



# Computational Intelligence for Characterization and Disaggregation of Residential Electricity Consumption

Mathías Esteban<sup>(✉)</sup>, Ignacio Fiori, Mateo Mujica, and Sergio Nesmachnow

Universidad de la República, Montevideo, Uruguay  
{mathias.esteban,ignacio.fiori,mateo.mujica,sergion}@fing.edu.uy

**Abstract.** This article presents the application of computational intelligence techniques for the characterization of electricity consumption in households. A specific variant of the residential energy disaggregation problem is solved, which proposes identifying the state changes in a electrical network, using a time series of aggregate household power consumption data. This article introduces and compares three classifiers to solve the problem: a Naive Bayes classifier, a K Nearest Neighbors algorithm, and a Long Short Term Memory neural network. The implemented classifiers are evaluated using the UK-DALE data repository. Experimental results show that the Long Short Term Memory network is the most accurate to deal with the characterization problem, achieving a successful rate of state changes up to 75% and values of *F1-score* close to 1.0 on certain appliances.

**Keywords:** Computational intelligence · Energy efficiency · Disaggregation

## 1 Introduction

Residential electricity consumption represents an important share of the energy demand worldwide. Some appliances contribute the most to electricity consumption (air conditioning  $\sim 30\%$ , electronic and kitchen appliances  $\sim 20\%$ , water heaters  $\sim 20\%$ , and cooling  $\sim 10\%$ ) [12]. Thus, energy management is a crucial issue in nowadays smart cities. Many strategies have been proposed to guarantee an increased access to the energy resources at affordable costs for citizens, while ensuring the preservation of natural resources and the protection of the environment [11]. In the search of a better use of energy resources, having accurate information about household electricity consumption is very useful to conceive energy plans and tariffs adapted to different sectors of the population and provide personalized advice towards achieving a better use of energy resources.

Installing individual meters to get the electricity consumption of each appliance in every house of a city is not a viable option nowadays. Thus, it is necessary to develop and apply techniques capable to offer this information using

the aggregate power consumption data that energy providers collect via global smart meters. This kind of techniques rely on the analysis of aggregate power signal to disaggregate the individual appliances loads [2].

In this line of work, this article presents the application of computational intelligence methods for solving the disaggregation problem by recognizing on/off appliances states and state changes using the aggregate consumption signal, and determine energy consumption patterns. Three methods are compared to solve the problem: a Naive Bayes classifier, a K Nearest Neighbors algorithm, and a Long Short Term Memory (LSTM) neural network. The experimental evaluation of the proposed algorithm is performed over synthetic datasets, specifically built using real energy consumption data from the well-known UK-DALE repository [6]. The main results demonstrate that the proposed LSTM neural network is able to compute accurate solution to the problem, achieving a successful rate of state changes up to 75% and values close to 1 of the of *F1-score* learning metric on certain appliances.

The research is developed within the project “Computational intelligence to characterize the use of electric energy in residential customers”, funded by the National the Uruguayan government-owned power company (UTE) and Universidad de la República, Uruguay. The project proposes the application of computational intelligence techniques for processing household electricity consumption data to characterize energy consumption, determine the use of appliances that have more impact on total consumption, and identify consumption patterns in residential customers.

The main contributions of this article include the design and implementation of three classifiers to solve the stated variant of the energy consumption disaggregation problem in residential households, and their experimental evaluation over a set of problem instances.

The article is structured as follows. Section 2 presents the main concepts related to energy disaggregation. A review of the main related work is presented in Sect. 3. The studied computational intelligence methods for energy consumption disaggregation are described in Sect. 4. The experimental evaluation of the proposed methods is reported in Sect. 5. Finally, Sect. 6 presents the conclusions and the main lines of future work.

## 2 Energy Disaggregation in Households

This section presents the main details about the energy disaggregation problem.

### 2.1 Overall Description

The energy disaggregation problem consists of disaggregating the overall energy consumption of a house into the individual consumption of a number of appliances. Energy disaggregation is a special case of a classification problem.

The problem was defined by Hart [5]. The goal of the problem is to identify which appliances are switched on and off in an electrical grid over a period of time

by analyzing the aggregate power measures. The main idea is that appliances operate at certain power levels during specific periods of time and change the aggregate signal in a specific way, therefore produce a characteristic signature. Other variables related to appliance usage may be taken into consideration, such as day of the week, temperature and weather information. This kind of techniques are called non-intrusive because do not require physical intervention on the grid components.

In this article, the binary variant of the problem is addressed, where appliances could be either on or off. The distinction between one state and the other is computed according to a set of threshold values called *standby consumption*, specific for each appliance. A more complex variant of the problem consists of accounting for different consumption states in the considered appliances. This is a significantly harder approach compared to the binary variant.

## 2.2 Mathematical Model

Consider a set of appliances available in a house  $E = \{e_1, e_2, e_3 \dots e_n\}$  and let  $X_t$  be the aggregate power consumption of the house at a given time step  $t$ .  $x_t$  can be expressed as the sum of the individual power consumption  $X_t^i$  of each appliance in use in that time step, according to Eq. 1, where  $y_i$  is the power demand of appliance  $e_i$ ,  $a_i(t)$  is a Boolean function returning weather or not appliance  $i$  is on at time  $t$ , and  $o(t)$  represents the noise in the grid.

$$X_t = \sum_{i=1}^n a_i(t)y_i + o(t) \quad (1)$$

The simplest (binary) variant of the problem assumes just two possible values for the power consumption of each appliance, i.e.,  $x_t^i = c_i \times a_i(t)$ , that is to say that the power consumption of appliance  $i$  is constant and does not depend on the activity being performed by the appliance.

Under the aforementioned conditions, the disaggregation problem consists in finding values  $a_i(t)$  associated to every appliance  $e_i$ .

## 3 Related Work

Several articles in the related literature have proposed methods for energy consumption disaggregation. The main related works are reviewed next.

One of the first publications on monitoring of energy consumption of household appliances was developed by Hart [5], who presented the concept of Non-Intrusive Load Monitoring (NILM), defined the energy disaggregation problem, and introduced a theoretical framework to standardize studies to solve it using software products. Instead of relegating the software to the task of data collection, Hart proposed an approach based on using a simple hardware and complex software for the analysis, thus eliminating permanent intrusion in homes.

The model proposed by Hart involves a combinatorial optimization (CO) problem to determine the vector  $a(t)$  from  $X_t$  and  $y_i^t$ , in order to minimize the error (Eq. 2).

$$\hat{a}(t) = \arg \min_a \left| X_t - \sum_{i=1}^n a_i(t) \cdot y_i^t \right| \quad (2)$$

Given that the proposed combinatorial optimization problem is NP-hard, heuristic algorithms and other soft computing methods have been applied to efficiently solve it. Even though heuristics allow computing reasonable solutions, their applicability is limited in practice due to the uncertainty on the data and the high sensitiveness of the problem. Another approach consists in treating NILM as a learning problem, to be solved with computational intelligence methods, either applying supervised or unsupervised learning.

The survey by Bonfigli et al. [2] described the main techniques used for the unsupervised NILM problem and the available datasets. One of the most popular unsupervised learning techniques is Hidden Markov Models (HMM), based on defining hidden states for the problem model, representing the operating state of each appliance (on, off, other intermediate states), and the observable result that accounts for the real state, according to the consumption data.

Kelly and Knottenbelt [6] studied Artificial Neural Networks (ANN) for disaggregation in the NILM problem. One ANN was trained for each appliance, to predict the power demanded by it. Three ANN architectures were studied: recurrent LSTM, denoising autoencoder (dAE), and a rectangle network. The ANN were trained using synthetic and real data from the UK-DALE dataset. dAE and the rectangle network outperformed baseline CO and FHMM methods in F1-score, precision, and mean absolute error. LSTM outperformed CO and FHMM in on/off appliances but had poor results in multi-state appliances.

Batra et al. [1] developed the `nilmtk` tool for handling energy consumption datasets and implementing algorithms for processing and disaggregation, and metrics. The REDD dataset for energy disaggregation was collected and introduced by Kolter and Johnson [7], who also studied HMM model for disaggregation. Prediction in the training set was 64.5%, while prediction in the evaluation set was just 47.7%.

Our recent articles [3, 4] presented a pattern detection algorithm to solve the energy disaggregation problem using aggregate consumption data. The experimental evaluation was performed over scenarios with different complexity. Results of the proposed pattern similarity method outperformed two built-in algorithms in the `nilmtk` framework, based on CO and a FHMM model.

## 4 Computational Intelligence Methods for Disaggregation

This section presents the proposed computational intelligence methods applied to solve the energy disaggregation problem.

### 4.1 Naive Bayes

Naive Bayes is a supervised classifier that makes predictions based on Bayes theorem. Naive Bayes assumes conditional independence between every pair of features, hence the “naive” term. If  $C$  is a set of possible classes for a vector of features  $X = (x_1, x_2, \dots, x_n)$ , the prediction  $c'$  is given by Eq. 3.

$$c' = \operatorname{argmax}_{c \in C} P(c) \prod P(x_i|c) \quad (3)$$

In this article, the Naive Bayes Gaussian implementation for  $P(x_i|c)$  in scikit-learn (scikit-learn.org) is used.

### 4.2 K Nearest Neighbors

The K Nearest Neighbors (KNN) algorithm classifies a vector  $X = (x_1, x_2, \dots, x_n)$  to the most repeated class in the  $k$  nearest neighbors of  $X$ , being  $k$  a highly sensitive parameter. In this article, the Euclidean distance between data points is used and neighbors relevance is weighted, so the closest to the sample have higher influence in the prediction.

### 4.3 Long Short Term Memory

LSTM neural networks are special kind of Recurrent Neural Networks (RNN). Recurrent networks differ from traditional feedforward ANN by accepting backwards connections and loops between neurons. This feature makes RNN specially good for time series analysis. However, because of backward connections, RNN are prone to suffer from the vanishing gradient problem. LSTM avoid this problem by replacing traditional neurons with more complex elements called cells, capable of remembering and forgetting pieces of information over iterations.

In this article, a five layer LSTM was implemented using the Keras library (keras.io). The implemented LSTM has the following architecture, selected after preliminary experiments that studied different configurations:

- An input layer, whose number of neurons is determined by the number of considered features of the problem.
- One LSTM bidirectional layer with 128 cell, TanH activation.
- One LSTM bidirectional layer with 256 cell, TanH activation.
- One LSTM simple layer with 256 cell, TanH activation.
- An output layer with one neuron fully connected with sigmoid activation.

The LSTM network is trained using Adam optimizer during 15 epochs with binary cross entropy as loss function. Different windows sizes were used for each application in the experimental evaluation.

## 5 Experimental Evaluation

This section reports the experimental evaluation of the proposed computational intelligence methods for energy disaggregation.

## 5.1 Evaluation Methodology

*Main Goals.* There are two main goals concerning experimental evaluation. The first one is to find the most suitable method for the energy disaggregation problem by comparing the achieved results. The second one aims to identify the features that best describe appliances behavior.

*Evaluated Features.* The main features in the considered variant of the NILM problem are the aggregate power and the difference of aggregate power between consecutive measurements. The main goal is to learn unique fluctuations in the signal that could be associated with a certain appliance being switched on or off. However, it is also interesting to evaluate if including other features to learn from lead to an improvement on the classification results. In this regard, the proposed approach complements raw aggregate power data with appliance-oriented features and human-oriented features. Appliance oriented features try to capture information about the appliances specific work cycle, for example, features like previously on, off-time and on-time. In turn, human-oriented features (e.g., hour, type of day, etc.) seek to capture the routine behavior of users.

In this article, the following features are considered:

- *Aggregate power:* The aggregate power consumption of the household.
- *Previously on:* A boolean value that indicates if the appliance was on on the previous time step.
- *On time:* The time in seconds that the appliance has been ON.
- *Off time:* The time in seconds that the appliance has been OFF.
- *Weekend:* A boolean value that indicates if the sample is from a weekend day or not.
- *Hour:* An integer value that represents the hour the sample has been taken.
- *Delta power:* The aggregate power difference between samples in consecutive timesteps.

*Evaluation Scenarios.* The studied computational intelligence methods are evaluated in five different scenarios that consider different features. Labeled data is required for supervised learning. Thus, each input vector should be assigned to either ON class or OFF class. Since UK-DALE is a timestamp-based dataset with different channels for appliances and total power, a mechanism must be established to label data. This article explores two labeling approaches. The first one only take into account aggregate samples that have matching measurements in the appliance data. This is a precise method, but tend to discard too many samples. On the other hand, as neural networks require as much data as possible to improve learning, a discrete time approach is also explored in order to avoid data loss. In this approach, the dataset is iterated over a time window, and each aggregate power window is labeled based on the majority of individual measurements in the appliance time windows.

Table 1 summarize the details of the five scenarios designed for the experimental evaluation, each of them using a different subset of the input features or a different approach to label data (continuous/discrete). Scenarios 1, 2, and 3

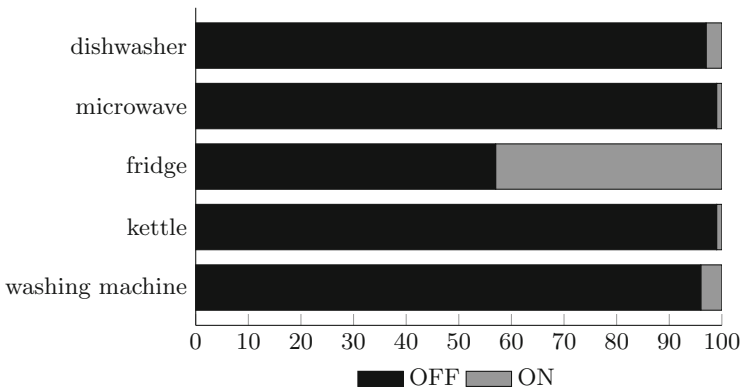
**Table 1.** Description of scenarios for the experimental evaluation.

Feature	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Aggregate power	✓	✓	✓	✓	-
Previously on	✓	-	✓	✓	✓
OFF time	✓	-	✓	✓	✓
ON time	-	-	-	-	✓
Weekend	✓	✓	✓	✓	✓
Hour	-	-	✓	✓	✓
Delta power	-	-	-	-	✓
Discrete time	-	-	-	✓	✓

only take into account matching samples between individual data and aggregate data from the dataset, and continuous time is considered. On the other hand, scenarios 4 and 5 are based on discrete time steps, considering an appliance is ON at certain step if the majority of its samples surpass the standby value.

*Metrics.* Well-known metrics for classification are used: accuracy, recall, precision and f1-score [10]. In the considered problem variant, in addition to the general classification results, it is also interesting to evaluate the capability of the studied models to detect the state changes, i.e., when appliances are turned ON or OFF. Thus, the percentage of correctly predicted state changes (*%psc*) is also reported.

*Data Instances.* The data samples used in this work were gathered from the UK-DALE dataset, which includes records of the aggregated and individual power demand from five houses of the United Kingdom, sampled at 1/6 Hz between 2012 and 2017.

**Fig. 1.** Data distribution in ON/OFF classes.

In the considered instances, the five appliances that contributes the most to the overall electricity consumption are selected for the experimental evaluation: dishwasher, microwave, fridge, kettle and Washing machine. Figure 1 shows the data distribution among classes ON and OFF for each appliance.

*Development and Execution Platform.* The studied classifiers were implemented in Python 3.6 using Keras and scikit-learn libraries. The experimental analysis was performed on high-end servers with two Xeon Gold 6138 processors (40 cores each), Nvidia Tesla P100 GPUs (12 GB memory), 128 GB RAM memory, and 10 GbE from National Supercomputing Center (Cluster-UY), Uruguay [9].

*Training.* Input data was divided into three data sets for training (80%), validation (10%), and testing (10%). Scenarios 1–3 include 2702444 energy consumption records for dishwasher, 2466515 for microwave, 2343539 for fridge, 2665720 for kettle samples, and 2583668 for washing machine. On the other hand, scenarios 4 and 5 include 4531129 energy consumption records for dishwasher, 4467068 for microwave, 4464742 for fridge, 4394752 for kettle samples, and 4494437 for washing machine. Experiments to find the best parameter configuration of the learning methods were performed on a validation instance. In KNN, parameters *neighbors.number* and *weights* (whether or not neighbor relevance is pondered by distance) were tuned. In LSTM networks, the timestamp (number of past

**Table 2.** Parameter configuration.

Appliance	Scenario	Neighbor_number	Weights	LSTM timesteps
Dishwasher	1	5	Uniform	-
	2	7	Distance	-
	3	7	Uniform	-
	4	-	-	5
	5	3	Uniform	3
Microwave	1	5	Uniform	-
	2	5	Uniform	-
	3	7	Uniform	-
	4	-	-	5
	5	3	Uniform	5
Fridge	1	5	Uniform	-
	2	7	Uniform	-
	3	5	Uniform	-
	4	-	-	3
	5	5	Uniform	3
Kettle	1	7	Uniform	-
	2	5	Uniform	-
	3	7	Uniform	-
	4	-	-	3
	5	7	Uniform	3
Washing machine	1	5	Uniform	-
	2	3	Uniform	-
	3	7	Uniform	-
	4	-	-	3
	5	3	Uniform	10



samples in memory) value was tuned. Table 2 reports the final values for each appliance and scenario.

## 5.2 Numerical Results

Tables 3, 4 and 5 reports the evaluation results for the studied classification methods over representative scenarios. The analysis focuses on the f1-score and %cpsc metrics, which provide quantitative information to assess the efficacy of the proposed methods. The best results for these metrics in each appliance are marked in bold face.

Table 3 reports the results of the Naive Bayes method in scenarios 1, 2, 3 and 5. These scenarios were considered in the evaluation in order to evaluate features impact on results (scenarios 1–3) and to compare against LSTM (scenario 5).

**Table 3.** Results of the Naive Bayes approach.

Appliance	Scenario	Accuracy	Precision	Recall	f1-score	% <i>cpsc</i>
Dishwasher	1	0.97	0.45	0.37	0.41	11
	2	0.97	0.43	0.37	0.40	11
	3	0.97	0.53	0.49	0.50	7
	5	0.99	0.80	0.88	0.84	5
Microwave	1	0.97	0.05	0.78	0.10	48
	2	0.96	0.03	0.60	0.06	<b>50</b>
	3	0.97	0.10	0.67	0.10	28
	5	0.99	0.18	0.79	0.29	29
Fridge	1	0.97	0.96	0.96	0.96	0
	2	0.58	0.60	0.04	0.08	2
	3	0.99	0.98	0.98	0.98	0
	5	0.98	0.98	0.98	0.98	0
Kettle	1	0.99	0.17	0.91	0.28	75
	2	0.99	0.17	0.91	0.28	75
	3	0.97	0.17	0.88	0.29	67
	5	0.99	0.35	0.69	0.47	41
Washing machine	1	0.96	0.31	0.36	0.33	4
	2	0.95	0.25	0.36	0.29	4
	3	0.96	0.37	0.35	0.36	3
	5	0.98	0.69	0.60	0.64	2

Results in Table 3 indicate that the Naive Bayes method just computed accurate values of the f1-score metric for the fridge in three out of four scenarios, while acceptable results were obtained for the dishwasher and the washing machine in just one scenario. Regarding the state changes, Naive Bayes only computed accurate results for kettle in three scenarios and for microwave in a single scenario. Results for other metrics (in particular, precision and recall) and other appliances were not accurate, suggesting that a simple Bayesian approach is not enough to solve the problem.

Table 4 reports the  $K$  Nearest Neighbors results for scenarios 1, 2, 3 and 5.

**Table 4.** Results of the  $K$  Nearest Neighbors method.

Appliance	Scenario	Accuracy	Precision	Recall	f1-score	% cpsc
Dishwasher	1	0.99	0.95	0.97	0.96	41
	2	0.97	0.61	0.17	0.26	6
	3	0.99	0.96	0.97	0.97	19
	5	0.99	0.97	0.98	0.97	31
Microwave	1	0.99	0.78	0.69	0.73	41
	2	0.99	0.37	0.25	0.30	19
	3	0.99	0.88	0.74	0.81	24
	5	0.99	0.68	0.65	0.67	22
Fridge	1	0.97	0.97	0.96	0.97	55
	2	0.64	0.57	0.62	0.60	37
	3	0.98	0.97	0.98	0.98	15
	5	0.99	0.98	0.98	0.98	49
Kettle	1	0.99	0.77	0.61	0.68	47
	2	0.99	0.35	0.19	0.25	20
	3	0.99	0.85	0.73	0.79	38
	5	0.99	0.86	0.82	0.84	53
Washing machine	1	0.99	0.75	0.82	0.78	23
	2	0.95	0.21	0.18	0.19	9
	3	0.99	0.83	0.87	0.85	11
	5	0.99	0.86	0.90	0.88	21

Results in Table 4 indicate that the  $K$  Nearest Neighbors method computed accurate results for the dishwasher and the fridge (for both appliances, in three out of four scenarios). Acceptable results were computed for microwave, kettle, and washing machine (three out of four scenarios for each appliance). Results in scenario 2 were significantly lower than for other scenarios. The efficacy on scenario 5 was significantly better than the Naive Bayes method, except for the fridge (where both methods computed accurate results). Regarding state

changes,  $K$  Nearest Neighbors improved over Naive Bayes, but just managed to achieve the better result in one scenario for the dishwasher. Correctly predicted state changes for fridge were significantly better than using Naive Bayes. These results indicate that, overall,  $K$  Nearest Neighbors is a better method to solve the problem when compared with Naive Bayes.

Table 5 reports LSTM results for scenarios 4 and 5. The experimental evaluation of LSTM focused on discrete time scenarios, because continue time scenarios (1–3) have a significantly lower number of training samples and LSTM performs poorly when few data is available due to the lack of a proper training.

**Table 5.** Results of the Long Short Term Memory neural network.

Appliance	Scenario	Accuracy	Precision	Recall	f1-score	% cpsc
Dishwasher	4	0.99	0.97	0.97	0.97	35
	5	0.99	0.99	0.97	<b>0.98</b>	35
Microwave	4	0.99	0.92	0.72	<b>0.81</b>	28
	5	0.99	0.80	0.77	0.78	40
Fridge	4	0.99	0.99	0.99	0.99	63
	5	0.99	0.99	0.99	<b>0.99</b>	<b>71</b>
Kettle	4	0.99	0.95	0.87	0.91	69
	5	0.99	0.94	0.90	<b>0.92</b>	<b>75</b>
Washing machine	4	0.99	0.88	0.92	0.90	24
	5	0.99	0.90	0.93	<b>0.91</b>	<b>31</b>

Results in Table 5 indicate that the LSTM neural network was able to compute accurate results for both studied scenarios, for four of the five studied appliances. Results of the f1-score metric were over 0.97 for dishwasher, 0.99 for fridge, over 0.91 for kettle, and over 0.90 for washing machine. The lowest f1-score values were computed for the microwave (0.81 and 0.78), which turns to be the home appliance with the highest variability of duration and utilization period. Similar results were computed for the other learning metrics, in which LSTM consistently outperformed both Naive Bayes and  $K$  Nearest Neighbors methods. The number of correctly predicted state changes significantly improved over the other reference learning methods too, achieving maximum of 75 for kettle and 71 for fridge.

*Overall Results.* Overall, results reported in Tables 3, 4 and 5 indicate that the studied methods computed different f1-score values and %cpsc for each studied appliance. For the dishwasher, the highest f1-score was 0.98, achieved by the LSTM neural network in scenario 5, whereas the highest percentage of state change detection was 41%, reached by KNN in scenario 1. In microwave, both KNN and LSTM computed the maximum value of f1-score (0.81) in scenarios 3 and 4, respectively. In turn, the highest state change percentage was achieved by

Naive Bayes in scenario 1. The LSTM neural network in scenario 5 achieved the highest values of f1-score and state change detection percentage for the fridge (0.99 and 71%, respectively), the kettle (0.92 and 75%, respectively), and the washing machine (0.91 and 31%, respectively). Overall, LSTM computed the best results for discrete time scenarios and KNN was the best method for continuous time scenarios. These results suggest that if only small data sets are available for training, LSTM neural networks are not able to learn properly the utilization patterns. In these cases, KNN should be used as a promising alternative, as it provides accurate results.

The graph bars in Figs. 2, 3, 4, 5 and 6 presents the comparison of the f1-score and %cpsc metrics for each appliance, for the methods and scenarios considered in the experimental evaluation (labels of the  $x$  axis correspond to [method]-s[scenario]).

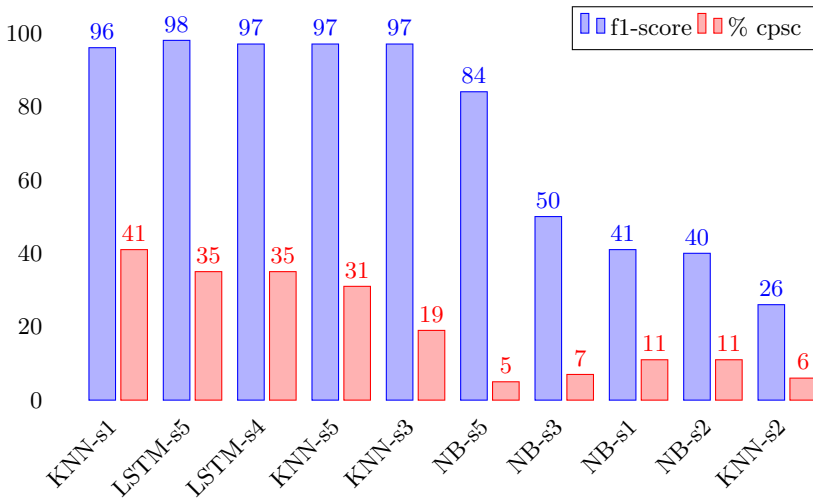
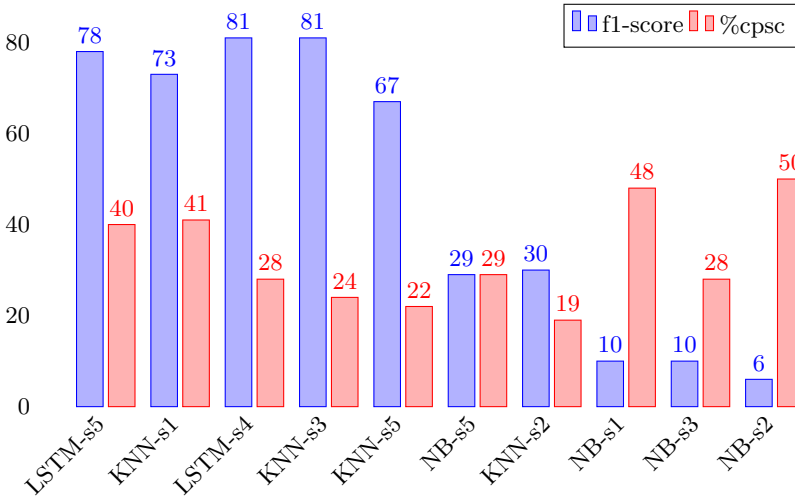


Fig. 2. Comparison of f1-score and %cpsc for the dishwasher.

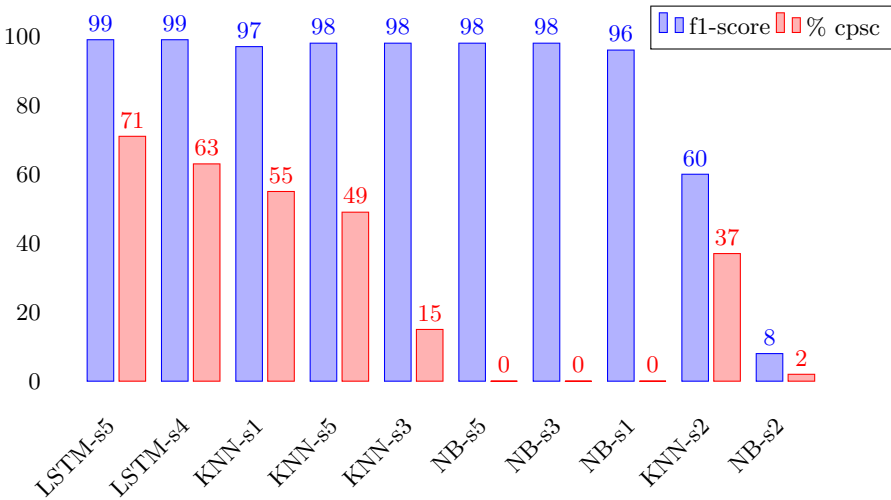
### 5.3 Summary

Several relevant conclusion from the analysis are summarized next.

*From the Appliance Perspective:* Target appliances have different properties (utilization time, utilization period, power demand, etc.) that impact on the classification task. A single pair of one classifier and one feature set may not be the



**Fig. 3.** Comparison of f1-score and %cpsc for the microwave.



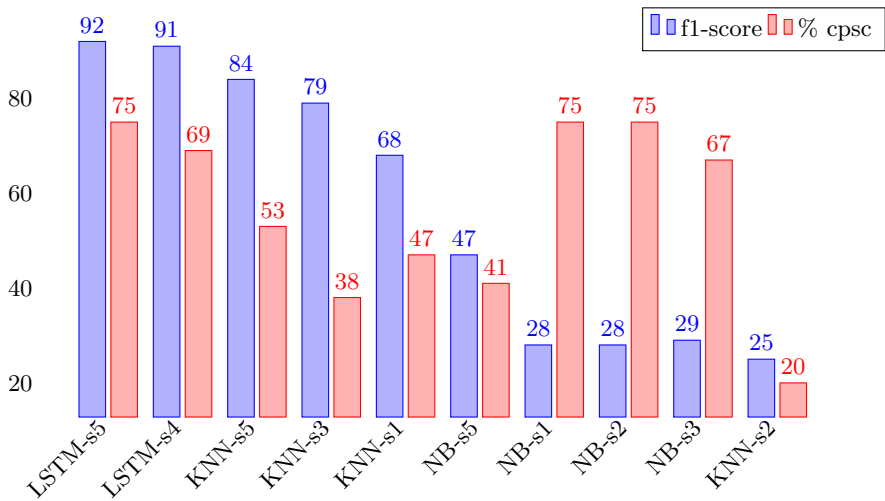
**Fig. 4.** Comparison of f1-score and %cpsc for the fridge.

best option to solve the problem for every appliance, so it is relevant to study results for each appliance separately.

Washing machine and dishwasher are human-activated appliances that have similar proportion of ON and OFF classes in the considered dataset. Both have multiple consumption levels over a single work cycle that could lead to misclassify power fluctuations as activations of other appliances in the house. On the other hand, once these appliances are switched on, they tend to operate during a fixed time. Results show that classification metrics were high for both KNN and LSTM. However, the percentage of correctly predicted state changes was poor compared to other appliances like fridge or kettle. Fridge activations are not human dependent. This appliance also has a well defined consumption pattern, with a balanced distribution between OFF and ON classes. This behavior explains high values of f1-score and correct state change predictions in LSTM networks. The kettle is a high power demand appliance that once is switched on, works over a short period of time. High values in both f1-scores and state change predictions could be associated with the impact this appliance has in aggregate signal. In turn, poor results achieved at microwave classification could be explained as it has the most variable working pattern of all appliances.

*From the Method Perspective:* In general, results show that the best performance among appliances is achieved by LSTM neural networks. This fact confirms the LSTM pattern recognition potential, especially considering that, in general, the next state of an appliance is determined by previous behaviour.

*From the Feature Perspective:* Poor results in scenario 2 confirmed that including features about previous states of appliances as input allows improving the



**Fig. 5.** Comparison of f1-score and %cpsc for the kettle.

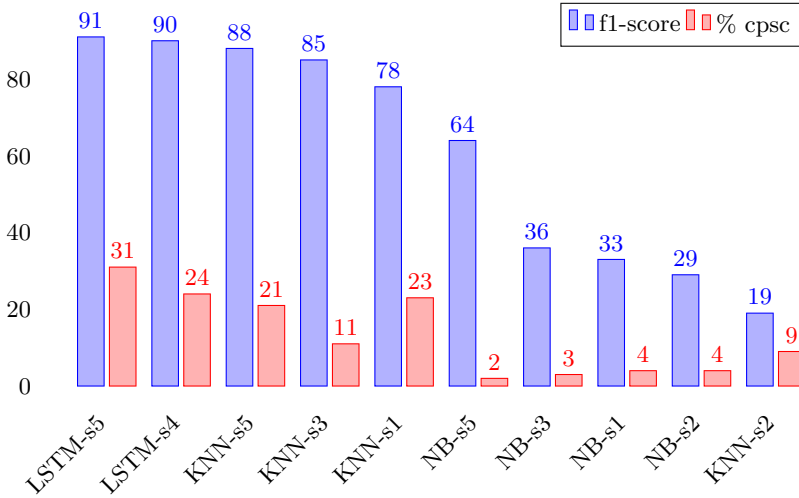


Fig. 6. Comparison of f1-score and %cpsc for the washing machine.

classification results. The LSTM improvement between scenarios 4 and 5 in every appliance allows concluding that *on time* and *delta power* features are helpful to detect on and off transitions.

## 6 Conclusions and Future Work

This article studied the application of computational intelligence techniques to solve the binary variant of the energy disaggregation problem. A Naive Bayes classifier, a  $K$  nearest neighbor classifier, and a LSTM neural network were evaluated on five different appliances considering real data from the UK-DALE dataset. Five different sets of input features were used.

Results of the experimental evaluation showed that LSTM neural networks are suitable for classification task and are able to reach high percentages of correctly predicted state changes for some appliances. The proposed methods can be integrated on smart energy management frameworks [8].

The main lines for future work are related to extending the analysis of relevant input features that improve classification, exploring different LSTM architectures, integrate the `nilmtk` framework into the developed methods, and analyze the generalization to unseen usage patterns.

## References

1. Batra, N., et al.: NILMTK: an open source toolkit for non-intrusive load monitoring. In: 5th International Conference on Future Energy Systems, pp. 265–276 (2014)

2. Bonfigli, R., Squartini, S., Fagiani, M., Piazza, F.: Unsupervised algorithms for non-intrusive load monitoring: an up-to-date overview. In: 15th International Conference on Environment and Electrical Engineering (2015)
3. Chavat, J., Graneri, J., Nesmachnow, S.: Household energy disaggregation based on pattern consumption similarities. In: Nesmachnow, S., Hernández Callejo, L. (eds.) ICSC-CITIES 2019. CCIS, vol. 1152, pp. 54–69. Springer, Cham (2020). [https://doi.org/10.1007/978-3-030-38889-8\\_5](https://doi.org/10.1007/978-3-030-38889-8_5)
4. Chavat, J., Nesmachnow, S., Graneri, J.: Non-intrusive energy disaggregation by detecting similarities in consumption patterns. *Revista Facultad de Ingeniería Universidad de Antioquia* (2020)
5. Hart, G.: Nonintrusive appliance load monitoring. *Proc. IEEE* **80**(12), 1870–1891 (1992)
6. Kelly, J., Knottenbelt, W.: The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Sci. Data* **2**(150007), 1–14 (2015)
7. Kolter, J., Johnson, M.: Redd: a public data set for energy disaggregation research. In: Workshop on Data Mining Applications in Sustainability, pp. 59–62 (2011)
8. Luján, E., Otero, A., Valenzuela, S., Mocskos, E.E., Steffenel, L.A., Nesmachnow, S.: An integrated platform for smart energy management: the CC-SEM project. *Revista Facultad de Ingeniería Universidad de Antioquia* (2019)
9. Nesmachnow, S., Iturriaga, S.: Cluster-UY: collaborative scientific high performance computing in Uruguay. In: Torres, M., Klapp, J. (eds.) ISUM 2019. CCIS, vol. 1151, pp. 188–202. Springer, Cham (2019). [https://doi.org/10.1007/978-3-030-38043-4\\_16](https://doi.org/10.1007/978-3-030-38043-4_16)
10. Sokolova, M., Japkowicz, N., Szpakowicz, S.: Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In: Sattar, A., Kang, B. (eds.) AI 2006. LNCS (LNAI), vol. 4304, pp. 1015–1021. Springer, Heidelberg (2006). [https://doi.org/10.1007/11941439\\_114](https://doi.org/10.1007/11941439_114)
11. Turner, W., Doty, S.: Energy management handbook. The Fairmont Press, Lilburn (2007)
12. U.S. Energy Information Administration (EIA): Energy use in homes, June 2020. <https://www.eia.gov/>