energies

MDPI

Article

A Knowledge Graph-Based Framework for Smart Home Device Action Recommendation and Demand Response †

Wenzhi Chen [1,*], Hongjian Sun [1], Minglei You [2], Jing Jiang [3] and Marco Rivera [2]

[1] The Department of Engineering, Durham University, Durham DH1 3LE, UK; hongjian.sun@durham.ac.uk
[2] The Department of Electrical and Electronic Engineering, University of Nottingham, Nottingham NG8 1BB, UK; minglei.you@nottingham.ac.uk (M.Y.); marco.rivera@nottingham.ac.uk (M.R.)
[3] The Department of Mathematics, Physics and Electrical Engineering, Northumbria University, Newcastle upon Tyne NE1 8ST, UK; jing.jiang@northumbria.ac.uk
* Correspondence: 2017140863@bupt.cn; Tel.: +44-07594395100

† This paper is a revised and expanded version of a paper entitled Accurate Action Recommendations and Demand Response for Smart Homes via Knowledge Graphs, which was presented at ICIT 2024 (The 25th IEEE International Conference on Industrial Technology in Bristol, the UK) from 25th to 27th of March, 2024.

Abstract: Within smart homes, consumers could generate a vast amount of data that, if analyzed effectively, can improve the convenience of consumers and reduce energy consumption. In this paper, we propose to organize household appliance data into a knowledge graph by using the consumers' usage habits, the periods of usage, and the location information for graph modeling. A framework, 'DARK' (Device Action Recommendation with Knowledge graphs), is proposed that includes three parts for enabling demand response. Firstly, a household device action recommendation algorithm is proposed that improves the knowledge graph attention algorithm to make accurate household appliance recommendations. Secondly, graph interpretable characteristics are developed in the DARK using trained graph embeddings. Finally, with the recommendation expectation, the consumers' comfort level and appliances' average power load are modeled as a multi-objective optimization problem in the DARK to participate in demand response. The results demonstrate that the proposed system can generate appliances' action recommendations with an average of 93.4% accuracy and reduce power load by up to 20% while providing reasonable interpretations for the device action recommendation results on the customized UK-DALE dataset.

Keywords: knowledge graph; smart home; demand response; recommendation system

Academic Editor: Marcin Kaminski

Received: 18 December 2024
Revised: 6 February 2025
Accepted: 8 February 2025
Published: 11 February 2025

Citation: Chen, W.; Sun, H.; You, M.; Jiang, J.; Rivera, M. A Knowledge Graph-Based Framework for Smart Home Device Action Recommendation and Demand Response. *Energies* **2025**, *18*, 833. https://doi.org/10.3390/en18040833

1. Introduction

With the advent of smart appliances, there has been a growing interest in replacing conventional automation with intelligent action recommendations for smart homes [1]. Accurate action recommendations can enhance the convenience of daily life, such as automatically turning on the dining area lights during meals, closing curtains at night, or turning on the TV/music based on residential habits, especially for the elderly or patients with mobility difficulties [2]. Since action recommendations are closely related to human activities, the interpretability of the algorithm is important [3]. However, to our knowledge, no paper currently focuses on interpretable models particularly designed for household appliance action recommendations, which is the aim of this paper.

While making recommendations, with the participation of a flexible load, the household appliances' demand response can improve power grid efficiency [4]. Many household appliances, such as fans, washing machines, and ovens, can be considered flexible loads with different power levels, and less energy consumption may lead to discomfort [5]. In this paper, the demand response means that residents (exchangeable with consumers) can balance comfort and energy consumption by setting the appropriate power level for each appliance, thus improving energy efficiency. Optimizing action recommendations to achieve demand response is another aim of this paper.

Research on household appliance action recommendations is limited and still in the early stages as of September 2024, as summarized in the literature review below.

In 2014, Rasch, Katharina et al. proposed a smart home recommendation system that continuously interpreted the user's current situation and recommended services that aligned with the user's habits. However, due to the limitations of the algorithms at that time, the accuracy was only about 60% [6]. In 2016, Chen et al. proposed a hybrid recommender system based on the Kalman Filter to predict the actions that users wanted to perform next in a smart home environment [7]. However, the recommendation accuracy was a major concern. In 2016, Belghini, Naouar et al. proposed a smart home recommendation system that offered personalized services using contextual information and physical sensor data; however, this paper only focused on the theoretical framework [8]. In 2021, Reyes-Campos et al. proposed a method for discovering resident behavior patterns using machine learning techniques and the Internet of Things [9]. In 2022, Jeon et al. proposed a precise action recommendation method for smart homes [10], which summarized the device control and temporal context of each action through a self-attention mechanism and extracted patterns related to the query from the sequence using an attention mechanism. Although the recommendation accuracy was good, both [9,10] did not consider energy optimization as well as the interpretability of their algorithms. In 2022, Varlamis et al. proposed a recommendation system that integrated sensor data, user habits, and user feedback to provide timely, personalized energy-saving suggestions [11]. This paper applied only to specific scenarios, such as predicting unnecessary lighting and air conditioning use or whether a room was occupied, which limited its applicability. In 2023, Yao et al. developed a recommendation system using GraphSAGE [12]. This system created a unique graph for each user based on the specific rules they employed in their smart devices. The system employed a federated training algorithm to ensure user data privacy. Both our approach and [12] utilized graph-based recommendation algorithms. The difference was that our recommended actions were triggered by electricity usage, leading to demand response-driven optimization. Additionally, our algorithm was based on a knowledge graph incorporating information from the edges, but this was not considered in [12]. In 2023, Ali et al. presented a method for an adaptive smart home system aimed at developing personalized automation systems that provide smart home services to users [13]. Similarly, this work did not consider the interpretability of the algorithm and further energy optimization. In 2024, Tahar et al. proposed a dynamic, context-aware recommender system for smart homes [14] but did not consider the interpretability of the recommendation system.

While the aforementioned recent studies focused on energy optimization to enhance the accuracy of recommendations, none have simultaneously considered both energy optimization and model interpretability during the recommendation process. Our research seeks to address this research gap, primarily focusing on the following aspects:

1. How to incorporate information to accurately recommend the next action of household appliances?
2. How to generate the interpretation for the recommended results of household appliance actions?
3. How to optimize energy consumption and comfort levels by setting the appropriate power for each household appliance?

In response to these questions, this paper establishes a framework 'DARK', representing Device Action Recommendations with Knowledge graph that accurately recommends and optimizes the appliances' next action with interpretive methods considering demand response optimization. Please note that 'DARK' is an acronym derived from the initials of 'Device', 'Action', 'Recommendations', and 'Knowledge', and it does not have any actual academic meaning. The main contributions of this paper are as follows:

1. We propose a modified KGAT algorithm by enhancing its sampling and aggregation. This algorithm predicts the next likely actions of appliances using a knowledge graph. We conducted comparative experiments with the traditional KGAT, DNN, CNN, and RNN algorithms, observing that our method demonstrates superior performance.
2. An interpretation method for the recommended results is proposed. This method utilizes embeddings for reasoning and analyzes the rationale behind each recommendation, enhancing the trustworthiness of our recommendation system.
3. Demand response optimization is carried out based on the expected recommended actions, taking into account the energy consumption and comfort level of consumers. This approach improves energy efficiency by effectively balancing energy consumption with comfort.

The remaining structure of this paper is as follows: Section 2 presents the system architecture, Section 3 introduces the proposed algorithm and method, Section 4 presents the simulation and results of the system, and Section 5 provides the conclusion.

2. System Composition

The overall architecture of DARK is built upon a household appliance system, as represented in Figure 1. The system comprises appliances, sensors, a small-scale database, and a home energy management system. Each appliance possesses the following attributes:

1. Rated power: Each appliance has its rated power. For example, a toaster has a rated power of 1500 W.
2. Appliance location: Appliances are distributed across different rooms, including the kitchen, living room, office, bedroom, utility room, children's room, and undefined locations.
3. Comfort impact: Each appliance has a different impact on comfort.
4. Habitual usage time: Each appliance has specific habitual usage times. For example, gas stoves are mostly used during mealtime, while lights are frequently used at night.
5. Habitual usage sequence: Some appliances have a habitual usage sequence. For instance, the most commonly used appliance following a hair dryer might be a hair straightener.

All the appliances are connected to the home's local area network, and sensors monitor them to collect energy consumption data. The data are sent to the local database for processing and transformed into embeddings. The data in the database are then used by the home energy management system to create recommendations. The recommended results are sent to the interpretation module to generate interpretations and to the optimization module to generate optimization results, considering both comfort level and energy consumption, assisting consumers in decision-making.

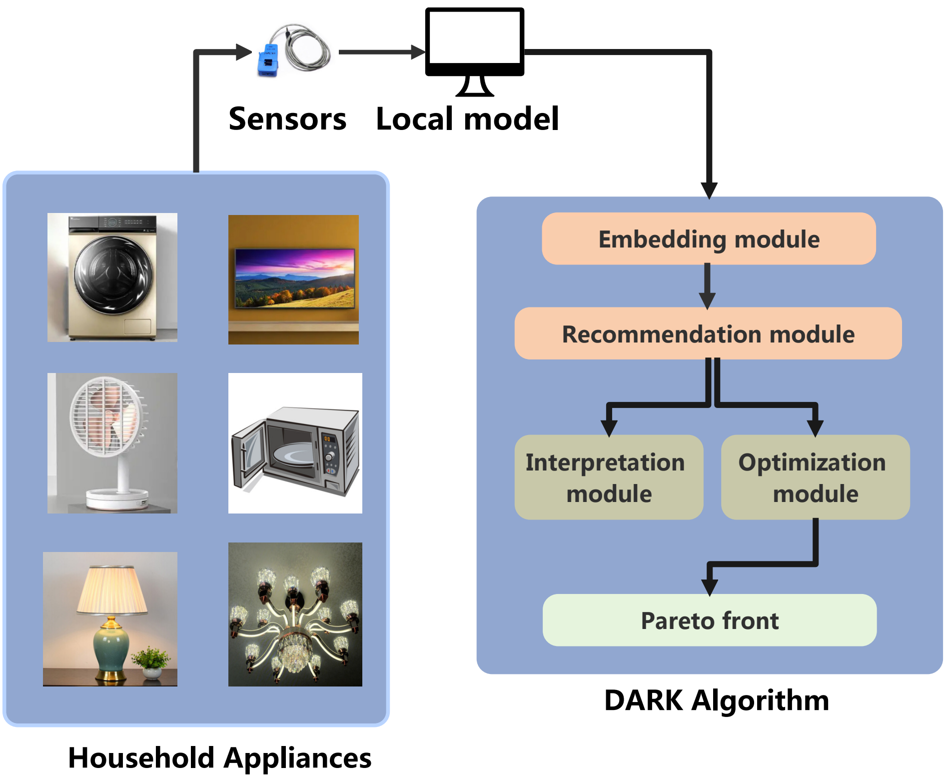

Figure 1. The proposed system architecture of DARK has different modules. The energy consumption information is collected by sensors and used in the local model. Then, the data are sent to different modules, including the embedding, recommendation, interpretation, and optimization modules.

Privacy protection is also essential. User data is considered private and should not be disclosed to third parties. To ensure user privacy, the database data is not shared externally. Instead, federated learning can be employed, only uploading the local model to the server. Federated learning ensures that the data remains only locally stored [15]; however, this is not the focus of this paper. This paper considers only standalone action recommendation systems without sharing any data, so there are no privacy leakage concerns.

3. Proposed Algorithm

This section introduces the proposed methods for graph building and the algorithms used in DARK's embedding, recommendation, interpretation, and optimization module.

3.1. Method for Building Knowledge Graph

It is assumed that each appliance has an associated sub-meter, so individual electricity usage data for each appliance can be obtained. If this is not achievable and only a main smart meter exists, non-intrusive load monitoring can be used for power disaggregation to obtain individual appliance electricity data. Then, the next step would be data processing for DARK, involving the following sequence of steps: format conversion, noise filtering, device selection, graph construction from conventional datasets, incorporating other attributes, and finally, negative sampling.

3.1.1. Format Conversion and Noise Filtering

First, the format should be transformed. The system needs to recognize and transform the timestamps into the year-month-day format and aggregate the data into certain intervals. The data may contain a large amount of noise, so filters are implemented to determine the on/off states of appliances as well as filter out momentarily switched-on appliances and electricity leakage. Due to the unique characteristics of each type of electrical appliance, the filtering thresholds for each type of appliance need to be customized according to the rated power. The thresholds (rated power) are shown in Table 1. The column 'Comfort Level' will be introduced later in the Section 4.4.

Table 1. Rated power and comfort level of home appliances.

Device	Rated Power (Watt)	Comfort Level c_i
Washing machine	1700	0.0001
Dishwasher	2350	0.0001
TV	130	0.0003
Kitchen lights	150	0.0003
HTPC	70	0.0003
Kettle	2400	0.0005
Toaster	1580	0.0005
Microwave	1510	0.0005
LCD office monitor	50	0.0003
Hi-Fi office stereo	15	0.0003
Breadmaker	580	0.0005
Living room amp	45	0.0003
Living room floor lamp	1100	0.0003
Hoover (vacuum cleaner)	2000	0.0001
Kitchen desktop lamp	40	0.0003
Bedroom desk lamp	80	0.0003
Living room side lamp	20	0.0003
Living room subwoofer	50	0.0003
Living room TV cabinet lamp	25	0.0003
Kitchen table lamp	20	0.0003
Kitchen phone stereo	20	0.0003
Utility room lamp	45	0.0003
Bedroom table lamp	60	0.0003
Coffee machine	1270	0.0003
Bedroom chargers	30	0.0003
Hair dryer	1680	0.0003
Straighteners	500	0.0003
Iron	1800	0.0003
Gas oven	60	0.0005
Child's table lamp	15	0.0003
Child's desk lamp	50	0.0003
Office desk lamp 1	30	0.0003
Office desk lamp 2	25	0.0003
Office desk lamp 3	20	0.0003
Office PC	240	0.0005
Office fan	50	0.0003
LED printer	900	0.0005

3.1.2. Domestic Devices Selection

Some appliances are not suitable for inclusion in the recommendation system. Figure 2 displays the typical power consumption of the fridge and washing machine for a day (24 h) [16]. The fridge exhibits a characteristic of prolonged operation, making it less suitable for recommendation system data. Please note that even after filtering, the refrigerator still exhibits significant periodic high peaks. These are not detailed in the UK-DALE dataset, but they can be assumed from the opening and closing of the refrigerator or the refrigerator's periodic defrost function. On the other hand, the washing machine shows distinct triggering patterns, making it a suitable candidate for inclusion in the recommendation system data. So appliances with long-term operation, such as refrigerators, routers, iPad chargers, etc., are excluded. Additionally, some appliances that are not under human control also need to be excluded. For example, the function of boilers is based on a preset automatic control system.

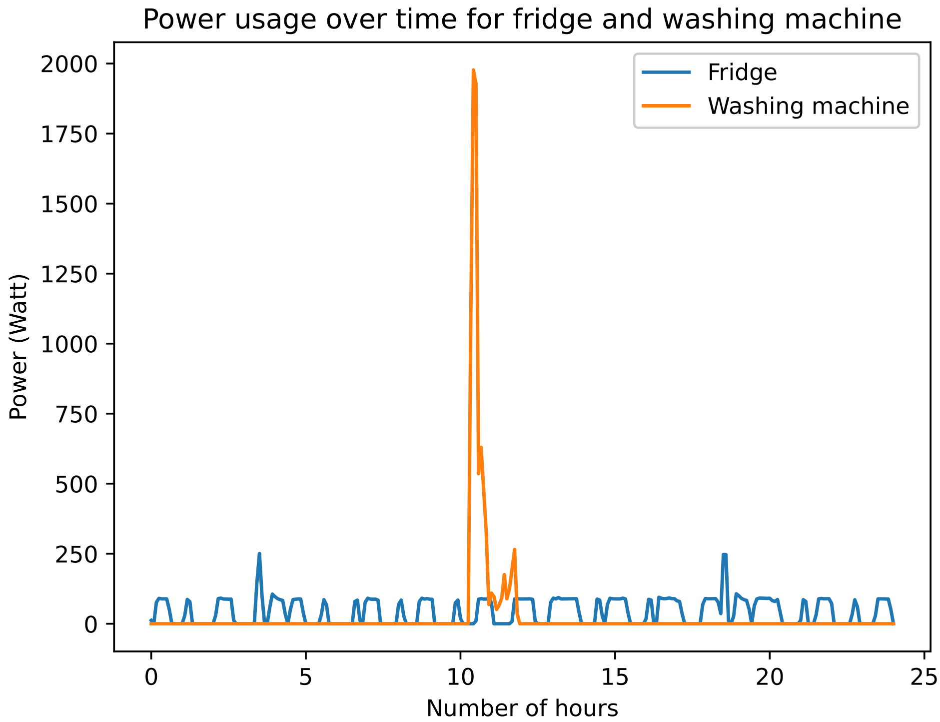

Figure 2. Power consumption for fridge and washing machine over time.

3.1.3. Graph Construction

Each appliance is treated as a node and adds an edge between two appliances if the appliances are used one after another, labeled as "used before/after". A knowledge graph is also established for the time zones where appliances are used, connecting time nodes with appliance nodes. The relationship can be represented as "used during the time period". Finally, a knowledge graph for the location of appliances can be established, connecting location nodes with appliance nodes and labeling the relationship as "located in the room". Other attributes, such as the rated power and comfort level, are added for each appliance. This information helps the graph to find the hidden relationships in recommendations and carry out demand response, as they connect the usage information with the energy consumption through a graph.

3.1.4. Negative Sampling

Negative sampling is performed on the dataset to accelerate convergence. For example, a gas oven is often used from 18:00 to 24:00, recorded as 'gas oven-used during the time-18:00'. This time can be replaced by 2:00 to 4:00 in the morning for producing negative examples, which have never happened, recorded as 'gas oven-never used during the time-3:00'. This method creates negative examples at a 1:1 ratio in numbers with positive examples. The reason for choosing a 1:1 ratio is that the model does not overly favor either positive or negative samples, which improves training stability and prevents overfitting to one side. This is also an empirical choice commonly adopted in the KGAT algorithm [17].

Compared to conventional data, by modeling the relationships between nodes in graph data and visualizing the connections between household data nodes and edges, one can intuitively understand how the proposed KGAT model performs reasoning and makes recommendations. This also imparts a structural nature to the model's decisions, beyond merely relying on the statistical properties of the data, which facilitates data mining [18]. Figure 3 illustrates the constructed knowledge graph, which is a part of the overall knowledge graph. It depicts a subgraph containing a kitchen light, microwave, and kettle, along with each appliance's attributes. For instance, the most commonly used time for the microwave is 6:00 to 12:00 on Tuesday. Similarly, the rated power of the kettle is 2000 Watts. There are also associations between appliances. For example, kitchen lights, the kettle, and the microwave are all in the kitchen, so they all point to the 'kitchen' point. Actions commonly taken after turning on the kitchen light include using the kettle or microwave, thus establishing relationships (edges on the graph) between appliances.

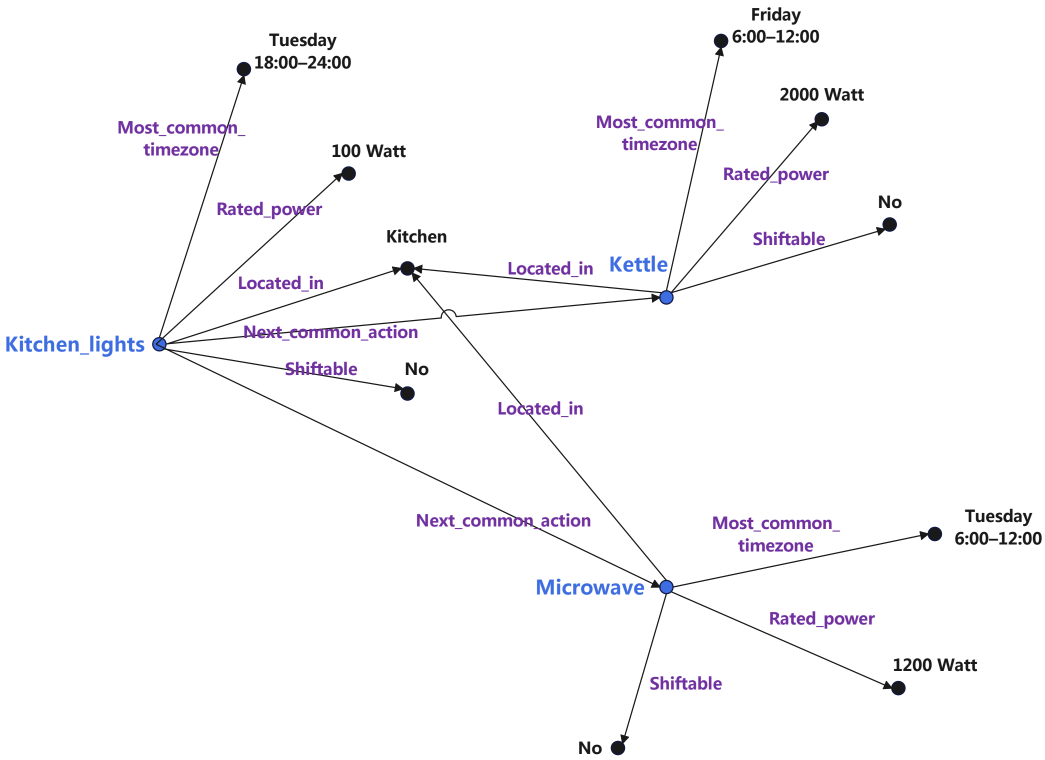

Figure 3. Constructed domestic appliances knowledge graph. Each household appliance has attributes such as shift ability, rated power, usage time, and appliance location. If there is a sequential order of usage between two appliances, a line is established between them.

3.2. Embedding Module

During the data processing, it is necessary to convert the data into embeddings, which use high-dimensional vectors to represent the discrete data. The transformation from data to embeddings facilitates the use of knowledge graph algorithms in subsequent steps. During backpropagation, its parameters are updated based on the gradient of the loss function to minimize the difference between predicted results and true labels. The following data were encoded to embeddings:

1. Appliances: Each appliance is assigned an embedding, resulting in a set of embeddings for different appliances.
2. Time zones: The daily time is divided into four time periods: midnight, morning, afternoon, and evening. Thus, a week can be divided into $4 \times 7 = 28$ time zones. These 28 time zones are encoded as a set of embeddings.
3. Locations: Appliances are distributed across different rooms, and an embedding is created for each room.

Please note that the reason for dividing daytime into four zones is that recommendation systems often have time dependencies. This allows time to be segmented into natural periods, such as late night for rest, morning for waking up and starting work, afternoon for work and study, and evening for entertainment and relaxation. Daily habits vary, as people typically engage in different activities on Mondays, Fridays, and weekends. Therefore, time was divided into $4 \times 7 = 28$ time zones. Compared to hourly segmentation, this approach reduces the computational burden on the recommendation system and lowers complexity. Additionally, merging data into broader time periods facilitates the discovery of potential features.

The embedding layer aims to train the embedding of (h, r, t) such that the embedding of the head e_h plus the embedding of relation e_r approaches the embedding of tail e_t for any positive (h, r, t) in distance (for example, L2 distance), which can be represented as

$$g_{(h,r,t)} = ||W_r e_h + e_r - W_r e_t||_2^2 , \tag{1}$$

where $|| \cdot ||_2^2$ means the squared results of the L2 distance. $g_{(h,r,t)}$ represents the positive triples, and the value of $g_{(h,r,t)}$ should be close to zero. However, for negative triples, the h, r or t can be replaced by other values that never happened. For example, $g_{(h,r,u)}$ is generated by replacing t with u arbitrarily, and the value of $g_{(h,r,u)}$ should be infinity.

The embedding module converts discrete inputs into continuous embeddings. These embeddings are then fed into the knowledge graph layer as input for training.

3.3. Recommendation Module

After encoding the original data into embeddings, the embeddings can be further used for recommendations. The recommendation module introduces the proposed DARK algorithm, an improved KGAT algorithm in the recommendation area. It also includes a comparison with the traditional KGAT algorithm.

3.3.1. Knowledge Graph Attention Networks

In KGAT, attention mechanisms are utilized to determine the aggregation weights of neighboring node embeddings in [17]

$$a_{(h,r,t)} = \mathrm{Softmax}((W_r e_t)^{\mathrm{T}} \tanh(W_r e_h + e_r)) , \tag{2}$$

where the $a_{(h,r,t)}$ is the calculated attention for knowledge graph aggregation. $\mathrm{Softmax}(\cdot)$ and $\tanh(\cdot)$ (hyperbolic tangent function) are activation functions. W_r is the relation transformation matrix. e_h, e_r, and e_t are the head, relation, and tail entity, respectively. Then, it aggregates the embeddings of neighboring nodes based on the weights in [17]

$$e_{Nh} = \sum_{(h,r,t) \in Nh} a_{(h,r,t)} \times e_t , \tag{3}$$

where e_{Nh} is the aggregated entity, which means how much information the tail entity e_t is going to pass to the head entity e_h, which is represented as [17]

$$agg = \mathrm{LeakyRelu}(W_{\mathrm{trans}}(e_h + e_{Nh})) , \tag{4}$$

where agg is the information aggregation vector, which contains weighted information with the head and tail entities and can then be used for recommendations by embedding the dot product with the aimed user embeddings. LeakyRelu is the activation function. W_{trans} is the transformation matrix [17].

3.3.2. Modified KGAT Algorithm

Different from the traditional KGAT algorithm, some improvements have been made to the action recommendation scenario. The pseudocode is shown as Algorithm 1. Based on the original KGAT algorithm, the proposed algorithm improves (1) the embedding aggregation process and (2) the embedding sampling method. These enhancements help improve recommendation performance. The detailed description is as follows:

Algorithm 1: The modified KGAT networks in the recommendation module of DARK

1 Load knowledge graph sequence embeddings $e^{se}_{(h,r,t,\text{time})}$, time zone embeddings $e^{ti}_{(h,r,t)}$, and location embeddings $e^{lo}_{(h,r,t)}$ from processed datasets.
2 Initialize the list L.
3 **for** *episode* $= 1, M$ **do**
4 **for** e^{se}_h, e^{se}_t *in* $e^{se}_{(h,r,t,time)}$ **do**
5 Sampling dataset s_h for e^{se}_h and s_t for e^{se}_t according to the distribution of $e^{se}_{(h,r,t)}$, $e^{ti}_{(h,r,t)}$ and $e^{lo}_{(h,r,t)}$.
6 Constructing adjacency matrics Adj_h with s_h and Adj_t with s_t.
7 **for** *each* e_h, e_r *and* e_t *in* Adj_h **do**
8 $a_{h\text{-}(h,r,t)} =$
9 $\text{Softmax}((W_r e_t)^{\text{T}} \tanh(W_r e_h + e_r))$
10 $e_{h\text{-}Nh} = \sum_{(h,r,t)\in Nh} a_{h\text{-}(h,r,t)} \times e_t$
11 $agg_h = \text{LeakyRelu}(W(e_h + e_{h\text{-}Nh}))$
12 **end for**
13 **for** *each* e_h, e_r *and* e_t *in* Adj_t **do**
14 $a_{t\text{-}(h,r,t)} =$
15 $\text{Softmax}((W_r e_t)^{\text{T}} \tanh(W_r e_h + e_r))$
16 $e_{t\text{-}Nh} = \sum_{(h,r,t)\in Nh} a_{t\text{-}(h,r,t)} \times e_t$
17 $agg_t = \text{LeakyRelu}(W(e_h + e_{t\text{-}Nh}))$
18 **end for**
19 $V = \text{concatenate}(agg_h, agg_t, e^{se}_{\text{time}})$
20 Store $O = \text{Relu}(\text{FullyConnectedLayer}(V))$ in L
21 **end for**
22 **end for**
23 Output: L

Firstly, the algorithm aims to predict the next potential appliance to be used based on known residents' appliance usage habits. In the traditional KGAT recommendation algorithm, only potentially recommended appliances undergo graph embedding aggregation because traditional KGAT treats the recommended appliance as an item and the known appliance as a user. However, in the household device action system, the known appliance usage can also be treated as an item and aggregated. This is expressed in lines 13–18 of Algorithm 1, where the subscript $t\text{-}Nh$ and $h\text{-}Nh$ means that $e_{h\text{-}Nh}$ is generated by t or h, similar to the subscripts $h\text{-}(h,r,t)$ and $t\text{-}(h,r,t)$.

In addition, each type of attribute, such as timezone, usage sequence, and location, is first collected, ensuring equal chances of different kinds of data. Then, a distribution-based sampling is used in line 5. These samples will inform the knowledge graph algorithms about the studied case and affect the aggregation results. For example, if a particular appliance is most commonly used on Tuesday mornings, the probability of sampling Tuesday mornings for the adjacency matrix will be higher. This is expressed in lines 5–6.

The output L contains aggregated embeddings agg_h for head embeddings and agg_t for tail embeddings. Then, the embedding L is used as the input for the fully connected layers for training.

3.4. Interpretation Module

After making recommendations in the recommendation module, the results can be sent to the interpretation module for further reasoning.

In recommendation algorithms, interpretability is crucial for making the recommendation results convincing. The purpose of this module is to provide further interpretations of the recommended results. It is known that the trained embeddings in graph neural networks can represent the meaning of the data [19], and in paper [20], Ai et al. used that for making interpretations. Based on this characteristic, we generate reasons for the recommended results, which consist of three types of reasons: (1) Habitual usage sequence, which means an appliance is often used after another one. (2) Habitual usage time, which means an appliance is often used at some specific time. (3) Usage at a habitual location, which means an appliance is used because it has location relationships with other appliances.

All the connections between the recommended appliances and the known appliances within two hops are identified in the graph, and we find the embeddings of all the nodes and relationships involved. Using the embedding approach described in the paper [19], we have

$$e_t \approx e_h + e_r \,, \tag{5}$$

where the head embedding e_h plus relation embedding e_r should approach the tail embedding e_t as much as possible after training. Ideally, they are equal.

Here we propose a new measure as shown in (6)–(11) for the recommendation results in the context of household appliance action recommendations. From (5), the embedding of the reasons can be calculated in (6)–(8).

$$e^{se}_{\text{total}} = (e^{se}_h + e^{se}_r)e^{se}_t \,, \tag{6}$$

$$e^{ti}_{\text{total}} = \frac{1}{N}\left(\sum_{j=1}^{N}(e^{ti}_{h,j} + e^{ti}_{r,j})e^{ti}_{t,j}\right) , \tag{7}$$

$$e^{lo}_{\text{total}} = (e^{lo}_h + e^{lo}_r)e^{lo}_t \,, \tag{8}$$

$$R(se) = \frac{e^{se}_{\text{total}}}{e^{se}_{\text{total}} + e^{ti}_{\text{total}} + e^{lo}_{\text{total}}} \,, \tag{9}$$

$$R(ti) = \frac{e^{ti}_{\text{total}}}{e^{se}_{\text{total}} + e^{ti}_{\text{total}} + e^{lo}_{\text{total}}} \,, \tag{10}$$

$$R(lo) = \frac{e^{lo}_{\text{total}}}{e^{se}_{\text{total}} + e^{ti}_{\text{total}} + e^{lo}_{\text{total}}} \,, \tag{11}$$

where the superscript se, ti, and lo refer to different reasons like sequence, timezone, and location. The values e^{se}_{total}, e^{ti}_{total}, and e^{lo}_{total} are the product values of the embeddings. From the characteristic embedding of TransE [20], a higher product value means that the reason has a stronger relationship with the answer, where e^{ti}_{total} is the top N average product of the embedding of the reason timezone. e^{se}_{total} and e^{lo}_{total} only exist if two household appliances have a sequential relationship or are placed in the same location. In (9)–(11), the explanation R of different reasons is calculated.

Please note that a major drawback of interpretation is that, generally, the correctness of the interpretation cannot be verified; thus, it is usually used as a reference rather than a precise interpretation. In this paper, the verification method only checks whether the interpretations align with common sense. If this is true, the interpretations are considered to be correct. Since interpretations themselves are estimations of the user's intentions, surveys can be created in the future to further verify the accuracy of these interpretations.

3.5. Optimization Module

Based on the results of the recommendation module, multi-objective optimization can be performed for demand response. This section primarily introduces the objective functions in the optimization module of DARK for optimizing energy consumption and the user's expected satisfaction.

When solving complex problems with multiple objectives and a large number of parameters, the Genetic Algorithm is a promising method, especially when these objectives involve a non-linear problem [21].

It is assumed that the power of each appliance is adjustable within a certain range, and the power of an appliance affects the comfort level as given by

$$f^{\text{comfort}} = -f^{\text{discomfort}} = -\sum_{i=1}^{N} c_i (P_i - P_i^{\text{real}})^2 \,, \tag{12}$$

where f^{comfort} is a comfort function, and $f^{\text{discomfort}}$ represents the sum of discomfort scores for N appliances. The rated power of appliance i is P_i^{real} with $i = 1, \ldots, N$. Power significantly above or below the rated power can result in discomfort. The average power load of appliances is calculated by

$$f_{\text{avgpower}} = \frac{1}{N} \sum_{i=1}^{N} P_i^{\text{real}}. \tag{13}$$

For (12) and (13), we address two competing objectives: maximizing comfort, which represents consumer needs, and minimizing average power load, a consideration crucial for supporting power grid operations. For this type of optimization problem, multi-objective optimization algorithms, such as genetic algorithms, can be used to find the Pareto front. The outcome of the optimization is to set the power level P_i for appliance i based on the expectation of the recommendation results. Through the optimization results, multiple feasible solutions between comfort level and average power load can be selected in the Pareto front [22].

4. Simulations Setup

This section introduces the simulation environment, dataset preprocessing, and the simulation results, including accuracy and interpretations. Finally, the results of the proposed demand response algorithm are introduced.

4.1. Dataset and Software

UK Domestic Appliance-Level Electricity is a household electricity dataset used for non-intrusive load recognition research [16]. The UK-DALE dataset provides high-precision power consumption data, including 54 household appliances such as refrigerators, washing machines, ovens, lighting, etc. It is based on electricity-triggered information. The detailed sampling frequency, reaching as high as 6 Hz, makes constructing the graph's structure possible and suitable. We used the dataset and transformed it into graph-structured data. Please note that this paper only considers electricity consumption and no other kinds of consumption, such as geothermal or gas.

For software, the PyTorch (https://pytorch.org/ accessed on 5 February 2025) framework was used for deep learning. The parameters are listed in Table 2.

Table 2. Parameters for the recommendation system.

Parameter in Recommendation System	Values
Device/Relation/Time embedding dimension	128
Embedding input nodes	3
KGAT input nodes	384
(Comparison) RNN/CNN/DNN input nodes	384
DNN hidden nodes	384 × 128
DNN output nodes	1
Learning rates	0.001
Training iterations in a round	50

4.2. Results for Recommendations

Our modified KGAT algorithm of DARK is compared with other algorithms, including the original KGAT, DNN, CNN, and RNN [23]. The recommended results are defined as the top three most likely appliances generated by the recommendation algorithm, and a recommendation score greater than 0.95 is also added to the recommendation list. Precision is determined by comparing it with the labels.

Figure 4 shows a selected instance where all other conditions were the same, representing typical algorithm performance. It can be observed that DNN and CNN yielded notably inferior results. However, they do show training effectiveness. Assuming the selection of 3 devices at random from a set of 37 devices, the probability of including a specific number is $\frac{C(36,2)}{C(37,3)}$, which is around 8.1%, and C means combination calculation in mathematics. The recommendation accuracy of DNN and CNN reaches approximately 30%. The performance of the RNN algorithm steadily improved over time, but the KGAT algorithm (averaged accuracy: 87.3%) reached the highest precision comparable to RNN in the first round. The improved KGAT achieved the highest accuracy (averaged accuracy: 93.4%) and provided the most accurate recommendations, which averaged 6.1% (accuracy improved) compared with the traditional KGAT algorithm, though the gap is not so apparent in the Figure 4. The related data are summarized in Table 3.

Table 3. Comparison of recommendation accuracy of different algorithms.

Algorithm	Highest Accuracy	Lowest Accuracy	Average Accuracy
KGAT	90.4%	83.3%	87.3%
DNN	32.9%	29.8%	31.1%
CNN	31.3%	26.5%	28.9%
RNN	88.9%	31.6%	77.6%
Proposed KGAT solution	94.7%	91.4%	93.4%

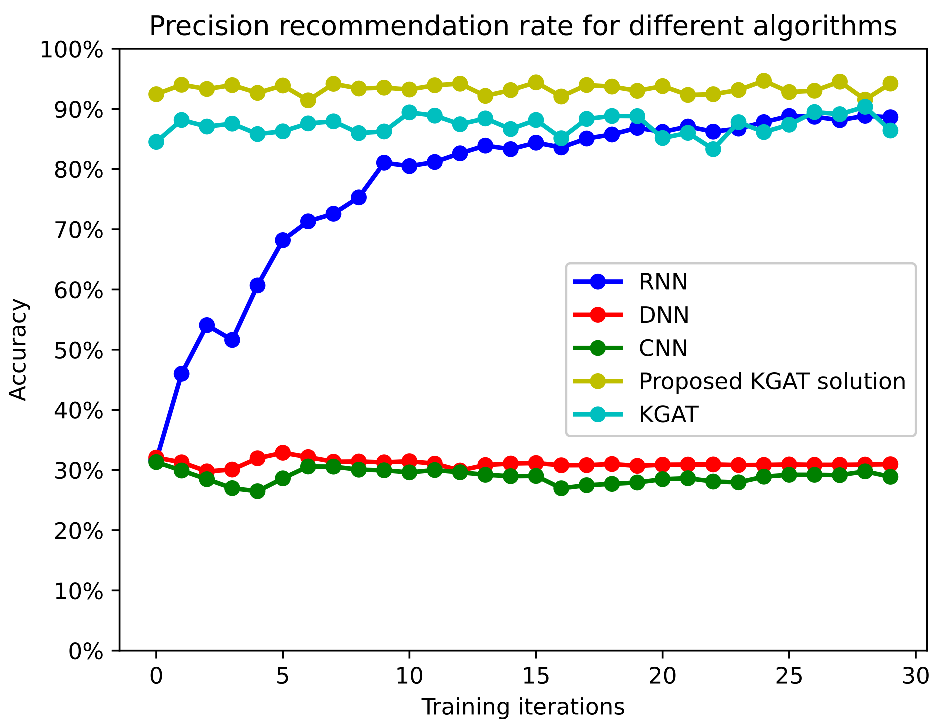

Figure 4. Recommendation accuracy vs training iterations for different algorithms.

4.3. Results for Interpretations

4.3.1. Reason-Timezone

Figure 5 represents the detailed next action recommendation after using an oven, and Figure 6 illustrates the recommended results for different appliances. On the left are the target appliances for recommendations, and on the right are the appliances most likely to be used next after using the target appliance in all instances. The results are sorted by the degree of recommendation from 0 to 1.

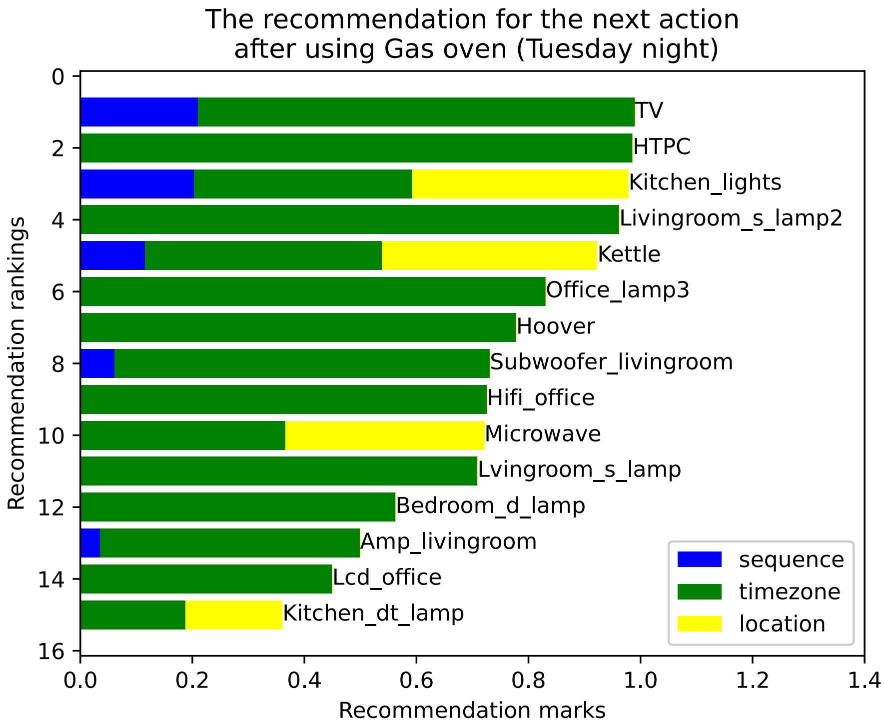

Figure 5. Next action's recommendation rankings and marks of various appliances after using an oven.

This means these two appliances are frequently used at the same time, as shown in Figure 5. According to the residents' habits on Tuesday nights, appliances such as lights, office lamps, and appliances for entertainment purposes, such as television, Hifi, and HPTC, are the most commonly used in our dataset. A similar result in Figure 6 is 'Utilityrm_lamp', and 'WashingMachine' are recommended because they have always been used at the same time.

4.3.2. Reason-Sequence

This reason means that 2 appliances are always used together. The recommended score on the horizontal axis is the inner product of embeddings from the recommendation algorithm. Appliances like 'hair_dryer' and 'straightener' are often used with each other because of the use sequence. Other similar results are 'Kitchen_phone_stereo' and 'Kitchen_lamp2', as well as 'Microwave' and 'Kitchen_dt_lamp'. The probability analysis is shown in Figure 6. This interpretation aligns with common sense, as 'hair_dryer' and 'straightener' are often used together after taking a shower. The use of 'Kitchen_phone_stereo' and 'Microwave' often requires lamps.

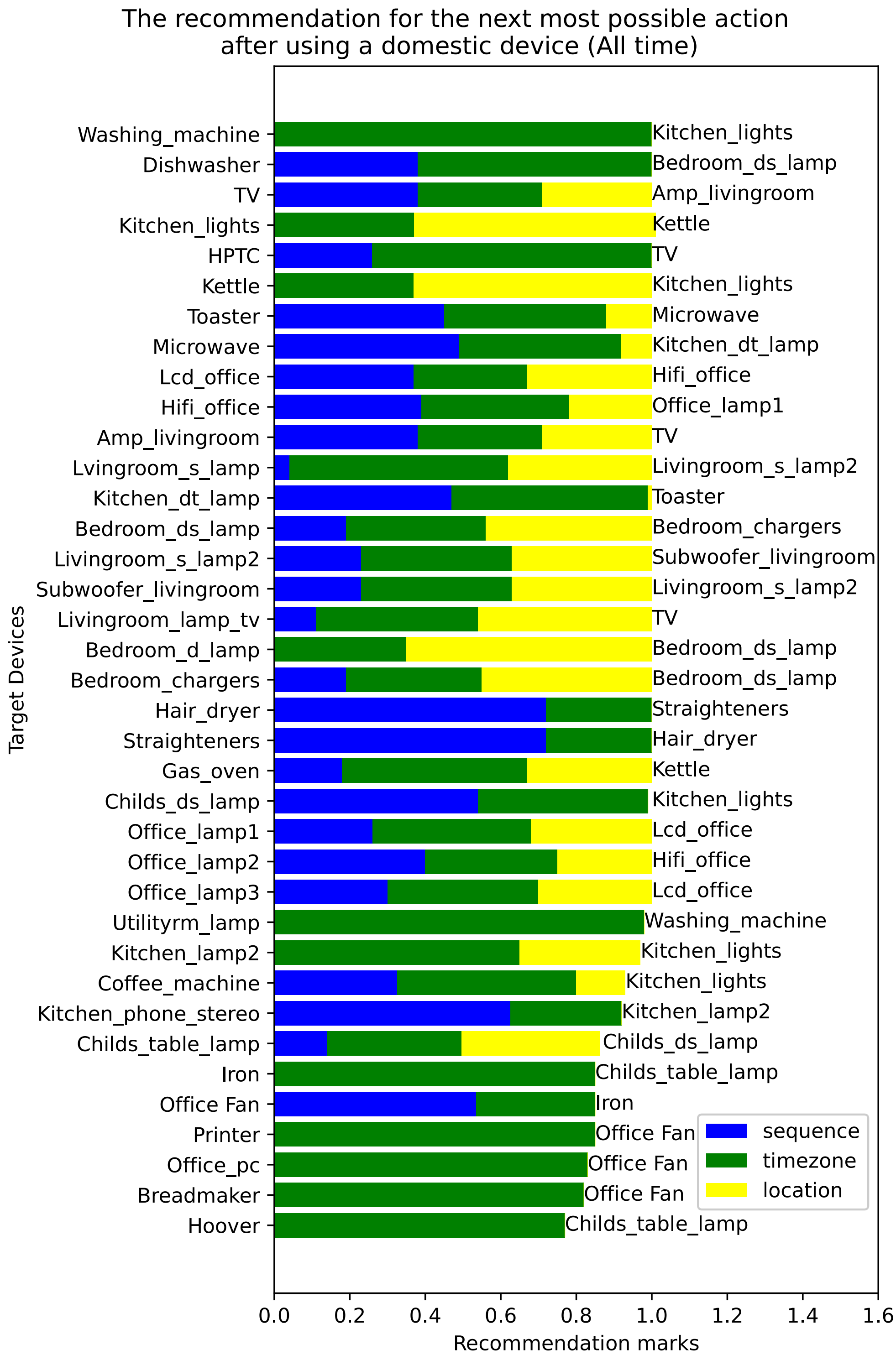

Figure 6. The most possible recommendation result for each domestic appliance; the left side represents appliances that have already been used, and the right side shows the appliances that are most likely to be used next.

4.3.3. Reason-Location

This reason means both appliances are in the same location. In Figure 5, when using the gas oven on Tuesday afternoon, the recommended appliance list includes the kettle, microwave, and other kitchen lights. This also aligns with common sense, as using appliances for cooking in the evening often involves turning on lights or using a kettle to boil water in the same location. In Figure 6, the recommendation results can be categorized into the following groups.

- Living Room Series: Such as 'TV', lamp for TV in the living room Livingroom_lamp_tv, amplifier in the living room 'Amp_livingroom', 'HTPC', and subwoofer 'Subwoofer_livingroom' are ranked as the top recommendations for each other, which align with daily usage logic, as using the TV often requires auxiliary devices like a lamp or speaker.
- Kitchen Series: Including lights 'Kitchen_lights' and 'Kitchen_dt_lamp', 'Kettle', 'Toaster', 'Gas_oven', 'Coffee_machine', and kitchen phone stereo 'Kitchen phone_stereo'. These appliances are recommended as the top choices for each other, which makes sense since they often collaborate. For example, cooking may involve turning on the lights.
- Bedroom Series Such as the lamp in the Bedroom 'Bedroom_d_lamp' and Bedroom's charger 'Bedroom_chargers' are recommended as top choices for each other, with the primary reason being location, aligning with typical usage patterns.
- Office Series: Such as three lamps in the office 'Office_lamp1', 'Office_lamp2' and 'Office_lamp3', LCD device 'Lcd_office', and Hifi device 'Hifi_office' are recommended as top choices for each other.

4.3.4. Example of Failed Recommendations

The last six items in Figure 6 are considered unsuccessful recommendation results due to a lack of data or unrepresentative data.

4.4. Results for Demand Response

It is assumed that the rated power of all appliances is adjustable within 0.8–1.2 times. In the simulation, assuming that there are three kinds of comfort levels in Table 1: The first kind is adjustable appliances, including washing machines and dishwashers, the parameter c_i in (12) is set as 0.0001, which means it will produce less uncomfortable if the power is adjusted. The second kind is mainly entertainment appliances, including TV, HTPC, etc. The parameter c_i is set as 0.0003. The third kind is essential household appliances, including gas ovens, office PCs, etc. The parameter c_i is set as 0.0005, which means it will cause discomfort if the demand response adjusts it. The value of c_i is manually set based on life experience. The values 0.0001, 0.0003 and 0.0005 are chosen because they can normalize the dissatisfaction score to a range of 0–400.

After optimization by the Genetic Algorithm with the objective dissatisfaction calculated in (12) and the averaged power calculated in (13), the Pareto front generated by the optimization algorithm converges. In genetic algorithms, the HV value (Hypervolume) represents the size of the hypervolume formed by the Pareto front and the reference point. A larger HV value indicates that the solution set covers a wider area in the objective space, implying better optimization performance. Figure 7 records the change in HV value over the optimization iterations, showing that the algorithm's performance gradually converges after 10,000 iterations. In addition, the genetic algorithm's convergence time was analyzed during the simulation process. Since the genetic algorithm optimizes based on the recommended list, the problem is fixed and relatively simple. The optimization times for 10 runs of the 20,000 generations genetic algorithm are as follows: 8.67, 8.63, 8.66, 8.66, 8.60, 8.57, 8.54, 8.57, 8.78, and 8.55 s. The average optimization time is only 8.62 s. The Pareto front is shown in Figure 8, where each point represents a set of device power settings.

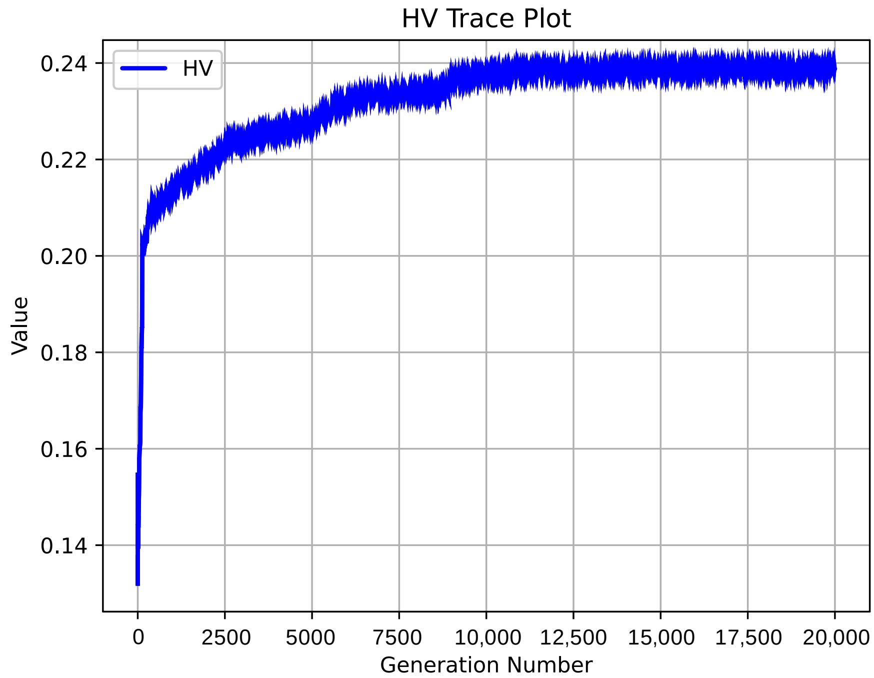

Figure 7. The hypervolume value of the genetic algorithm over 20,000 generations optimization.

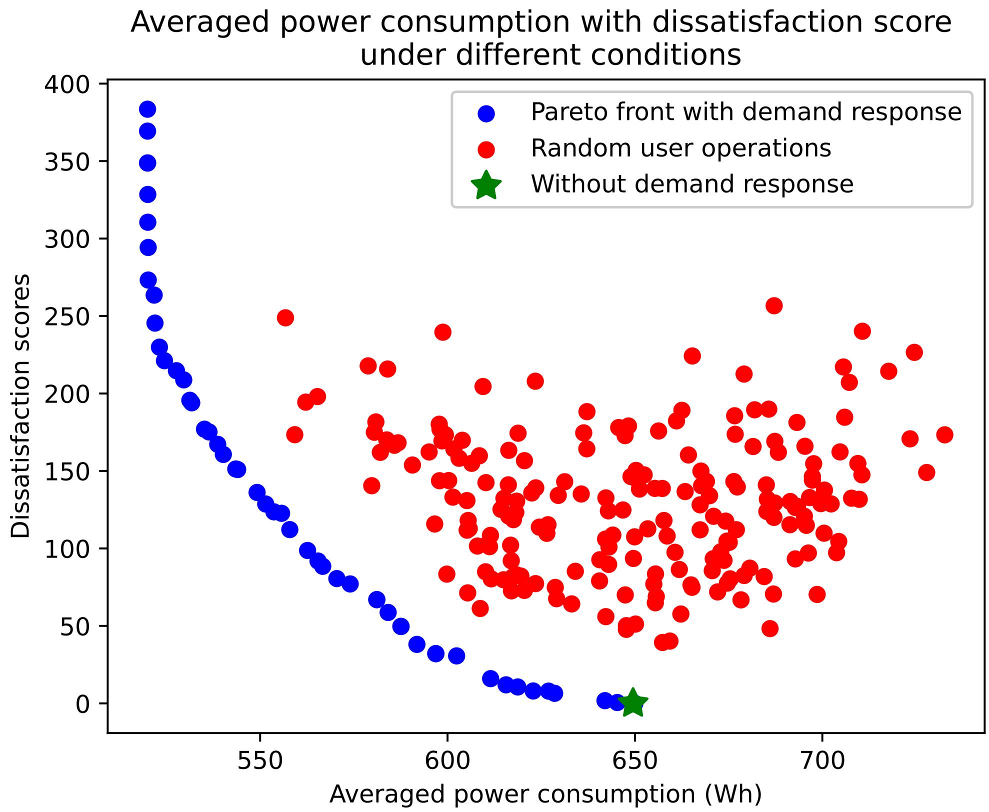

Figure 8. Pareto front for bi-objective optimization in demand response. The optimized Pareto points (marked in blue) significantly outperform the unoptimized points (marked in red) in terms of both users' expected satisfaction and average power consumption.

In Figure 8, the blue points represent the optimized Pareto fronts, which have the most efficient energy consumption and comfort balance. The red points represent the values obtained by simulating user operations, assuming that the power range of each electrical appliance is a random multiple of 0.8 to 1.2 times the rated power; calculate the energy consumption and discomfort level (average power between 560 and 740 watts). It can be seen that the optimized Pareto front blue points can achieve lower discomfort levels than the red points while saving energy consumption. The green star dots represent rated appliances' power (average power = 650 watts and discomfort level = 0). It can be seen that compared to the green dots. The optimized power load can be reduced by up to $(650 \text{ watt} - 520 \text{ watt})/650 \text{ watt} = 20\%$ with the proposed algorithm (The average power of the rightmost blue point is 520 watts).

According to the user's preference, the system can assign power settings to each appliance based on the Pareto fronts toward a higher comfort level or energy saving.

5. Conclusions

The proposed DARK framework integrates a knowledge graph-based recommendation system with interpretative analysis and multi-objective optimization for enabling demand response. Our results indicate that the modified KGAT algorithm effectively predicts the next actions of appliances and provides comprehensible interpretations of these recommendations. By fine-tuning the expected outcomes, we successfully achieve a balance between consumers' comfort and average power load, represented by a Pareto front.

One limitation of the simulations in this paper is that, due to the constraints of the dataset and the complexity of customizing data, only the UK-DALE dataset was used. This does not demonstrate the broad applicability of the algorithm but serves only as a starting point for future research. To obtain an exact percentage, much more research should be conducted in the future, including studies involving inaccurate data.

Future research will aim to develop more complex recommendation graphs (or use different datasets) and test the robustness of the algorithm in practical applications.

Author Contributions: Methodology, W.C., H.S. and M.Y.; Software, W.C.; Validation, W.C. and H.S.; Investigation, M.Y.; Writing—original draft, H.S.; Writing—review and editing, H.S., M.Y., J.J. and M.R.; Supervision, H.S., J.J. and M.R. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie Grant Agreement No. 872172 TESTBED2 project, Royal Society IEC\NSFC\223152, Royal Society Short Industry Fellowship and European Space Agency Project 12893212. The Agencia Nacional de Investigación y Desarrollo (ANID) FONDECYT Regular grant number 1220556, the Centre for Multidisciplinary Research on Smart and Sustainable Energy Technologies for Sub-Antarctic Regions under Climate Crisis ANID/ATE220023, ANID FOVI230169, Fondap SERC 1523A0006 and IRCF 24932270 project from the University of Nottingham.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional neural network
DARK	Device Action Recommendation with Knowledge graph
DNN	Deep neural network
KGAT	Knowledge graph attention network
RNN	Recurrent neural network
TransE	Translating embeddings for modelling multi-relational data
UK	The United Kingdom
UK-DALE	UK domestic appliance-level electricity

References

1. Orfanos, V.A.; Kaminaris, S.D.; Papageorgas, P.; Piromalis, D.; Kandris, D. A comprehensive review of IoT networking technologies for smart home automation applications. *J. Sens. Actuator Netw.* **2023**, *12*, 30. [CrossRef]
2. Das, S.; Namasudra, S.; Deb, S.; Ger, P.M.; Crespo, R.G. Securing IoT-based Smart Healthcare Systems by using Advanced Lightweight Privacy-Preserving Authentication Scheme. *IEEE Internet Things J.* **2023**, *10*, 18486–18494. [CrossRef]
3. Javed, A.R.; Ahmed, W.; Pandya, S.; Maddikunta, P.K.R.; Alazab, M.; Gadekallu, T.R. A survey of explainable artificial intelligence for smart cities. *Electronics* **2023**, *12*, 1020. [CrossRef]
4. Bakare, M.S.; Abdulkarim, A.; Zeeshan, M.; Shuaibu, A.N. A comprehensive overview on demand side energy management towards smart grids: Challenges, solutions, and future direction. *Energy Inform.* **2023**, *6*, 4. [CrossRef]
5. Sridhar, A.; Honkapuro, S.; Ruiz, F.; Stoklasa, J.; Annala, S.; Wolff, A.; Rautiainen, A. Toward residential flexibility—Consumer willingness to enroll household loads in demand response. *Appl. Energy* **2023**, *342*, 121204. [CrossRef]
6. Rasch, K. An unsupervised recommender system for smart homes. *J. Ambient. Intell. Smart Environ.* **2014**, *6*, 21–37. [CrossRef]
7. Chen, H.; Xie, X.; Shu, W.; Xiong, N. An efficient recommendation filter model on smart home big data analytics for enhanced living environments. *Sensors* **2016**, *16*, 1706. [CrossRef] [PubMed]
8. Belghini, N.; Gouttaya, N.; Bennani, W.B.; Sayouti, A. Pervasive recommender system for smart home environment. *Int. J. Appl. Inf. Syst.* **2016**, *10*, 1–7. [CrossRef]
9. Reyes-Campos, J.; Alor-Hernández, G.; Machorro-Cano, I.; Olmedo-Aguirre, J.O.; Sánchez-Cervantes, J.L.; Rodríguez-Mazahua, L. Discovery of resident behavior patterns using machine learning techniques and IoT paradigm. *Mathematics* **2021**, *9*, 219. [CrossRef]
10. Jeon, H.; Kim, J.; Yoon, H.; Lee, J.; Kang, U. Accurate action recommendation for smart home via two-level encoders and commonsense knowledge. In Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, 17–21 October 2022; pp. 832–841.
11. Varlamis, I.; Sardianos, C.; Chronis, C.; Dimitrakopoulos, G.; Himeur, Y.; Alsalemi, A.; Bensaali, F.; Amira, A. Smart fusion of sensor data and human feedback for personalized energy-saving recommendations. *Appl. Energy* **2022**, *305*, 117775. [CrossRef]
12. Yao, Y.; Kamani, M.M.; Cheng, Z.; Chen, L.; Joe-Wong, C.; Liu, T. FedRule: Federated Rule Recommendation System with Graph Neural Networks. *arXiv* **2022**, arXiv:2211.06812.
13. Ali, S.M.; Augusto, C.; Windridge, D.; Ward, E. A user guided personalization methodology to facilitate new smart home occupancy. *Univers. Access Inf. Soc.* **2023**, *22*, 869–891. [CrossRef] [PubMed]
14. Dilekh, T.; Benharzallah, S.; Mokeddem, A.; Kerdoudi, S. Dynamic Context-Aware Recommender System for Home Automation Through Synergistic Unsupervised and Supervised Learning Algorithms. *Acta Inform. Pragensia* **2024**, *13*, 38–61. [CrossRef]
15. Wen, J.; Zhang, Z.; Lan, Y.; Cui, Z.; Cai, J.; Zhang, W. A survey on federated learning: Challenges and applications. *Int. J. Mach. Learn. Cybern.* **2023**, *14*, 513–535. [CrossRef]
16. Kelly, J.; Knottenbelt, W. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Sci. Data* **2024**, *2*, 150007 [CrossRef] [PubMed]
17. Wang, X.; He, X.; Cao, Y.; Liu, M.; Chua, T.S. KGAT: Knowledge graph attention network for recommendation. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Anchorage, AK, USA, 4–8 August 2019; Volume 1, pp. 950–958.
18. Peng, C.; Xia, F.; Naseriparsa, M.; Osborne, F. Knowledge graphs: Opportunities and challenges. *Artif. Intell. Rev.* **2023**, *56*, 13071–13102. [CrossRef] [PubMed]
19. Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; Yakhnenko, O. Translating embeddings for modeling multi-relational data. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 1–14.
20. Ai, Q.; Azizi, V.; Chen, X.; Zhang, Y. Learning heterogeneous knowledge base embeddings for explainable recommendation. *Algorithms* **2018**, *11*, 137. [CrossRef]
21. Mirjalili, S.; Mirjalili, S. Genetic algorithm. *Evol. Algorithms Neural Netw. Theory Appl.* **2019**, *1*, 43–55.
22. Tarascio, V.J. Vilfredo Pareto and the translation of his Manuel. *J. Econ. Lit.* **1974**, *12*, 91–96. [CrossRef]
23. Sharifani, K.; Amini, M. Machine learning and deep learning: A review of methods and applications. *World Inf. Technol. Eng. J.* **2023**, *10*, 3897–3904.