

APPLICATION PAPER

# An explainable multi-agent recommendation system for energy-efficient decision support in smart homes

Alona Zharova , Annika Boer, Julia Knoblauch, Kai Ingo Schewina  and Jana Vihs

Chair of Information Systems, Humboldt-Universität zu Berlin, Berlin, Germany

**Corresponding author:** Alona Zharova; Email: [alona.zharova@hu-berlin.de](mailto:alona.zharova@hu-berlin.de)

**Received:** 27 February 2023; **Revised:** 08 November 2023; **Accepted:** 22 January 2024

**Keywords:** energy efficiency; explainable AI; recommendation systems; residential buildings

## Abstract

Transparent, understandable, and persuasive recommendations support the electricity consumers' behavioral change to tackle the energy efficiency problem. This paper proposes an explainable multi-agent recommendation system for load shifting for household appliances. First, we extend a novel multi-agent approach by designing and implementing an Explainability Agent that provides explainable recommendations for optimal appliance scheduling in a textual and visual manner. Second, we enhance the predictive capacity of other agents by including weather data and applying state-of-the-art models (i.e., k-nearest-neighbors, extreme gradient boosting, adaptive boosting, Random Forest, logistic regression, and explainable boosting machines). Since we want to help the user understand a single recommendation, we focus on local explainability approaches. In particular, we apply post-model approaches local, interpretable, model-agnostic explanation and SHapley Additive exPlanations as model-agnostic tools that can explain the predictions of the chosen classifiers. We further provide an overview of the predictive and explainability performance. Our results show a substantial improvement in the performance of the multi-agent system while at the same time opening up the "black box" of recommendations.

## Impact Statement

This application paper addresses the explainability side of the load-shifting recommendations aiming at energy efficiency in residential households. Seeing the transparent and understandable recommendations daily will increase the awareness of residents of their energy consumption and will encourage more climate-related actions (supporting SDG 13). The shifted load will facilitate energy efficiency in the grid (SDG 7), foster energy innovation toward sustainable development (SDG 9), reduce the environmental impact, and strengthen the households' sustainability, making them inclusive, safe, and resilient (SDG 11).

## 1. Introduction

Europe faces a double urgency to increase energy efficiency: on the one hand, caused by the war in Ukraine, and on the other hand, due to the continuous rise in electricity consumption (European Commission, 2022). Tackling the energy efficiency problem through consumers' behavioral change is an obvious, however, challenging solution. People often need guidance and sometimes a soft nudge to put the intentions into actions (Frederiks et al., 2015), for instance, to change the timing of appliance usage. Recommender systems can suggest energy-efficient actions to facilitate such behavioral changes. To increase trust in the recommendation system, and, thus, the acceptance rate of recommendations,

users need to understand why and how the model makes its predictions (Luo et al., 2021; Sayed et al., 2022). Thus, the recommendation system should be explainable.

Existing recommendation systems for enhancing energy efficiency in residential buildings vary in approach and implementation. Pinto et al. (2019) introduce a multi-agent case-based reasoning system that focuses on load curtailing. Ran and Leng (2019) propose a load-shifting strategy for optimizing energy costs. Jimenez-Bravo et al. (2019) present a multi-agent system that offers load-shifting recommendations. Sinha and De (2016) develop a load-shifting algorithm that prioritizes time-of-day tariffs, offering potential cost savings and scalability. Machorro-Cano et al. (2020) introduce a home energy management system that generates recommendations based on behavioral patterns.

However, the existing research on explainability in recommender systems for energy-efficient smart homes is very scarce (Himeur et al., 2021). Zhang and Chen (2020) provide a thorough literature review on explainability in recommender systems for other application domains. However, most existing approaches are not applicable to the smart home area because of the missing data structures. Sardanios et al. (2021) design an explainable context-aware recommendation system for a smart home ecosystem. They show that displaying the advantages and the reasoning behind recommendations leads to a 20% increase in acceptance rate. To the best of our knowledge, the issue of explainability in multi-agent recommendation systems for energy-efficient smart homes has not been studied yet.

Our contributions are twofold. First, we suggest an explainable multi-agent recommendation system for energy efficiency in private households. In particular, we extend a multi-agent approach of Riabchuk et al. (2022) by designing and implementing an Explainability Agent that provides explainable recommendations for optimal appliance scheduling in a textual and visual manner. Second, we enhance the predictive capacity of other agents by including weather data and applying state-of-the-art models. We also provide an overview of predictive and explainability performance.

## 2. Explainable multi-agent recommendation system

Riabchuk et al. (2022) introduce a utility-based context-aware multi-agent recommendation system that provides load-shifting recommendations for household devices for the next 24 h. Their system includes six agents: Price Agent (prepares external hourly electricity prices), Preparation Agent (prepares data for the other agents), Availability Agent (predicts the hourly user availability for the next 24 h), Usage Agent (calculates the devices' usage probabilities for the prediction day), Load Agent (extracts the typical devices' loads), and Recommendation Agent (collects the inputs from the other agents and provides recommendations). The multi-agent architecture is flexible and can be easily integrated into existing smart home systems. However, the cost of the simplicity of the approach (i.e., they use logistic regression for the availability and usage predictions) is a relatively low prediction accuracy.

We address the limitations in Riabchuk et al. (2022) by enhancing the performance of the Availability and the Usage Agents. In particular, we apply the K-nearest neighbors (KNN), extreme gradient boosting (XGBoost), adaptive boosting (AdaBoost), and Random Forest to predict the availability and usage probabilities in the smart home environment. Furthermore, we use logistic regression (Logit) and explainable boosting machines (EBM) as inherently explainable models designed for interpretability (Nori et al., 2019).

We justify our choice of advanced algorithms for the following reasons. First, the size of the dataset, the complex feature space, and the nonlinear energy consumption patterns make it difficult for a linear model such as linear regression to effectively capture the complexity. Second, our goal goes beyond predictive accuracy. We aim at providing explainable recommendations to consumers, thereby encouraging behavioral changes to enhance energy efficiency. Advanced models such as Random Forest, XGBoost, and EBM offer a balance between predictive power and interpretability that is crucial for our application.

The explainability models are divided into local and global, depending on their capability to explain a particular instance or the entire model. Since we want to help the user understand a single recommendation, we focus on explainers of local models aiming at uncovering the reasons for the decision of a

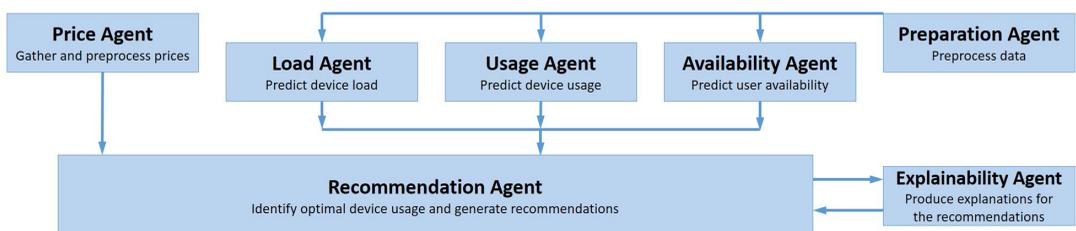
black-box model for a specific instance. We apply post-model approaches local, interpretable, model-agnostic explanation (LIME) (Ribeiro et al., 2016) and SHapley Additive exPlanations (SHAP) (Lundberg and Lee, 2017) as model-agnostic tools that can explain the predictions of the chosen classifiers. In particular, LIME uses simple linear models to explain the black-box model and focuses on explaining the prediction of a single instance of data in an interpretable and faithful manner. SHAP explains the prediction of a black-box model by computing the contribution of each feature to the prediction.

We propose an Explainability Agent that is called within the Recommendation Agent (see Figure 1). To create an explanation for a recommendation, the Explainability Agent extracts feature importance from the explainability models and provides them to the Recommendation Agent. To inform the user about what causes the specific recommendation besides price, we design the explanation for a single device recommendation to include two parts: usage and availability explanations. The usage explanation shows which features lead to the specific device usage prediction for the day, whereas the availability explanation describes which features drive the user availability prediction for the hour. We do not include an explanation for the Load Agent since we do not consider the extracted typical load profile of every shiftable device as informative to the users. As a result, the Recommendation Agent suggests the cheapest starting hour within the hours of user availability for the shiftable devices that are likely to be used on the prediction day with an explanation in text and visual form. The system provides no recommendations if the predictions for the availability and usage probabilities are below the thresholds.

### 3. Experimental design

We use the REFIT electrical load measurement dataset (Murray et al., 2017). It contains the energy consumption in Watts of nine appliances used in 20 households in the United Kingdom as well as the aggregate energy consumption in each household over the period 2013–2015. Energy consumption readings are recorded at 8-s intervals. We perform a number of preparation steps, including data aggregation; data preparation (i.e., examining data for skewness, identifying outliers, handling outliers, scaling the data); feature creation; and weather data integration. The energy consumption data have skewed distributions, likely due to periods when appliances are switched off and not using energy. However, this skewness is considered acceptable for the analysis since the focus is on periods when appliances are actively in use. We handle the outliers in the energy consumption data through outlier truncation method to ensure that extreme values do not influence the analysis.

To capture hidden patterns, improve predictions, and reveal insights beyond the raw data, we create several types of features from the energy consumption dataset. The device usage features are binary variables that indicate whether a device has been used in a given time period. These features are based on the scaled energy consumption of each device and involve setting a threshold to account for noise in the consumption data. The last usage feature calculates how many time periods have passed since the last usage of a device, providing insight into the frequency and recency of device use. The user availability feature is a binary variable that indicates whether the user was available during a given time period. It is based on



**Figure 1.** Architecture of the explainable multi-agent recommendation system.

the use of appliances that require user interaction (e.g., tumble dryer, washing machine, dishwasher). The Time features include information such as the hour of the day, day of the week (i.e., index), day name, month index, month name, and whether the period falls on a weekend. The time lag features represent time shifts for specific periods and features. For example, the availability lag and hour lag features provide insight into how usage and time-related variables change over time intervals (e.g., lag of 1 h, 2 h, and 5 h). These additional features enhance the analysis of energy consumption and user behavior, facilitating the exploration of device usage, availability, and temporal patterns to gain insight into the factors that affect household energy consumption.

Furthermore, the weather may be crucial for the use of some appliances in a household. Therefore, we extend our dataset with specific weather features from the Meteostat (2022). In particular, we use the following features: dew point (dwpt), relative humidity (rhum), air temperature (temp), average wind direction (wdir), and average wind speed (wspd). The missing values are imputed using the KNN algorithm.

We apply an exhaustive grid search over the algorithms mentioned in the previous section, excluding the EBM. The latter is computationally expensive. Overall, 87 parameter combinations are tested twice (with and without weather data) to quantify the benefit of including additional data. We use a KernelExplainer for explaining Logit, AdaBoost, KNN, and Random Forest. For XGBoost, we use the fastest version of the TreeExplainer—interventional feature perturbation (Lundberg et al., 2020).

To evaluate the performance of the multi-agent model, we apply multiple metrics depending on the task. The Usage and the Availability Agents perform a classification task, and therefore, we evaluate their area-under-the-ROC-curves (AUC). We evaluate each day by predicting probabilities of availability and usage and compare them to the true values. The Load Agent is evaluated by investigating the mean squared error (MSE) of the predicted load profile to real usages. We refer to Riabchuk et al. (2022) for details. We do not make any changes to the prediction approach of the Load Agent and, therefore, we do not report its performance evaluation.

With the evaluation of the explainability approaches, we aim at a quantitative comparison of the individual explanations that the models offer. For this purpose, we use three metrics within the Explainability Agent to reflect how well the explainability approaches work in giving accurate local explanations (Carvalho et al., 2019): accuracy, fidelity, and efficiency. Accuracy shows how well the explainable model predicts unseen instances compared to the real outcome. We use the AUC as a measure to compare the true labels with the predictions from the explainers. Fidelity determines how close the prediction from the explainable model is to the black-box model's prediction. In other words, fidelity describes how well the explainability model can imitate the prediction of the black-box model. Additionally, we calculate the mean absolute explainability error (MAEE) for every approach to measure how close is the decision of the black-box model to the explainability approach. The MAEE represents how well the explainability approaches are calibrated. The efficiency metric describes the algorithmic complexity of the explainability approaches when calculating the local explanations. For this purpose, we measure the time that each method needs to calculate all the local explanations for a day and average the values for all calculations.

#### 4. Results

Table 1 provides the performance evaluation results for the Availability and Usage Agents without weather data. The models show relatively stable performance for the Availability Agent, with AdaBoost slightly outperforming the others. For the Usage Agent, we observe a much greater variation in performance across devices, with the highest AUC achieved by the KNN model. The inclusion of weather data (see Table 2) leads to slight performance improvements for the Availability Agent, but a significant performance increase for the Usage Agent. Without the additional data, most models achieve an AUC of around 0.7 for the Usage Agent. However, the inclusion of the weather data allows for a substantial increase in performance toward 0.9. Complex models in particular benefit considerably from the inclusion of the data, outperforming the approach of Riabchuk et al. (2022).

**Table 1.** Performance evaluation results (in AUC) for the Availability and the Usage Agents with tuned hyperparameters excluding weather data

Model	Availability	Usage TD	Usage WM	Usage DW
Logit	0.838	0.702	0.689	0.676
Random Forest	0.835	0.713	0.683	0.715
AdaBoost	<b>0.849</b>	0.710	0.701	0.680
KNN	0.826	<b>0.738</b>	<b>0.718</b>	<b>0.779</b>
XGBoost	0.845	0.712	0.682	0.710
EBM	0.846	0.707	0.683	0.700

Abbreviations: TD—tumble dryer, WM—washing machine, DW—dishwasher.  
The best-performing model is in bold.

**Table 2.** Performance evaluation results (in AUC) for the Availability and the Usage Agents with tuned hyperparameters including weather data

Model	Availability	Usage TD	Usage WM	Usage DW
Logit	0.836	0.728	0.703	0.684
Random Forest	<b>0.859</b>	<b>0.928</b>	<b>0.932</b>	<b>0.929</b>
AdaBoost	0.850	0.779	0.777	0.788
KNN	0.830	0.760	0.758	0.785
XGBoost	0.852	0.922	0.925	0.922
EBM	0.854	0.803	0.822	0.842

Abbreviations: TD—tumble dryer, WM—washing machine, DW—dishwasher.  
The best-performing model is in bold.

**Table 3.** Performance evaluation results (in AUC) for Random Forest for the Availability and the Usage Agents for 10 households

Household	Availability	Usage TD	Usage WM	Usage DW	Usage WM2	Usage WD
1	0.795	0.949	0.923	0.942		
2	0.827		0.892	0.932		
3	0.859	0.927	0.934	0.927		
4	0.860		0.929		0.871	
5	0.816	0.910				
6	0.826		0.914	0.948		
7	0.881	0.950	0.945	0.945		
8	0.675		0.921			
9	0.765		0.926	0.944		0.920
10	0.820		0.950			

Abbreviations: DW, dishwasher; TD, tumble dryer; WD, washdryer; WM, washing machine; WM2, second washing machine.

Specifically, the Random Forest model profits the most from additional data and consistently outperforms other models for both agents and devices. We conclude that it is not sufficient to use either a more complex model or weather data. Instead, it is the combination of both that leads to significant performance improvements.

To assess the stability of the Random Forest model, we further analyze the AUC across 10 households in Table 3 (see Table A1 in Appendix for performance evaluation results with logistic regression). The results indicate consistent performance for both the Availability and the Usage Agents, with an average AUC of 0.812 and 0.918, respectively, across all households. Based on these findings, we utilize the Random Forest model with the tuned hyperparameters obtained from the grid search for the prediction tasks of both agents.

#### 4.1. Explainability

We examine the results of the explainability evaluation by using the predictions from LIME and SHAP (see Tables 4 and 5). SHAP generally performs better according to the higher accuracy and fidelity compared to LIME. Most strikingly, since the forecasts of the prediction algorithms and SHAP are so similar, the fidelity is almost perfect. In other words, SHAP works very well for mimicking the local behavior of the prediction model. The accuracy of LIME for the Usage Agent is not as high. The possible reason for this is that the predictions, for example, for AdaBoost, are very close to the chosen cutoff of 0.5 to create target values of the prediction. For example for the AdaBoost model, LIME is only off by an average of 0.0311 AUC points, sometimes exceeding the cutoff and assigning the values to the wrong class. For the Availability Agent, the poorer calibration of LIME does not have too much of an impact as the prediction values are more extreme. Furthermore, SHAP produces predictions faster than LIME in most cases. Therefore, we choose SHAP over LIME for our explainability task because of its higher performance on the metrics.

Since there is no significant difference between the fidelity of SHAP across the models, we apply SHAP with one of the most consistent and stable algorithms. The AdaBoost model shows the highest

**Table 4.** Explainability evaluation results for LIME and SHAP for the Availability Agent for 10 households

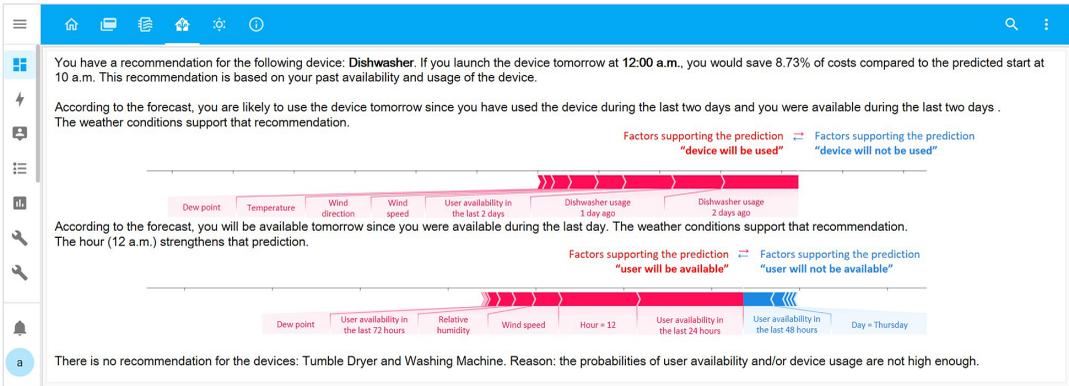
Model	LIME				SHAP			
	Accuracy	Fidelity	MAEE	Duration*	Accuracy	Fidelity	MAEE	Duration*
Logit	0.8371	0.9890	0.0265	73.748	0.8362	1	0	2.147
KNN	0.8295	0.9747	0.0194	79.695	0.8386	1	0	20.759
AdaBoost	0.8369	0.9349	0.0035	50.975	0.8479	1	0	52.103
Random Forest	0.8411	0.9761	0.0313	107.363	0.8451	1	0	64.853
XGBoost	0.8378	0.7971	0.1848	310.551	0.8366	1	0	0.717

\*The duration is only comparable within the model since the evaluation was run on different machines with different background tasks.

**Table 5.** Explainability evaluation results for LIME and SHAP for the Usage Agent for 10 households

Model	LIME				SHAP			
	Accuracy	Fidelity	MAEE	Duration*	Accuracy	Fidelity	MAEE	Duration*
Logit	0.6761	0.8215	0.1454	2.582	0.7049	1	0	0.084
KNN	0.6928	0.8101	0.1858	2.864	0.7611	1	0	0.852
AdaBoost	0.5757	0.5934	0.0311	1.859	0.8254	1	0	0.813
Random Forest	0.7101	0.7988	0.1203	3.682	0.8726	1	0	0.957
XGBoost	0.6763	0.6516	0.2705	10.222	0.9353	1	0	0.230

\*The duration is only comparable within the model since the evaluation was run on different machines with different background tasks.



**Figure 2.** Exemplary explainable recommendation.

performance for the Availability Agent, followed by the Random Forest with a slightly lesser performance. For the Usage Agent, the best model is the XGBoost, closely followed by the Random Forest. Thus, the Random Forest model has a consistently good performance across agents and is therefore selected as a final model for the prediction task.

The reported SHAP runtimes within the Random Forest model for the Availability and Usage Agents are 64.853 and 0.957, respectively (see Tables 4 and 5). To expedite the explainability evaluation (i.e., decrease the duration) while retaining accuracy, we conduct further runs with the TreeExplainer. This led to a significant reduction in the SHAP runtimes for Availability and Usage Agents to 0.558 and 0.0357, respectively. It is noteworthy that this alteration in runtime has minimal impact on the other metrics. Consequently, we utilize SHAP and Random Forest with TreeExplainer as the ultimate model.

#### 4.2. Explainable recommendation

The recommendation for the next 24 h is provided once a day at the same time specified by the user. To create an explanation for the recommendation, we create a feature importance ranking using SHAP values. We provide two different explanations for the Availability and the Usage Agents. We separate the features into two categories for each of the agents: features based on weather data and non-weather features. The Recommendation Agent embeds the two most important features of each group into an explanation sentence. The usage explanation is provided for each device (if its usage is recommended) since their predictions differ. Additionally, we adapt the plots provided by SHAP to inform the user about the specific impact of the features. We only display the most important features to shorten the explanation. We show an exemplary explainable recommendation in Figure 2.

### 5. Future work

In our future work, we aim to create an integration of the explainable multi-agent recommendation system for the existing open-source smart home platform Home Assistant. With around 330,000 active users worldwide (Home Assistant, 2024), it provides a viable real-world environment for testing our solution within selected households. We are going to modify the recommendation system to work with real-time data streams from the smart home and include a user feedback option. This research work on an explainable multi-agent recommendation system lays a solid foundation for further evaluation of the interpretability, comprehensibility, and understandability of the recommendations using user feedback.

### 6. Conclusions

This paper presents an explainable multi-agent recommendation system aimed at enhancing energy efficiency in private households. Our empirical results show significant performance improvements by

incorporating weather data into the recommendation system. In particular, we observe notable improvements in AUC values, especially for the Usage Agent, where performance increases from around 0.7 to approximately 0.9. This highlights the importance of leveraging weather data to improve energy efficiency recommendations.

In the context of explainability, our evaluation highlights the superiority of SHAP over LIME, with SHAP exhibiting higher accuracy and fidelity. Particularly, SHAP's ability to closely mimic the local behavior of the prediction model contributes to its selection as the preferred explainability method. Furthermore, our findings consistently identify the Random Forest model as a strong performer across different agents and devices. The ability of this model to consistently achieve high performance underscores its suitability for the prediction tasks within the recommendation system.

To conclude, our study demonstrates that generating load-shifting recommendations for household appliances as explainable enhances the transparency and trustworthiness of the system. The comprehensible and persuasive nature of these recommendations facilitates consumers behavior change, making them more inclined to address the energy efficiency problem.

**Acknowledgments.** The authors would like to thank Lucas Pereira for his great mentorship within the Climate Change AI (CCAI) program while preparing the first version of this paper for the submission to NeurIPS 2022 Tackling Climate Change with Machine Learning Workshop. The authors wish to express their sincere gratitude and warm appreciation to Stefan Lessmann for many insightful discussions on the topic. The authors are deeply thankful to all participants of the Information Systems Research Seminar at the Humboldt-Universität zu Berlin for the valuable feedback.

**Author contribution.** Conceptualization: A.Z.; Data curation: A.B., K.I.S.; Formal analysis: A.B., J.K., K.I.S.; Investigation: A.B., J.K., K.I.S., J.V.; Methodology: J.K.; Resources: A.Z.; Software: A.B., J.K., K.I.S., J.V.; Supervision: A.Z.; Validation: A.Z.; Visualization: J.K., J.V.; Writing—original draft: A.B., J.K., K.I.S., J.V.; Writing—review and editing: A.Z. All authors approved the final submitted draft.

**Data availability statement.** The REFIT data set we used is available at <https://doi.org/10.1038/sdata.2016.122>. The weather data from the Meteostat are available at <https://github.com/meteostat/meteostat-python>. We provide a comprehensive tutorial in Jupyter Notebook with the replication code in GitHub: [https://github.com/Humboldt-WI/Explainable\\_multi-agent\\_RecSys](https://github.com/Humboldt-WI/Explainable_multi-agent_RecSys) for all the steps described in this paper and beyond.

**Funding statement.** This work received no specific grant from any funding agency, commercial, or not-for-profit sectors.

**Competing interest.** The authors declare none.

**Ethics statement.** The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

## References

- Carvalho DV, Pereira EM and Cardoso JS (2019) Machine learning interpretability: A survey on methods and metrics. *Electronics* 8(8), 832.
- European Commission (2022) REPowerEU: A Plan to Rapidly Reduce Dependence on Russian Fossil Fuels and Fast Forward the Green Transition. 18 May 2022. Available at [https://ec.europa.eu/commission/presscorner/detail/en/IP\\_22\\_3131](https://ec.europa.eu/commission/presscorner/detail/en/IP_22_3131) (accessed 08 November 2023).
- Frederiks ER, Stenner K and Hobman EV (2015) Household energy use: Applying behavioural economics to understand consumer decision-making and behaviour. *Renewable and Sustainable Energy Reviews* 41, 1385–1394.
- Himeur Y, Alsalemi A, Al-Kababji A, Bensaali F, Amira A, Sardianos C, Dimitrakopoulos G and Varlamis I (2021) A survey of recommender systems for energy efficiency in buildings: Principles, challenges and prospects. *Information Fusion* 72(2021), 1–21.
- Home Assistant (2024) Home Assistant Analytics. Active Home Assistant Installations. Available at <https://analytics.home-assistant.io/> (accessed 04 March 2024).
- Jimenez-Bravo DM, Perez-Marcos J, De la Iglesia DH, Villarrubia Gonzalez G and De Pa JF (2019) Multi-agent recommendation system for electrical energy optimization and cost saving in smart homes. *Energies* 12(7), 1317.
- Lundberg SM, Erion G, Chen H, DeGrave A, Prutkin JM, Nair B, Katz R, Himmelfarb J, Bansal N and Lee S-I (2020) From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence* 2(1), 56–67.
- Lundberg SM and Lee SI (2017) A unified approach to interpreting model predictions. In *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 4768–4777.
- Luo F, Ranzi G, Kong W, Liang G and Dong ZY (2021) Personalized residential energy usage recommendation system based on load monitoring and collaborative filtering. *IEEE Transactions on Industrial Informatics* 17(2), 1253–1262.

- Machorro-Cano I, Alor-Hernández G, Paredes-Valverde MA, Rodríguez-Mazahua L, Sánchez-Cervantes JL and Olmedo-Aguirre JO** (2020) HEMS-IoT: A big data and machine learning-based smart home system for energy saving. *Energies* 13(5), 1097.
- Meteostat** (2022) GitHub repository meteostat/meteostat-python. Access and analyze historical weather and climate data with Python. Available at <https://github.com/meteostat/meteostat-python> (accessed 08 November 2023).
- Murray D, Stankovic L and Stankovic V** (2017) An electrical load measurements dataset of United Kingdom households from a two-year longitudinal study. *Scientific Data* 4(1), 160122.
- Nori H, Jenkins S, Koch P and Caruana R** (2019) InterpretML: A unified framework for machine learning interpretability. Preprint, [arXiv:1909.09223](https://arxiv.org/abs/1909.09223).
- Pinto T, Faia R, Navarro-Caceres M, Santos G, Corchado JM and Vale Z** (2019) Multiagent-based cbr recommender system for intelligent energy management in buildings. *IEEE Systems Journal* 13(1), 1084–1095.
- Ran X and Leng S** (2019) Enhanced robust index model for load scheduling of a home energy local network with a load shifting strategy. *IEEE Access* 7, 19943–19953.
- Riabchuk V, Hagel L, Germaine F and Zharova A** (2022) Utility-based context-aware multi-agent recommendation system for energy efficiency in residential buildings. Preprint, [arXiv:2205.02704](https://arxiv.org/abs/2205.02704).
- Ribeiro MT, Singh S and Guestrin C** (2016) Why should I trust you? Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. Association for Computing Machinery, New York, NY, USA, pp. 1135–1144.
- Sardianos C, Varlamis I, Chronis C, Dimitrakopoulos G, Alsalemi A, Himeur Y, Bensaali F and Amira A** (2021) The emergence of explainability of intelligent systems: Delivering explainable and personalized recommendations for energy efficiency. *International Journal of Intelligent Systems* 36(2), 656–680.
- Sayed A, Himeur Y, Alsalemi A, Bensaali F and Amira A** (2022) Intelligent edge-based recommender system for internet of energy applications. *IEEE Systems Journal* 16(3), 5001–5010.
- Sinha A and De M** (2016) Load shifting technique for reduction of peak generation capacity requirement in smart grid. In *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, pp. 1–5.
- Zhang Y and Chen X** (2020) Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval* 14(1), 1–101.

## Appendix

**Table A1.** Performance evaluation results for logistic regression for the 10 households from Riabchuk et al. (2022)

Household	Availability AUC		Usage AUC		Load MSE		
	0	1	1	2	0	1	2
1	0.72	0.52	0.45	0.49	1,224.45	511.49	3,021.88
2	0.77	0.60	0.80	–	320.81	14,578.80	–
3	0.80	0.65	0.65	0.65	24,907.50	890.33	314.84
4	0.82	0.67	0.52	–	0.80	1,793.77	–
5	0.76	0.68	–	–	37,982.21	–	–
6	0.77	0.53	0.66	–	424.78	6,032.65	–
7	0.83	0.79	0.80	0.81	43,673.35	336.83	6,845.26
8	0.62	0.77	–	–	0.35	–	–
9	0.67	0.60	0.57	0.78	30,711.30	14.36	5,480.13
10	0.77	0.74	–	–	647.57	–	–

The table is reproduced here from Riabchuk et al. (2022) to enable the direct comparison with our results in Tables 3. Legend for mapping the shiftable devices 0–2 is provided in Table A2.

**Table A2.** Legend for mapping the shiftable devices to an integer index from Riabchuk et al. (2022)

Household	Devices		
	0	1	2
1	Tumble dryer	Washing machine	Dishwasher
2	Washing machine	Dishwasher	–
3	Tumble dryer	Washing machine	Dishwasher
4	Washing machine (1)	Washing machine (2)	–
5	Tumble dryer	–	–
6	Washing machine	Dishwasher	–
7	Tumble dryer	Washing machine	Dishwasher
8	Washing machine	–	–
9	Washer dryer	Washing machine	Dishwasher
10	Washing machine	–	–