

# Electrical Load Classification with Open-Set Recognition

Dániel István Németh \*  and Kálmán Tornai 

Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, 1083 Budapest, Hungary

\* Correspondence: nemeth.daniel.istvan@hallgato.ppke.hu

**Abstract:** Full utilization of renewable energy resources is a difficult task due to the constantly changing demand-side load of the electrical grid. Demand-side management would solve this crucial problem. To enable demand-side management, knowledge about the composition of the grid load is required, as well as the ability to schedule individual loads. There are proposed Smart Plugs presented in the literature capable of such tasks. The problem, however, is that these methods lack the ability to detect if a previously unseen electrical load is connected. Misclassification of such loads presents a problem for load estimation and scheduling. Open-set recognition methods solve this problem by providing a way to detect samples not belonging to any class used during the training of the classifier. This paper evaluates the novel application of open-set recognition methods to the problem of load classification. Two approaches were examined, and both offer promising results. A Support Vector Machine based approach was first evaluated. The second, more robust method used a modified OpenMax-based algorithm to detect unseen loads.

**Keywords:** convolutional neural networks; electrical load classification; open-set recognition; smart grid; smart home; smart plug

## 1. Introduction

Utilization of renewable energy sources is becoming more and more challenging, as the contribution of non-controllable electricity producers such as wind and solar power becomes a significant part of electricity production in the electrical grid. Demand-side management is required in order to fully utilize non-controllable renewables in order to adjust the load side to match the available supply and involve traditional power plants as little as possible. In our previous work in [1], we presented a dimmer-based Smart Plug prototype capable of classifying the connected electrical load using less than 10 s of measurement data. Using traditional classification methods such as Support Vector Machines (SVM) or Neural Networks (NN) only works if measurement data are available for all electrical loads that will be plugged into the Smart Plug. In a real-world system, this is not feasible. To solve the closed nature of traditional classification methods, we examined a one-class SVM-based open-set recognition method, and our results are presented in [2]. Since then, a refined prototype was designed with on-device classification capabilities [3], and the collected dataset was also extended with measurement data from new electrical loads.

This paper shows the results for one-class SVM-based open-set recognition on the extended dataset and the performance of a novel and modified OpenMax-based [4] open-set recognition method. Both methods present promising results.

## 2. Literature Review

Open-set recognition is a novel topic made necessary by the closeness of traditional classification methods and the difficulty in acquiring samples for every class during the training phase. In some cases, it would not be possible to even determine the number of possible classes. Outlier detection, however, already has well-established use cases. The difference between open-set recognition (Novelty Detection) and Outlier detection is the availability of samples during the training phase. In outlier detection, knowledge and/or



**Citation:** Németh, D.I.; Tornai, K. Electrical Load Classification with Open-Set Recognition. *Energies* **2023**, *16*, 800. <https://doi.org/10.3390/en16020800>

Academic Editor: Hossam A. Gaber

Received: 24 November 2022

Revised: 15 December 2022

Accepted: 19 December 2022

Published: 10 January 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

samples are available about the outliers. An outlier detection system for the power usage of buildings is presented in [5]. However, research is lacking in the application of open-set recognition methods in the field of Smart Plugs and smart grids.

There are multiple approaches to electrical load classification in the literature. Loads can be measured in one common point (Non-Intrusive Load Monitoring—NILM), typically using Smart Meters [6], in groups, such as per room or connector, and individually as well (Intrusive Load Monitoring—ILM) using Smart Plugs. For NILM, the disaggregation of the total load curve is required in order to determine individual loads and their power consumption. This is not needed if all the loads are measured individually in the case of ILM. Currently available Smart Plugs on the market lack load classification capabilities [7], but the load-controlling features are already beneficial for accessibility [8]. A number of different Smart-Plug-based load classification solutions are proposed in the literature. Electric Load Signatures [9] can be recorded for individual loads and later used for classification. Alternatively, characteristic features of the loads can be observed, such as in [10] using Voltage–Current trajectories, or the load’s response to signal manipulation can be recorded [1,3,11]. Classification and measurement speed is also an important factor. Detection methods based on time series data are generally slower than those based on a small burst of measurement [1,3,10,11].

A real-time NILM solution is presented in [12,13], which uses the active power measured with 100 Hz sampling frequency. These data are used in real time to detect when a new load is turned on. Although this is a fast method to detect electrical loads from a single measurement point, individual appliance control is impossible without additional hardware and measuring devices at each outlet.

Smart Plugs are just one component of a larger system spanning an entire household or even the electrical grid. The main goal of such a system is to provide an automated way to manage and control the energy usage of the household. Controlling the load on the demand side (also called demand-side management—DSM) has several benefits for both utility providers and residents. Electricity providers can provide incentives to use less energy during peak hours using higher prices. There can be several types of pricing schemes, such as Critical Peak Pricing, Real-Time Pricing, or Time-Of-Use pricing [14]. In these cases, the load scheduling is left to the customers. The peak-to-average ratio (PAR) is a key metric as electricity production levels need to be able to handle the load during peak hours, while this production capacity is not utilized off peak. Reducing this ratio helps the better utilization of electricity production capacity. Utilities can also offer Direct Load Control methods, where the customer allows the energy provider to turn off high-power but interruptible loads such as HVAC in exchange for benefits such as reduced pricing. With the introduction of non-controllable energy providers such as wind turbines and solar panels, load balancing became more challenging. To fully utilize renewable energy, DSM is required.

In [15], home appliances are separated into two categories: schedulable and non-schedulable loads. The operation of nonschedulable loads cannot be deferred (i.e., TV, lights). Schedulable loads can also be interruptible (i.e., HVAC and battery charging) or noninterruptible (i.e., dishwasher). Based on this and similar load grouping methods, several simulation-based studies have been published in the literature [15–22]. Significant cost savings of up to 29% [16], 25% [15,17], 23% [23], and 19% [18] can be achieved. PAR can be significantly reduced [18] (29%). Ref. [24] summarizes various publications of DSM research which offer cost savings in the range of 5% to 42%. Home Energy Management systems have been proposed which utilize solar power as well as energy storage solutions [16,18,20,22] while achieving up to 46% PAR reduction [18]. In [19], 46% PAR reduction was achieved by using DSM in multiple households. In [20], using batteries, 48% electricity cost reduction was achieved with 7.69% PAR reduction. In [25], it is discussed that further studies into using data-driven, real-time optimization methods for DSM are needed.

In all these cases examined, however, the problem of actually identifying the household electrical loads has not been discussed in detail. In most cases, a static list of devices with simulated load profiles were examined. In a real-world scenario, electrical loads can be plugged into any outlet, and manual reconfiguration of the system would not be practical. In addition, the load profile and list of possible electrical loads is not known in advance. Knowledge about the loads in the household is crucial for forecasting energy requirements and scheduling the (schedulable) appliances accordingly. Although current Smart Plugs on the market can control and measure the connected appliance, they cannot detect the type of load connected. Smart Plugs with load classification capabilities have been presented in the literature [1,3,11]. Still, recognizing all types of loads is impossible due to the limited number of loads the models were trained on. Novelty detection is required to prevent the misclassification of previously unseen electrical loads. This paper presents a novel application of open-set recognition methods in the field of electrical load classification.

### 2.1. Previous Work—Smart Plug Prototype

The authors' previous work is published in [1–3]. In [1], a Smart Plug is presented that is capable of classifying the connected electrical load using less than 10 s of measurement data. The Smart Plug uses a dimmer to cut the voltage supply of the load, and the response is recorded. Multiple measurement methods, called measurement profiles, are presented. The Smart Plug cuts the voltage supply of the connected load with 10% to 75% cutoff ratios. An example of these cutoff ratios can be seen in Figure 1. The load's power consumption is measured during the dimming for  $h$  cycles. Then, the load receives uninterrupted power for  $d$  number of AC cycles before continuing the measurement with the next cutoff ratio. A measurement profile was defined in the following way:

$$\{< r, \text{percentage\_min} - \text{percentage\_max} >, h, d\}$$

where  $< r, \text{percentage\_min} - \text{percentage\_max} >$  determines the cutoff ratios used (from smallest to largest),  $h$  controls the number of AC cycles measured with each cutoff ratio, and  $d$  represents the number of delay/idle cycles between measuring with different cutoff ratios. Different measurement profiles were examined to determine how the classification accuracy is affected with fewer data available. This enables the reduction in the measurement data size for faster on-device load classification, as well as faster measurements. The measurement profiles examined in [1] are shown in Table 1.

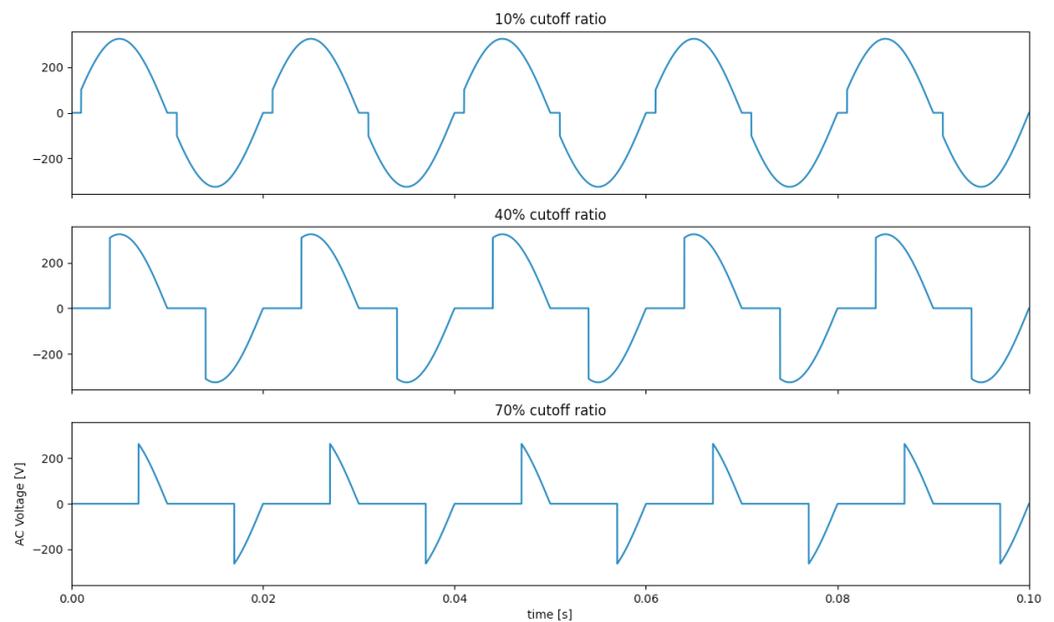
**Table 1.** Measurement Profiles and their properties. The measurement profiles are displayed in the following format:  $\{< r, \text{percentage\_min} - \text{percentage\_max} >, h, d\}$ .

Name	Profile	Measurement Time	Measurement Matrix Size
TEST_ORIG	$\{< 14, 10\text{--}75\% >, h = 20, d = 16\}$	488 cycles (9.76 s)	$14 \times 20$
TEST_HALVED	$\{< 7, 15\text{--}75\% >, h = 10, d = 8\}$	118 cycles (2.36 s)	$7 \times 10$
TEST_FOUR	$\{< 4, 15\text{--}75\% >, h = 8, d = 4\}$	44 cycles (0.88 s)	$4 \times 8$
TEST_TINY	$\{< 2, 10\text{--}75\% >, h = 12, d = 4\}$	28 cycles (0.56 s)	$2 \times 12$

The measurement times were calculated using the following formula:

$$t[\text{cycles}] = h \cdot r + d \cdot (r - 1) \quad (1)$$

An example of a measurement matrix measured with the TEST\_ORIG profile is shown in Figure 2. Each cell of the matrix represents the power measurement for one AC cycle. In addition to the power measurement, the RMS voltage and current values are also recorded for the measured AC cycles. Thus, the number of rows in the measurement matrix is determined by  $r$  and the number of columns by  $h$ . The performances of Fully Connected Neural Networks (FCNN), Convolutional Neural Networks (CNN), and Support Vector Machines were evaluated on the dataset. CNN-based classification offered the best accuracy rates.



**Figure 1.** Voltage cutoff method used by the Smart Plug to extract a characteristic response from the connected electrical load. The voltage curves are shown for cases when different cutoff ratios are applied. The leading part of the sinusoidal is cut, and the start is synchronized to the zero crossing of the AC voltage curve.



**Figure 2.** Power measurement matrix of an LED light bulb, measured using the TEST\_ORIG profile. Each cell represents the AC power consumed by the load in a single AC period during the measurement. The matrix is filled in a row-major way during the measurement. The size of the measurement matrix is determined by  $r$  and  $h$ .

In [3], a refined Smart Plug is presented which is capable of on-device Neural Network inference. The loads measured in [3] are the ones used in this paper as well:

- usb5V1A—A 5 W generic USB adapter.
- usbapple5V1A—A 5 W Apple USB adapter.
- ipad10W—A 10 W Apple USB adapter for iPad.
- batterycharger4A—A four-ampere “smart” lead–acid battery charger.

- batterycharger800 mA—An 800 mA traditional lead–acid battery charger.
- fan—A fan.
- hairdryer—A hairdryer.
- ledbulb—LED Light Bulb.
- ledspotlight—LED-based spotlight.
- incandescentbulb—An incandescent light bulb.
- irlamp—An infrared heat lamp.
- laptop—A laptop charger charging the laptop.
- monitor—An LCD screen.
- solderingiron—A soldering iron.

The same types of loads are grouped into a common class:

- USB—usb5V1A, usbapple5V1A, ipad10W;
- incandescent—incandescentbulb, irlamp.

So, in total, 14 loads were measured, and 11 classes were distinguished. A total of 250 samples were collected with each measurement profile from the 14 electrical loads.

It was shown that CNN inference on the Smart Plug is possible. Using an optimized model, the measurement and classification can be conducted in a little over ten seconds using the TEST\_ORIG profile, while only 2.5 s are required using the TEST\_HALVED measurement profile. In both cases, above 99.96% (average) accuracy rate was possible.

A one-class SVM-based open-set recognition method is evaluated in [2]. The dataset was the same as in [1]. For the classification, the following ten features were extracted from the measurement matrices:

- The mean and standard deviation of the matrix elements.
- The mean and standard deviation of the standard deviations of matrix rows.
- The mean and standard deviation of the standard deviations of matrix columns.
- Mean of  $2 \times 2$  submatrices on the four corners of the original matrix, divided by the mean of the entire matrix.

These ten features are also used in this paper for the one-class SVM classification, as well as during the modified OpenMax algorithm.

### 3. Clustering Using Traditional Closed Classification

Using traditional closed-set classification methods, an unknown load is classified into one of the existing classes present using training. This presents a problem, as the unknown load (if it does not belong to one of the training classes) is guaranteed to be misclassified. Applying load control to an incorrect type of load or making incorrect grid load predictions can have serious consequences. Although, in practice, the closeness of classification needs to be extended, examining how a closed classification method handles unseen samples can offer insight into how close one load may be to the others from the viewpoint of the classifier. This was examined on the measured loads listed in Section 2.1. The loads were clustered using the output of a Convolutional Neural Network (CNN). The CNN used the raw power measurement matrices as the input. The following methodology was used to run the closed CNN classification and achieve load clustering:

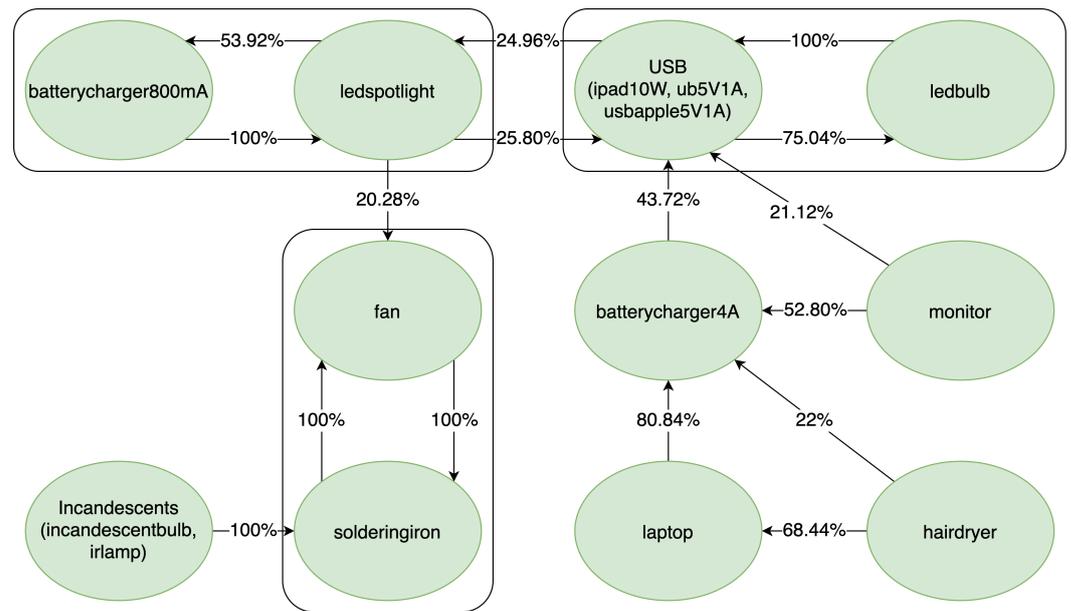
1. Select one of the 11 classes as the unknown class.
2. Train a CNN using 150 samples from each of the 10 known classes.
3. Run CNN inference of the 250 samples from the unknown class and record the output.
4. Repeat these steps with another unknown class.
5. If all classes were used as unknown, switch to another measurement profile and rerun steps 1–4.

The results of the experiment are shown in Figure 3. It can be seen that there are some loads where the samples are classified into a single class, while in other cases, the samples are misclassified into a range of different classes. For example, batterycharger4A has a measurement matrix with a high standard deviation for each cell, and it has samples classified into 8 of the 10 known classes. Some classes can be thought of as a group, such as

the fan and soldering iron, where both of them were classified into the other class. Based on the data, we can cluster the devices depending on where their samples had been classified to when the model had no knowledge of the class. It can be seen in Figure 4, where the clustering is shown for the TEST\_ORIG profile. The USB and ledbulb form one such cluster. The reason for it may be the similar power usage level and both loads having similar power supplies.

	USB	batterycharger4A	batterycharger800mA	fan	hairdryer	ledbulb	ledspotlight	incandescents	laptop	monitor	solderingiron
<b>TEST_ORIG</b>											
ipad10W		0	0	0	0	250	0	0	0	0	0
usb5V1A		0	0	0	0	157.1	92.9	0	0	0	0
usbapple5V1A		0	0	0	0	155.7	94.3	0	0	0	0
batterycharger4A	109.3		0.3	1.4	0.7	46.4	33	0	32.1	26.8	0
batterycharger800mA	0	0		0	0	0	250	0	0	0	0
fan	0	0	0		0	0	0	0	0	0	250
hairdryer	0	55	0	0		0	0	23.9	171.1	0	0
ledbulb	250	0	0	0	0		0	0	0	0	0
ledspotlight	64.5	0	134.8	50.7	0	0		0	0	0	0
incandescentbulb	0	0	0	0	0	0	0		0	0	250
irlamp	0	0	0	0	0	0	0	0	0	0	250
laptop	0	202.1	0	0	20.2	0	0	15.7		12	0
monitor	52.8	132	11.9	27.7	0	1.1	21.4	0	3.1		0
solderingiron	0	0	0	250	0	0	0	0	0	0	
<b>TEST_HALVED</b>											
ipad10W		59	0	0	0	191	0	0	0	0	0
usb5V1A		0	0	0	0	250	0	0	0	0	0
usbapple5V1A		0	0	0	0	250	0	0	0	0	0
batterycharger4A	86.4		0	1.7	3.4	45.1	12.8	2.5	55.5	42.6	0
batterycharger800mA	8.1	0		0.1	0	0	241.8	0	0	0	0
fan	0	0	0		0	0	0	0	0	0	250
hairdryer	0	200.1	0	0		0	0	0	49.9	0	0
ledbulb	224.8	24.8	0	0	0		0.2	0	0	0.2	0
ledspotlight	0	18.5	75.6	155.9	0	0		0	0	0	0
incandescentbulb	0	0	0	0	0	0	0		0	0	250
irlamp	0	0	0	0	0	0	0	0	0	0	250
laptop	0	250	0	0	0	0	0	0		0	0
monitor	9.1	206.8	1	0.7	0	25.7	6.6	0.1	0		0
solderingiron	0	0	0	250	0	0	0	0	0	0	
<b>TEST_FOUR</b>											
ipad10W		203.4	0	0	0	21.9	0	0	24.7	0	0
usb5V1A		37.1	0	0	0	212.9	0	0	0	0	0
usbapple5V1A		41.9	0	0	0	208.1	0	0	0	0	0
batterycharger4A	65.5		0	0.3	10.6	56.1	20.1	8.5	77.6	11	0.3
batterycharger800mA	20	0		78.1	0	0	151.2	0	0	0.7	0
fan	0	0	0		0	0	0.2	0	0	0	249.8
hairdryer	0	150	0	0		0	0	0	100	0	0
ledbulb	49.3	199.6	0	0	0		0	0	0	1.1	0
ledspotlight	0	0	115.3	85.1	0	49.6		0	0	0	0
incandescentbulb	0	0	0	11.6	0	0	0		0	22.5	215.9
irlamp	0	0	0	1.4	0	0	0	0	0	25	223.6
laptop	0	250	0	0	0	0	0	0		0	0
monitor	15.1	58.8	0.1	0	0	156.3	19.6	0.1	0		0
solderingiron	0	0	0	250	0	0	0	0	0	0	

**Figure 3.** Classification of unknown electrical loads (left column) to the other classes with the CNN model. Each row shows (the average of 10 runs) where the CNN model would classify the unknown load.



**Figure 4.** Clustering of load classes based on where the CNN model would classify the unknown class. Only those mappings that contained (on average) at least 50 samples (20% of all samples) are displayed.

#### 4. Open-Set Recognition

In the previous section, we demonstrated what happens when closed classification methods encounter samples from unseen classes. Open-set recognition solves the problem of misclassification by providing means of detecting samples not belonging to any class present during the training of the model.

In [26], the difficulties and the task of open-set recognition are discussed. The goal is to find an optimal function  $f$  minimizing the loss function  $L(x, y, f(x))$ :

$$\arg \min_f \left\{ R_I(f) := \int_{\mathbb{R}^d \times \mathbb{N}} L(x, y, f(x)) P(x, y) \right\} \quad (2)$$

where  $d$  is the dimension of the feature vectors,  $y$  is the label of the target class, and  $x$  represents the measurements ( $x \in \mathbb{R}^d, y \in \mathbb{N}$ ). The loss function has the following properties:  $L(x, y, f(x)) \geq 0$  and  $L(x, y, y) = 0$ . Minimizing the risk function  $R_I$  is a difficult task, as  $P(x, y)$ , the probability distribution over  $\mathbb{R}^d \times \mathbb{N}$ , is not known.

In [26], it is shown that knowledge about the known samples allows assumptions about the other parts of feature space. For example, regions located far from any known samples are less likely to contain samples from a known class.

The closed nature of classification algorithms is often disregarded in the literature. There can be various levels of openness, and thus different levels of difficulty are involved when considering open-set recognition methods. In [26], openness is defined as:

$$openness = 1 - \sqrt{\frac{2 \cdot |training\ classes|}{|testing\ classes| + |target\ classes|}} \quad (3)$$

Training classes represent the number of classes available during the training phase of the model. The number of testing classes represents the number of classes distinguished during the testing phase. The number of real-world classes to be distinguished is the target classes. Openness is 0 if samples from all classes are available during the training phase. In those cases, open-set recognition methods are not needed. However, as more unknown classes are introduced, the openness score approaches 1. In the case of load classification,

the number of loads not present during training is huge compared with the number of classes that were sampled.

In [4], OpenMax is introduced as a method to enable the rejection of fooling samples in images and solve the closed nature of traditional Deep Neural Network based classification. OpenMax works by using information from values of the penultimate layer's neurons of the DNN to adjust the SoftMax layer's output probabilities, while also introducing another element to the output: the probability that the input is unknown. After training the Neural Network, for each known class, the means of the Activation Vectors (AVs) in the penultimate layer of the CNN are calculated. A Weibull distribution is then fitted to the norms of the differences between the AVs and the MAVs (Mean Activation Vectors). Then, during inference, the top N classes of the SoftMax layers are considered, and the probabilities are revised using the Weibull CDF probability for the selected classes. The unknown probability value is also calculated. The output of the N revised probabilities and the unknown value are normalized using the SoftMax method, and the argument of the maximum probability is selected as the decision.

#### *4.1. Applying Open-Set Recognition to the Problem of Electrical Load Classification*

No knowledge is available about the set of unknown loads when considering applying open-set detection methods to the problem of electrical load classification. When designing and choosing the parameters of the open-set-enabled classifier, only knowledge about the known classes is available, as well as the domain-specific problem statements and goals:

- Several different types of electrical loads exist, and it is infeasible to measure all of them.
- Distinguishing between individual loads achieving the same function (i.e., USB charger) is not required.
- The result of the classification is used to predict future grid load. Misclassification of a true unknown load as one of the known loads results in inaccurate predictions. Incorrectly classifying a true known load as unknown is less problematic, as no incorrect assumptions are made about the appliance's future energy usage in those cases.
- Load control is applied to the connected electrical load via the Smart Plug. Applying load scheduling or interrupting the load's operation because it is falsely classified as one of the known schedulable/interruptible loads can have serious consequences (i.e., turning off a device such as a hairdryer, lights, or computer).
- If a load is classified as unknown, load control is not applied, and the load can be profiled by collecting measurements. Thus, incorrectly classifying a known load as unknown does not have severe consequences compared with misclassifying an unknown load as known.

To summarize, misclassification of an unknown load should be avoided. Classifying a known load as unknown may be acceptable, provided it makes misclassification of an unseen load less likely.

### **5. Open-Set Recognition Using One-Class SVM**

A one-class SVM classifier can be trained to predict whether a sample belongs to a class that was used during training. With this method, it is possible to detect if a known or unknown load is connected to the device. Since the one-class SVM classifier does not provide information about the specific known class, one of the classification methods shown previously ([1,3]—FCNN, SVM, CNN, and LSTM) can be used to determine the class of the sample which was detected by the one-class SVM classifier as a known sample.

A one-class SVM classifier with an RBF (Radial Basis Function) kernel was trained, where the kernel parameter was adjusted in accordance with the requirements of the load classification problem presented in Section 4.1. The ten features introduced in Section 2.1 were used as the input of the classifier. The evaluation methodology was the following: One class was selected as the unknown class; the other classes constituted the set of known loads. Of the 250 samples per known class, 100 were used for training the model. The

classifier was trained on the known training samples. Then, the model's performance was evaluated using all 250 samples of the unknown class and the 150 unused samples from each known class. This was repeated for each class as the unknown class. The average accuracy rates (of 100 runs) are shown in Figure 5. The performance of the one-class SVM classifier was evaluated for the TEST\_ORIG, TEST\_HALVED and TEST\_FOUR measurement profiles. The accuracy rates for the true known and unknown samples are shown in different columns. We can see that a nearly 100% rate can be achieved for the unknown samples, while for the known samples, the accuracy rates are also high, above 96% for the TEST\_ORIG measurement profile.

Unknown class	TEST_ORIG		TEST_HALVED		TEST_FOUR	
	Known	Unknown	Known	Unknown	Known	Unknown
USB	96.45%	100.00%	96.30%	100.00%	95.54%	100.00%
batterycharger4A	96.93%	100.00%	96.96%	100.00%	96.13%	100.00%
batterycharger800mA	96.48%	100.00%	96.38%	100.00%	95.48%	100.00%
fan	96.43%	100.00%	96.36%	100.00%	95.56%	100.00%
hairdryer	96.79%	100.00%	96.60%	100.00%	95.94%	100.00%
ledbulb	96.44%	98.70%	96.38%	100.00%	95.43%	100.00%
ledspotlight	96.39%	100.00%	96.32%	100.00%	95.58%	100.00%
incandescents	96.49%	100.00%	96.49%	100.00%	95.51%	100.00%
laptop	97.16%	100.00%	97.11%	100.00%	96.29%	100.00%
monitor	97.45%	100.00%	97.50%	100.00%	97.00%	100.00%
solderingiron	96.53%	100.00%	96.41%	100.00%	95.57%	100.00%
Average:	<b>96.69%</b>	<b>99.88%</b>	<b>96.62%</b>	<b>100.00%</b>	<b>95.82%</b>	<b>100.00%</b>

**Figure 5.** Average accuracy rates (of 100 runs) for SVM-based open-set recognition. The Known columns show the average accuracy rate for classifying all known samples when the class in the left column was selected as the unknown. The Unknown column shows the average accuracy rate of classifying the 250 unknown samples (when the class in the left column was selected as the unknown).

## 6. Modified OpenMax Method for Open-Set Recognition

Although the one-class SVM classification presents promising results, using the ten features from Section 2.1 may not be enough in the future if the list of measured electrical loads is extended. If an unknown sample was sufficiently close to a known sample in feature space, misclassification would happen.

The modified OpenMax-based method presented in this section is more robust, using both CNN classification and the ten extracted features. The original OpenMax method [4] was examined on the dataset, but the results are much worse than the ones using the one-class SVM shown in Section 5. The problem may be that some loads, such as batterycharger4A, featured a high standard deviation in the norm from the MAV. To solve that, instead of using the MAVs and the AVs for the Weibull CDF, the ten features introduced in Section 2.1 were used for fitting the Weibull CDF for each class. This method yielded better classification results. However, in this case, normalized features such as the ratio of  $2 \times 2$  submatrices compared with the mean of the entire matrix contributed much less to the total norm than, for example, the mean of the input matrix values. To solve this, a mean feature vector was calculated from the known training samples, and the feature vectors were divided by this mean feature vector element-wise before the norm calculation and fitting. So, the norm was calculated the following way:

Calculating the mean feature vector for the training samples used ( $TS$ ) for training the CNN model:

$$\mu = \text{mean}(TS_{\text{known}}) \quad (4)$$

Calculating the mean AVs for each class using the training samples from that particular class only:

$$\mu_i = \text{mean}\left(\frac{TS_{\text{known and class}=i}}{\mu}\right) \quad (5)$$

Calculating the norm value for a training sample ( $j$ ) belonging to class  $i$ :

$$\left\| \frac{TS_j}{\mu} - \mu_i \right\| \quad (6)$$

After fitting the Weibull CDF using the above normed values, during inference for a new test sample ( $S$ ), the norm is calculated for each top class  $k$  using this method:

$$\left\| \frac{S}{\mu} - \mu_k \right\| \quad (7)$$

This way, there is some level of normalization, so that all ten features contribute around the same amount (on average) to the norm value for a sample.

High accuracy rates were achieved using this modified OpenMax method and setting the correct coefficients and  $N$  to be 2 (number of TOP classes to consider). The CNN consisted of two convolutional layers with 30 and 40 channels. The second layer used the SoftMax activation function to normalize the output. The convolutional layers were followed by a dropout layer (20%) and a fully connected layer with 20 neurons before the final SoftMax layer. The input consisted of the power measurement matrix of a sample.

The evaluation methodology was similar to the one-class SVM method. For each measurement profile (from TEST\_ORIG, TEST\_HALVED, TEST\_FOUR), the following steps were executed:

1. Select one class as the unknown class.
2. Train the CNN model using 150 samples from each known class; no samples from the unknown class.
3. Calculate mean feature vector, MAVs, and fit Weibull CDF for each known class.
4. Evaluate model performance with the 100 unused known samples from each (known) class.
5. Evaluate model performance with the 250 unknown samples from the omitted class.
6. Repeat from step 2 for the next class as the unknown class.

The summary of the results can be seen in Figure 6, where the averages of 10 runs are shown. We can see that the accuracy rates are slightly worse than with the one-class SVM method, but some of the errors may not be problematic with respect to load control. For the unknown samples in the case of the TEST\_ORIG and TEST\_HALVED measurement profiles, the only error occurred with the batterycharger4A load. For the TEST\_ORIG case, 15.32% of the samples were classified as laptops, while 0.2% of the samples were classified as a monitor. In the case of load scheduling, this misclassification would not be a problem, as turning off lead–acid battery charging will not cause problems compared with other loads used, such as a hairdryer or a kitchen appliance.

Unknown class	TEST_ORIG		TEST_HALVED		TEST_FOUR	
	Known	Unknown	Known	Unknown	Known	Unknown
USB	94.60%	100.00%	94.99%	100.00%	93.62%	67.27%
batterycharger4A	95.22%	84.48%	95.36%	73.96%	94.91%	64.16%
batterycharger800mA	95.07%	100.00%	94.68%	100.00%	94.90%	100.00%
fan	94.56%	100.00%	94.85%	100.00%	95.36%	100.00%
hairdryer	94.54%	100.00%	93.45%	100.00%	93.70%	100.00%
ledbulb	95.43%	100.00%	94.71%	100.00%	95.02%	99.52%
ledspotlight	94.99%	100.00%	94.98%	100.00%	95.04%	57.60%
incandescents	94.88%	100.00%	94.65%	100.00%	94.59%	100.00%
laptop	95.98%	100.00%	95.44%	100.00%	95.33%	100.00%
monitor	95.52%	100.00%	95.67%	100.00%	93.18%	100.00%
solderingiron	94.89%	100.00%	94.90%	100.00%	95.14%	100.00%
Average:	<b>95.06%</b>	<b>98.59%</b>	<b>94.88%</b>	<b>97.63%</b>	<b>94.62%</b>	<b>89.87%</b>

**Figure 6.** Average accuracy rates (of 10 runs) for CNN-based OpenMax open-set recognition using Weibull CDF fitted to the mean of the normed features introduced in Section 2.1. The Known columns show the average accuracy rate for classifying all known samples when the class in the left column was selected as the unknown. The Unknown column shows the average accuracy rate of classifying the 250 unknown samples (when the class in the left column was selected as the unknown).

## 7. Discussion of the Results

The results for both SVM-based and OpenMax-based open-set recognition show that the methods presented in this paper can be used to achieve accurate open-set recognition. The ability to detect previously unseen loads solves one of the often overlooked challenges of the real-world application of DSM solutions presented in the literature. The modified OpenMax-based method presented slightly lower accuracy rates, but it is a more robust method. Both solutions are suitable for detecting previously unseen electrical loads, and thus avoiding misclassification of the connected load.

For a Smart Plug such as the one introduced in [3], which is capable of on-device CNN inference, and thus electrical load classification, accommodation of further logic to enable open-set recognition would be possible. Modifying the firmware to support Weibull distribution calculation and OpenMax final layer calculation can be achieved, which will enable accurate on-device classification with open-set recognition. A Smart Plug capable of classifying a connected load in 10 s and detecting and automatically profiling unseen appliances can be a key component in a Smart Home system designed to utilize renewable resources fully.

Compared with existing proposals in the literature, our method can achieve both fast load classification and individual load control. Open-set recognition capabilities are necessary for applying load control in a real-world scenario, where training samples are not available for every electrical load in the household. Other Smart-Plug-based methods presented in the literature lack open-set recognition capabilities [9,11] or only differentiate a limited set of classes based on load characteristics [10], which would not be suitable for DSM applications. NILM methods [12,13] do not offer load control capabilities required for DSM.

With the proposed method in this paper, automatic load profiling also becomes possible. If the Smart Plug detects a previously unseen load, it can automatically profile it so that only the scheduling information will require human interaction.

The load detection and control capabilities can be used for DSM applications not only on the household level but also at the utility level. Understanding the load composition of households and the usage patterns of individual loads can be a key component in grid load estimation for energy providers. All the benefits of DSM presented in Section 2 become possible, such as better utilization of renewable energy resources and PAR reduction.

## 8. Conclusions and Future Work

In this paper, the applications of open-set recognition methods were discussed in the field of electrical load classification. Accurate detection of unseen loads is possible with both methods examined. This novel application of open-set recognition can enable the creation of Home Energy Management systems capable of functioning in a real-world scenario. Accurate and fast load detection is crucial for load forecasting and for effective demand-side management.

The methods presented in this paper can be applied to existing Smart Plugs in the literature (such as [3]), allowing fast on-device load classification along with open-set recognition capabilities. Smart Plugs capable of classifying a connected load within seconds and also detecting and automatically profiling unseen appliances can be a key component in a Smart Home system designed to enable full utilization of renewable resources. Renewable energy resource utilization is an especially relevant field of research due to the energy crisis in Europe. Demand-side management is a key step needed as the world transitions to using renewable energy sources. We hope our results presented in this paper become a key component to achieve that.

**Author Contributions:** Conceptualization, D.I.N. and K.T.; methodology, D.I.N.; software, D.I.N.; validation, D.I.N. and K.T.; formal analysis, D.I.N.; investigation, D.I.N.; resources, K.T. and D.I.N.; data curation, D.I.N.; writing—original draft preparation, D.I.N.; writing—review and editing, D.I.N. and K.T.; visualization, D.I.N.; supervision, K.T.; project administration, K.T.; funding acquisition, K.T. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the New National Excellence Program of the Ministry for Culture and Innovation from the source of the National Research, Development and Innovation Fund grant number ÚNKP-22-1-I-PPKE-7 and by the Fund FK 137 608 and Fund TKP2021-NVA-26 of the Hungarian National Research, Development and Innovation Office.

**Data Availability Statement:** A sample dataset for demo purposes is available at <https://github.com/ndaneil/electrical-load-classification-openset-demo>, accessed on 23 November 2022. It is for demo purposes only and should otherwise not be used without explicit permission from the authors.

**Acknowledgments:** Supported by the ÚNKP-22-1 New National Excellence Program of the Ministry for Culture and Innovation from the source of the National Research, Development, and Innovation Fund. This work has also been supported by the Fund FK 137 608 and Fund TKP2021-NVA-26 of the Hungarian National Research, Development and Innovation Office.

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

## Abbreviations

The following abbreviations are used in this manuscript:

SVM	Support Vector Machines
CNN	Convolutional Neural Network
FCNN	Fully Connected Neural Network
ILM	Intrusive Load Monitoring
NILM	Non-Intrusive Load Monitoring
DSM	Demand-Side Management
HVAC	Heating, Ventilation, and Air Conditioning
PAR	Peak-to-Average Ratio
HEM	Home Energy Management
RMS	Root Mean Square
DNN	Deep Neural Network
MAV	Mean Activation Vector
CDF	Cumulative Distribution Function
LSTM	Long Short-Term Memory

## References

1. Németh, D.; Tornai, K. SP4LC: A Method for Recognizing Power Consumers in a Smart Plug. In Proceedings of the 11th International Conference on Smart Cities and Green ICT Systems-SMARTGREENS, INSTICC, SciTePress, Online, 27–29 April 2022; pp. 69–77. [\[CrossRef\]](#)
2. Németh, D.I.; Tornai, K. Detecting Unknown Electrical Loads Using open-set recognition. In Proceedings of the 2022 IEEE 10th International Conference on Smart Energy Grid Engineering (SEGE), Oshawa, ON, Canada, 10–12 August 2022; pp. 7–11. [\[CrossRef\]](#)
3. Nemeth, D.I.; Tornai, K. [In review] Fast Electrical Load Classification using a Dimmer-based Smart Plug. In *Proceedings of the Smart Cities, Green Technologies, and Intelligent Transport Systems*; Springer International Publishing: Berlin, Germany, 2023.
4. Bendale, A.; Boulton, T.E. Towards Open Set Deep Networks. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 1563–1572. [\[CrossRef\]](#)
5. Li, X.; Bowers, C.P.; Schnier, T. Classification of Energy Consumption in Buildings With Outlier Detection. *IEEE Trans. Ind. Electron.* **2010**, *57*, 3639–3644. [\[CrossRef\]](#)
6. Ridi, A.; Gisler, C.; Hennebert, J. A Survey on Intrusive Load Monitoring for Appliance Recognition. In Proceedings of the 2014 22nd International Conference on Pattern Recognition, Stockholm, Sweden, 24–28 August 2014; pp. 3702–3707.
7. Gomes, L.; Sousa, F.; Vale, Z. An Intelligent Smart Plug with Shared Knowledge Capabilities. *Sensors* **2018**, *18*, 3961. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Mtshali, P.; Khubisa, F. A Smart Home Appliance Control System for Physically Disabled People. In Proceedings of the 2019 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 6–8 March 2019; pp. 1–5.
9. da Veloso, A.F.S.; de Oliveira, R.G.; Rodrigues, A.A.; Rabelo, R.A.L.; Rodrigues, J.J.P.C. Cognitive Smart Plugs for Signature Identification of Residential Home Appliance Load using Machine Learning: From Theory to Practice. In Proceedings of the 2019 IEEE International Conference on Communications Workshops (ICC Workshops), Shanghai, China, 20–24 May 2019; pp. 1–6.
10. Du, L.; He, D.; Harley, R.G.; Habetler, T.G. Electric Load Classification by Binary Voltage–Current Trajectory Mapping. *IEEE Trans. Smart Grid* **2016**, *7*, 358–365. [\[CrossRef\]](#)
11. Petrović, T.; Morikawa, H. Active sensing approach to electrical load classification by smart plug. In Proceedings of the 2017 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT), Torino, Italy, 26–29 September 2017; pp. 1–5.
12. Athanasiadis, C.L.; Papadopoulos, T.A.; Doukas, D.I. Real-time non-intrusive load monitoring: A light-weight and scalable approach. *Energy Build.* **2021**, *253*, 111523. [\[CrossRef\]](#)
13. Athanasiadis, C.; Doukas, D.; Papadopoulos, T.; Chrysopoulos, A. A Scalable Real-Time Non-Intrusive Load Monitoring System for the Estimation of Household Appliance Power Consumption. *Energies* **2021**, *14*, 767. [\[CrossRef\]](#)
14. Iqbal, S.; Sarfraz, M.; Ayyub, M.; Tariq, M.; Chakraborty, R.K.; Ryan, M.J.; Alamri, B. A Comprehensive Review on Residential Demand Side Management Strategies in Smart Grid Environment. *Sustainability* **2021**, *13*, 7170. [\[CrossRef\]](#)
15. Mahapatra, B.; Nayyar, A. Home energy management system (HEMS): concept, architecture, infrastructure, challenges and energy management schemes. *Energy Syst.* **2022**, *13*, 643–669. [\[CrossRef\]](#)
16. Bouakkaz, A.; Mena, A.J.G.; Haddad, S.; Ferrari, M.L. Efficient energy scheduling considering cost reduction and energy saving in hybrid energy system with energy storage. *J. Energy Storage* **2021**, *33*, 101887. [\[CrossRef\]](#)
17. Setlhaolo, D.; Xia, X.; Zhang, J. Optimal scheduling of household appliances for demand response. *Electr. Power Syst. Res.* **2014**, *116*, 24–28. [\[CrossRef\]](#)
18. Noor, S.; Yang, W.; Guo, M.; van Dam, K.H.; Wang, X. Energy Demand Side Management within micro-grid networks enhanced by blockchain. *Appl. Energy* **2018**, *228*, 1385–1398. [\[CrossRef\]](#)
19. Caprino, D.; Della Vedova, M.L.; Facchinetti, T. Peak shaving through real-time scheduling of household appliances. *Energy Build.* **2014**, *75*, 133–148. [\[CrossRef\]](#)
20. Javaid, N.; Hafeez, G.; Iqbal, S.; Alrajeh, N.; Alabed, M.S.; Guizani, M. Energy Efficient Integration of Renewable Energy Sources in the Smart Grid for Demand Side Management. *IEEE Access* **2018**, *6*, 77077–77096. [\[CrossRef\]](#)
21. Ayan, O.; Turkay, B. Domestic electrical load management in smart grids and classification of residential loads. In Proceedings of the 2018 5th International Conference on Electrical and Electronic Engineering (ICEEE), Istanbul, Turkey, 5 May 2018; pp. 279–283. [\[CrossRef\]](#)
22. Hosseini, S.M.; Carli, R.; Dotoli, M. Robust Optimal Energy Management of a Residential Microgrid Under Uncertainties on Demand and Renewable Power Generation. *IEEE Trans. Autom. Sci. Eng.* **2021**, *18*, 618–637. [\[CrossRef\]](#)
23. Zhang, D.; Shah, N.; Papageorgiou, L.G. Efficient energy consumption and operation management in a smart building with microgrid. *Energy Convers. Manag.* **2013**, *74*, 209–222. [\[CrossRef\]](#)
24. Sharda, S.; Singh, M.; Sharma, K. Demand side management through load shifting in IoT based HEMS: Overview, challenges and opportunities. *Sustain. Cities Soc.* **2021**, *65*, 102517. [\[CrossRef\]](#)

25. Lu, X.; Zhou, K.; Zhang, X.; Yang, S. A systematic review of supply and demand side optimal load scheduling in a smart grid environment. *J. Clean. Prod.* **2018**, *203*, 757–768. [[CrossRef](#)]
26. Scheirer, W.J.; de Rezende Rocha, A.; Sapkota, A.; Boulton, T.E. Toward open-set recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1757–1772. [[CrossRef](#)] [[PubMed](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.