieves almost the same training times per epoch with the SAED-dot (the fastest of the two SAED models), whereas Seq2Point achieves faster inference times. On the other hand, in terms of model size, SAED is significantly smaller. The explanation hides in the structural differences between Convolutional and Recurrent Neural Networks.

Concisely, using the Attention mechanism on lightweight ANN architectures led to the creation of fast-trainable models with good generalization capabilities. As a result, Attention may be a powerful tool in the task of energy disaggregation with Neural Network architectures. In order to achieve even faster training-inference times, Attention could be combined with CNN layers instead of RNNs.

## Declarations

**Funding** - This research has been co-financed by the European Regional Development Fund of the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call *RESEARCH-CREATE-INNOVATE*(project code: *T1EDK-00343(95699)-Energy Controlling Voice Enabled Intelligent Smart Home Ecosystem*).

**Conflicts of interest/Competing interests** - Authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Ethics approval** - Not Applicable

**Consent to participate** - Not Applicable

**Consent for publication** - Not Applicable

**Availability of data and material** - All data presented in the article and supporting information is available from the corresponding authors.

**Code availability** - <https://github.com/Virtsionis/SelfAttentiveEnergyDisaggregator>

**Authors' contributions** - Nikolaos Virtsionis Gkalinikis: Conceptualization, Methodology, Investigation, Writing, Revision; Christoforos Nalmpantis: Conceptualization, Methodology, Coordination, Revision, Supervision; Dimitris Vrakas: Conceptualization, Coordination, Supervision.

## References

1. Armel, K.C., Gupta, A., Shrimali, G., Albert, A.: Is disaggregation the holy grail of energy efficiency? the case of electricity. *Energy Policy* **52**, 213–234 (2013)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. *ICLR* (2015)
3. Basu, K., Debusschere, V., Bacha, S.: Load identification from power recordings at meter panel in residential households. In: 2012 XXth International Conference on Electrical Machines, pp. 2098–2104. *IEEE* (2012)

4. Basu, K., Debusschere, V., Bacha, S.: Residential appliance identification and future usage prediction from smart meter. In: IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society, pp. 4994–4999. IEEE (2013)
5. Batra, N., Kelly, J., Parson, O., Dutta, H., Knottenbelt, W., Rogers, A., Singh, A., Srivastava, M.: Nilmtk: an open source toolkit for non-intrusive load monitoring. In: Proceedings of the 5th international conference on Future energy systems, pp. 265–276 (2014)
6. Cheng, J., Dong, L., Lapata, M.: Long short-term memory-networks for machine reading. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing pp. 551–561 (2016)
7. D’Incecco, M., Squartini, S., Zhong, M.: Transfer learning for non-intrusive load monitoring. IEEE Transactions on Smart Grid **11**(2), 1419–1429 (2020). DOI 10.1109/TSG.2019.2938068
8. Fang, Z., Zhao, D., Chen, C., Li, Y., Tian, Y.: Non-intrusive appliance identification with appliance-specific networks. In: 2019 IEEE Industry Applications Society Annual Meeting, pp. 1–8 (2019). DOI 10.1109/IAS.2019.8912379
9. Firth, S., Kane, T., Dimitriou, V., Hassan, T., Fouchal, F., Coleman, M., Webb, L.: Refit smart home dataset (2017). DOI 10.17028/rd.lboro.2070091.v1. URL [https://repository.lboro.ac.uk/articles/dataset/REFIT\\_Smart\\_Home\\_dataset/2070091/1](https://repository.lboro.ac.uk/articles/dataset/REFIT_Smart_Home_dataset/2070091/1)
10. Hart, G.W.: Nonintrusive appliance load monitoring. Proceedings of the IEEE **80**(12), 1870–1891 (1992)
11. Houidi, S., Fourer, D., Auger, F., Sethom, H.B.A., Miègeville, L.: Comparative evaluation of non-intrusive load monitoring methods using relevant features and transfer learning. Energies **14**(9) (2021). DOI 10.3390/en14092726. URL <https://www.mdpi.com/1996-1073/14/9/2726>
12. Jack, K., William, K.: The uk-dale dataset domestic appliance-level electricity demand and whole-house demand from five uk homes. Sci. Data **2**, 150007 (2015)
13. Kaselimi, M., Doulami, N., Doulami, A., Voulodimos, A., Protopapadakis, E.: Bayesian-optimized bidirectional lstm regression model for non-intrusive load monitoring. In: ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2747–2751 (2019). DOI 10.1109/ICASSP.2019.8683110
14. Kelly, J., Knottenbelt, W.: 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, pp. 55–64 (2015)
15. Kim, H., Marwah, M., Arlitt, M., Lyon, G., Han, J.: Unsupervised Disaggregation of Low Frequency Power Measurements, pp. 747–758. SIAM (2011). DOI 10.1137/1.9781611972818.64. URL <https://epubs.siam.org/doi/abs/10.1137/1.9781611972818.64>
16. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. ICLR (2015)
17. Klemenjak, C., Faustine, A., Makonin, S., Elmenreich, W.: On metrics to assess the transferability of machine learning models in non-intrusive load monitoring. arXiv preprint arXiv:1912.06200 (2019)
18. Klemenjak, C., Makonin, S., Elmenreich, W.: Towards comparability in non-intrusive load monitoring: on data and performance evaluation. In: 2020 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), pp. 1–5. IEEE (2020)
19. Kolter, J.Z., Jaakkola, T.: Approximate inference in additive factorial hmms with application to energy disaggregation. In: Artificial intelligence and statistics, pp. 1472–1482 (2012)
20. Kolter, J.Z., Johnson, M.J.: Redd: A public data set for energy disaggregation research. In: Workshop on data mining applications in sustainability (SIGKDD), San Diego, CA, vol. 25, pp. 59–62 (2011)
21. Krystalakos, O., Nalmpantis, C., Vrakas, D.: Sliding window approach for online energy disaggregation using artificial neural networks. In: Proceedings of the 10th Hellenic Conference on Artificial Intelligence, pp. 1–6 (2018)
22. Kukunuri, R., Aglawe, A., Chauhan, J., Bhagtani, K., Patil, R., Walia, S., Batra, N.: Edgenilm: Towards nilm on edge devices. In: Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, BuildSys ’20, p. 90–99. Association for Computing Machinery, New York, NY, USA (2020). DOI 10.1145/3408308.3427977. URL <https://doi.org/10.1145/3408308.3427977>

23. Luong, M.T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* pp. 1412–1421 (2015)
24. Mahapatra, B., Nayyar, A.: Home energy management system (hems): concept, architecture, infrastructure, challenges and energy management schemes. *Energy Systems* pp. 1–27 (2019)
25. Mauch, L., Yang, B.: A new approach for supervised power disaggregation by using a deep recurrent lstm network. In: *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 63–67. IEEE (2015)
26. Naghibi, B., Deilami, S.: Non-intrusive load monitoring and supplementary techniques for home energy management. In: *2014 Australasian Universities Power Engineering Conference (AUPEC)*, pp. 1–5. IEEE (2014)
27. Nalmpantis, C., Vrakas, D.: Machine learning approaches for non-intrusive load monitoring: from qualitative to quantitative comparison. *Artificial Intelligence Review* **52**(1), 217–243 (2019)
28. Nalmpantis, C., Vrakas, D.: Signal2vec: Time series embedding representation. In: *International Conference on Engineering Applications of Neural Networks*, pp. 80–90. Springer (2019)
29. Nalmpantis, C., Vrakas, D.: On time series representations for multi-label nilm. *NEURAL COMPUTING & APPLICATIONS* (2020)
30. Parson, O., Ghosh, S., Weal, M.J., Rogers, A.C.: Non-intrusive load monitoring using prior models of general appliance types **26** (2012). URL <https://ojs.aaai.org/index.php/AAAI/article/view/8162>
31. Piccialli, V., Sudoso, A.: Improving non-intrusive load disaggregation through an attention-based deep neural network. *Energies* **14**, 847 (2021). DOI 10.3390/en14040847
32. Ruano, A., Hernandez, A., Ureña, J., Ruano, M., Garcia, J.: Nilm techniques for intelligent home energy management and ambient assisted living: A review. *Energies* **12**(11) (2019). DOI 10.3390/en12112203. URL <https://www.mdpi.com/1996-1073/12/11/2203>
33. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* **15**(1), 1929–1958 (2014)
34. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: *Advances in neural information processing systems*, pp. 3104–3112 (2014)
35. Symeonidis, N., Nalmpantis, C., Vrakas, D.: A benchmark framework to evaluate energy disaggregation solutions. In: *International Conference on Engineering Applications of Neural Networks*, pp. 19–30. Springer (2019)
36. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, u., Polosukhin, I.: Attention is all you need. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, p. 6000–6010. Curran Associates Inc., Red Hook, NY, USA (2017)
37. Virtsionis Gkalinikis, N., Nalmpantis, C., Vrakas, D.: Attention in recurrent neural networks for energy disaggregation. In: A. Appice, G. Tsoumakas, Y. Manolopoulos, S. Matwin (eds.) *Discovery Science*, pp. 551–565. Springer International Publishing, Cham (2020)
38. Yue, Z., Witzig, C.R., Jorde, D., Jacobsen, H.A.: Bert4nilm: A bidirectional transformer model for non-intrusive load monitoring. In: *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring, NILM’20*, p. 89–93. Association for Computing Machinery, New York, NY, USA (2020). DOI 10.1145/3427771.3429390. URL <https://doi.org/10.1145/3427771.3429390>
39. Zhang, C., Zhong, M., Wang, Z., Goddard, N., Sutton, C.: Sequence-to-point learning with neural networks for nonintrusive load monitoring. *AAAI* (2018)
40. Zoha, A., Gluhak, A., Imran, M.A., Rajasegarar, S.: Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. *Sensors* **12**(12), 16838–16866 (2012). DOI 10.3390/s121216838. URL <https://www.mdpi.com/1424-8220/12/12/16838>