

Article

# Bats: An Appliance Safety Hazards Factors Detection Algorithm with an Improved Nonintrusive Load Disaggregation Method

Wei Wang <sup>1,†</sup>, Zilin Wang <sup>1,†</sup>, Yanru Chen <sup>1</sup>, Min Guo <sup>1</sup>, Zhengyu Chen <sup>1,†</sup>, Yi Niu <sup>1</sup>, Huangeng Liu <sup>2</sup>  
and Liangyin Chen <sup>1,3,\*</sup>

- <sup>1</sup> College of Computer Science, Sichuan University, Chengdu 610065, China; wang.david.wei@stu.scu.edu.cn (W.W.); wangzilin@stu.scu.edu.cn (Z.W.); yanruchen@stu.scu.edu.cn (Y.C.); guomin@stu.scu.edu.cn (M.G.); chenzhengyu@stu.scu.edu.cn (Z.C.); niuyi@stu.scu.edu.cn (Y.N.)
- <sup>2</sup> School of Mechanical Electronic and Information Engineering, China University of Mining and Technology, Beijing 100083, China; lhg@student.cumt.edu.cn
- <sup>3</sup> Institute for Industrial Internet Research, Sichuan University, Chengdu 610065, China
- \* Correspondence: chenliangyin@scu.edu.cn
- † These authors contributed equally to this work.

**Abstract:** In an electrical safe microenvironment, all kinds of electrical appliances can be operated safely to ensure the safety of life and property. The significance of safety hazard factors detection is to detect safety hazards in advance, to remind the administrators to exclude risk, to reduce the unnecessary loss, and to ensure that the electrical operation is healthy and orderly before the occurrence of accidents. In this paper, batteries are selected as the primary research subject of safety detection because batteries are used more and more in the Internet of Things (IOT), and they often cause fire in the process of discharging and charging. The existing algorithms need to be embedded into the specialized sensor for each important electrical appliance. However, they are limited by the actual deployment, so it is extremely difficult to spread widely. According to the opinions above, an improved load disaggregation algorithm based on dictionary learning and sparse coding with optimal dictionary matrix period is proposed to detect potential safety hazards of battery loads. For safety-related electrical applications, doing so can increase interpretability. Through experiments, we test this algorithm on the REDD dataset, and compare it with the baseline algorithms (combinatorial optimization, factorial hidden Markov model, basic discriminative dictionary sparse coding algorithm) to achieve a degree of trust. The Mean Absolute Error (MAE) value is 8.26, which drops by 70%. The Root Mean Square Error (RMSE) value is 97.75, which is also better than those baseline algorithms.

**Keywords:** Internet of Things; power supply safety; safety hazard factors detection



**Citation:** Wang, W.; Wang, Z.; Chen, Y.; Guo, M.; Chen, Z.; Niu, Y.; Liu, H.; Chen, L. Bats: An Appliance Safety Hazards Factors Detection Algorithm with an Improved Nonintrusive Load Disaggregation Method. *Energies* **2021**, *14*, 3547. <https://doi.org/10.3390/en14123547>

Academic Editor: Georgios Christoforidis

Received: 21 April 2021  
Accepted: 11 June 2021  
Published: 15 June 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 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/>).

## 1. Introduction

All the time, electrical fire accidents occur frequently, causing many economic property losses and casualties [1]. The statistical report on fire losses and casualties in the United States from 1980 to 2016 shows that the total number of fire hazards has decreased by half, but the losses of property and human have not significantly reduced [2]. The key reason to electrical safety problems is that there are not enough monitoring datasets about unsafe electrical appliances and interpretable monitoring algorithms [3,4]. Moreover, there are two ways to detect safety hazard factors, direct monitoring and nonintrusive load disaggregation [5,6]. Although the power consumption of individual appliances can be directly and accurately monitored, direct monitoring of battery load is difficult to deploy in practice due to hardware budget constraints and installation space constraints. With widespread use of smart meters and charging/recharging appliances, the safety hazard factors issue regarding electrical appliances is still an intractable challenging problem. The existing methods can be categorized as several main types of interpretable nonintrusive load disaggregation

algorithms, including combinatorial optimization (CO) [7], factorial hidden Markov model (FHMM) [8], and dictionary learning sparse coding (SC) [4] algorithms.

The combinatorial optimization algorithms in load disaggregation of safety detection problem are like the famous knapsack problem and subset summation problem. These algorithms are essential to find the optimal combination from a finite set of appliances for power value. The goal of these algorithms is to assign power value to every appliance in order to minimize the error between the sum of the estimated value and the real aggregated data. The combinatorial optimization algorithms mentioned above have been used as the benchmark for these algorithms [4,7,9]. The combinatorial optimization algorithm is used to express the electric appliance model as switching state and finite multi-state [10]. Although the combinatorial optimization non-intrusive load disaggregation algorithm can explain the electrical load mathematically, it is still not suitable in battery safety scenarios because of low accuracy for lack of time correlation.

With the deepening of the research, although the continuity principle of the close time is considered, the problem of the state transition probability is not considered [11–13]. The factorial hidden Markov algorithms model each electrical appliance as a single Markov chain. The states of multiple electrical appliances form multiple Markov chains and evolve simultaneously, which are not only constrained by the aggregate power, but also influenced by the transition probability. The factorial hidden Markov model algorithm will use the clustering algorithm to cluster more discrete electrical states into several basic states, thus there is less representation of the state of each appliance [8,14–16]. The models from multiple appliances are combined to create a superstate hidden Markov model (SSHMM), which can represent states of all the appliances. The aggregated power is passed through the SSHMM, which returns the states of every appliance included in this aggregated power. When the number of appliances is smaller, these factorial hidden Markov models can be solved exactly (FHMMExact) [17]. The factorial hidden Markov algorithms have some explanatory abilities, but they have low precision because it expresses less state of each electrical appliance [18], so it is not suitable for battery safety hazard detection as well.

The current load disaggregation algorithms only focus on the consumed load, rarely involving the supplied load [19]. The existing load disaggregation algorithms [20] are not suitable for the expression of battery safety hazard factors. Recent studies on interpretable load disaggregation have been dominated by dictionary learning algorithms [21,22], such as energy disaggregation via discriminative sparse coding (DDSC) [4]. However, the dictionary matrix period of these algorithms is not optimal, so it leads to lower accuracy. At the same time, the dictionary learning and sparse coding can be not applied directly to our scenarios because of nonnegative matrix properties in the dictionary learning algorithm. Inspired by these algorithms, for battery safety hidden danger scenarios, this paper needs to build a new topology expression diagram for battery loads not mentioned before, and to build an improved detection algorithm with higher precision.

The contributions of this paper are as follows:

- For the battery load safety hazard factors' detection scenarios, we propose a new topological representation diagram in the load disaggregation domain firstly. In order to tackle the issue of nonnegative matrix properties, we design the shift strategy of battery load.
- The optimal dictionary matrix period algorithm is constructed for improved the accuracy of the improved dictionary learning and sparse coding. In addition, adding the aggregation constraint further improves the algorithm.
- Based on the above two points, we propose a new safety hazard factors detection algorithm for battery load.
- Compared with three baseline algorithms including CO, FHMMExact, and DDSC, our algorithm Bats is more accurate than them on the Reference Energy Disaggregation Dataset (REDD) [23].

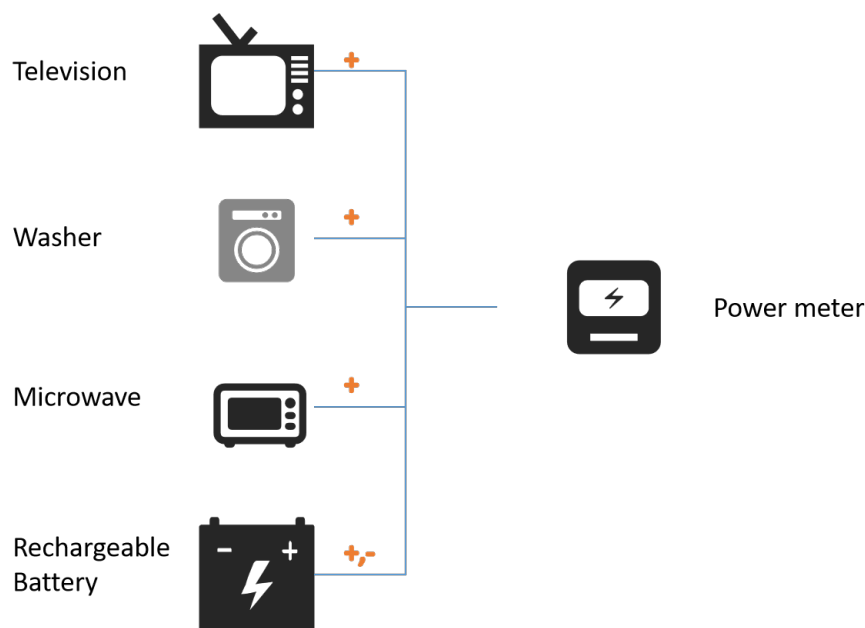
The remainder of this paper is organized as follows. In Section 2, we describe the basic model of the nonintrusive load disaggregation algorithm, the basic process of dictionary

learning and sparse coding. Based on the above basic model, we propose an algorithm about how to further build the model for dictionary learning and sparse coding in the scene of nonintrusive load disaggregation, and we finally make some improvements according to the characteristics of battery safety hazard factors and the requirement for accuracy. Finally, we describe the modeling process for the problem. In Section 3, we describe the establishment and analysis of the model, deducing the detailed process of the algorithm. Section 4 is the experimental part which shows the simulation and analysis results. Section 6 is the concise summary of this paper, and the possible research directions in the future are prospected.

## 2. Problem Modeling

### 2.1. Preliminary

In the electrical microenvironment, the topologically logic relationship among the electrical appliances is shown in Figure 1. It is assumed that the power consumption is mathematically positive, and the power production is mathematically negative. The mathematically positive load includes the TV set, washing machine, microwave oven, refrigerator, and the energy storage battery in charging time slot, etc., while the negative load refers to the energy storage battery in the discharge that provides the power consumption time slot. In the case of the whole time slot, the energy storage battery shows partially positive and negative power consumption curve with alternating operation mode. The other commonly used electrical appliances continue to maintain the original operation mode of power consumption. For the convenience of reading, most of the important mathematical symbols are adopted in this paper, and the corresponding description is listed in Table 1.



**Figure 1.** New topological logic diagram of electrical microenvironment with potential safety hazards.

In this way, we can define the aggregate power consumption on the meter or the plug board as  $y(t)$ ,  $t \in (1, 2, \dots, T)$ , where  $T$  is the length of the entire time slot. Suppose that there are  $N$  electrical appliances, the  $i$ th electrical equipment in power consumption expression of time slot  $t$  is  $x_i(t)$ ,  $t \in (1, 2, \dots, T)$ ,  $i \in (1, 2, \dots, N)$ ; therefore,

$$y(t) = x_1(t) + \sum_{k=2}^N x_k(t) + \epsilon(t) \quad (1)$$

where  $x_1(t)$  represents the energy storage battery with the mathematically positive or negative power consumption,  $x_i(t)$ ,  $t \in (2, \dots, N)$  represents other kinds of commonly

used electrical appliances, and  $\epsilon(t)$  represents the random noise of system operation and measurement.

**Table 1.** Academic terms covered in this paper.

Symbol	The Meaning of the Symbol
$Y$	Aggregation power
$X_i, i \in (1, 2, \dots, N)$	Power of $i$ th appliance
$y(t), t \in (1, 2, \dots, T)$	Aggregation power in $t$
$x_i(t)$	Actual power of $i$ th appliance in $t$
$\epsilon(t)$	Power noise in $t$
$\hat{y}(t), t \in (1, 2, \dots, T)$	Estimated aggregation power in $t$
$\hat{x}_i(t), t \in (2, \dots, N)$	Estimated power of $i$ th appliance in $t$
$t \in (1, 2, \dots, T)$	Time slot
$i \in (1, 2, \dots, N)$	Index of all appliances
$j \in (1, 2, \dots, M)$	The atom numbers of dictionary
$B$	Fundamental matrix of Fourier transform
$A$	Coefficient matrix of Fourier transform
$D$	Dictionary representation of aggregate
$D_i, i \in (1, 2, \dots, N)$	Dictionary representation of $i$ th appliance
$D_{1:k}$	$\{1 : k\}$ Appliance dict concatenation
$\mathbf{d}_i^{(j)}$	The subvector of dictionary
$C$	Aggregation sparse coefficient
$\hat{C}$	Estimated aggregation sparse coefficient
$C^*$	Optimal aggregation sparse coefficient
$C_i, i \in (1, 2, \dots, N)$	$i$ th appliance sparse coefficient
$C_{1:k}$	$\{1 : k\}$ Appliance sparse coefficient
$\hat{C}$	Estimated aggregation sparse coefficient

Due to practical constraints, we cannot provide all electrical appliances, charging and discharging appliances with some safe, expensive and large power sensors, or configure the corresponding signal processing and data transmission module. Generally speaking, only a single power sensor device is deployed at the entrance of the electrical micro-environment, such as a building, a room, or a station. In the case of a few power sensor points' deployment, we can only measure and record the aggregate power data expressed as  $y_t, t \in (1, 2, \dots, T)$ , but we often hope to get the power consumption of these electrical appliances, which is inferred and computed by signal processing and data mining methods. Inspired by the theory of pattern recognition and numerical estimation, it is assumed that the electrical appliances estimated value  $\hat{x}_i(t), t \in (2, \dots, N)$ . In addition, then add the estimated individual electrical appliance power consumption values to get a new aggregate estimated value  $\hat{y}(t)$ , which is compared with the original aggregate power consumption data  $y_t$ . The goal of estimation is to make the error between the estimated aggregate value and the actual aggregate value as small as possible, which can be formally expressed as:

$$\min \|y(t) - \sum_{i=1}^N \hat{x}_i(t)\|^2 \quad (2)$$

There are many specific modeling and solving methods for general expression (2). Inspired by the expression of Fourier series, any waveform theory can be expressed as the combination of fundamental waves. For load disaggregation modeling methods about the safety hazard factors detection, in order to obtain explicable results, we use the following form to express:

$$\min \|y(t) - BA\|^2 \quad (3)$$

where  $y(t)$  is the signal to be decomposed,  $B$  is the Fourier basis functions, and  $A$  is the coefficients of the basis functions. However, on the whole, the actual deployment of the electrical IOT safety monitoring systems is affected by some factors, such as a lower hardware budget of the sensor, higher computation cost of signal processing and data

analysis, more storage and communication consumption of the back end system, and a bigger amount of connected internet data. Because such systems generally do not acquire high-frequency load data information in real time, it is unrealistic to directly use the Fourier transform method to implement high-frequency signal processing. In this case, the selection of similar and alternative solutions has become one of the basic modelings and solutions.

In this paper, the dictionary learning method is adopted to learn the basis matrix of each electrical appliance from the time-series data sampled at low frequency, and the activation matrix corresponding to the basis matrix is similar to the basis function and coefficient in the Fourier transform method. In Equation (4),  $D$  represents the basis matrix, and  $C$  represents the sparse coefficient matrix. The error between the expression of the estimated aggregate obtained by the combination of  $D$  and  $C$  can be measured by the Euclidean distance. Dictionary learning and sparse coding theory show that  $C$  should be as sparse as possible. While keeping the error small, sparse means to make as many zero items in the sparse coefficient matrix as possible,

$$\min_C \|y(t) - DC\|_2^2 + \lambda \|C\|_1 \quad (4)$$

where  $\|y(t) - DC\|_2^2$  is Euclidean distance expression, and  $\lambda \|C\|_1$  is the penalty term.  $\ell_1$  norm or  $\ell_2$  norm regularization helps reduce overfitting and implements sparse solution problems.  $\ell_1$  norm regularization uses proximal gradient descent. Tibshirani et al. [24] described the reason why the  $\ell_1$  norm is chosen for the penalty item. The  $\ell_1$  norm represents a rectangle in the coordinate system of the solution, intersecting a circle constructed by the quadratic function of the objective function, usually on the coordinate axis. In the coordinate system of the solution,  $\ell_2$  norm presents a circle with the origin of coordinates as the center of the circle, and intersects the circle constructed by the quadratic function of the objective function. Generally, it will not intersect on the coordinate axis. It can be seen from the properties of the solution of activation matrix or sparse representation that the sparsity of the solution of  $\ell_1$  norm type is better than that of  $\ell_2$  norm. In this way, we can construct the objective function of sparse coding for dictionary learning of  $\ell_1$  norm type, and finally solve the safety hazard factors detection problem to satisfy the requirements of the system.

## 2.2. Determination of Objective Function

In this section, Equation (4) is further transformed into an objective function with coefficients. According to the derivation process of Taylor's Equation, the transformation and derivation of L-Lipschitz condition realization problem are figured out in [25].

Assuming that  $\nabla$  represents a differential operator and the objective function is  $f(z)$ , the problem is constructed as the following objective function:

$$\min_z f(z) + \lambda \|z\|_1 \quad (5)$$

where  $z$  is the independent variable. In addition, find the smallest  $z$  value by minimizing the objective function  $f(z)$ . If the objective function  $f(z)$  is differentiable, and  $\nabla f$  meets the L-Lipschitz condition, where  $L > 0$  is constant, the gradient inequality is formed as follows:

$$|\nabla f(z') - \nabla f(z)|_2^2 \leq L \|z' - z\|_2^2 \quad (\forall z, z') \quad (6)$$

The second-order Taylor expansion of the objective function  $f(z)$  is carried out near  $z_k$ , and its expression is approximately as follows:

$$\begin{aligned} \hat{f}(z) &\simeq f(z_k) + \nabla f(z_k)(z - z_k) + \frac{L}{2} \|z - z_k\|_2^2 \\ &= \frac{L}{2} \left\| z - \left( z_k - \frac{1}{L} \nabla f(z_k) \right) \right\|_2^2 + \text{CST} \end{aligned} \quad (7)$$

where  $\hat{f}(z)$  is the estimated value of the object function,  $z_k$  is the specific value of  $z$ ,  $L$  is constant coefficient value, and  $CST$  is constant. The minimum value will be obtained at  $z_{k+1}$ :

$$z_{k+1} = z_k - \frac{1}{L} \nabla f(z_k) \quad (8)$$

Through the gradient descent method,  $f(z)$  can be minimized by iterative computations. With each step of gradient descent iteration, the quadratic function  $\hat{f}(z)$  is minimized to get

$$z_{k+1} = \arg \min_z \frac{L}{2} \left\| z - \left( z_k - \frac{1}{L} \nabla f(z_k) \right) \right\|_2^2 + \lambda \|z\|_1 \quad (9)$$

where  $L$  can be simplified as constant value 1.

### 3. Model Solution and Algorithm Analysis

This section includes the detailed derivation process of the optimization objective function. Firstly, the dictionary learning algorithm of single appliance explains the dictionary belonging to every appliance. In addition, the load disaggregation error function states the iteration optimization methods. Then, the optimal dictionary matrix period algorithm can find one of the maximum periods for the improved dictionary learning sparse coding algorithm.

#### 3.1. The Optimization Objective

Based on the input data  $X_i$ , the basis matrix  $D_i$  and the activate matrix  $C_i$ , we can build the minimization objective function as follows:

$$P1 : \min_{D_i \geq 0, C_i \geq 0} \frac{1}{2} \|X_i - D_i C_i\|_2^2 + \lambda \|C_i\|_1, i = 1, \dots, N$$

$$\text{s.t.} \quad \|c_i^{(j)}\|_2 \leq 1, j = 1, \dots, M \quad (10)$$

$$\sum_{t=1}^T X_i(t) \leq \text{const}_i$$

where the coefficient value  $\frac{1}{2}$  is the coefficient of second derivative term of the Taylor series expansion Equation (7),  $\text{const}_i$  is the constraint value of the  $i$ th appliance, i.e., the maximum power value. In this way, when the objective function is doing gradient descending, by adjusting the regularization coefficient  $\lambda$ , the relationship between sparse coding error and sparsity of sparse matrix  $C_i$  is balanced. In addition, the sparse expression obtained finally meets the sparse condition. At the same time, we add the coefficient of sparse matrix normalized constraints  $\|c_i^{(j)}\|_2 \leq 1, j = 1, \dots, M$ , in order to balance the sub-dictionary of relations among the weights of all atoms. The second constraint condition is the load disaggregation value finally solved, which needs to satisfy a condition. This condition is that each electrical appliance has a threshold value of cumulative sum of power consumption, and the whole operation process cannot exceed the threshold value.

From Equation (10), we can see that the objective function to be optimized includes two optimization variables:  $D_i$ , which is the dictionary basis matrix of the  $i$ th electrical appliances, and  $C_i$  which is the corresponding sparse expression. According to the description of the literature in [26], the natural solution to the problem is to fix a variable and solve another variable. At this time, the convex optimization theory can be used to obtain the corresponding solution through the derivation.

The values  $D_i$  and  $C_i$  can be solved by the above alternative optimization method; then, they are used to construct the dictionary of  $[1 : k]$  electrical appliances by means of matrix concatenation. The dictionary construction of all electrical appliances is formed as follows, and its formal expression is as follows:



$$\begin{aligned}\hat{\mathbf{D}}_{1:k} &= \arg \min_{\mathbf{D}_{1:k} \geq 0} \|\mathbf{Y} - \mathbf{D}_{1:k} \mathbf{C}_{1:k}\|_2^2 + \lambda \|\mathbf{C}_{1:k}\|_1 \\ &= \arg \min_{\mathbf{D}_{1:k} \geq 0} F(\mathbf{Y}, \mathbf{D}_{1:k}, \mathbf{C}_{1:k})\end{aligned}\quad (11)$$

Assuming the fixed dictionary  $D$ , the estimated sparse code  $\hat{C}$  can be calculated by  $\arg \min_{\mathbf{C}_{1:k} \geq 0} F(\mathbf{Y}, \mathbf{D}_{1:k}, \mathbf{C}_{1:k})$ . In Equation (11),  $F(\mathbf{Y}, \mathbf{D}_{1:k}, \mathbf{C}_{1:k})$  is equivalent to  $\|\mathbf{Y} - \mathbf{D}_{1:k} \mathbf{C}_{1:k}\|_2^2 + \lambda \|\mathbf{C}_{1:k}\|_1$ . After calculating the estimated sparse code  $\hat{C}$ , the estimated power value of the  $i$ th electrical appliance can be obtained:

$$\hat{\mathbf{X}}_i = \mathbf{D}_i \hat{\mathbf{C}}_i \quad (12)$$

In this way, the problem P1 is transformed into the problem P2, and the load disaggregation error can be expressed as follows:

$$\begin{aligned}P2 : \min E(\mathbf{X}_{1:k}, \mathbf{D}_{1:k}) &= \sum_{i=1}^k \frac{1}{2} \|\mathbf{X}_i - \mathbf{D}_i \hat{\mathbf{C}}_i\|_2^2 \\ \text{s.t. } \hat{\mathbf{C}}_{1:k} &= \arg \min_{\mathbf{C}_{1:k} \geq 0} F(\mathbf{Y}, \mathbf{D}_{1:k}, \mathbf{C}_{1:k}) \\ &\|\mathbf{c}_i^{(j)}\|_2 \leq 1, j = 1, \dots, M \\ &\sum_{t=1}^T X_i(t) \leq \text{const}_i\end{aligned}\quad (13)$$

where  $E(\cdot)$  is the error function.  $j \in (1, 2, \dots, M)$  is the atom numbers of dictionary.  $\mathbf{C}_{1:k}$ ,  $\{1 : k\}$  is the concatenation formation of the appliance sparse coefficients.

However, because it is not easy to solve the error function minimization, a new penalty term is introduced here to convert the problem P2 into the problem P3, which is not a convex optimization problem, and it can be solved by using the gradient descent algorithm:

$$\begin{aligned}P3 : \min E_{reg}(\mathbf{X}_{1:k}, \mathbf{D}_{1:k}) &= E(\mathbf{X}_{1:k}, \mathbf{D}_{1:k}) + \lambda (\|\hat{\mathbf{C}}_{1:k}\|_1) \\ &= \sum_{i=1}^k \frac{1}{2} \|\mathbf{X}_i - \mathbf{D}_i \hat{\mathbf{C}}_i\|_2^2 + \lambda (\|\hat{\mathbf{C}}_{1:k}\|_1) \\ \text{s.t. } \hat{\mathbf{C}}_{1:k} &= \arg \min_{\mathbf{C}_{1:k} \geq 0} F(\mathbf{Y}, \mathbf{D}_{1:k}, \mathbf{C}_{1:k}) \\ &\|\mathbf{c}_i^{(j)}\|_2 \leq 1, j = 1, \dots, M \\ &\sum_{t=1}^T X_i(t) \leq \text{const}_i\end{aligned}\quad (14)$$

where  $E_{reg}(\cdot)$  is the error function with regularization item.

The power consumption data matrix  $\mathbf{X}_i$  of the  $i$ th electrical appliance is used for sparse coding iteration to obtain the optimal activation coefficient matrix  $\mathbf{C}^*$ :

$$\mathbf{C}_i^* = \arg \min_{\mathbf{C}_i \geq 0} \frac{1}{2} \|\mathbf{X}_i - \mathbf{D}_i \mathbf{C}_i\|_2^2 + \lambda (\|\hat{\mathbf{C}}_{1:k}\|_1) \quad (15)$$

where  $\mathbf{C}_i^*$  is the optimal activation coefficient matrix of the  $i$ th appliance. Then, concatenate these coefficient matrices into a bigger matrix.

Then, the iteration of gradient descent is performed to update the rules as follows:

$$\tilde{\mathbf{D}} \leftarrow \tilde{\mathbf{D}} - \alpha \left( (\mathbf{Y} - \tilde{\mathbf{D}} \hat{\mathbf{C}}) \hat{\mathbf{C}}^T - (\mathbf{Y} - \tilde{\mathbf{D}} \mathbf{C}^*) \mathbf{C}^{*T} \right) \quad (16)$$

where the update rate or learning rate is  $\alpha$ , which is the step size of each step of gradient descent. The solution with smaller error is found by controlling the step size in the process of solving this problem.

For each atom learned by updating iteratively the dictionary, for the convenience of further interference, the sub-vectors of all the learned dictionary  $D$  are normalized as follows:

$$\mathbf{d}_i^{(j)} \leftarrow \mathbf{d}_i^{(j)} / \|\mathbf{d}_i^{(j)}\|_2, i \in (1, 2, \dots, N), j \in (1, 2, \dots, M) \quad (17)$$

where  $\mathbf{d}_i^{(j)}$  is the  $j$ th vector of the  $i$ th appliance's learned dictionary.

### 3.2. Dictionary Learning Period Optimal Algorithm

$X_i(t), i = 1, \dots, N, t = 1, \dots, T$  is the power value of the  $i$ th appliance.  $N_{period}$  is the total period number of every appliance.  $T_i, i = 1, \dots, N$  is the average period of the  $i$ th appliance. In addition, the unit of  $T_i$  is the number of samples and the interval of two samples is 60 s. Through data exploring, it is found that different appliances have different operation periods as Figure 2 shows. Is it possible to consider different period window size, which may lead to different precision? The answer is yes.

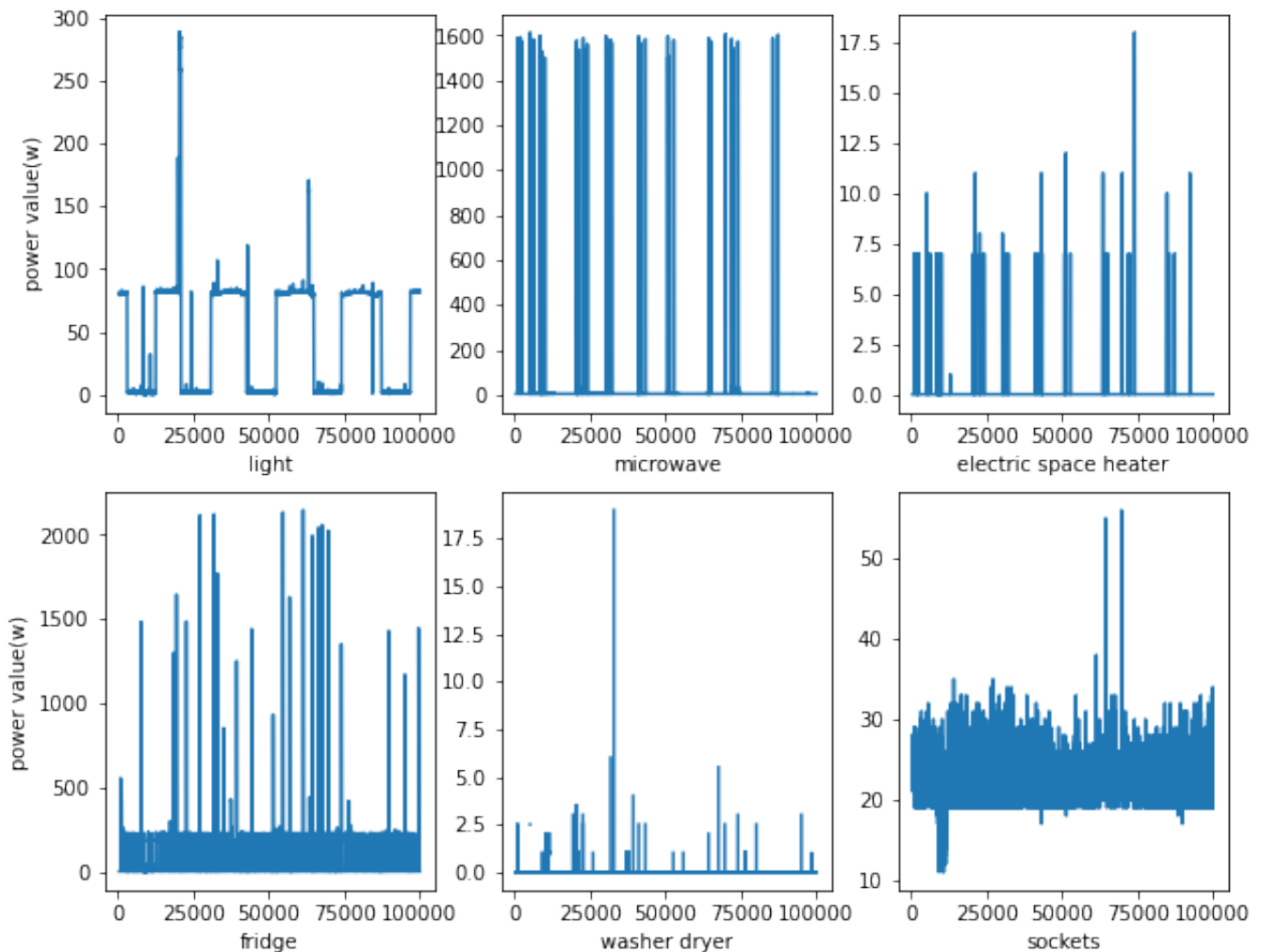


Figure 2. Appliance period characteristics in REDD dataset.



Therefore, as Algorithm 1 describes, we compute every appliance typical period time statistically. Then, for every appliance, if the power value is greater than the threshold, then mark this timestamp as the start point of power period. Calculate the interval between two marked start points of power value. Finally, select one of maximum period times as the optimal dictionary learning period:

---

**Algorithm 1:** Dictionary Learning Period Optimal algorithm

---

```

Init  $T_i = 0, i = 1, \dots, N$ ;
1. statistically computing the every appliance period;
while  $\forall X_i(t), i = 1, \dots, N$  do
    while  $t=1, \dots, T$  do
        if single power value  $X_i(t) \geq \text{threshold}$  then
            Mark the start of the power period;
            while meet next new start point do
                |  $T_i ++$ ;
            end
        end
         $t ++$ ;
    end
     $i ++$ ;
end
2. selecting one of the maximum for these appliance periods;
 $m^* = \max(T_i), i = 1, \dots, N$ .

```

---

### 3.3. Improved Dictionary Learning Sparse Coding Algorithm

In Algorithm 2, firstly, the optimal window size  $M^*$  is calculated by Algorithm 1, which is used as the segmentation method of time series data  $X_i$ . In addition, the time series data can be converted into matrix form. Then, the power values with a negative value are moved to positive values by adding some shift value. The same shift values are also added to the aggregate power consumption data. Thus, the new battery power values and aggregate power values are formed. Then, the positive values are initialized for  $D_i$  and  $C_i$ , and normalized for  $D$ . The dictionary learning algorithm is learned for each appliance to train the dictionary and the corresponding sparse code of every appliance. The learned dictionary is concatenated into a new sparse coding matrix. Update the dictionary according to the learning rate  $\alpha$ . After adding the aggregate constraints, the optimization is iterated continuously. Through the above process, the optimal sparse coding matrix is obtained. Finally, in the dataset, the dictionary is multiplied by the sparse code matrix to obtain the predicted load decomposition power. The flowchart about Algorithms 1 and 2 is shown in Figure 3.

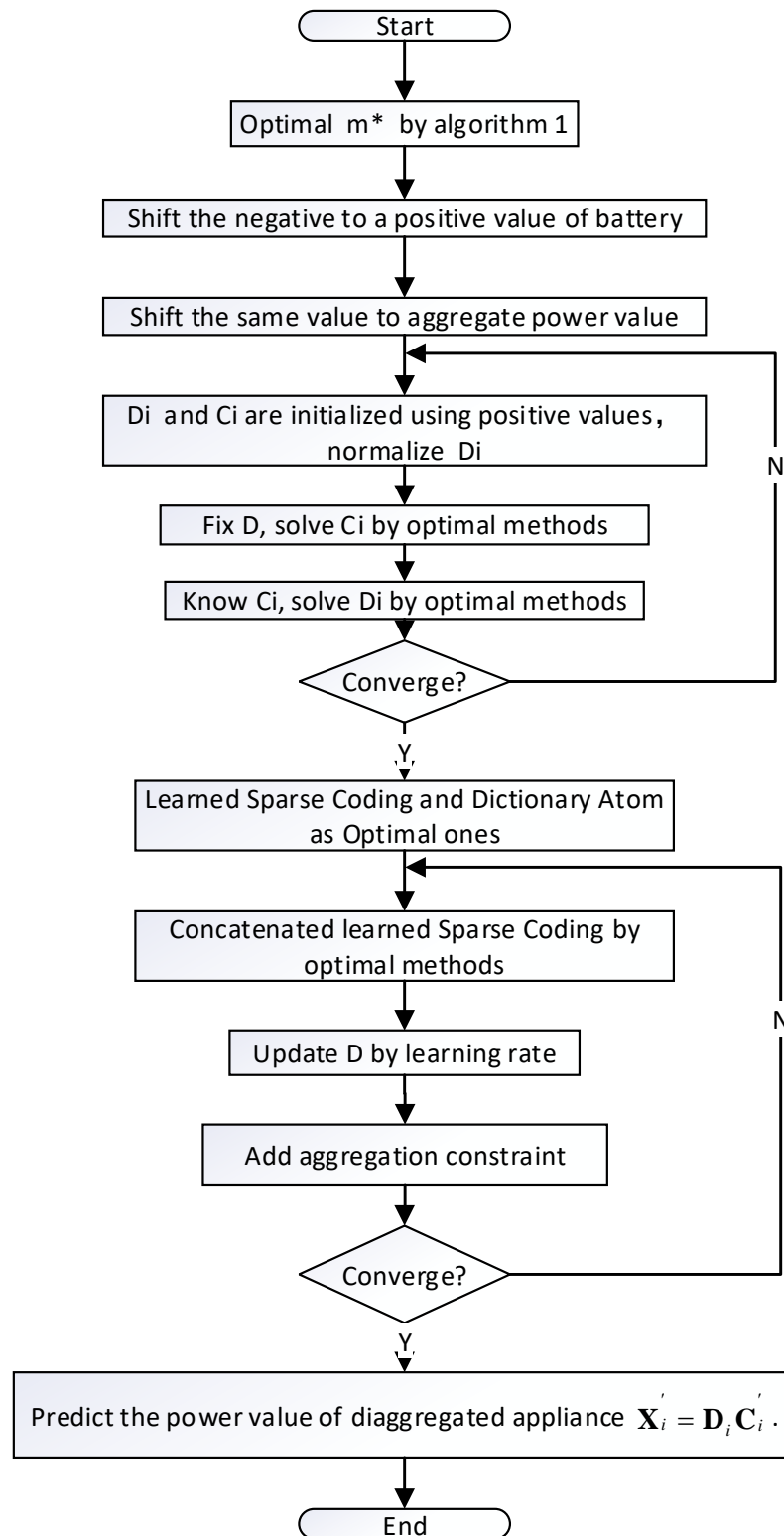


Figure 3. Flowchart about Algorithms 1 and 2.

**Algorithm 2:** Improved dictionary learning sparse coding algorithm**Input:** Input data and constant.The power value of the  $i$ th electrical appliance  $\mathbf{X}_i \in \mathbb{R}^{T \times m}, i = 1, \dots, N$ ;Optimal  $m$  via Algorithm 1;The regularization parameter  $\lambda \in \mathbb{R}_+$ ;The gradient step  $\alpha \in \mathbb{R}_+$ ;**Output:** Prediction value of an appliance  $\hat{\mathbf{X}}'_i$ .

1. Shift the negative to a positive value of battery's power consumption;

2.  $D_i$  and  $C_i$  are initialized using positive values, and the column vector of  $D_i$  is normalized so that  $\|\mathbf{d}_i^{(j)}\|_2 = 1, i \in (1, 2, \dots, N), j \in (1, 2, \dots, M)$ ;3. **while** Iterate for each appliance until convergence **do**

(a)

$$\mathbf{C}_i \leftarrow \arg \min_{\mathbf{C} \geq 0} \|\mathbf{X}_i - \mathbf{D}_i \mathbf{C}\|_2^2 + \lambda \|\mathbf{C}_{1:k}\|_1$$

(b)

$$\mathbf{D}_i \leftarrow \arg \min_{\mathbf{D} \geq 0, \|\mathbf{d}^{(j)}\|_2 \leq 1} \|\mathbf{X}_i - \mathbf{D} \mathbf{C}_i\|_2^2$$

**end**4. Optimal Sparse coding value as  $\mathbf{C}_{1:k}^* \leftarrow \mathbf{C}_{1:k}$ ;Estimated Dictionary as  $\tilde{\mathbf{D}}_{1:k} \leftarrow \mathbf{D}_{1:k}$ ;5. **while** Iterate to convergence **do**

(a)

$$\hat{\mathbf{C}}_{1:k} \leftarrow \arg \min_{\mathbf{C}_{1:k} \geq 0} F(\mathbf{Y}, \tilde{\mathbf{D}}_{1:k}, \mathbf{C}_{1:k})$$

(b)

$$\tilde{\mathbf{D}} \leftarrow \left[ \tilde{\mathbf{D}} - \alpha \left( (\mathbf{Y} - \tilde{\mathbf{D}} \hat{\mathbf{C}}) \hat{\mathbf{C}}^T - (\bar{\mathbf{X}} - \tilde{\mathbf{D}} \mathbf{C}^*) (\mathbf{C}^*)^T \right) \right]$$

(c) For all  $i, j$ :

$$\mathbf{d}_i^{(j)} \leftarrow \mathbf{d}_i^{(j)} / \|\mathbf{d}_i^{(j)}\|_2$$

(d) Add aggregation constraint:

$$\sum_{t=1}^T X_i(t) \leq \text{const}_i$$

**end**6.  $\hat{\mathbf{C}}'_{1:k} \leftarrow \arg \min_{\mathbf{C}_{1:k} \geq 0} F(\mathbf{Y}', \tilde{\mathbf{D}}_{1:k}, \mathbf{C}_{1:k})$ ;7. Predict the power value of disaggregated appliance,  $\hat{\mathbf{X}}'_i = \mathbf{D}_i \hat{\mathbf{C}}'_i$ .**4. Experimental Results and Analysis****4.1. Datasets for Experiments**

The Reference Energy Disaggregation Dataset (REDD) is a representative, public, and freely available dataset that has frequently been utilized to explore all kinds of nonintrusive load disaggregation algorithms [23]. In order to facilitate the experiments, this paper modified and synthesized the battery simulation data based on REDD datasets. At the same time, four electrical appliances, such as battery, fridge, sockets, and light, were selected from building 1 to carry out the experiments of the improved dictionary learning and sparse coding algorithms. The dataset for battery devices is from synthetic data. Currently, there is no existing real dataset of battery electrical appliances for our experiments. The data can be obtained through simulation and then transformed to a completely positive condition for verification.

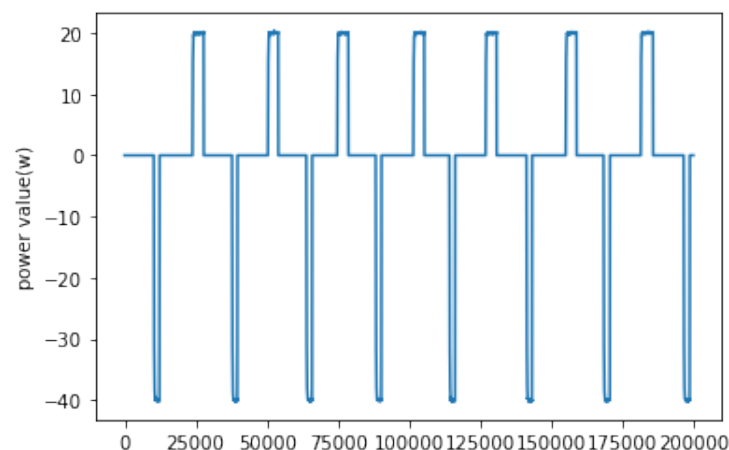
As Table 2 shows, the time of the REDD dataset is selected to execute 80% for training and 20% for testing. In fact, the training dataset lasted from 18 April 2011 to 25 February

2012. The test dataset lasted from 25 February 2012 to 25 May 2012. According to the method of 1-min resample period, we get a new sample dataset.

**Table 2.** Experiment case for REDD datasets.

Index	Item	Time	Duration
1	train start time	2011.04.18	313 days
2	train end time	2012.02.25	
3	test start time	2012.2.25	90 days
4	test end time	2012.5.25	

As Figure 4 shows, battery charging is embodied as external consumption power consumption, and the maximum is 20 W. In addition, the minimum value is  $-40$  W, indicating that the average power consumption of external power supply is 40 W. To facilitate the experiments, the first phase of the experimental simulation requires a relatively ideal record of battery charging and discharging. It is further proved that the power consumptions of charging and discharging is close to each other in geometric area. In the future, we will test our algorithm on the real scenarios of battery loads.



**Figure 4.** Ideal simulation of battery charging and discharging curves and their data sets.

#### 4.2. Experimental Setting

These algorithms are implemented in Python based on the NILMTK [17,27]. This experiment is run on a desktop computer with GPU 1080i, Intel Core i5-10400 CPU, 2.9 GHz CPU physical frequency, and 16 GB memory capacity of a Windows 10 operating system. In order to make operation convenient and meet the needs of comparison experiments, this paper builds a virtual machine environment of Anaconda, which is a professional platform in the domain of data science research. It creates an isolated operating environment and installs all kinds of Python installation packages, such as numpy, matplotlib, cvxpy, hmmlearn, scikit-Learn, TensorFlow, Keras, and so on.

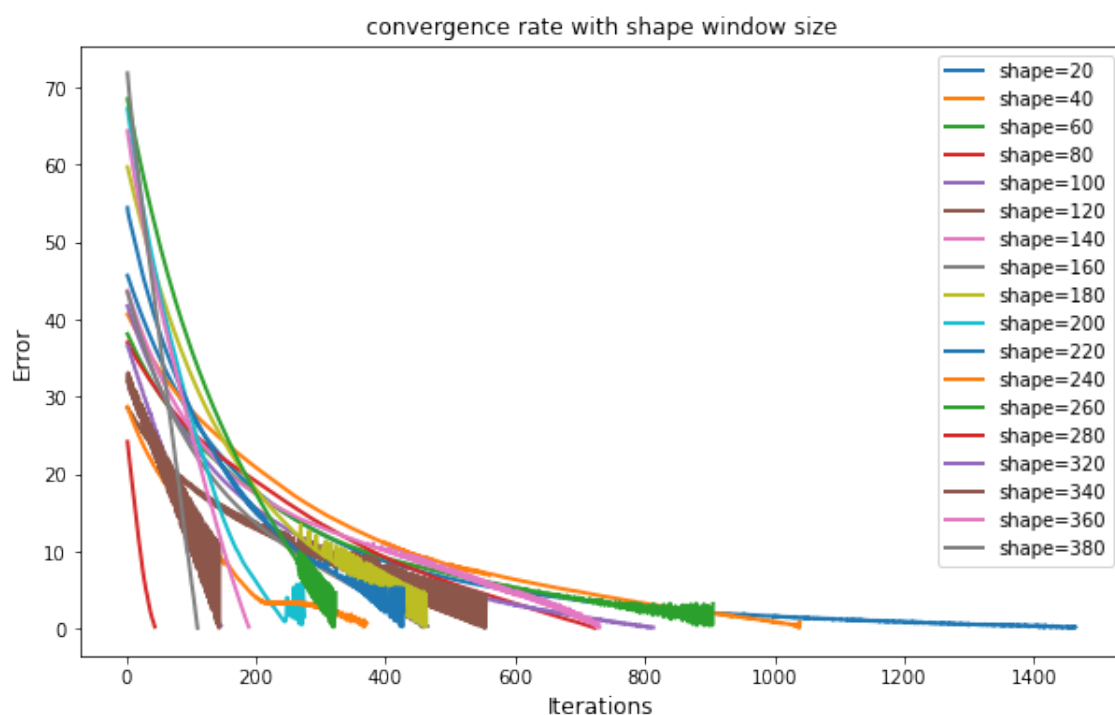
The baseline algorithms including CO, FHMMExact, and DDSC are selected as the experimental comparison objects. For the convenience of the experiment, we adopt the state-of-the-art load disaggregation framework in the domain of load disaggregation research, NILMTK-Contrib, as an important means of our evaluation. In this unified framework, three algorithms can obtain data, preprocess data, train models, and test them.

Furthermore, in the process of a simulation experiment, this paper selects the following specific hyper-parameters and corresponding explanations in Table 3. The regularization coefficient  $\lambda$  is 20. The learning rate of dictionary learning  $\alpha$  is  $10^{-12}$ . The max iterative step is 10,000. Certainly, as the operation converges gradually, the calculation will not generally

run to the maximum number of iterations as Figure 5. In all these experiments, the atom number of dictionary learning  $n$  is 10.

**Table 3.** Experiment parameters.

Index	Parameter	Parameter Name	Parameter Value
1	$\lambda$	coefficient	20
2	$\alpha$	update step	$10^{-12}$
3	$const_i$	constraint const	0.1
4	T	time length	entire dataset
5	Step	max iterative step	10,000
6	n	atom number	10
7	error	objective function error	0.1
8	m	default matrix shape	120 or variable



**Figure 5.** Disaggregation convergence rate with shape window size in the REDD dataset.

#### 4.3. Convergence Analysis

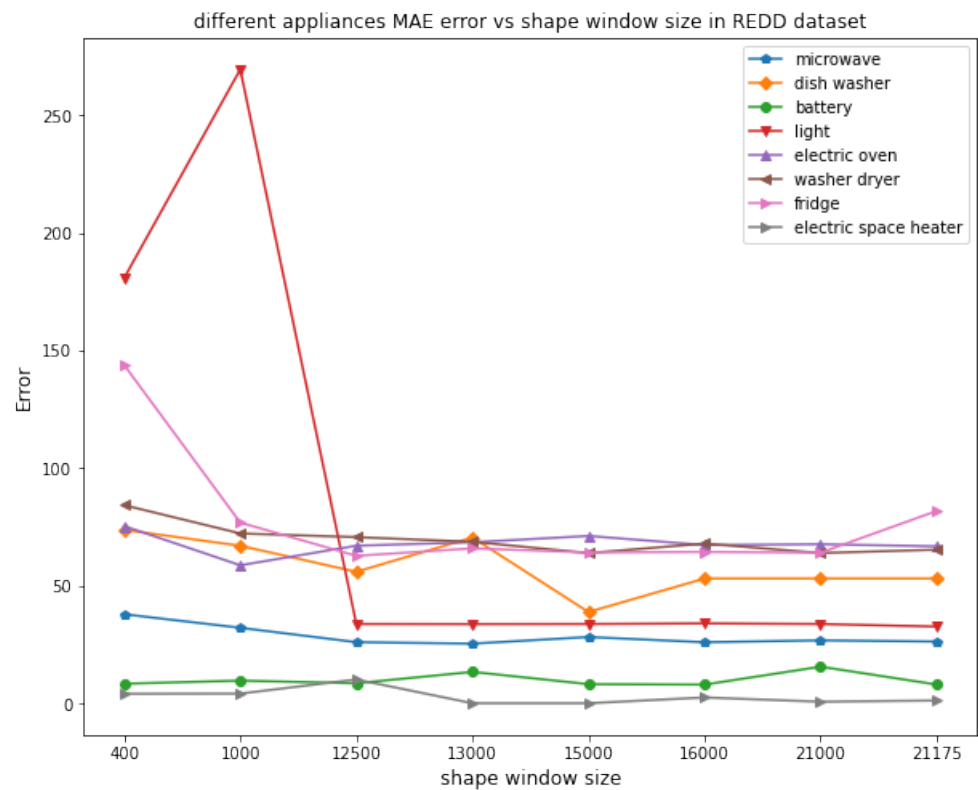
Experimenting with Algorithm 1, the window segmentation experiment shows that different window sizes have different convergence rates, where the unit of convergence rate is the number of iterations. As Figure 5 is shown and Table 4 is displayed, under different segmentation windows, the optimization objective or the error function converge over time. In general, with the increase of iteration times, the convergence effect is getting better and better, while the error function is getting smaller and smaller. Therefore, the choice of the window size is very important for the effect of the convergence rate. As Table 4 says, convergence experiments were performed in the window size, ranging from 20 to 380. The result is that the minimum windows for convergence are 280 samples, and the corresponding iterations are 40 iterations. At the same time, the maximum one for convergence is 20 samples, and the corresponding iterations are 1465.

In this subsection, we investigate the convergence and carry out error analysis of the dictionary learning method for load disaggregation. In Equation (11), we present the error between the previous object function value and the current object function value. In Figure 5, the error reduction is shown along with the number of iterations. From Figure 6,

we can see the error variation of microwave as the shape window size increases. If the shape window size changes, the MAE value is not always better than the other algorithms. Thus, the optimal shape windows size is considered as one of the most important factors.

**Table 4.** Statistical measures on minimum, maximum, average, and standard deviation values of errors for different presented cases.

Index	Shape Size	Iteration Numbers	Remarks
1	20	1465	maximum
2	40	1040	-
3	60	906	-
4	80	724	-
5	100	814	-
6	120	556	-
7	140	731	-
8	160	466	-
9	180	464	-
10	200	276	-
11	220	429	-
12	240	372	-
13	260	324	-
14	280	45	minimum
15	320	147	-
16	340	145	-
17	360	190	-
18	380	111	-
minimum	280	40	-
maximum	20	1464	-
average	-	511	-
standard deviation	-	376	-



**Figure 6.** Error change of microwave with shape window size in the REDD dataset.

#### 4.4. Metrics

Compared with three baseline algorithms CO, FHMMExact, and DDSC, we need to perform their measurement comparisons using the same metrics.

##### 4.4.1. MAE

In statistics, the Mean Absolute Error (MAE) is the measurement error of a pair of observations which express the same phenomenon. For instance, as described in this paper, the mean absolute value error is expressed as follows:

$$MAE_i = \frac{1}{T} \sum_{t=1}^T |\hat{x}_i(t) - x_i(t)| \quad (18)$$

where  $\hat{x}_i(t)$  is the estimated power consumption of the  $i$ th electrical appliance in the time slot  $t$ , while  $x_i(t)$  is the real power consumption of the  $i$ th electrical appliance in the time slot  $t$ .  $T$  is the length of the entire dataset, and the MAE is the mean power consumption error of the  $i$ th electrical appliance in the entire dataset. As a traditional indicator in the domain of pattern recognition or measurement, this indicator can be seen in all kinds of literature, and it is the most important indicator of load disaggregation.

##### 4.4.2. RMSE

The Root Mean Square Error (RMSE) is often used to quantify the measurement error of a pair of observations. Its mathematical expression is as follows:

$$RMSE_i = \sqrt{\frac{1}{T} \sum_t (\hat{x}_i(t) - x_i(t))^2} \quad (19)$$

where the  $RMSE_i$  is the Root Mean Square Error value of the  $i$ th electrical appliance,  $\hat{x}_i(t)$  is the estimated power consumption of the  $i$ th electrical appliance in the time slot  $t$ , and  $x_i(t)$  is the real power consumption of the  $i$ th electrical appliance in the time slot  $t$ ,  $T$  is the time length of the whole training and testing set, and the  $RMSE_i$  is the root mean square error of the  $i$ th electrical appliance in the whole time period. The root mean square error is calculated, which corresponds to the Euclidean distance or Euclidean norm, and could also be called  $\ell_2$  norm, pronounced  $\| \cdot \|_2$  or  $\| \cdot \|$ .

#### 4.5. Experiment Result Analysis

Through experiments, Bats (this paper), CO, FHMMExact, and DDSC are compared. Composite data: use a curve similar to the heat pump data and offset the negative number to the positive number line. Aggregated data: battery data, refrigerator data, etc. are integrated for safety hazard factors detection.

According to the REDD dataset, as the basis of dataset, synthetic data are about batteries while the power data of other appliances lasted from 18 April 2011 to 25 May 2012. These algorithms include CO, FHMMExact, DDSC, and Bats as the interpretable algorithms, and you can select some of the main electrical appliances.

It can be seen from Table 5 that, compared with the CO and FHMMExact algorithms in the REDD dataset, the MAE value of the Bats algorithm is at the same level as combinatorial optimization in terms of battery electrical disaggregation results. The MAE values of all appliances are smaller than the other two algorithms. The optimal windows shape size of dictionary learning has a positive effect.

From Table 6, it can be seen that the root mean square error of the Bats algorithm is improved to some extent compared with that of CO, FHMMExact, and DDSC algorithm. The superiority of Bats over the CO and FHMMExact algorithm is due to capturing complex dictionary atoms from the aggregate power data and learning more sequential relationships in the power trace.



**Table 5.** MAE in the REDD dataset.

Appliance Name	CO	FHMMExact	DDSC	Bats
battery	52.79	27.72	15.86	<b>8.26</b>
microwave	86.29	57.92	63.20	<b>28.29</b>
dish washer	113.13	70.58	113.05	<b>38.92</b>
light	69.40	38.05	73.50	<b>33.83</b>
electric oven	149.30	114.04	82.02	<b>71.24</b>
washer dryer	98.04	102.70	92.69	<b>63.97</b>

**Table 6.** RMSE in the REDD dataset.

Appliance Name	CO	FHMMExact	DDSC	Bats
battery	220.97	99.17	114.14	<b>97.75</b>
microwave	263.99	158.54	268.37	<b>157.13</b>
dish washer	265.07	220.22	389.93	<b>174.31</b>
light	101.14	64.87	132.43	<b>59.35</b>
electric oven	573.47	474.11	457.36	<b>453.57</b>
washer dryer	391.21	393.78	441.01	<b>390.07</b>

At present, according to the analysis of experimental data, the Bats algorithm has a better effect than a combinatorial optimization algorithm and factorial hidden Markov model. However, from the perspective of data trends, the effect after negative translation is basically explained, indicating that the Bats algorithm can be used to detect battery safety trends.

As a new and important application scenario, it is also very valuable. Through the lasso and lars algorithm, the Bats algorithm converges well to a relatively smaller value, for example,  $error = 0.1$ . In the future, the effect of sparse coding algorithm for dictionary learning in load disaggregation, especially in the case of battery load, can be analyzed in depth from the perspectives of initialization value, learning rate, and gradient.

## 5. Discussion

As is mentioned above in Section 1, although the power consumption of appliances can be monitored directly and accurately, this paper should focus on safety hazard monitoring methods under nonintrusive load decomposition scenarios in reality. Considering safety monitoring, we select several relatively explicable approaches based on optimization theory rather than black-box neural network algorithms, such as CO, FHMMExact, and DDSC. As Table 7, the CO algorithm has no regard for temporal correlation of power consumption data, the CO and FHMMExact transform power consumption data into state data and first three algorithms have not been designed for battery scenarios. Therefore, in this paper, when we design an algorithm for our research scenarios, dictionary learning algorithms can overcome the loss of time correlation for CO and the loss of information based on the state algorithm for FHMMExact. For battery scenarios, we design some algorithm improvements including the shift strategy of battery load and optimal dictionary matrix period algorithm. At the same time, we add an aggregation constraint.

**Table 7.** Comparison of several algorithms.

Algorithms	Temporal Correlation	State or Non State Based	Specially for Battery	Black-Box
CO [7]	No	State based	No	NO
FHMMExact [17]	Yes	State based	No	NO
DDSC [4]	Yes	Non State based	No	NO
Seq2point [28]	Yes	Non State based	No	Yes
DTDl [21]	Yes	Non State based	No	Yes
Bats	Yes	Non state Based	Yes	NO

However, there are some limitations in our proposed algorithm Bats. The first main limitation of Bats is that only simulation tests are performed. In the future, we will establish a test-bed to verify the algorithm. The second main limitation of Bats is that, in the process of dictionary learning and sparse coding, all the input training and testing dataset is low frequency power consumption data that are not beneficial for the algorithm's efficiency. In the future, we will take the high frequency dataset into account.

## 6. Conclusions

As unsafe electrical appliances including batteries are important for life and property, direct monitoring can ensure the safety monitoring timely. However, in some scenarios, there is no choice but to adopt the nonintrusive load disaggregation methods. Based on the principle that charging power is mathematically positive and discharging power is negative, we propose a new topological representation diagram. Inspired by the idea of Fourier transform algorithms and dictionary learning algorithms, we present an electrical safety hazard factors algorithm by using the improved dictionary learning and sparse coding methods, including an optimal dictionary matrix period. In our algorithm, we build the minimization objective function and adopt the gradient descent algorithm to fix the approximate solution for this problem. Compared with three baseline algorithms CO, FHMMExact, and DDSC, our algorithm is more accurate than them on the dataset REDD. The MAE value is 8.26, which drops by 70%. The RMSE value is 97.75, which is also better than these baseline algorithms. In conclusion, our algorithm has achieved some degree of feasibility but still has some room to improve. In future work, we will continue to explore higher precision algorithms while maintaining the interpretability for safety-related tasks.

**Author Contributions:** Conceptualization, W.W., L.C.; Formal analysis, M.G., Z.W. and W.W.; Investigation, Y.C., W.W. and Z.C.; Methodology, M.G., Y.N. and W.W.; Software, Z.W. and W.W.; Supervision, L.C.; Validation, H.L.; Writing—original draft, W.W. and Z.W.; Funding, L.C.; Writing—review and editing, W.W., Y.C. and H.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by the National Natural Science Foundation of China under Grant No. 62072319, in part by the Foundation of Science and Technology on Communication Security Laboratory (No. 6142103190415), and in part by the Key Research and Development Program of the Science and Technology Department of Sichuan Province under Grant No. 2020YFG0254.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data sharing is not applicable to this article.

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

## References

1. Beláň, A.; Cintula, B.; Cenký, M.; Janiga, P.; Bendík, J.; Eleschová, Ž.; Šimurka, A. Measurement of Static Frequency Characteristics of Home Appliances in Smart Grid Systems. *Energies* **2021**, *14*, 1739. [[CrossRef](#)]
2. Campbell, R. *Home Electrical Fires: Supporting Tables*; National Fire Protection Association: Quincy, MA, USA, 2019.
3. Wang, B.; Dehghanian, P.; Wang, S.; Mitolo, M. Electrical Safety Considerations in Large-Scale Electric Vehicle Charging Stations. *IEEE Trans. Ind. Appl.* **2019**, *55*, 6603–6612. [[CrossRef](#)]
4. Kolter, J.; Batra, S.; Ng, A. Energy Disaggregation via Discriminative Sparse Coding. *Adv. Neural Inf. Process. Syst.* **2010**, *23*, 1153–1161.
5. Schirmer, P.A.; Mporas, I.; Sheikh-akbari, A. Energy Disaggregation Using Two-Stage Fusion of Binary Device Detectors. *Energies* **2020**, *13*, 2148. [[CrossRef](#)]
6. Song, J.; Wang, H.; Du, M.; Peng, L.; Zhang, S.; Xu, G. Non-Intrusive Load Identification Method Based on Improved Long Short Term Memory Network. *Energies* **2021**, *14*, 684. [[CrossRef](#)]
7. Hart, G. Nonintrusive appliance load monitoring. *Proc. IEEE* **1992**, *80*, 1870–1891. [[CrossRef](#)]
8. Henaou, N.; Agbossou, K.; Kelouwani, S.; Dubé, Y.; Fournier, M. Approach in Nonintrusive Type I Load Monitoring Using Subtractive Clustering. *IEEE Trans. Smart Grid* **2017**, *8*, 812–821. [[CrossRef](#)]
9. Batra, N.; Singh, A.; Whitehouse, K. If You Measure It, Can You Improve It? The Value of Energy Disaggregation. *BuildSys* **2015**, 191–200. [[CrossRef](#)]

10. He, K.; Jakovetic, D.; Zhao, B.; Stankovic, V. A Generic Optimisation-Based Approach for Improving Non-Intrusive Load Monitoring. *IEEE Trans. Smart Grid* **2019**, *10*, 6472–6480. [[CrossRef](#)]
11. Machlev, R.; Levron, Y.; Beck, Y. Modified Cross-Entropy Method for Classification of Events in NILM Systems. *IEEE Trans. Smart Grid* **2019**, *10*, 4962–4973. [[CrossRef](#)]
12. Bhotto, M.Z.A.; Makonin, S.; Bajić, I.V. Load Disaggregation Based on Aided Linear Integer Programming. *IEEE Trans. Circuits Syst. II Express Briefs* **2017**, *64*, 792–796. [[CrossRef](#)]
13. Piga, D.; Cominola, A.; Giuliani, M.; Castelletti, A.; Rizzoli, A.E. Sparse Optimization for Automated Energy End Use Disaggregation. *IEEE Trans. Control Syst. Technol.* **2016**, *24*, 1044–1051. [[CrossRef](#)]
14. Anderson, K. Non-Intrusive Load Monitoring: Disaggregation of Energy by Unsupervised Power Consumption Clustering. Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 2014.
15. Mengistu, M.; Girmay, A.; Camarda, C. A Cloud-Based On-Line Disaggregation Algorithm for Home Appliance Loads. *IEEE Trans. Smart Grid* **2019**, *10*, 3430–3439. [[CrossRef](#)]
16. Liu, Q.; Kamoto, K.; Liu, X.; Sun, M.; Linge, N. Low-Complexity Non-Intrusive Load Monitoring Using Unsupervised Learning and Generalized Appliance Models. *IEEE Trans. Consum. Electron.* **2019**, *65*, 28–37. [[CrossRef](#)]
17. Batra, N.; Kukunuri, R. Towards reproducible state-of-the-art energy disaggregation. In Proceedings of the 5th International Conference on Future Energy Systems, Xi'an, China, 3–6 November 2010; pp. 193–202.
18. Nalmpantis, C.; Vrakas, D. Machine learning approaches for non-intrusive load monitoring: From qualitative to quantitative comparison. *Artif. Intell. Rev.* **2019**, *52*, 217–243. [[CrossRef](#)]
19. Zhao, J.; Jung, T.; Wang, Y.; Li, X. Achieving differential privacy of data disclosure in the smar. In Proceedings of the IEEE INFOCOM 2014—IEEE Conference on Computer Communications 2014, Toronto, ON, Canada, 27 April–2 May 2014; pp. 504–512.
20. Athanasiadis, C.; Dimitrios, D.; Theofilos, P.; Antonios, C. A Scalable Real-Time Non-Intrusive Load Monitoring System for the Estimation of Household Appliance Power Consumption. *Energies* **2021**, *14*, 767. [[CrossRef](#)]
21. Khodayar, M.; Wang, J.; Wang, Z. Energy Disaggregation via Deep Temporal Dictionary Learning. *IEEE Trans. Neural. Netw. Learn Syst.* **2020**, *31*, 1696–1709. [[CrossRef](#)] [[PubMed](#)]
22. Lyu, H.; Strohmeier, C.; Menz, G. COVID-19 time-series prediction by joint dictionary learning and online NMF. *arXiv* **2020**, arXiv:2004.09112.
23. Kolter, J.; Johnson, M. *REDD: A Public Data Set for Energy Disaggregation Research*; Workshop on Data Mining Applications in Sustainability SIGKDD: San Diego, CA, USA, 2011.
24. Tibshirani, R. Regression Shrinkage and Selection Via the Lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **1996**, *58*, 267–288. [[CrossRef](#)]
25. Boyd, S. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004.
26. Julien, M.; Francis, B.; Jean, P.; Guillermo, S. Online dictionary learning for sparse coding. In Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009; pp. 689–696.
27. Batra, N.; Kelly, J.; Parson, O. NILMTK: An open source toolkit for non-intrusive load monitoring. In Proceedings of the 5th International Conference on Future Energy Systems, Cambridge, UK, 11–13 June 2014; pp. 265–276.
28. Zhang, C.; Zhong, M.; Wang, Z.; Goddard, N.; Sutton, C. Sequence-to-Point Learning with Neural Networks for Non-Intrusive Load Monitoring. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32, pp. 2604–2611.