Machine Learning Techniques for Regression in Energy Disaggregation

Christos Konstantopoulos(✉), Spyros Sioutas, and Konstantinos Tsichlas

Department of Computer Engineering and Informatics, University of Patras,
Patras, Greece
chris.konstanto@upatras.gr, {sioutas,ktsichlas}@ceid.upatras.gr

Abstract. Non-Intrusive Load Monitoring (NILM) or Energy disaggregation may be the holy grail of energy efficiency. The impact of energy disaggregation at the commercial level of home customers is the increased utility customer engagement and the reduced energy usage. The goal at this level is to itemize the consumer's energy bill, analyze the energy usage and cost per household appliance and make personalized and prioritized energy savings recommendations. All these should be viable through a single sensor per household that monitors the total energy consumption and other related quantities. Energy disaggregation is a set of computational approaches for extracting end-use appliance level data from an aggregate energy signal without any plug-level sensors. In the present work, we used a smart meter designed by Meazon S.A. to monitor the energy consumption of a house for 70 days and use basic machine learning methods for regression. To this end, we use an extensive set of features to train our models apart from using only active power. Furthermore, we make comparisons with respect to accuracy and training time between Decision Tree, Random Forest and *k*-NN machine learning methods.

Keywords: NILM · Energy disaggregation · Supervised machine learning · Decision Tree · Random Forest · *k*-NN

1 Introduction

Non-Intrusive Load Monitoring (NILM), also referred to as Energy Disaggregation, was firstly introduced by George W.Hart, Ed Kern and Fred Schweppe in the early 1980s. Household energy consumption signals are analyzed and decomposed into various sub-signals, which correspond to the energy consumption of individual appliances. The difficulties arising in this problem involve the existence of multiple sources of uncertainty, such as "noise" present in the background, multiple devices with almost the same energy consumption and similar behavior as well as appliances with complex energy profiles (e.g., multiple states).

The problem of energy disaggregation can be formulated as follows [1]:

$$X_t = \sum_{i=1}^{N} y_t^i + \sigma(t)$$

Published by Springer Nature Switzerland AG 2022
I. Maglogiannis et al. (Eds.): AIAI 2022, IFIP AICT 646, pp. 356–366, 2022.
https://doi.org/10.1007/978-3-031-08333-4_29

The term $\sigma(t)$ represents devices which contribute to the total consumption at time t but are not taken into account as well as the background noise. The set $X = \{X_1, X_2, ..., X_T\}$ includes the central consumption of N total devices at different time instances $t = \{1, 2, ..., T\}$. Therefore, the objective is to find the contribution y_t^i of the i-th device separately, where $i \in \{1, 2, ..., T\}$.

NILM is constantly gaining attention due to the development of smart grids as well as the development of cheap smart meters with enhanced capabilities. This is also due to the undeniable advantages of NILM allowing us to face certain challenges. Among them the following can be distinguished [15,16]:

- **Detailed information about consumption:** the main advantage for customers is that the analytical energy consumption will allow them to adopt an energy saving behavior. In addition, real-time information on running devices could be a useful tool, i.e., provide reminders to turn off certain devices before consumers leave home, especially those that are likely to cause serious damage or those that require excessive energy.
- **Separate device power consumption:** This allows the consumers to identify the devices that consume the most energy in their home and in general the contribution of each device in the total energy consumption. In fact, [17] estimates savings of 9% to 20% by implementing an energy consumption strategy based on these power analytics.
- **Detection of dysfunctional devices:** a precise device usage archive is useful for checking device status and detecting faulty devices.
- **Illegal load detection:** detection of abnormal loads in households is more accurate and can be used to report potential energy theft in public and private buildings.
- **Environmental intelligence**: allows for other detection approaches without the need to apply new sensors. Instead of turning all devices into smart devices, which is very expensive and not environment-friendly, a single smart meter can provide the necessary information to implement various policies.

1.1 Impact of NILM

Many studies demonstrate the impact of NILM on consumption behavior [2]. This research has shown a potential saving (theoretically) of 15% in energy consumption. However, a later analysis of 36 studies over 15 years [3] shows that it is possible to influence people's decision to reduce their consumption by up to 12%, providing real-time information or even daily or weekly information as depicted in Fig. 1.

Even though the first commercialization of a NILM system was done in 1996 by the company Enetics Inc, the generalized use of the NILM system remains low, as the process is time-consuming, costly due to the difficulties in installation but also highly inaccurate on a larger scale. For these reasons, NILM is still considered unreliable, which prevents its widespread adoption. Nevertheless, there are many start-ups and companies that try to commercialize NILM.

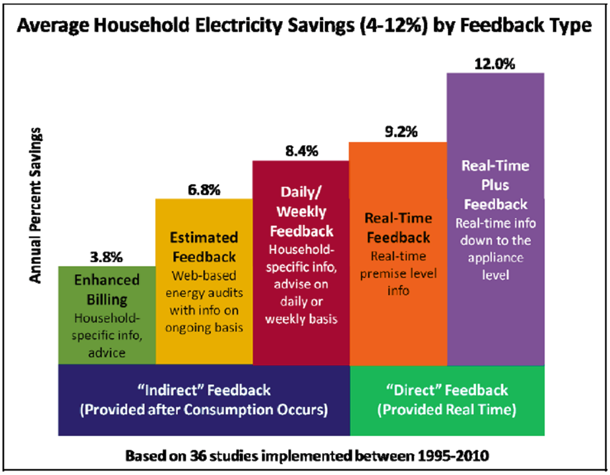

Fig. 1. Analysis of the impact of different feedback methods on behavior change [3]

1.2 Our Contributions

In this paper, we process the data generated by a smart meter (designed by Meazon S.A.) that was installed in the central electricity switchboard of a house. Our goal is to identify the appliances that operate in each time instance and compute the total energy consumed by each appliance. To this end, experiments are carried out using three different machine learning techniques for energy disaggregation from the aggregate signal, which are detailed in Sect. 4. Our contribution lies in the use of a set of features to train our models rather than using only active power. This set contains "Active Power", "Angle between V and I", "Reactive Power" and "Crest Factor". Our goal is to measure and compare the accuracy of the predictions of the methods as well as their performance. After the evaluation of the above experiments with the usual metrics in the Regression field (MSE, MAE, RMSE), an attempt was made to compare these methods in terms of their effectiveness and efficiency.

2 Related Work

Many different algorithmic approaches have been used for NILM that contain among others machine learning, signal processing and deep learning methods with artificial neural networks. The latter are increasingly being used as they have proven their effectiveness in various settings. Initially, one of the most used techniques are different variants of Hidden Markov models (HMMs) such as [4,17,18,20], presenting clearly satisfactory results. Other approaches include signal processing techniques (Dynamic Time Warping) [5,5,21], and Graph Signal Processing [6,7].

Machine learning techniques have been extensively used, such as Decision Trees [21], Random Forests [8,22] or even genetic algorithms like [9,19,23–25]. In addition, Support Vector Machines have been used in [11,26] and in [10]. In [11], the authors experimented with k-NN and Naive-Bayes [10]. Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) have also been used as in [12–14].

Our paper is structured as follows. In Sect. 3 we discuss the dataset and in Sect. 4 we discuss our experimental setup and our methodology with respect to the experimental evaluation. In Sect. 5, we provide our experimental findings while we conclude in Sect. 6.

3 Data Preprocessing

The sampling rate of the smart meter is at 1Hz. In Fig. 2, we observe a sample of the energy consumption (60 min) of all the appliances in the house as well as of the energy consumption of the boiler and the oven separately.

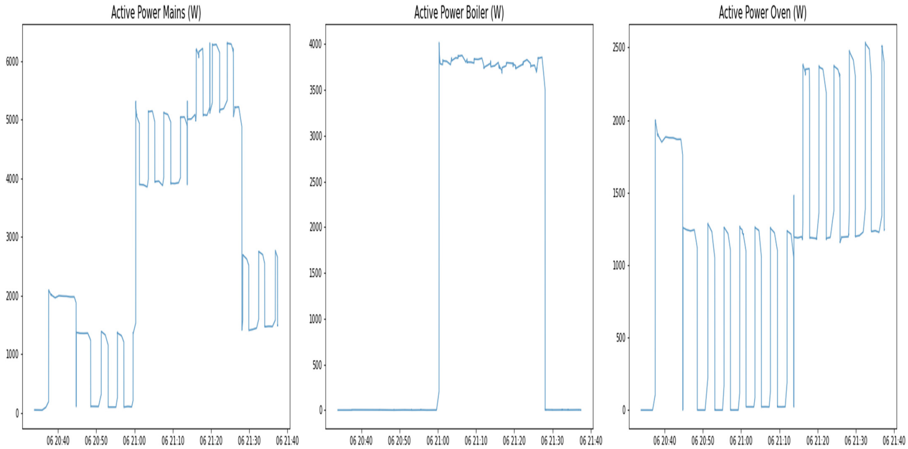

Fig. 2. Energy consumption of appliances from 06/06/21 20:40 to 06/06/21 21:40

The data set includes NaN (Not a Number) values at many positions. In order for the models to be able process the data, it's necessary to solve this missing values problem. We tried three different approaches, at first by replacing NaN with zero values, then with the mean value of each column and at last with the median of each column. The results of the algorithms were almost the same in all three cases so we adopted the median approach. The reason that all methods had the same effect was that the NaN values were existent for very small time intervals, and thus their impact on the effectiveness was non-observable.

The dataset has 2.539.386 measurements (rows) that correspond to 70 days, and dimension of 12 (columns) that correspond to the 3 appliances (mains, oven and boiler) with 4 features each. We manually split the dataset into training, test and validation. The training set has 1.523.632 measurements, the validation set has 507.877 and the test data has 507.877 measurements.

4 Experimental Results

The experiments were performed on a 64-bit operating system with an Intel Core i7-10700 processor and with 16 GB RAM.

We are setting the values of the main consumption on the X variable and the data from the boiler (or oven) on the Y. In order to improve the effectiveness of the algorithms and select the best possible classifiers, some tuning experiments were performed in order to achieve the maximum possible performance. The behavior of each machine learning method is determined by certain parameters.

The tuning experiments aim at selecting the most appropriate values for these parameters, which are evaluated on the basis of evaluation metrics and aim to extract the best prediction model of each algorithm. In addition, to avoid overfitting, the cross-fold validation method is used, which is a sampling process used to evaluate machine learning models in a limited data sample. The basic parameter tested in the DT and RF methods is minimum samples split, which specifies the minimum number of samples needed to separate a node. A range with different values for this parameter was initialized for each method. We execute a repeated procedure and as result we have a unique classifier for each different value of minimum sample split. The best classifier is the one with the lowest value of the evaluation metrics.

4.1 Decision Tree

For DT, the range of values for minimum samples split was between 2 and 400 with step 5. Afterwards, we use the best estimator in order to make predictions and the results are shown in Fig. 3. The performance of the model is observed through graphs, comparing the actual values of the oven consumption (blue line) with those predicted by the model (red line).

In general, the model's predictions are successful and for long time intervals they match the actual consumption, but in some other time intervals the model does not perform well. This may be the result of the complicated behavior of the specific device.

In the next experiment the same model is far more accurate for the boiler as it is confirmed by Fig. 4. As we notice, the model's predictions are quite successful and they match the actual consumption. The first diagram of Fig. 4 depicts the predicted consumption for a complete operation cycle of the appliance (from ON state until OFF state). In the second diagram we can observe in more detail the predicted consumption versus the real consumption during the operation of the boiler (this is why we use a different scale on the y axis). The model is highly accurate and generally follows the changes of the real consumption.

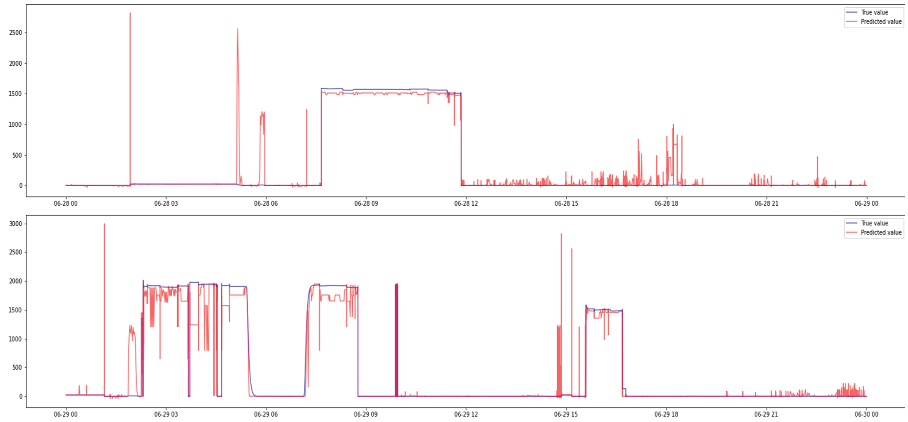

Fig. 3. Real vs Predicted consumption of Decision Trees for the oven.

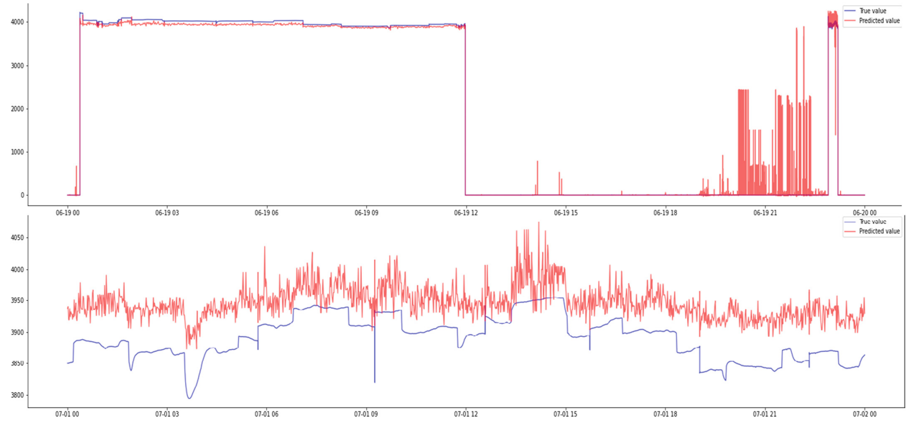

Fig. 4. Real vs Predicted consumption of Decision Trees for the boiler.

4.2 Random Forest

The same set of experiments were also conducted for a Random Forest-based regressor. The only difference is the chosen range of the minimum sample split variable. It is set between 2 and 30 with step 2 for efficiency reasons.

Figure 5 depicts the results of regression with the RF model for the oven. The model's predictions are successful in general and match the actual consumption in many cases. The results are almost the same with the DT model, although we set a lower value for minimum samples split.

Figure 6 presents the experimental results for the boiler. The model's predictions are quite successful and they match to a great extent the actual consumption. The first diagram of Fig. 6, depicts the predictions for the whole operation cycle of the boiler (from the ON state until OFF state). In the second diagram

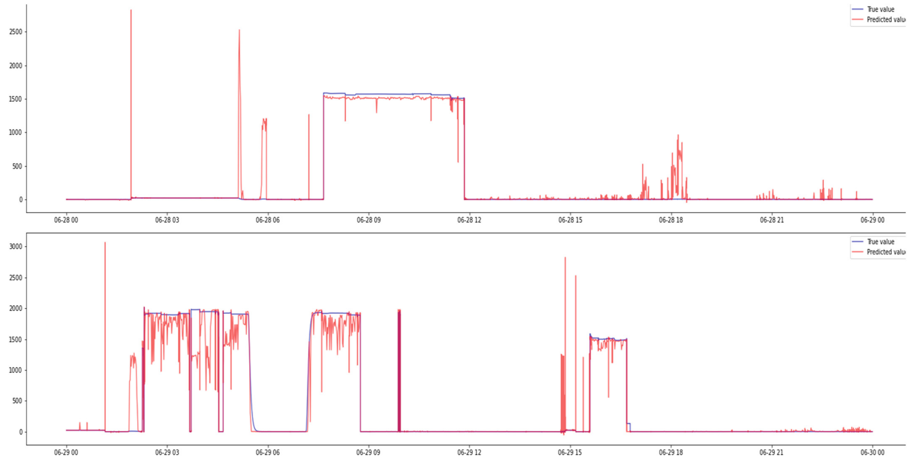

Fig. 5. Real vs Predicted consumption of Random Forest for the oven.

it is depicted in greater detail the predicted versus the actual consumption when the boiler is operating.

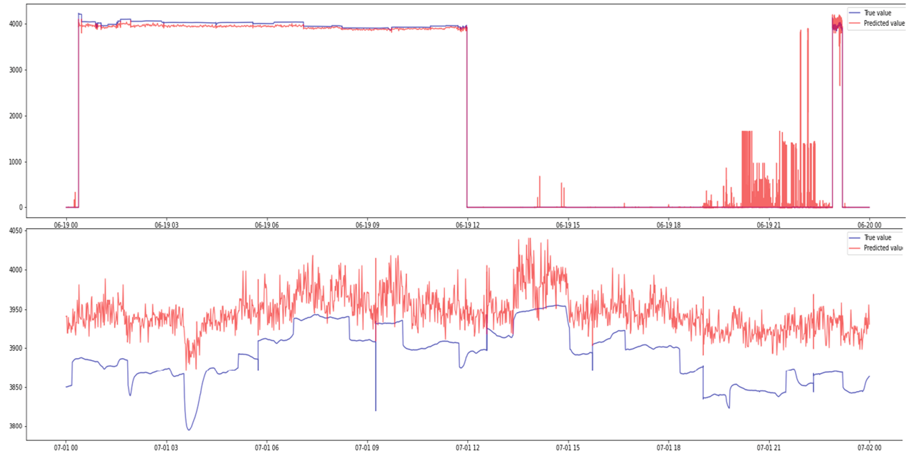

Fig. 6. Real vs Predicted consumption of Random Forest for the boiler.

4.3 *k*-NN

The tuning of the k-NN regressor is related to the number of neighbors. The range of k value is ranging between 2 and 30. Each value returns a different classifier/estimator and we keep the best one by calculating their RMSE value. The best value for the oven is $k = 29$. The results of this model for the oven are depicted in Fig. 7 and for the boiler in Fig. 8. The k-NN model performs similarly to the previously two models. The results for the boiler are better as expected, although the best results were achieved by using $k = 20$ neighbors.

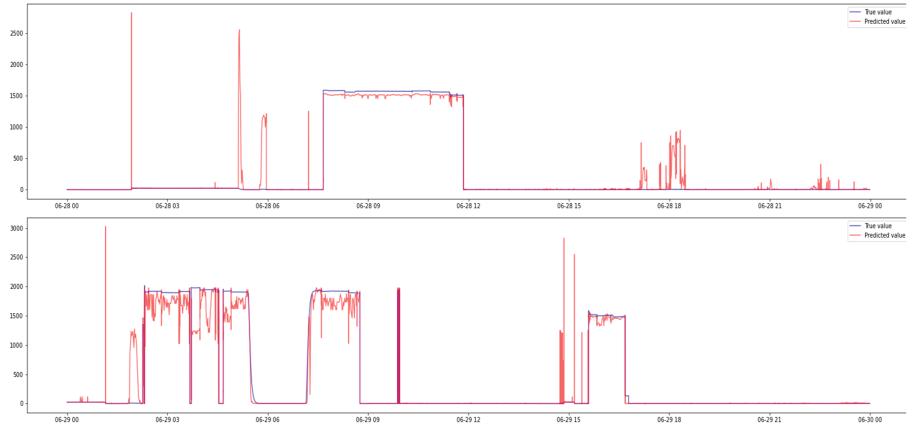

Fig. 7. Real vs Predicted consumption of k-NN for the oven.

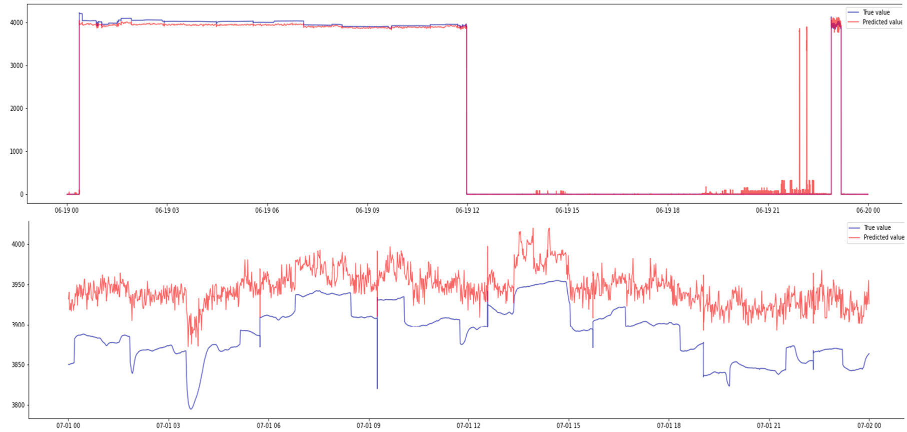

Fig. 8. Real vs Predicted consumption of k-NN for the boiler.

5 Results and Performance

In this section we present the performance of our models in terms of the Mean-Square-Error (MSE), Mean-Average-Error (MAE) and Rooted-Mean-Square-Error (RMSE). In addition, the execution time of tuning, training and regression are reported. In Table 1 we provide the error measures for the oven while in Table 2 we provide the error measures for the boiler.

Table 1. Performance: Decision Tree vs Random Forest vs k-NN (Oven)

Oven application			
Metrics	DT	RF	k-NN
MSE	878.13	882.89	1860.53
MAE	3.25	2.88	3.91
RMSE	29.63	29.71	43.13

Table 2. Performance: Decision Tree vs Random Forest vs k-NN (Boiler)

Boiler application			
Metrics	DT	RF	k-NN
MSE	184.03	179.42	202.8
MAE	1.44	1.43	1.12
RMSE	13.57	13.39	14.24

DT and RF are superior to k-NN, concerning the case of the oven as it can be seen from Table 1. The differences between DT and RF is almost insignificant with respect to accuracy. In the case of the boiler, all three models have the almost the same behavior with the RF marginally outperforming the others. With respect to execution time, DT is significantly faster than RF and k-NN for both appliances. In the case of the oven, DT requires 18 min and 7 s to train in contrast to RF, which needs 1 h, 16 min and 46 s, and k-NN, which is the slowest method, requiring 28 h and 15 min. The same holds for the case of the boiler where DT requires 21 min and 10 s, RF requires 1 h, 39 min and 34 s, and the k-NN requires 28 h and 6 min.

6 Conclusions and Future Work

In the present work, we study the regression problem for energy disaggregation in a real-world dataset. More precisely, we used a smart meter to collect measurements related to the aggregate consumption of a household. Three different supervised machine learning methods are applied: 1) Decision Tree, 2) Random Forest and 3) k-NN. Our main goal and the significant difference to the existing literature, is that we take advantage of a set of automatically (by the smart meter) generated features to train our models. In particular, we do not only use active power data, but our models are also trained with "Angle between V and I", "Reactive Power" and "Crest Factor" features. Our findings indicate that regression in energy disaggregation is possible and these features enhance the accuracy of our models. The experimental comparison shows that DT and RF are far more accurate than k-NN, although the differences between them seems to be insignificant. Concerning the training time, DT is the superior method, outperforming RF and k-NN.

This research constitutes a first step towards a more profound understanding of regression in energy disaggregation. In the future, we intend to focus on extensive experimental evaluation for deep learning methods and more specifically with Neural Networks. Our main focus is to make use of CNNs and RNNs.

Acknowledgements. 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:T2E DK-00127)».

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Ταμείο
Περιφερειακής Ανάπτυξης

ΕΠΑνΕΚ 2014-2020
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ
ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ
ΚΑΙΝΟΤΟΜΙΑ

ΕΣΠΑ
2014-2020
ανάπτυξη - εργασία - αλληλεγγύη

References

1. Anthony, F., et al.: A Survey on Non-Intrusive Load Monitoring Methodies and Techniques for Energy Disaggregation Problem. arXiv:abs/1703.00785 (2017)
2. Fischer, C.: Feedback on household electricity consumption: a tool for saving energy? Energy Effi. **1**, 79–104 (2008). https://doi.org/10.1007/s12053-008-9009-7
3. Karen, E.-M., Kat, D., John, L.: Advanced metering initiatives and residential feedback programs: a meta-review for household electricity-saving opportunities. American Council for an Energy-Efficient Economy (2010)
4. Kolter, J., Johnson, M.: REDD: A Public Data Set for Energy Disaggregation Research. Artificial Intelligence, p. 25 (2011)
5. Liu, B., Luan, W., Yu, Y.: Dynamic time warping based non-intrusive load transient identification. Appl. Energy **195**, 634–645 (2017). https://doi.org/10.1016/j.apenergy.2017.03.010
6. Zhao, B., Stankovic, L., Stankovic, V.: On a training-less solution for non-intrusive appliance load monitoring using graph signal processing. IEEE Access **4**, 1784–1799 (2016). https://doi.org/10.1109/ACCESS.2016.2557460
7. He, K., Stankovic, L., Liao, J., Stankovic, V.: Non-intrusive load disaggregation using graph signal processing. IEEE Trans. Smart Grid **9**(3), 1739–1747 (2018). https://doi.org/10.1109/TSG.2016.2598872
8. Schirmer, P.A., Mporas, I.: Statistical and electrical features evaluation for electrical appliances energy disaggregation. Sustainability **11**, 3222 (2019). https://doi.org/10.3390/su11113222
9. Hock, D., Kappes, M., Ghita, B.V.: Non-intrusive appliance load monitoring using genetic algorithms. IOP Conf. Ser. Mater. Sci. Eng. **366**, 012003 (2018). https://doi.org/10.1088/1757-899X/366/1/012003
10. Yang, C.C., Soh, C.S., Yap, V.V.: A systematic approach in appliance disaggregation using k-nearest neighbours and native Bayes classifiers for energy efficiency. Energ. Effi. **11**(1), 239–259, 012003 (2017). https://doi.org/10.1007/s12053-017-9561-0
11. Altrabalsi, H., Stankovic, V., Liao, J., Stankovic, L.: Low-complexity energy disaggregation using appliance load modelling. AIMS Energy **4**(1), 1–21, 012003 (2016). https://doi.org/10.3934/energy.2016.1.1
12. Jack, K., William, K.: Neural NILM: Deep Neural Networks Applied to Energy Disaggregation (2015). https://doi.org/10.1145/2821650.2821672

13. Xia, M., Liu, W., Wang, K., Xu, Z., Xu, Y.: Non-intrusive load disaggregation based on deep dilated residual network. Electric Power Syst. Res. **170**, 277–285, 012003 (2019). https://doi.org/10.1016/j.epsr.2019.01.034
14. Kim, J., Le, T.-T.-H., Kim, H.: Nonintrusive load monitoring based on advanced deep learning and novel signature. Comput. Intell. Neurosci. **2017**, 1–22, 012003 (2017). https://doi.org/10.1155/2017/4216281
15. Revuelta, H.J., et al.: Non Intrusive Load Monitoring (NILM): A State of the Art. PAAMS (2017)
16. Behzad, N., Sadaf, M., Fabio, R.: Data Analytics for Energy Disaggregation: Methods and Applications (2018). https://doi.org/10.1016/B978-0-12-811968-6.00017-6
17. Kim, H., Marwah, M., Arlitt, M., Lyon, G., Han, J.: Unsupervised disaggregation of low frequency power measurements. Proc. SIAM Conf. Data Mining. **11**, 747–758, 012003 (2011). https://doi.org/10.1137/1.9781611972818.64
18. Mingjun, Z., Nigel, G., Charles, S.: Signal Aggregate Constraints in Additive Factorial HMMs, with Application to Energy Disaggregation. Advances in Neural Information Processing Systems, vol. 4 (2014)
19. Zhang, G., Wang, G., Farhangi, H., Palizban, A.: Residential electric load disaggregation for low-frequency utility applications. In: 2015 IEEE Power and Energy Society General Meeting, pp. 1–5 (2015). https://doi.org/10.1109/PESGM.2015.7286502
20. Parson, O., Ghosh, S., Weal, M., Rogers, A.: Using hidden Markov models for iterative non-intrusive appliance monitoring. Neural Information Processing Systems workshop on Machine Learning for Sustainability, Sierra Nevada, Spain (2011)
21. Georgia, E., Lina, S., Vladimir, S.: Power disaggregation of domestic smart meter readings using dynamic time warping. In: ISCCSP 2014–2014 6th International Symposium on Communications, Control and Signal Processing, Proceedings, pp. 36–39 (2014). https://doi.org/10.1109/ISCCSP.2014.6877810
22. Gong, F., Liu, C., Jiang, L., Li, H., Lin, J.Y., Yin, B.: Load disaggregation in non-intrusive load monitoring based on random forest optimized by particle swarm optimization. In: 2017 IEEE Conference on Energy Internet and Energy System Integration (EI2), pp. 1–6 (2017). https://doi.org/10.1109/EI2.2017.8245609
23. Dominik, E., Anita, S., Wilfried, E.: Evolving Non-Intrusive Load Monitoring (2013). https://doi.org/10.1007/978-3-642-37192-9_19
24. Baranski, M., Voss, J.: Genetic algorithm for pattern detection in NIALM systems. In: 2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583), vol. 4, pp. 3462–3468 (2004). https://doi.org/10.1109/ICSMC.2004.1400878
25. Dominik, E., Wilfried, E.: EvoNILM -evolutionary appliance detection for miscellaneous household appliances. In: GECCO 2013 - Proceedings of the 2013 Genetic and Evolutionary Computation Conference Companion (2013). https://doi.org/10.1145/2464576.2482733
26. Kenneth, B.S.: Model-Driven Analytics of Energy Meter Data in Smart Homes (2014)