

# Deep Learning-based Electric Appliances Identification from their Switching-On Current Waveforms

Yassine Chemingui<sup>1</sup>, Adel Gastli<sup>1</sup>, Mahdi Houchati<sup>2</sup>

<sup>1</sup>Electrical Engineering Dept. Qatar University  
Doha, Qatar

yassine.chemingui@qu.edu.qa; adel.gastli@qu.edu.qa

<sup>2</sup>Iberdrola Innovation Middle East  
Doha, Qatar

**Abstract** - The field of non-intrusive load monitoring offers a multitude of methods for investigating and diagnosing energy demand per appliance. Thus, energy-aware strategies can be derived and implemented. With the widespread of smart meters, the rich information of the main current variation is within reach for many households. Through continuous analysis of the main current waveform, switching-on loads can be identified, and energy-saving practices can be devised. This paper proposes a deep learning model, a Convolutional Siamese neural network for appliance classification based on the WHITED raw high-frequency current dataset. The model is trained on pairs of appliance, measuring their similarity. Based on that, the appliance is identified. With minimal data preprocessing, an F1 macro measure of 0.95 was achieved on the training appliances, and a 0.79 score on previously unseen devices.

**Keywords:** Non-intrusive load monitoring, Appliance classification, Convolutional Siamese networks, Household appliance recognition

## 1. Introduction

In a world seeking a transition from fossil fuel resources, energy consumption monitoring and load identification are essential for establishing a fundamental understanding of energy use. Through it, problems can be exposed (e.g. abnormal changes in energy consumption), opportunities are brought to light (pinpoint energy-saving practices), and the effectiveness of the energy strategies is assessed. Given these valuable insights, consumers adjust their behaviour and utilities leverage such information to optimize production and billing plans and improve client engagement. Due to its importance, energy consumption monitoring and load identification became an important study area. In addition, since privacy is a major concern, Non-Intrusive Load Monitoring (NILM) techniques have become the most relevant methods. Besides privacy, compared to Intrusive Load Monitoring (ILM), these are easily deployed and require only a unique point of measurement, the smart meter. The widespread installation of smart meters gave momentum to the field and brought in several datasets to boost research [1][2][3][4][5]. The single point measurement from the main feed is examined for specific appliance signatures. Every electrical device has its own and unique features in terms of consumer behaviour. These unique features known as Load Signatures (LS) include voltage waveform (VW), current waveform (CW), active and reactive power (P & Q), ...etc. Thus several datasets were built, each considering some of these features. Even though most of the research is focused on low-frequency data, high frequency data is to our concern. In this paper, Neural Networks' capacity of automatic feature extraction is exploited to recognize different appliances and identify previously unknown devices.

The rest of the paper is organized as follows: In Section 2, related work is presented; section 3 presents the proposed methodology; section 4 presents the experimental results; and section 5 concludes the paper.

## 2. Literature Review

Initially proposed by Hart [6], several NILM studies have advanced the field to its state today. Varieties of methods were developed to learn the hidden patterns for load disaggregation. Techniques range from supervised, when labeled appliance data is present, to unsupervised algorithms when individual appliance data is absent. Hidden Markov Models (HMM) and Graph Signal Processing[7][8], for instance, were applied for unsupervised disaggregation. HMM's basic idea is to investigate a defined number of hidden states (ON/OFF and other intermediate appliance states) to analyze consumption.

Different methods were derived based on HMM[9][10], [11]. On the other side, supervised learning literature is much richer. In [12], several methods were highlighted including Artificial Neural Networks (ANNs) with a focus on Multilayer Perceptron and Convolutional Neural Networks (ConvNets). Also, Support Vector Machines (SVM), K-Nearest Neighbors(k-NN), and naïve Bayes classifiers were used[13][14]. The breakthroughs of Deep Learning motivated the widespread of ANNs application in a variety of fields including NILM. These networks are universal function approximators and thus can be fine-tuned to specific domains. A ConvNet combined with sequence-to-point was proposed by Zhang et al.[15]. In [16], authors compared Denoising AutoEncoder (DAE) with long short term memory network (LSTM) and gated recurrent unit (GRU) recurrent models and HMM for load disaggregation. DAE was the better option. Nevertheless, Recurrent models showed promise and were successfully developed in [17][18]. High frequency data is scarcely considered [19] and in general, hand-designed features by domain experts are considered for intelligent classifiers[19][20]. However, in [21], the authors trained several ConvNet-based binary classifiers per appliance to reach the state of the art results on WHITED [4]. In [22], based on Siamese neural networks, they clustered known devices per category and leveraged the resulting groups to detect unidentified appliances. The model learned from appliances' VI trajectory images. The same authors used, before, VI-trajectory-based appliance signatures to train a ConvNet for appliance identification [23].

In this paper, a Convolutional Siamese Neural Network model is trained to learn household appliances' similarities. The learned metric function was utilized to classify appliances. Learning from minimally processed current measurements to identify appliances and recognize newly seen equipment is the main contribution of this work.

### **3. Methodology**

#### **3.1. Approach**

Designing features from available data to feed the machine learning model is key for the model's performance. The information must be in the data for an algorithm to extract it. Irrelevant features result in low performance however the data were analyzed. As the mantras say: "Garbage In, Garbage Out". This process is time-consuming and relies on expert knowledge. Even though feature engineering highlights key information, leverages the expertise, and lightens black-box optimization (model Explainability), it is a difficult and open-ended process with humans in the loop. Besides, data is scarce in general and there are not enough samples to experiment on.

Deep learning is capable of extracting higher-level features from raw input. But, it doesn't hurt to add some designed expert features. The rise of deep learning is associated with the ImageNet winning convolutional neural network. This success is computer vision's milestone in replacing hand-designed features by applying successive convolutional layers on raw images to extract information. These models, on their own, learn to find edges and shapes. However, these models require enormous amounts of labeled data, which is not available in most real-world cases. There might be an abundance of data, but it is unstructured and mostly unlabeled data. Classification constrained with limited samples is a hard task and the standard deep learning framework performs badly when data is scarce. In contrast, humans can learn new concepts from a few examples based on some preexisting knowledge. Inspired by this, the one-shot learning paradigm was conceived. Given an instance of data, such models are capable of recognizing it in inference, hence one-shot. Variants considering k instances are called k-shot learning where k is small. In order to classify all appliances, huge datasets must be collected and taking into account the plethora of devices and manufacturers, it is very unlikely to achieve such goals, thus regular deep learning methods are inefficient in recognizing new devices, and their performance is limited to the available data. If new samples are gathered from a new device, such models require retraining to add the new class. The solution of one-shot learning consists of the adoption of Siamese neural networks. Through the Siamese network, combining pairs as input, the number of samples grows exponentially, and with only a few instances of a class, the neural net learns efficiently the optimal subset of features for distance measuring. Instead of training directly to classify input data, these networks learn the distance function between given input pairs, (sometimes triplets). They are trained to differentiate between the same and different input categories. These methods are also labeled as distance metric learning. They are simply trying to learn a feature embedding so that the distances learned in that space capture semantic similarity. Learning distance is crucial for several applications of matching, retrieval, recognition ...etc. Siamese and Triplet networks are usually the go-to architectures. In deep metric learning, a low-dimensional feature representation is learned in a way to condense similar classes in a close neighborhood

and separate them from other class samples. Inspired by [24][25], a pair-based Siamese network is applied for distinguishing appliances and classifying old and new devices.

### 3.2. Convolutional Neural Networks

Unlike standard feedforward neural networks, applying convolution on patches of adjacent values reduces operations and generates features that are more meaningful. By sharing weights, a patch of filters is valid at whatever position of the input. It is impractical to connect high-dimensional inputs to the following neurons. In the case of spatial relationships in the data like the order relationship between words in text or the time steps of a time series, ConvNets are very useful. ConvNets proved to handle well one-dimensional data even though they were initially developed specifically for images. During the 5 seconds data, the appliance startup signature is located in a smaller duration interval. Since, it is not known at what point in time, exactly, the device will start, the convolutional framework is an interesting candidate for detecting the event.

Let  $f$  and  $g$  be respectively the input vector and the kernel. Their corresponding lengths are  $n$  and  $m$ . The convolution ( $f * g$ ) of  $f$  and  $g$  is defined as follows:

$$(f * g)(i) = \sum_j^m g(j) \cdot f(i - j + \frac{m}{2}) \quad (1)$$

The convolution operation slides the kernel over the input. The overlapping values of the kernel and the input are multiplied and summed up. The result of the operation is the output at the position in the input where the kernel was centered.

A convolutional layer scans the input through a set of filters/kernels. The current signature is examined by sliding the filter window. At each pass, the filter convolves the values in the corresponding region. In the end, an activation map of the input belonging to filters response from every position is generated focusing solely on important fields. This way the filters learn to match patterns and detect local features. Then, pooling layers are added. They downscale the resulting convolutions preserving locality and discarding irrelevant information. Hence, they reduce computation and prevent overfitting.

### 3.3. Model

The Siamese architecture applies the same neural network on input pairs, Fig. 1, the resulting vectors are compared on an L1 distance lambda layer before feeding the difference to the last layer composed of one neuron with sigmoid activation, as shown in Eq.(2).

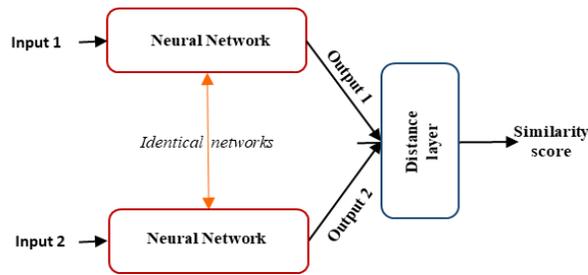


Fig. 1: Siamese Architecture

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

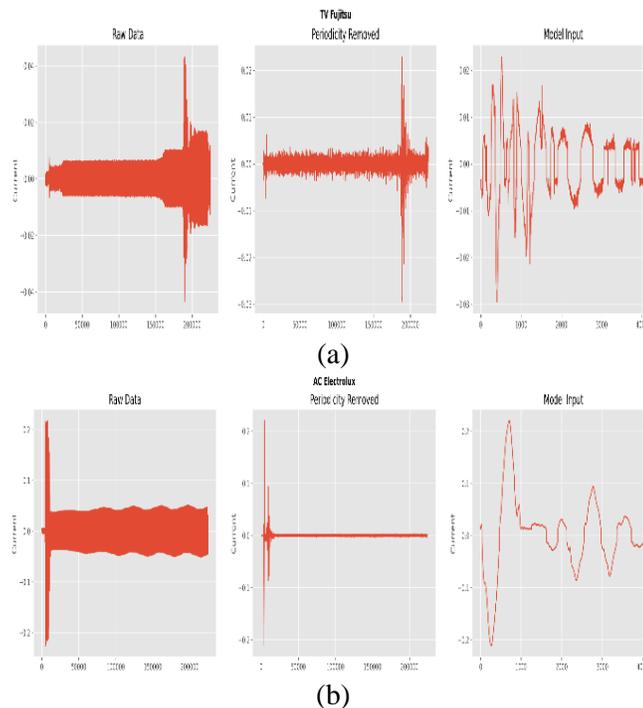
The output is the similarity measure value between  $[0, 1]$  where 1 means the inputs match. It can be interpreted in terms of probability. For example, there is a 70% (0.7) chance the two inputs belong to the same category.

A typical architecture is developed with no special caveats. A 1D Convolutional layer with 64 filters of size 128 and ReLU activation, Eq. (3), activation followed by 1D Maxpooling with a pool size of 5. The output is then fed to another 1D convolutional layer with 128 filters of size 64 and ReLU activations followed by 1D Maxpooling with a pool size of 5. The resulting output is flattened and connected to a 512-neurons regular layer with ReLU activations. L2 regularization penalty is applied to both convolutional layers kernels/filters. The model is developed with Keras library in a python 3.7 environment. This work showcases that with minimal preprocessing and a fairly standard 1D ConvNet, it is possible to recognize appliances' type and recognize new devices. The current signature is rich enough to identify loads.

$$ReLU(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (3)$$

### 3.4. Data

WHITED is a high-frequency public dataset of current and voltage measurements of appliance start-up sampled at 44 kHz from several locations. It is comprised of 1100 5-second recordings from 110 appliances corresponding to 46 appliance types. Our goal is to extract as much information as possible from only raw input or minimally preprocessed samples. Only the current signal is considered. The data periodical term is removed and each sample is reduced to an input size element of  $S=\{2048, 4096, 8192\}$ . Each region has its frequency (50Hz /60Hz), and the periodical pattern is subtracted accordingly since during the steady state the current waveform is periodic. The frequency  $f$  is extracted by applying the fast Fourier transform algorithm. The period is then deduced to remove the periodical term from the data:  $x(t) = x(t) - x(t-T)$  where  $f = 1/T$ . Afterward, only the highest  $M \in S$  values, which is redeemed the most important, are kept. The operation could be interpreted as zooming in for meaningful values. The sample values are sorted, and the indices of the highest M values are retained. The remaining values are deleted while preserving the original order. Taking more values do not improve performance. The original sample vector length is 22400. The majority of these float numbers are redundant or too close to zero that they convey no useful information. The values are small and centered around zero that it was unnecessary to normalize the data. Fig. 2 illustrates the transformation with the input size of 4096. The shape and structure of a sample are preserved and it is hypothesized that it embeds the appliance signature. When visually inspecting data, unique patterns per appliance are clear.



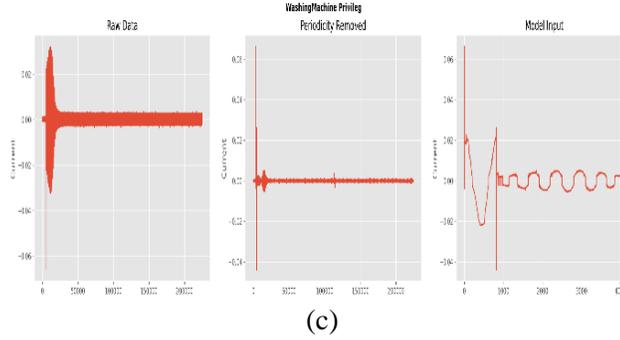


Fig. 2: Samples showing raw data, periodicity removed and downsampling of current waveforms for: (a) TV Fujitsu, (b) AC Electrolux and (c) Washing Machine Privileg

### 3.5. Training

The classification problem is mapped to a binary classification task in which pairs of input are evaluated for similarity. For multiple iterations, batches of 64 inputs are processed. Half of it correspond to pairs of the same appliance and the other contains pairs of different classes. Plenty of data is available it is hard to cover all possible pairs. From  $N$  samples of a class,  $N(N-1)/2$  similar pairs can be created and thus circumvent the initial data scarcity. For dissimilar pairs, the numbers escalate quickly. After every 200 iterations, the performance is evaluated with one-shot tasks. The optimizer utilized is Adam with a learning rate of 0.0005 and the binary crossentropy loss is used:

$$Loss = \frac{1}{M} \sum_{i=1}^M y_i * \log \log (\hat{y}_i) + (1 - y_i) * \log \log (1 - \hat{y}_i) \quad (4)$$

where  $\hat{y}_i$  is the  $i^{\text{th}}$  model output and  $y_i$  is its corresponding target.

### 3.6. Model Evaluation

Three models are trained with the same previously described architecture on different input sizes. The following is the list of training appliances: *AC, CFL, Charger, Coffee Machine, Deep Fryer, Fan, Fridge, Hair Dryer, Iron, Kettle, Laptop, Light Bulb, Microwave, Rice Cooker, Sandwich Maker, Stove, Toaster, TV, Vacuum Cleaner, Washing Machine*. The trained model performance is evaluated on known training appliances and unknown appliances, testing appliances. The following appliances are selected for testing: *Air Pump, Bench Grinder, Cable Modem, Cable Receiver, Desktop PC, Desoldering, Drilling Machine*

#### 3.6.1 Classification

When classifying a new input, it is preprocessed before feeding it to the Siamese network for comparison with other samples from known appliances. If  $n$  categories are available,  $n$  pairs are constructed where each pair is composed of the new input and a sample from the known appliances. The new input is classified as the appliance's type corresponding to the maximum similarity score. In the case of the testing data, only a sample is collected from the device's switching-on. The following Fig. 3 describes the classification process.

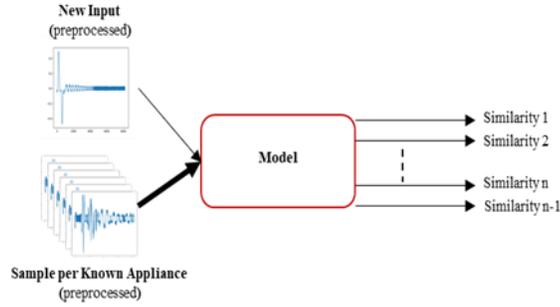


Fig. 3: Similarity calculation between new input and known

### 3.6.2 Classification

The macro F1 score is the evaluation metric. It calculates the mean of binary scores, the F1 score per each class. This assumes that all classes are equally important to emphasize the importance of model performance in classes with less frequent data.

$$F_1 macro = \frac{1}{N} \sum_{i=1}^N F_{1 i} \quad (5)$$

where  $N$  is the number of classes and the F1 score per each class is calculated as:

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall} \quad (6)$$

The precision and recall for each class are determined as:

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

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

where:

- True Positives (TP): when the prediction and the ground truth are both positive.
- False Negatives (FN): when the prediction is negative but the ground truth is positive.
- False Positives (FP): when the prediction is positive but the ground truth is negative.

## 4. Results

Three models were trained with different input sizes: Model\_1 (2048), Model\_2 (4096), Model\_3 (8192). They have the same architecture and they were trained each for 40.000 iterations. In each iteration, a batch of 64 samples is fed to the model. The three models perfectly classify samples from 15 training appliances but the scores drop particularly for recognizing two appliances: *Toaster* and *Kettle*. The model with the larger input performs well on those appliances compared to the other two models. These two models often misclassify the *Toaster* as *Iron*. Model\_1, with 2048 input size, confuses the *Toaster* mostly with the *Coffee Machine*, and Model\_2, with 4096 input size, misclassifies the *Toaster* sometimes as a *Coffee Machine* and sometimes as a *Sandwich Maker*. Further details are shown in the annex. However, these devices have similar functioning principles. Table 1 summarizes F1 macro scores per model on seen and unseen devices.

Table1: F1 macro scores per model

	Model_1	Model_2	Model_3
<b>Training Appliances</b>	0.95	0.94	0.98
<b>Testing Appliances</b>	0.76	0.79	0.53

Even though Model\_3 performs best on training data it overfits and its accuracy deteriorates in front of previously unknown appliances. Having more data points comes with the risk of including noisy information. The other two models have similar performance but Model\_2 is better at recognizing new devices. Here, in Tables II and III, the F1 scores per training appliance and per testing appliances for the three models are presented.

Table 2: F1 scores per model per training appliance

Model	Appliance	Model_1	Model_2	Model_3
		2048	4096	8192
AC		1	1	1
CFL		1	1	1
Charger		1	1	1
Coffee Machine		0.95	0.9	1
Deep Fryer		1	1	1
Fan		1	1	0.96
Fridge		1	1	1
Hair Dryer		0.94	0.94	1
Iron		0.9	0.88	0.76
Kettle		0.77	0.74	0.99
Laptop		1	1	1
Light Bulb		1	1	1
Microwave		1	1	1
Rice Cooker		1	1	1
Sandwich Maker		1	1	1
Stove		1	1	1
Toaster		0.64	0.7	0.94
TV		1	1	1
Vacuum Cleaner		1	1	1
Washing Machine		1	1	1

Table 3: F1 scores per model per testing appliance

Model	Appliance	Model_1	Model_2	Model_3
		2048	4096	8192
Air Pump		1	1	0.8
Bench Grinder		1	1	0.9

<b>Cable Modem</b>	0.7	0.6	0.2
<b>Cable Receiver</b>	1	0.9	0.4
<b>Desktop PC</b>	0.6	0.9	0.6
<b>Desoldering</b>	1	0.6	1
<b>Drilling Machine</b>	0.3	0.6	0.2

## 4. Conclusion

In this paper, a Deep Siamese Model is developed for appliance classification. The model is composed of a standard 1D convolutional neural network. Based on raw current input, most of the appliances are correctly identified. The developed model is capable of recognizing newly introduced devices from a few samples. It proves the generality of our model and the promise of load identification with minimal raw input data. We reach a 0.94 F1 macro score on the specific set of appliances the model was trained on and a 0.79 F1 macro score classifying previously unknown appliances

## Acknowledgements

This publication was made possible by the National Priority Research Program (NPRP) grant [NPRP10- 1203-160008] from the Qatar National Research Fund (a member of Qatar Foundation) and the co-funding by IBERDROLA QSTP LLC. The findings achieved herein are solely the responsibility of the authors

## References

- [1] K. Anderson, A. F. Ocneanu, D. Benitez, D. Carlson, A. Rowe, and M. Bergés, “BLUED : A Fully Labeled Public Dataset for Event-Based Non-Intrusive Load Monitoring Research,” in Proceedings of the 2nd KDD Workshop on Data Mining Applications in Sustainability (SustKDD), 2012, no. October 2011.
- [2] J. Z. Kolter and M. J. Johnson, “REDD : A Public Data Set for Energy Disaggregation Research,” SustKDD Work., vol. xxxxx, no. 1, 2011.
- [3] J. Kelly and W. Knottenbelt, “The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes,” Sci. Data, vol. 2, 2015, doi: 10.1038/sdata.2015.7.
- [4] M. Kahl, A. U. Haq, T. Kriechbaumer, and H.-A. Jacobsen, “WHITED - A Worldwide Household and Industry Transient Energy Data Set,” in 3rd International Workshop on Non-Intrusive Load Monitoring, 2016.
- [5] J. Gao, S. Giri, E. C. Kara, and M. Bergés, “PLAID: A public dataset of high-resolution electrical appliance measurements for load identification research,” in BuildSys 2014 - Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings, 2014, doi: 10.1145/2674061.2675032.
- [6] G. W. Hart, “Nonintrusive appliance load monitoring,” Proc. IEEE, vol. 80, no. 12, pp. 1870–1891, 1992, doi: 10.1109/5.192069.
- [7] B. Zhao, K. He, L. Stankovic, and V. Stankovic, “Improving Event-Based Non-Intrusive Load Monitoring Using Graph Signal Processing,” IEEE Access, vol. 6, 2018, doi: 10.1109/ACCESS.2018.2871343.
- [8] K. He, L. Stankovic, J. Liao, and V. Stankovic, “Non-Intrusive Load Disaggregation Using Graph Signal Processing,” IEEE Trans. Smart Grid, vol. 9, no. 3, 2018, doi: 10.1109/TSG.2016.2598872.
- [9] J. Z. Kolter and T. Jaakkola, “Approximate inference in additive factorial HMMs with application to energy disaggregation,” in Journal of Machine Learning Research, 2012, vol. 22.
- [10] S. Patten, “Unsupervised disaggregation for non-intrusive load monitoring,” in Proceedings - 2012 11th International Conference on Machine Learning and Applications, ICMLA 2012, 2012, vol. 2, doi: 10.1109/ICMLA.2012.249.
- [11] O. Parson, S. Ghosh, M. Weal, and A. Rogers, “An unsupervised training method for non-intrusive appliance load monitoring,” Artif. Intell., vol. 217, 2014, doi: 10.1016/j.artint.2014.07.010.
- [12] A. Ruano, A. Hernandez, J. Ureña, M. Ruano, and J. Garcia, “NILM techniques for intelligent home energy management and ambient assisted living: A review,” Energies, vol. 12, no. 11. 2019, doi: 10.3390/en12112203.

- [13] C. C. Yang, C. S. Soh, and V. V. Yap, "A systematic approach in appliance disaggregation using k-nearest neighbours and naive Bayes classifiers for energy efficiency," *Energy Effic.*, vol. 11, no. 1, 2018, doi: 10.1007/s12053-017-9561-0.
- [14] A. Ashari, I. Paryudi, and A. Min, "Performance Comparison between Naïve Bayes, Decision Tree and k-Nearest Neighbor in Searching Alternative Design in an Energy Simulation Tool," *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, no. 11, 2013, doi: 10.14569/ijacsa.2013.041105.
- [15] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," in *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 2018.
- [16] J. Kelly and W. Knottenbelt, "Neural NILM," in *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments - BuildSys '15*, 2015, pp. 55–64, doi: 10.1145/2821650.2821672.
- [17] T.-T.-H. Le, J. Kim, and H. Kim, "Classification performance using gated recurrent unit recurrent neural network on energy disaggregation," in *2016 International Conference on Machine Learning and Cybernetics (ICMLC)*, Jul. 2016, pp. 105–110, doi: 10.1109/ICMLC.2016.7860885.
- [18] L. Mauch and B. Yang, "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 2015*, 2016, doi: 10.1109/GlobalSIP.2015.7418157.
- [19] M. Kahl, A. U. Haq, T. Kriechbaumer, and H. A. Jacobsen, "A comprehensive feature study for appliance recognition on high frequency energy data," in *e-Energy 2017 - Proceedings of the 8th International Conference on Future Energy Systems*, 2017, doi: 10.1145/3077839.3077845.
- [20] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive Load Monitoring approaches for disaggregated energy sensing: A survey," *Sensors (Switzerland)*, vol. 12, no. 12, 2012, doi: 10.3390/s121216838.
- [21] D. Jorde, T. Kriechbaumer, and H. A. Jacobsen, "Electrical Appliance Classification using Deep Convolutional Neural Networks on High Frequency Current Measurements," in *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids, SmartGridComm 2018*, 2018, doi: 10.1109/SmartGridComm.2018.8587452.
- [22] L. De Baets, C. Develder, T. Dhaene, and D. Deschrijver, "Detection of unidentified appliances in non-intrusive load monitoring using siamese neural networks," *Int. J. Electr. Power Energy Syst.*, vol. 104, 2019, doi: 10.1016/j.ijepes.2018.07.026.
- [23] L. De Baets, J. Ruyssinck, C. Develder, T. Dhaene, and D. Deschrijver, "Appliance classification using VI trajectories and convolutional neural networks," *Energy Build.*, vol. 158, 2018, doi: 10.1016/j.enbuild.2017.09.087.
- [24] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, 2005, vol. I, doi: 10.1109/CVPR.2005.202.
- [25] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014, doi: 10.1109/CVPR.2014.220.

## Annex

Here, the confusion matrices of each model are included. This information presents a more detailed view of the classification of each appliance, and where the models got it wrong.

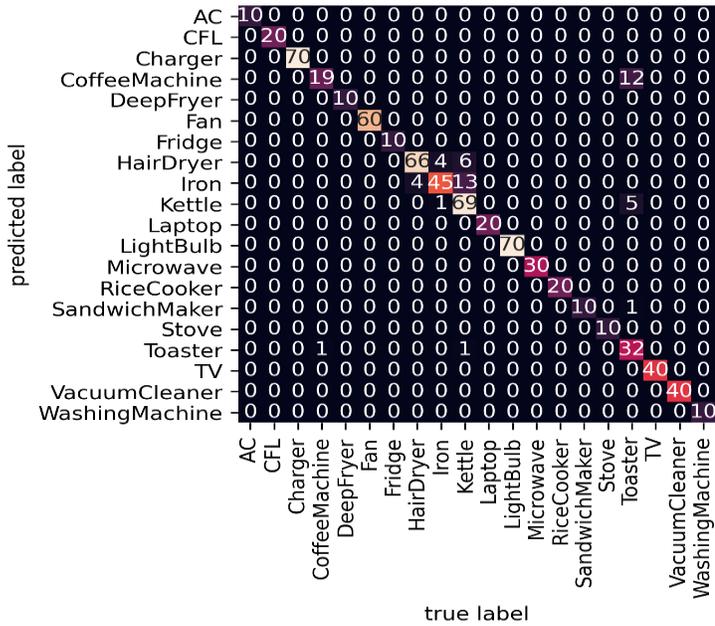


Fig. 4: Model\_1 confusion matrix on training appliances

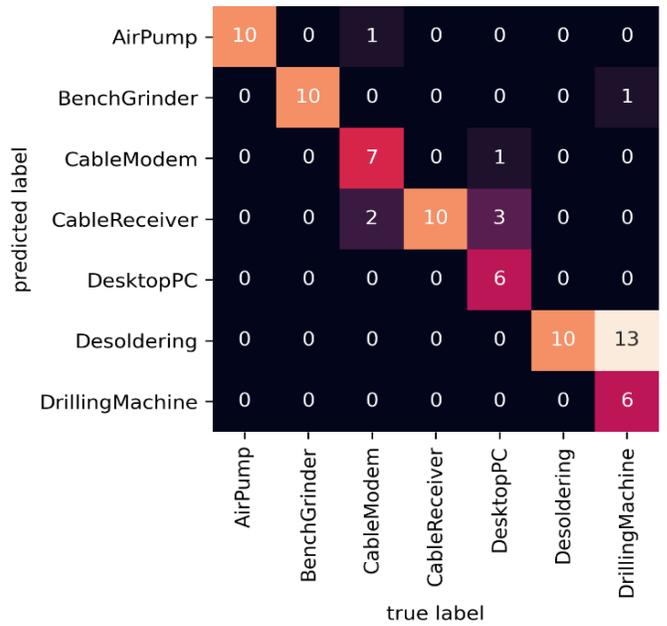


Fig. 5: Model\_1 confusion matrix on testing appliances

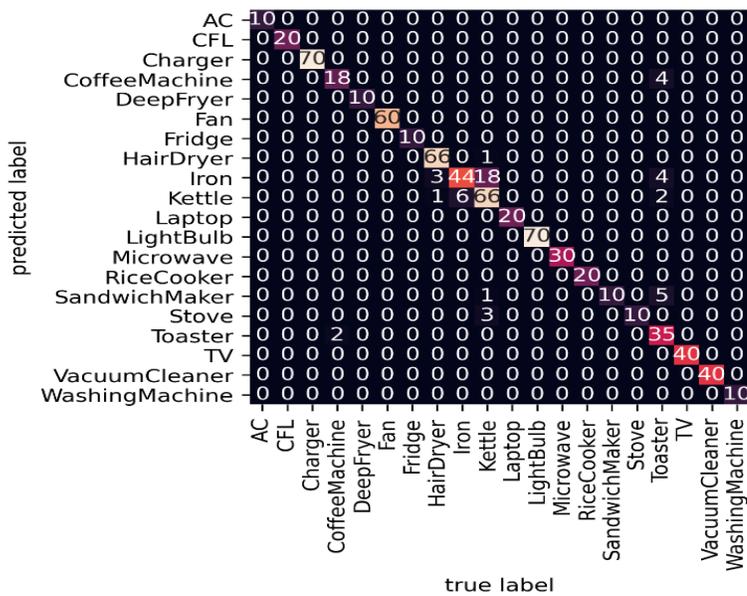


Fig. 6: Model\_2 confusion matrix on training appliances

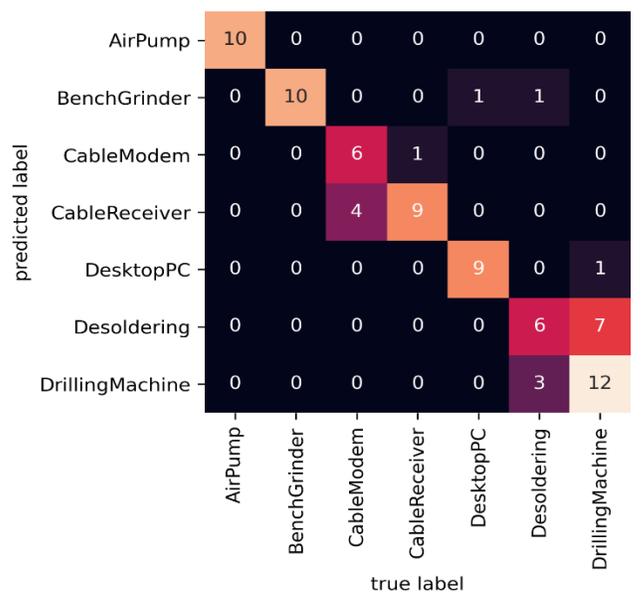


Fig. 7: Model\_2 confusion matrix on testing appliances

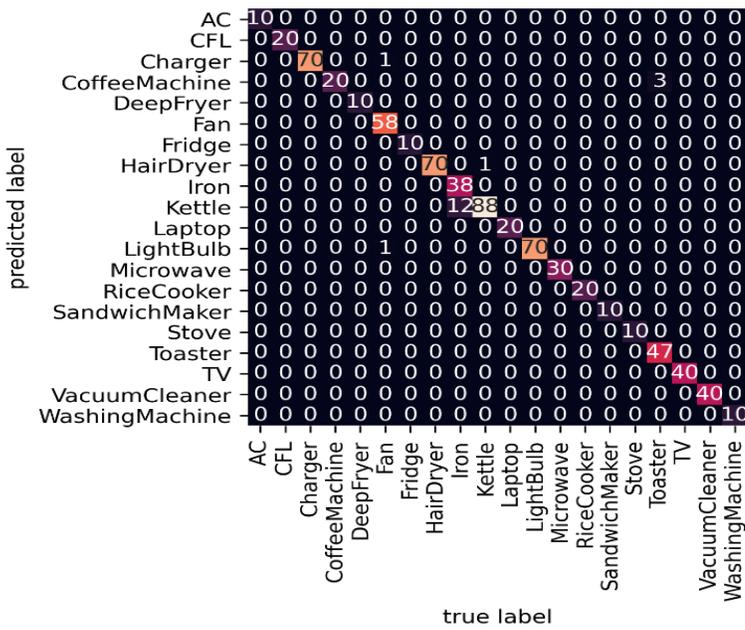


Fig. 8: Model\_3 confusion matrix on training appliances

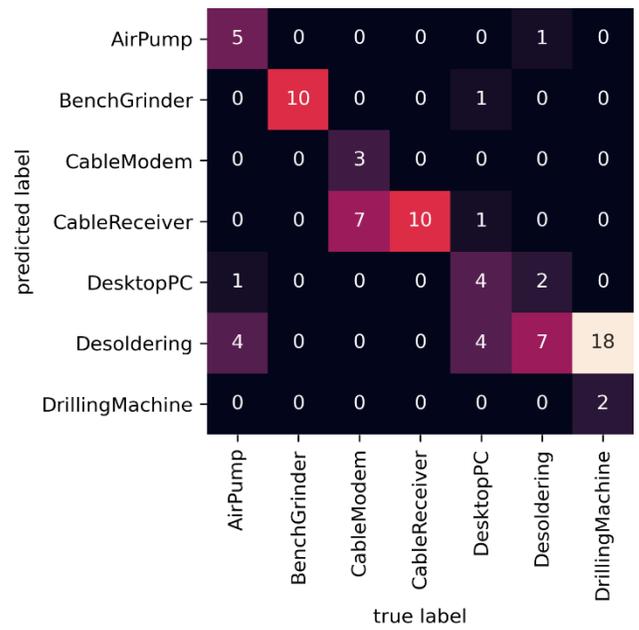


Fig. 9: Model\_3 confusion matrix on testing appliances