IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 24 January 2026, accepted 13 February 2026, date of publication 20 February 2026, date of current version 25 February 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3666653

RESEARCH ARTICLE

Adaptive Integration of Steady-State Change Features for Enhanced Multi-Label Energy Disaggregation

BUNDIT BUDDHAHAI[1], TRAN ANH TUAN[1], WARANYU WONGSEREE[2], AND STEPHEN MAKONIN[3], (Senior Member, IEEE)

[1]Informatics Innovation Center of Excellence, School of Informatics, Walailak University, Nakhon Si Thammarat 80160, Thailand
[2]Faculty of Digital Technology, Chitralada Technology Institute, Bangkok 10300, Thailand
[3]School of Engineering Science, Simon Fraser University, Burnaby, BC V5A 1S6, Canada

Corresponding author: Bundit Buddhahai (bundit.bu@mail.wu.ac.th)

ABSTRACT Accurate identification of appliance operations is crucial for effective energy management in smart energy monitoring or energy disaggregation. This paper proposes a multi-label classification approach that utilizes adaptive feature integration of steady-state features (current (I), active power (P), reactive power (Q), power factor (PF)) and steady-state change features (ΔI, ΔP, and ΔQ) obtained from low-frequency meter data. Recursive Feature Elimination with Cross-Validation (RFECV) is employed as the feature selection technique to determine the optimal subset of features for enhancing individual appliance identification. Experiments are conducted on two datasets, AMPds and ECO, consisting of high-power appliances that consume a significant amount of power when turned on (e.g., Wall Oven, Clothes Dryer, Kettle) and low-power appliances that consume less power (Fridge, Freezer). Results demonstrate significant performance improvements on the AMPds dataset with the additional selective features, increasing the F-score for the Wall Oven from 0.56 to 0.70, outperforming Forward Sequential Feature Selection (FSFS) as a comparator of feature selection method and a benchmarking Denoising Autoencoder (DAE) neural network model. In contrast, the ECO dataset yields marginal performance improvement for most appliances due to subtler steady-state change signatures. Additional sensitivity analyses indicate that the approach can achieve superior performance compared to using the baseline features under the data sampling frequencies of 5 and 10 minutes, noisy levels of 1% and 5%, and appliance concurrency up to 3 appliances. The proposed approach of deploying basic features and adapting to variations in measurement setups is suitable for use in resource-constrained environments.

INDEX TERMS Adaptive feature selection, energy disaggregation, multi-label classification, smart meter.

I. INTRODUCTION

Energy disaggregation, also known as non-intrusive load monitoring (NILM), refers to a system that estimates the power consumption of individual appliances using aggregate electricity measurements, typically from a single smart meter [1]. NILM significantly enhances household energy efficiency and facilitates smart-grid applications by offering detailed insights into appliance-level energy consumption without requiring extensive or costly sensor deployments. The NILM research has evolved from rule-based approaches to advanced machine learning techniques, including deep learning, to enhance appliance identification accuracy. Despite significant advancements in accuracy, modern deep learning-based methods, including DAE networks [2] and Convolutional Neural Networks (CNNs) [3], [4], have faced limitations of interpretability, high computational complexity, and substantial training data requirements. Thus, there is an ongoing demand for NILM solutions that could achieve high accuracy of load identification while being interpretable, computationally efficient, and suitable for practical use in real-world scenarios.

To address these challenges, this paper proposes a multi-label classification method that utilizes steady-state

The associate editor coordinating the review of this manuscript and approving it for publication was Zhengmao Li.

features (current, active power, reactive power, and power factor) alongside steady-state change features (ΔI, ΔP, and ΔQ) derived from low-frequency smart meter data. Recursive Feature Elimination with Cross-Validation (RFECV) is applied as a feature selection technique to select the subsets of these interpretable features for optimal predictive performance, independently for each appliance label [5]. Previous feature selection techniques used in the NILM task included the use of explainability metrics (e.g., SHAP values) to identify feature contributions of appliance prediction [6]. However, the approach was a guided method that might not guarantee obtaining the feature set for optimal model performance. The other approach used Principal Component Analysis (PCA) to capture the topmost variance features for appliance identification from raw electrical parameters [7], [8]. This approach, however, provided the transformed feature components that were not directly interpretable, and it might discard some meaningful electrical features. The RFECV approach offers better feature interpretability through cross-validation to guard against overfitting. In addition, the RFECV has better feature stability (consistency of selected features when data or model slightly changes) than SHAP, which can be more sensitive to model randomness and correlated features. It can also generalize the model better than PCA when training across datasets since it ties stability to predictive performance rather than solely to variance structure. However, the RFECV can be computationally more expensive than the other two approaches during training since it requires retraining the model many times (once per feature removal step $\times$ each cross-validation fold) [9]. The approach was successfully applied to other domains of machine learning, such as disease diagnostics and security detection [10].

Experiments conducted on the AMPds and ECO benchmark datasets show substantial improvements, particularly for high-power appliances, and demonstrate clear advantages over Forward Sequential Feature Selection (FSFS). Additionally, benchmarking against a DAE-based method shows that while DAEs might excel at modeling complex appliance behaviors, the proposed approach delivers competitive performance with better feature interpretability and significantly lower computational complexity. The key contributions of this work can be summarized as follows:

(1) We propose a supervised multi-label NILM framework using steady-state and steady-state change features and validate the approach on two public low-frequency metering datasets.

(2) We systematically analyze feature importance and select the optimal subset of features for individual appliances using RFECV to enhance the performance of appliance identification.

(3) We comprehensively benchmark our approach against the established steady-state approach, another feature selection technique, and a deep learning method on real data and simulated data as the sensitivity analysis of the approach. It demonstrates overall improvement of predictive performance.

(4) We provide insights into the appliance-specific effectiveness of steady-state change features, guiding future research toward appliance-adaptive NILM approaches.

The remainder of this paper is structured as follows. Section II provides a review of related NILM literature. Section III describes the proposed methodology which includes feature extraction, RFECV-based feature selection, and a multi-label classification approach. Experimental results and discussions are presented in Section IV. Finally, conclusions and suggestions for future research directions are provided in Section V.

II. RELATED WORKS

NILM was introduced in the late 1980s, employing heuristic algorithms to identify distinctive appliance power signatures from aggregate electrical signals [1]. Initial research in the NILM predominantly utilized steady-state change features, specifically step changes in active and reactive power. These features provided the computational efficiency and suitability for low-frequency measurements. Although these heuristic methods offered practical benefits, they encountered limitations when multiple appliances shared similar power transitions or overlapping steady-state signatures [11]. To overcome the limitations, researchers explored transient features, i.e., the unique signal fluctuations captured during appliance state transitions. These features could improve differentiation among appliances with similar power characteristics [12]. Transient feature-based methods, however, required high-frequency data acquisition, typically in the kHz range, resulting in increased costs and hardware complexity [13]. Additionally, transient signals could vary significantly between appliance operations, further complicating the reliability of appliance detection.

Hybrid approaches have been developed by integrating steady-state measurements (such as active power, reactive power, current, and power factor) with features that indicate steady-state changes. These approaches aimed to improve appliance identification at lower sampling rates by leveraging additional electrical parameters available from digital smart meters. Thus, the approaches could facilitate the practical implementation of large-scale NILM systems. Other machine learning methods, such as Hidden Markov Models (HMMs) and factorial HMMs, further improved NILM performance by modeling the temporal dynamics and multiple appliance states [14]. These approaches, however, required careful feature engineering and struggled with the computational complexity in probabilistic modeling.

Recent system development has increasingly focused on deep learning to learn complex patterns directly from large datasets. The key network architectures include CNNs [15], [16], [17], variant Recurrent Neural Networks (RNNs) for sequential power data [18], [19], and DAEs [20]. In addition, the Transformer-based networks are also employed to

learn complex appliance features and behaviors of industrial machines [21], [22]. Although deep learning-based techniques demonstrated improved performance, they often faced challenges of interpretability and computational complexity. These challenges could hinder their deployment in real-time or embedded systems [23]. The requirement for extensive labeled datasets and substantial training resources could also present additional practical limitations.

The NILM research community has acknowledged the inherent multi-label classification nature by formulating the task as a multiple-output problem [24], [25], [26]. The multi-label classification frameworks explicitly represent simultaneous appliance states, enhancing performance relative to single-label methods in detecting the co-occurrence of appliances. This approach aligns naturally with real-world scenarios, thereby improving the practicality of NILM applications.

Based on the limitations of deep learning and feature selection techniques in machine learning, this paper proposes a multi-label classification framework for low-frequency smart metering. By integrating steady-state and steady-state change features with RFECV as the adaptive feature selection for modeling each appliance, the method balances the computational efficiency, interpretability, and performance, providing a scalable NILM solution for practical deployment.

III. PROPOSED METHODS

The NILM task involves breaking down the total electricity consumption data within a building to identify the operational status of individual appliances. The task can be formulated as a multi-label classification problem [27], with the aggregate measurement data (X) as the input. The output (Y) reflects the operating status of each appliance, indicating whether it is "ON" or "OFF" with binary values of "1" or "0," respectively. The relationship between the input and output can be represented as a multi-label dataset (D) of n instances, k input features, and m output labels as $D = \{(X, Y)\}$. The representation of a multi-label dataset is illustrated in Fig.1, where x_{ij} denotes the j^{th} feature of the i^{th} data instance and y_{ij} denotes the j^{th} output label of the i^{th} data instance. The y_{ij} can be either "1" or "0" ($y_{ij} \in \{1, 0\}$) to represent the relevance to that input feature.

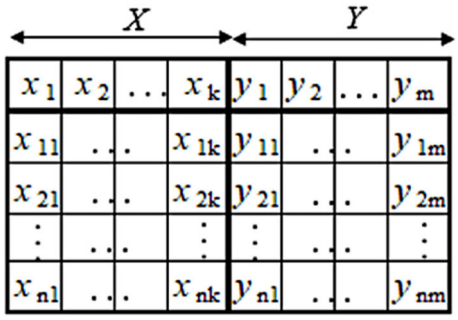

FIGURE 1. Illustration of the input and output of a multi-label dataset.

A. RFECV

The RFECV is a wrapper-based feature selection method that determines the optimal subset of features. It recursively removes weak features based on feature importance (RFE) and uses cross-validation (CV) to identify the feature subset that achieves the best generalization performance [28]. Let X denotes the complete set of features ($X = \{x_1, x_2, .., x_k\}$) and let $C(X_t)$ is the cross-validated performance when using the subset $X_t \subseteq X$ at iteration t. RFECV eliminates the feature x_{min} that contributes the least according to the model internal metric $I(x)$ (such as feature coefficients or impurity importance). It updates the feature set by excluding the x_{min}. These steps can be represented by (1).

$$\begin{aligned} x_{\min} &= \arg\min_{x \subseteq X_t} I(x), \\ X_{t+1} &= X_t \setminus \{x_{\min}\} \end{aligned} \tag{1}$$

The process continues until all possible subsets are evaluated. The optimal subset of features ($X*$) is chosen as the one that maximizes the cross-validation score C, as (2).

$$X^* = \arg\max_{X_t} C\,(X_t) \tag{2}$$

B. FEATURE IMPORTANCEANALYSIS

Feature importance analysis is essential for interpreting how much each feature contributes to the model's predictive capability. For tree-based ensemble methods (e.g., Random Forest or XGBoost), the feature importance is typically calculated by measuring the total reduction in an impurity metric such as the Gini impurity based on splits on that feature. The feature importance (J(x)) for feature x can be computed as (3).

$$J(x) = \frac{1}{N} \sum_{t=1}^{N} \Delta i_t(x) \tag{3}$$

where N and $\mathbf{\Delta i_t(x)}$ are the total number of nodes and the reduction in impurity at node t due to feature x, respectively. The calculation is repeated for all features and can be implemented through the XGBoost's plot_importance() function or the feature importances attribute of the tree-based model through Python's scikit-learn library.

C. MODEL COMPLEXITY

The complexity of the machine learning model can be analyzed using structural and computational complexity metrics. The structural complexity of tree-based ensemble methods can be quantified using metrics such as the average number of nodes per tree. A higher number of nodes indicates a more complex decision structure. The complexity metric is the average number of nodes across all trees, defined by C_{ondes} as (4).

$$C_{\text{ondes}} = \frac{1}{T} \sum_{t=1}^{T} N_t \tag{4}$$

where T is the total number of trees in the model and N_t is the number of nodes in tree t. The calculation can be implemented

through the estimator_ attribute to obtain the count of nodes for each tree and get the average number.

Training time is another metric that measures the time required for model fitting. It serves as a measure of the computational demands associated with different set of features.

IV. EXPERIMENTAL SETUP

We experiment on two datasets, AMPds [29] and ECO [30]. Fig. 2 shows the key processes of data processing and experiments, including creating a multi-label dataset, performing feature analyses as described in Section III, and evaluating model performance by benchmarking it against other approaches. The process can be explained in the following subsections.

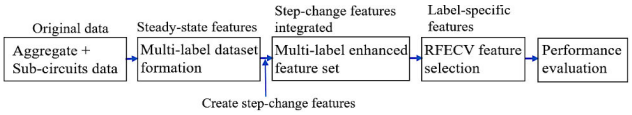

FIGURE 2. Key process on data processing and experiments.

A. DATA PROCESSING

The AMPds dataset is a collection of electrical parameters from 20 sub-circuits and one aggregate circuit from a house in Canada. The measurement parameters include active power (P), reactive power (Q), power factor (PF), current (I), and voltage (V). The data interval for each sample is one minute, and the data collection period is two years. Six appliance labels each contain a single appliance, and the features are used in the experiment because they could indicate which appliances were turned on. The data description for each circuit is summarized in Table 1.

TABLE 1. AMPds dataset description of appliances with a single appliance within each label.

Label	Appliance type	Power range (W)
CDE	Clothes Dryer	80-5000
CWE	Clothes Washer	50-1000
DWE	Dish Washer	30-800
FGE	Kitchen Fridge	30-1400
HPE	Heat Pump	50-2500
WOE	Wall Oven	80-3500

The ECO dataset is a collection of six houses of measurement data from Switzerland over eight months. We use the data from House no. 2 for the experiment because it has the least proportion of unmetered data (the difference between the aggregate power and the sum of individual labels' power data). The measurement parameters include current (I), voltage (V), and phase shift (ϕ) at the aggregate measurement by a second data interval. The sub-circuit measurement contains only the active power data. We process the aggregate data to include the features of current (I), active power (P), and reactive power (Q) (omitting PF due to dataset interpretability), and resample all data to one-minute intervals. We focus on four appliances that consume significant amounts of power, as listed in Table 2.

TABLE 2. ECO dataset description of appliances under experiment.

Label	Power range (W)
Dishwasher	30-2200
Fridge	20-120
Freezer	10-100
Kettle	100-1800

The multi-label data for both datasets are created based on the concept presented in Section III. The original feature set consists of the steady-state features (X) measured at the aggregate circuit, including the baseline parameters of I, P, Q, and PF (PF for AMPds data), which could reflect, to some extent, the operation of appliances. The binary output value for each label is created by converting of power consumption data (P) into a power binary value (''1'': ON/ ''0'': OFF) based on the power ON threshold (P_{ath}) for each appliance. The basic rule is that if the current P data equals or is greater than the P_{ath}, the binary value for that instance is ''1''; otherwise, it is ''0''. The p_{ath} value for each label is determined by identifying the power data that could make the current (I) value jump from zero or near zero to the next step value [31].

The additional steady-state change features (ΔX) include the step change of the steady-state feature from the current instance to the previous one ($x_i - x_{i-1}$). There are three step change features, including ΔI ($i_i - i_{i-1}$), ΔP ($p_i - p_{i-1}$), and ΔQ ($q_i - q_{i-1}$), excluding the step change of the power factor since the feature value is varied within the range of 0 to 1, its step changes would be less informative to detect appliance events compared to ΔI, ΔP, and ΔQ data that has bigger base values. Fig. 3 represents the structure of enhanced multi-label data for the AMPds dataset.

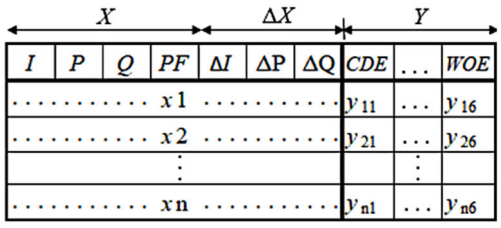

FIGURE 3. AMPds dataset with steady-state and steady-state change features.

B. EVALUATION TOOLS

The multi-label dataset can be processed to learn a model using the Problem Transformation or Algorithm Adaptation approach [27]. The former approach transforms multi-label data into individual or label subsets to create single-label problems. It then manipulates each problem using the conventional classification algorithm as the base classifier to learn the model. The latter approach adapts the single-output classification algorithm to handle the multi-output data.

This work uses the Problem Transformation approach with Boost [32] as the base classifier to learn the model. The classifier is an ensemble method of gradient-boosted decision trees that sequentially trains new trees to minimize the residual errors from previous ones. It features parallelized GPU-based computing to analyze large-scale data effectively.

We use Python's scikit-multilearn library (version 0.2) [33], which currently supports running the multi-label data classification framework through the Problem Transformation approach. It includes data modeling to create the classifier and splitting the training and test data using a stratification method within the library, which ensures an even distribution of data between the two portions. We also use the scikit-learn library (version 1.6) for other data processing, including RFECV feature selection, cross-validation, and model complexity estimation.

C. EVALUATION METRICS

In NILM classification tasks, the model typically aims to detect when an appliance turns on or off, where the samples ON ("1") are rare but important, as they signify the state of energy consumption of that appliance. For such imbalanced data ("1": minority class, "0": majority class), using the accuracy metric is not appropriate, as the model could be misled into predicting all samples "OFF" but still achieve a high accuracy. We use Precision, Recall, and F-score indices to evaluate the model performance as they specifically measure the model ability to correctly identify the minority class (power "ON") data [34], which is the key information to obtain in the NILM task. The calculation of Precision, Recall, and F-score values for each appliance label is presented in (5), (6), and (7), respectively. *TP*, *FP*, and *FN* are the number of samples under the predictive conditions of true positive, false positive, and false negative, respectively.

$$\text{Precision(P)} = \frac{TP}{TP + FP} \tag{5}$$

$$\text{Recall(R)} = \frac{TP}{TP + FN} \tag{6}$$

$$\text{F-score} = \frac{2 \times P \times R}{P + R} \tag{7}$$

V. RESULTS AND DISCUSSION

This section describes the experimental results obtained from each dataset. It aims to analyze the data and determine the optimal set of features for each appliance label. The experiment is conducted separately for each dataset and described as follows.

A. AMPDS DATASET

The selected appliance labels exhibit the power data distribution shown in Fig. 4. All data labels are highly imbalanced, with most data points labeled "0" (appliances are in the OFF state). The heat pump HPE, wall oven WOE, and clothes dryer CDE are high-power appliances, with power consumption data of approximately 2000, 3000, and 4500 W, respectively, at the ON state. The fridge FGE has relatively low-power labels, with power ON data typically around 50-150 W. The clothes washer CWE and dishwasher DWE consume moderate power, at approximately 250 W and 800 W, respectively.

The original dataset has four baseline features (I, P, Q, and PF) and the enhanced feature set has seven features (with the addition of ΔI, ΔP, and ΔQ). The experiments for feature importance analysis and feature selection are described as follows.

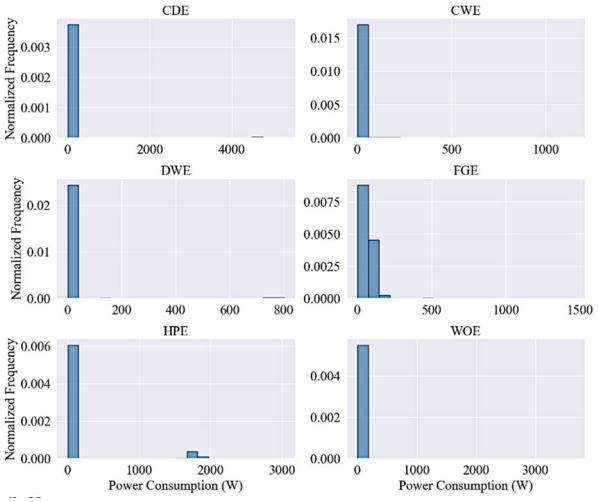

FIGURE 4. AMPds power data distribution of the selected load labels.

1) FEATURE IMPORTANCE ANALYSIS

Feature importance analysis provides a brief overview of each feature's significance to the model predictive capability. We evaluate the feature importance using the *feature_importance* function in the scikit-learn library, which is typically used for single-output data. For multi-label data, we need to evaluate each label individually, and the feature importance values for the enhanced feature set are presented in Table 3. The results show that the baseline features are the key information for the model, with their values mostly higher than those of the steady-state change features. However, the enhanced features are significant to some extent to the model, which could be further analyzed to select the optimal subset.

TABLE 3. AMPds feature importance values for each label.

Label	Feature						
	I	P	Q	PF	ΔI	ΔP	ΔQ
CDE	0.098	0.058	0.768	0.041	0.006	0.006	0.024
CWE	0.197	0.075	0.148	0.367	0.074	0.069	0.070
DWE	0.347	0.176	0.140	0.160	0.065	0.056	0.056
FGE	0.415	0.164	0.146	0.166	0.037	0.038	0.034
HPE	0.115	0.124	0.121	0.471	0.047	0.060	0.063
WOE	0.176	0.075	0.400	0.104	0.063	0.118	0.063

2) FEATURE SELECTION BY RFECV

Feature selection provides an in-depth analysis of features, resulting in a selected subset of features for optimal predictive performance. The RFECV analyzes the entire data for the enhanced feature set and uses K-fold cross-validation ($K = 5$) with default model parameter settings (no nested CV) to evaluate the F-score as a measure of the model generalization performance using a selective subset of features, one for each label. The method provides the *ranking_* parameter where lower numbers indicate more important features and the optimal subset of features is assigned the value "1" for each feature. In other words, the feature whose score is not "1" will be excluded from the optimal feature set. The results of RFECV feature selection for each label, represented by the *ranking_*score, are shown in Fig. 5. The optimal feature set for each appliance are interpreted and presented in Table 4.

TABLE 4. AMPds optimal feature set for each appliance label.

Label	Optimal feature set
CDE	[I, P, Q, PF, ΔP, ΔQ]
CWE	[I, P, Q, PF, ΔI, ΔP, ΔQ]
DWE	[I, P, Q, PF, ΔI, ΔP, ΔQ]
FGE	[I, P, Q, PF]
HPE	[I, P, Q, PF, ΔP, ΔQ]
WOE	[I, P, Q, PF, ΔI, ΔP, ΔQ]

The high-power and moderate-power labels typically exhibit a more noticeable signal of power step changes than the low-power labels. Thus, including steady-state change features would benefit the predictive capability of those labels. The exclusion of ΔI for CDE and HPE suggests that the ΔI feature alone cannot significantly improve predictive performance; it should be associated with the step changes of power signals (ΔP and ΔQ). In other words, the ΔI feature might have a lower proportion of step changes than ΔP and ΔQ, which could make it less significant to the model capability.

3) MODEL PREDICTIVE PERFORMANCE

We compare the model's predictive performance between the optimal feature set and the baseline features. The evaluation is conducted through cross-validation (K=5) using the metrics of F-score, Precision, and Recall for each appliance label as shown in Table 5.

Most appliance labels show better predictive performance with the optimal feature set than with the baseline feature, especially for WOE, which notably improved across all metrics (e.g., 0.56 to 0.70 for F-score). The additive steady-state change features for this appliance might provide discriminative information that was previously missed by using the steady-state feature alone. The CWE and DWE gain moderate improvement, which indicates that the steady-state change features help capture subtle information of the cyclical operation events (e.g., washing, water heating), which complements the performance improvement. The CDE and HPE show marginal improvements, suggesting that the steady-state feature alone could make their operational signal clearer and more stable. The classification performance for these appliances already reaches a high level with the baseline features, leaving little room for improvement. For FGE, the steady-state change features are insignificant and are not included in the optimal feature set; therefore, the model performance value does not change.

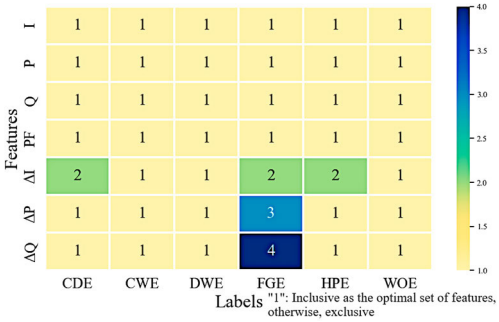

FIGURE 5. AMPds ranking score by RFECV feature selection.

TABLE 5. AMPds model predictive performance by the optimal features and the baseline features.

Label	Metric	Cross-validation performance	
		Optimal features (proposed method)	Baseline features
CDE	F-score	**0.980** ± 0.004	0.967 ± 0.004
	Precision	**0.983** ± 0.007	0.962 ± 0.004
	Recall	**0.975** ± 0.003	0.972 ± 0.006
CWE	F-score	**0.673** ± 0.009	0.625 ± 0.008
	Precision	**0.790** ± 0.015	0.782 ± 0.010
	Recall	**0.564** ± 0.010	0.515 ± 0.012
DWE	F-score	**0.492** ± 0.013	0.453 ± 0.021
	Precision	**0.680** ± 0.014	0.671 ± 0.017
	Recall	**0.392** ± 0.025	0.353 ± 0.015
FGE	F-score	0.764 ± 0.006	0.764 ± 0.006
	Precision	0.804 ± 0.006	0.804 ± 0.006
	Recall	**0.738** ± 0.006	0.736 ± 0.010
HPE	F-score	**0.989** ± 0.001	0.987 ± 0.001
	Precision	**0.987** ± 0.002	0.985 ± 0.001
	Recall	**0.993** ± 0.002	0.990 ± 0.002
WOE	F-score	**0.704** ± 0.020	0.563 ± 0.049
	Precision	**0.806** ± 0.025	0.740 ± 0.020
	Recall	**0.593** ± 0.035	0.447 ± 0.050

To track performance improvements from the enhancement features, we evaluate how the F-score values (Cross-validation (K=5)) improve as we add individual features in the order of the optimal set of features for each label. As shown in Fig. 6, the enhanced features can gradually improve the model performance when added individually. The performance value reaches the maximum when the number of added features meets the number of features in the optimal feature set. Therefore, the 6th and 7th performance data of CDE yield the same value. The result is also applied for HPE, since both have six features for the optimal set. For FGE, the 4th to the 7th performance data would also maintain

the same value. Another notice of the plot in Fig. 5 is that, for the CDE and WOE, adding the 3rd feature (Q) would significantly improve the model performance. This result is consistent with the feature importance values of these two labels (Table 3), with Q as the highest. It also signifies that the Q is the most impactful feature for these labels.

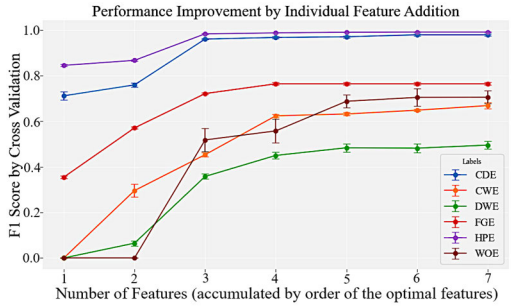

FIGURE 6. AMPds performance improvement by individual feature addition.

We conduct the performance benchmarking by comparing the use of the RFECV for feature selection with two other comparators, including 1) Forward Sequential Feature Selection (FSFS) [35] and 2) DAE neural network [3] as the learning framework. The FSFS is another feature selection technique that uses a greedy approach to add a feature at a time based on cross-validation performance. We use the same enhanced feature set as in previous experiments, but the results of selective features using the technique may differ. The DAE is a deep learning model that aims to construct the clean power signature of each appliance from the aggregate data, and it has shown to outperform the other network architectures [3]. The work implements and evaluates the model using NILMTK [36], a toolkit for experimentation of the NILM approach. We also use the XGBoost as the base classifier for RFECV and FSFS with default model setting (key model hyperparameters: maximum depth of a tree (*max_depth*): 3, number of boosting trees (*n_estimators*): 100). For the DAE, the network architecture mentioned in [3] is applied where the encoder and decoder part uses one convolutional layer for 1D data (Conv1D). The loss function and optimizer use 'mse' and 'adam', respectively, as this is a regression problem of predicting power consumption for each appliance. Table 6 presents the results of the performance benchmarking for the three approaches using F-score, Precision, and Recall indices via K-fold cross-validation (K =5).

The benchmarking results show that the predictive performance of the proposed approach outperforms competitors, particularly for high- and moderate-power appliance labels. For the FSFS, the number of selected features per appliance label is lower than that selected by RFECV, resulting in inferior performance. In addition, the RFECV can automatically choose the optimal number of features through iteratively removing the least important features from the full feature set. Whereas, the FSFS adds features one by one to improve the model performance. The FSFS mechanism can yield good results when features are independent, but it may miss benefits from feature interactions (e.g., adding ΔQ yields no improvement, whereas adding Q with ΔQ could yield enhancements). Thus, the global backward feature elimination of the RFECV, which evaluates the feature interactions, can provide an optimal set of features and better model performance than the FSFS. For the DAE, the model learns sets of sequential power data (P) rather than leveraging other features to identify loads, resulting in inferior performance for most appliance labels.

TABLE 6. AMPds performance benchmarking by different approaches.

Label	Cross-validation performance (F-score/ Precision/ Recall)		
	Proposed method	FSFS	DAE
CDE	**0.980** ± 0.004	0.970 ± 0.004	0.792 ± 0.005
	0.983 ± 0.007	0.974 ± 0.006	0.833 ± 0.011
	0.975 ± 0.003	0.962 ± 0.003	0.782 ± 0.012
CWE	**0.673** ± 0.009	0.588 ± 0.015	0.525 ± 0.008
	0.790 ± 0.015	0.654 ± 0.012	0.604 ± 0.010
	0.564 ± 0.010	0.495 ± 0.015	0.472 ± 0.018
DWE	0.492 ± 0.013	0.375 ± 0.015	**0.573** ± 0.012
	0.680 ± 0.014	0.554 ± 0.012	0.604 ± 0.021
	0.392 ± 0.025	0.369 ± 0.018	**0.412** ± 0.014
FGE	0.764 ± 0.006	0.731 ± 0.004	**0.787** ± 0.015
	0.804 ± 0.006	0.764 ± 0.003	**0.812** ± 0.012
	0.738 ± 0.006	0.697 ± 0.004	0.725 ± 0.016
HPE	**0.989** ± 0.001	0.972 ± 0.001	0.882 ± 0.013
	0.987 ± 0.002	0.968 ± 0.012	0.875 ± 0.015
	0.993 ± 0.002	0.977 ± 0.018	0.886 ± 0.010
WOE	**0.704** ± 0.020	0.692± 0.025	0.536 ± 0.018
	0.806 ± 0.025	0.778 ± 0.017	0.628 ± 0.022
	0.593 ± 0.035	0.567 ± 0.020	0.428 ± 0.016

4) MODEL COMPLEXITY

Although integrating more features could enhance predictive performance, the technique involves an additional process of RFECV feature selection and the addition of new features. These enhancements typically increase the model complexity which can be expressed as O(k×C×T) where k, C, and T are the number of features, the number of CV folds, and the training cost per sub-model, respectively. We evaluate the model complexity through model training time that integrates the RFECV process into the training pipeline, and the average number of tree nodes across the 100 trees (the default number). Table 7 shows the comparison of model complexity between the baseline features and the optimal feature set. The additional features do not significantly affect the average number of nodes per tree since the derivative features are usually less significant than the steady-state features. Therefore, the derivatives may not provide incremental gain and are not chosen for node splits. The number of nodes for the approach thus does not significantly change compared to the use of baseline features.

Although the proposed system increases initial computational time for training, practical NILM applications typically separate offline training and feature selection from real-time load identification. Therefore, it remains acceptable for real-world deployment by balancing complexity and performance based on appliance-specific benefits.

TABLE 7. AMPds model complexity between the baseline features and the optimal features.

Label	Baseline model complexity		Optimal model complexity	
	Train.time(s)	Avg.#nodes	Train.time(s)	Avg.#nodes
CDE	0.59	31.8	11.42	31.7
CWE	0.44	69.9	11.04	78.8
DWE	0.40	60.8	10.64	74.4
FGE	0.32	88.6	16.70	88.6
HPE	0.35	82.5	23.16	85.6
WOE	0.27	37.4	22.36	44.2

To quantify the effectiveness of the enhanced features relative to model complexity, we evaluate the model complexity index, defined as the increase in training time ($\Delta T/T$) per unit increase in F-score ($\Delta F/F$). The ΔT and ΔF refer to the increment of training time and F-score performance gain from the baseline features to the enhanced features, respectively. The T and F refer to the training time and F-score value of the baseline feature set, respectively. The model complexity index for each appliance label is presented in Table 8. High-power appliances, such as clothes dryer (CDE) and wall oven (WOE), tend to have lower model complexity indices than low-power appliances. Those appliances typically achieve better predictive performance, which could make a higher value of the term $\Delta F/F$, resulting in a lower model complexity index. The index value could help determine the cost-effective appliance type for monitoring, especially if training resources are a constraint.

TABLE 8. AMPds model complexity index.

Label	$\Delta T/T$	$\Delta F/F$	Model complexity index $(\Delta T/T)/(\Delta F/F)$ $(\times 10^3)$
CDE	18.35	0.013	1.36
CWE	24.09	0.077	0.31
DWE	25.60	0.086	0.30
FGE	51.18	0.002	19.50
HPE	65.17	0.002	32.16
WOE	81.81	0.250	0.33

5) SENSITIVITY ANALYSIS

To examine sensitivity analysis [34] or to evaluate how well the proposed approach generalizes across variations in the measurement setup, we experiment on two scenarios. 1) Varying data sampling frequency: Resample the original steady-state data from 1 minute to 5 minutes and 10 minutes. 2) Varying data noisy level: Add Gaussian noise to the original data (I, P, and Q) for 1% and 5%. Those two scenarios are also conducted by the cross-validation method. The comparative F-score results for the proposed method (enhanced features) and the baseline features method are presented in Table 9 and 10, respectively.

TABLE 9. AMPds sensitivity analysis on 5-minute and 10-minute data sampling frequency.

Label	5-min. sampling frequency		10-min. sampling frequency	
	Proposed method	Baseline features	Proposed method	Baseline features
CDE	**0.943** ± 0.015	0.894 ± 0.017	**0.887** ± 0.038	0.847 ± 0.021
CWE	**0.447** ± 0.038	0.373 ± 0.019	**0.397** ± 0.051	0.325 ± 0.067
DWE	**0.269** ± 0.055	0.256 ± 0.057	**0.222** ± 0.039	0.211 ± 0.067
FGE	**0.728** ± 0.005	0.727 ± 0.007	**0.778** ± 0.006	0.776 ± 0.006
HPE	**0.962** ± 0.006	0.926 ± 0.012	**0.932** ± 0.006	0.893 ± 0.010
WOE	**0.602** ± 0.116	0.491 ± 0.134	**0.498** ± 0.091	0.445 ± 0.165

TABLE 10. AMPds sensitivity analysis on 1% and 5% noisy level data.

Label	1% noisy level		5% noisy level	
	Proposed method	Baseline features	Proposed method	Baseline features
CDE	**0.979** ± 0.004	0.966 ± 0.006	**0.977** ± 0.005	0.962 ± 0.006
CWE	**0.653** ± 0.012	0.607 ± 0.004	**0.597** ± 0.010	0.565 ± 0.006
DWE	**0.457** ± 0.014	0.436 ± 0.013	**0.360** ± 0.020	0.358 ± 0.022
FGE	**0.754** ± 0.004	0.754 ± 0.004	**0.686** ± 0.002	0.679 ± 0.002
HPE	**0.976** ± 0.002	0.974 ± 0.003	**0.972** ± 0.002	0.970 ± 0.001
WOE	**0.697** ± 0.022	0.547 ± 0.056	**0.712** ± 0.028	0.544 ± 0.030

The results of the sensitivity analysis show that the proposed approach, which uses optimal feature sets, can achieve better model performance than the baseline features across all appliance labels. The coarser the sampling frequency and the noisier the level, the worse the data quality, resulting in lower performance values. For the case of sampling frequency variation, even though the change of time interval between samples, the differentiated feature set (ΔI, ΔP, and ΔQ) could still be informative for both 5-minute and 10-minute sampling frequencies. However, the benefit of the enhanced features tends to have less effect at higher time intervals (>10-minute frequency) because many power events and data details are diminished. For the case of noisy data, the added noise is applied to all samples, which makes the differentiation between consecutive data or the enhanced features almost unchanged. Thus, the added feature set would still improve the model performance. However, the noise level should be within a certain limit due to the variation in sampling frequency. At higher noisy levels, the values of the enhanced feature set can also vary greatly, making it difficult for the model to capture the useful signatures of appliances.

In addition, we evaluate the stress test or robustness of the approach when applied to a train-test data distribution shift, where variations of data measurement and appliance

usage behavior may occur. We split the original data into training and test sets and simulate the test set under two test cases. 1) Test data with a deviated sampling frequency from the training set. 2) Test data with simulated concurrency of appliances (a combination of appliances being turned ON at a given period). For the second case, we define a set of appliance concurrency levels ranging from 1 to 3, with a total of 16 combinations and 200 samples each. The simulated data uses the average value of the original I, P, and Q of each appliance and derives the PF (for AMPds data). The average-power representations are sufficient for the approach to capture steady-state-level features and allow controlling the variation of concurrent conditions, which is difficult to achieve with real data. The performance results, as measured by F-score, of the first case are presented in Table 11, and for the second case in Table 12, with benchmarking against the DAE approach.

TABLE 11. AMPds on simulated test data with deviated sampling frequency.

Label	3 minutes		5 minutes	
	Proposed method	Baseline features	Proposed method	Baseline features
CDE	**0.892**	0.865	**0.791**	0.739
CWE	**0.505**	0.450	**0.409**	0.364
DWE	**0.336**	0.312	**0.269**	0.252
FGE	0.548	**0.575**	0.548	**0.563**
HPE	**0.905**	0.885	**0.861**	0.856
WOE	**0.550**	0.468	**0.215**	0.201

TABLE 12. AMPds on simulated test data with concurrency of appliances.

Label	Proposed method	Baseline features	DAE
CDE	0.498	**0.507**	0.395
CWE	0.208	**0.235**	0.202
DWE	**0.258**	0.105	0.105
FGE	**0.430**	0.239	0.430
HPE	**0.537**	0.515	0.352
WOE	**0.276**	0.110	0.205

Results of the train-test deployment with synthetic test data show that the greater the shift of test data distribution from the training data, the greater the drop in predictive performance, but this is normal in the machine learning paradigm. However, the approach of using the enhanced features could still yield better model performance than using the baseline features for most appliances. For the first test case, at a more deviated data resolution (larger time interval), the enhanced features tend not to provide a benefit or even worsen the model performance due to the large discrepancy between the training and test data. For the second test case, the synthetic test data are constructed from the average I, P, and Q data with simulated combinations of activated appliances. The recombination of test data also creates large discrepancies with the training data, leading to a worse model performance than in the first test case. However, the proposed approach still delivers better performance than the DAE approach which only uses the power data (P) as a feature. The limitation of the evaluation is that the simulation isolates the factor of data distribution shifts, whereas real deployments may experience multiple shifts simultaneously (e.g., low sampling + high concurrency, high noise + high sampling). Therefore, the simulation results are interpreted as a controlled stress test, which motivates awareness of data distribution shifts in real applications.

B. ECO DATASET

The baseline steady-state feature set of the ECO dataset is the [I, P, Q]. With the additional steady-state change features, we present the key results using the same experimental procedure as in the previous experiment.

1) FEATURE IMPORTANCE ANALYSIS

We obtain the feature importance values for each label, shown in Table 13. The result aligns with the AMPds outputs, where the steady-state features provide significant information for the model. The steady-state changes are complementary features for enhancing the model performance.

TABLE 13. ECO feature importance values for each label.

Label	Feature					
	I	P	Q	ΔI	ΔP	ΔQ
Dishwasher	0.159	0.473	0.211	0.050	0.056	0.051
Fridge	0.505	0.209	0.145	0.049	0.042	0.050
Freezer	0.163	0.306	0.435	0.035	0.027	0.033
Kettle	0.164	0.578	0.061	0.057	0.100	0.040

2) FEATURE SELECTION BY RFECV

The ranking score matrix presented in Fig. 7 shows that the optimal feature set for Dishwasher and Kettle are [I, P, Q, ΔP, ΔQ] and [I, P, Q, ΔI, ΔP], respectively. This result suggests that the Dishwasher and Kettle, which are high-power appliances (with clear steady-state change features), do not require all steady-state change features since some of them can already provide unique and distinct signatures for optimal performance. The Fridge and Freezer are low-power appliances (with less clear steady-state change features) and require all parameters for the optimal feature set. The result suggests that these labels require more information (the steady-state features alone may not be sufficient) for the model to achieve optimal predictive performance. This scenario aligns with the results of the AMPds dataset, where high-power loads do not require all steady-state change features to achieve the optimal performance.

3) MODEL PREDICTIVE PERFORMANCE

The comparison of model performance (F-score, Precision, and Recall) for the optimal feature set and the baseline feature set using K-fold cross-validation ($K = 5$) is shown in Table 14. The optimal feature set, which combines the steady-state change features, provides a marginal performance improvement over the baseline steady-state features. The relatively small performance gains compared to the AMPds dataset could result from a couple of reasons. 1) Characteristics and behavior of appliances are different, which might affect the significance of the steady-state change features. 2) Data resampling might cause mismatches in data alignment, leading to notable errors in computing the difference between consecutive data or the steady-state change values.

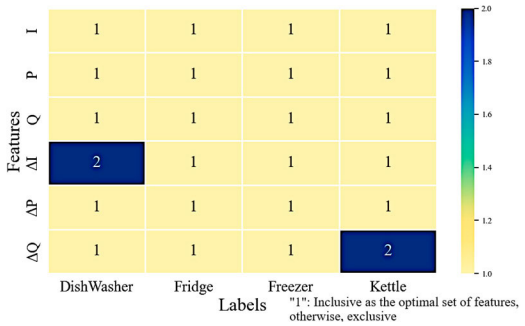

FIGURE 7. ECO ranking score by RFECV feature selection.

TABLE 14. ECO model predictive performance by the optimal features and the baseline features.

Label	Metric	Cross-validation performance	
		Optimal features (proposed method)	Baseline features
Dishwasher	F-score	**0.749** ± 0.013	0.747 ± 0.014
	Precision	**0.857** ± 0.016	0.856 ± 0.016
	Recall	**0.664** ± 0.010	0.662 ± 0.016
Fridge	F-score	**0.930** ± 0.002	0.918 ± 0.001
	Precision	**0.932** ± 0.002	0.922 ± 0.002
	Recall	**0.928** ± 0.003	0.914 ± 0.003
Freezer	F-score	**0.965** ± 0.001	0.960 ± 0.001
	Precision	**0.954** ± 0.002	0.950 ± 0.008
	Recall	**0.978** ± 0.002	0.970 ± 0.001
Kettle	F-score	**0.577** ± 0.044	0.562 ± 0.030
	Precision	0.697 ± 0.102	0.697 ± 0.102
	Recall	**0.503** ± 0.032	0.452 ± 0.022

Fig. 8 presents performance tracking against the addition of individual features are added. The first three features are the steady-state features (I, P, and Q), and the last three features are the steady-state change features. According to the optimal feature set for each label, the Dishwasher and Kettle achieve the optimal model performance with five features (excluding ΔI and ΔQ, respectively).

The performance benchmarking through F-score, Precision, and Recall using the proposed method, FSFS feature selection, and DAE learning framework produces the results shown in Table 15. The results suggest that the model performance of the proposed approach still outperforms the other comparators for most appliance labels. However, the FSFS approach extracts the optimal features of only the steady-state features ($[I, P, Q]$) for all labels, yielding inferior performance.

4) MODEL COMPLEXITY

The computational time during the training phase and the average number of tree nodes constructed for the two feature sets are presented in Table 16, yielding results similar to those for the AMPds data. The timing difference between the two feature sets depends on the number of samples, features, and variations within each dataset. However, as mentioned previously, the increase in training time due to the additional RFECV process would occur once (or infrequently). It would not affect the load inference in practical online applications. The model complexity index for each appliance is presented in Table 17, where the high-power appliance, such as a kettle, has a low model complexity index, which is a similar result to that obtained from the AMPds dataset.

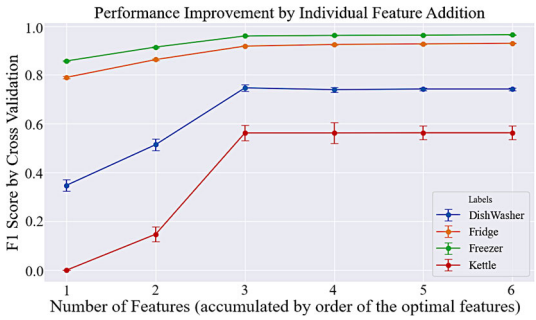

FIGURE 8. ECO performance improvement by individual feature addition.

TABLE 15. ECO performance benchmarking by different approaches.

Label	Cross-validation performance (F-score/ Precision/ Recall)		
	Proposed method	FSFS	DAE
Dishwasher	**0.749** ± 0.013	0.747 ± 0.014	0.722 ± 0.011
	0.857 ± 0.016	0.851 ± 0.012	0.833 ± 0.010
	0.670 ± 0.010	0.660 ± 0.008	0.635 ± 0.008
Fridge	**0.930** ± 0.002	0.918 ± 0.001	0.805 ± 0.012
	0.932 ± 0.002	0.922 ± 0.002	0.816 ± 0.014
	0.928 ± 0.003	0.898 ± 0.004	0.792 ± 0.015
Freezer	**0.965** ± 0.001	0.960 ± 0.001	0.852 ± 0.005
	0.954 ± 0.002	0.948 ± 0.003	0.840 ± 0.006
	0.978 ± 0.002	0.973 ± 0.005	0.862 ± 0.003
Kettle	0.577 ± 0.044	0.562 ± 0.032	**0.602** ± 0.020
	0.697 ± 0.102	0.655 ± 0.022	0.697 ± 0.027
	0.503 ± 0.032	0.513 ± 0.018	**0.515** ± 0.030

5) SENSITIVITY ANALYSIS

The two sensitivity evaluations of (1) adjusting data sampling frequency and (2) adding noise to data are examined for the robustness of the proposed approach. Table 18 and Table 19 show the performance results using F-scores of the two evaluations, where the proposed approach of adding enhanced features can also achieve better model performance than using the baseline features.

TABLE 16. ECO model complexity between the baseline features and the optimal features.

Label	Baseline model complexity		Optimal model complexity	
	Train.time(s)	Avg.#nodes	Train.time(s)	Avg.#nodes
Dishwasher	0.17	52.7	6.88	88.6
Fridge	0.18	89.7	6.78	99.8
Freezer	0.18	84.6	6.24	93.1
Kettle	0.24	39.4	5.88	48.6

TABLE 17. ECO model complexity index.

Label	ΔT/T	ΔF/F	Model complexity index $(\Delta T/T)/(\Delta F/F)$ $(\times 10^3)$
Dishwasher	39.471	0.003	14.742
Fridge	36.667	0.013	2.805
Freezer	33.667	0.005	6.464
Kettle	23.500	0.027	0.880

TABLE 18. ECO sensitivity analysis on 5-minute and 10-minute data sampling frequency.

Label	5-min. sampling frequency		10-min. sampling frequency	
	Proposed method	Baseline features	Proposed method	Baseline features
Dishwasher	**0.576** ± 0.023	0.570 ± 0.025	**0.604** ± 0.053	0.567 ± 0.065
Fridge	**0.890** ± 0.005	0.870 ± 0.003	**0.880** ± 0.006	0.863 ± 0.009
Freezer	**0.951** ± 0.002	0.944 ± 0.003	**0.962** ± 0.002	0.959 ± 0.002
Kettle	**0.381** ± 0.052	0.351 ± 0.015	**0.407** ± 0.085	0.367 ± 0.050

TABLE 19. ECO sensitivity analysis on 1% and 5% noisy level data.

Label	1% noisy level		5% noisy level	
	Proposed method	Baseline features	Proposed method	Baseline features
Dishwasher	**0.729** ± 0.008	0.717 ± 0.012	**0.661** ± 0.016	0.636 ± 0.018
Fridge	**0.921** ± 0.002	0.908 ± 0.002	**0.875** ± 0.003	0.852 ± 0.003
Freezer	**0.961** ± 0.001	0.956 ± 0.001	**0.940** ± 0.001	0.931 ± 0.001
Kettle	**0.580** ± 0.039	0.570 ± 0.031	**0.572** ± 0.030	0.553 ± 0.026

For the variable train-test deployment with simulated test data, two cases are conducted: (1) simulated test data with a deviated sampling frequency and (2) simulated test data with varied appliance concurrency. The results are presented in Table 20 and Table 21, respectively. Most appliances still gain some benefits from the proposed enhanced features approach, while some appliances (moderate-power consumption) might fluctuate and give worse performance than using the baseline features.

TABLE 20. ECO on simulated test data with deviated sampling frequency.

Label	3 minutes		5 minutes	
	Proposed method	Baseline features	Proposed method	Baseline features
Dishwasher	**0.670**	0.632	**0.535**	0.510
Fridge	**0.755**	0.737	**0.702**	0.685
Freezer	0.855	**0.865**	0.790	**0.795**
Kettle	**0.445**	0.430	**0.361**	0.348

TABLE 21. ECO on simulated test data with concurrency of appliances.

Label	Proposed method	Baseline features	DAE
Dishwasher	**0.518**	0.507	0.435
Fridge	0.218	**0.235**	0.220
Freezer	**0.158**	0.120	0.134
Kettle	**0.430**	0.239	0.200

The key findings of these experiments can be summarized as follows:

1. The high-power appliances (clothes dryer, wall oven) from AMPds data gain a substantial performance improvement by the additional selective steady-state change features (distinct step-change values). However, the heat pump (AMPds), dishwasher, and kettle (ECO) obtained a marginal performance improvement. For low-power appliances, the Fridge from AMPds does not require any steady-state changes to achieve optimal features, because their values are so low that they are not impactful on the model. However, the Fridge and Freezer of ECO data require all steady-state change features to achieve optimal performance, with only a marginal performance gain. The result suggests that the steady-state features alone may be sufficient for the optimal feature set of low-power appliances. In addition, these results suggest that the approach's effectiveness is appliance-specific and could depend on the dataset characteristics.

2. The feature selection by RFECV reveals that the steady-state features are more significant than the steady-state changes for both datasets. This result is evident in the selection of all steady-state features, while some steady-state change features are also selected as complements to construct the optimal feature set.

3. Although the addition of steady-state change features could enhance model performance, it significantly increases computational complexity in the training phase. In practical data training scenarios, selective and adaptive feature utilization strategies can optimize both model performance and computational efficiency.

4. Through the results of sensitivity analysis, the proposed enhanced feature set shows benefit for performance improvement in applications where there are variations in measurement setup. The stress test analyses show the degraded performance when large train-test data distribution

shifts are introduced, highlighting the need to account of the metering configuration when deploying the approach.

While feature interpretability and performance improvement are the key advantages of the proposed approach, there are some limitations or drawbacks when deploying the system in real applications. (1) The system is dependent on steady-state features and their differentiation values. It might work well for appliances with stable steady-state data (the differentiations are also stable), for example fridges, heaters, or kettles. However, other variable- or multi-state-power appliances (with differentiated features that are not stable), such as a washing machine or a dishwasher, might not get much benefit from the approach (as shown in the experimental results). To address this issue, additional timing features that can learn how the data evolves might be added, such as, a rolling mean of the steady-state or the period of variable data. The added features could enhance adaptive feature selection's ability to learn various appliance type from power data profiles (binary, multi-state, or variable-state). (2) The differentiated features can be noisy when used with low-power appliances like a lamp or electronic equipment because their feature values are low and fluctuate. Additionally, when the sampling frequency is relatively low (e.g., greater than 10 minutes), the approach may not deliver the expected performance boost. Thus, the measurement setup is a constraint that should be adhered to when applying the system in a real application.

VI. CONCLUSION

This paper investigates the effectiveness of integrating steady-state and steady-state change features derived from low-frequency metering data for NILM applications. The proposed approach utilizes a multi-label classification framework to learn the operating status (ON/OFF) of appliances. The RFECV feature selection technique can identify an optimal subset of features for each appliance, providing a model that outperforms the method using steady-state features alone and a comparative feature selection method, Forward Sequential Feature Selection (FSFS), across Precision, Recall, and F-score metrics. The experimental results show substantial performance gains for certain high-power appliances in the AMPds dataset (e.g., Wall Oven), as well as moderate and marginal improvements in the ECO dataset, depending on appliance operation and dataset characteristics. In addition, benchmarking against the DAE neural network shows that while the deep learning approach is superior at modeling appliances with more complex operations, the proposed feature-based method can achieve comparable or superior overall performance with greater interpretability of features and lower computational complexity.

Through sensitivity analysis, the proposed approach is applicable under a range of measurement setup variations. The experiments are conducted by changing data sampling frequency to 5 minutes and 10 minutes, creating data with noisy levels of 1% and 5%, as well as simulating test data of deviated sampling frequencies and concurrency of appliance operations. However, the approach can benefit from a limited range of these variations, as at higher levels, the loss of data resolution or data discrepancy is too high for the model to capture true signatures.

Future research directions should explore hybrid NILM frameworks that integrate the complementary strengths of deep learning and interpretable feature-based approaches. Additionally, adaptive strategies capable of automatically selecting the optimal features tailored to specific appliance types and operational behaviors would enhance practicality for real-world smart grid applications.

REFERENCES

[1] P. A. Schirmer and I. Mporas, "Non-intrusive load monitoring: A review," *IEEE Trans. Smart Grid*, vol. 14, no. 1, pp. 769–784, Jan. 2023, doi: 10.1109/TSG.2022.3189598.

[2] R. Bonfigli, A. Felicetti, E. Principi, M. Fagiani, S. Squartini, and F. Piazza, "Denoising autoencoders for non-intrusive load monitoring: Improvements and comparative evaluation," *Energy Buildings*, vol. 158, pp. 1461–1474, Jan. 2018, doi: 10.1016/j.enbuild.2017.11.054.

[3] J. Kelly and W. Knottenbelt, "Neural NILM: Deep neural networks applied to energy disaggregation," in *Proc. 2nd ACM Int. Conf. Embedded Syst. Energy-Efficient Built Environments*, Seoul, South Korea, Nov. 2015, pp. 55–64, doi: 10.1145/2821650.2821672.

[4] A. Pirnat, B. Bertalanič, G. Cerar, M. Mohorčič, and C. Fortuna, "Energy efficient deep multi-label ON/OFF classification of low frequency metered home appliances," *IEEE Access*, vol. 12, pp. 51966–51981, 2024, doi: 10.1109/ACCESS.2024.3382830.

[5] M. Ravinder and V. Kulkarni, "Optimizing household energy consumption prediction with recursive feature elimination and LSTM-keras," in *Proc. IEEE Int. Conf. Smart Power Control Renew. Energy (ICSPCRE)*, Rourkela, India, Jul. 2024, pp. 1–6, doi: 10.1109/ICSPCRE62303.2024.10675266.

[6] R. S. Mollel, L. Stankovic, and V. Stankovic, "Explainability-informed feature selection and performance prediction for nonintrusive load monitoring," *Sensors*, vol. 23, no. 10, p. 4845, May 2023, doi: 10.3390/s23104845.

[7] A. Moradzadeh, O. Sadeghian, K. Pourhossein, B. Mohammadi-Ivatloo, and A. Anvari-Moghaddam, "Improving residential load disaggregation for sustainable development of energy via principal component analysis," *Sustainability*, vol. 12, no. 8, pp. 1–14, Apr. 2020, doi: 10.3390/su12083158.

[8] A. Yaniv and Y. Beck, "Enhancing NILM classification via robust principal component analysis dimension reduction," *Heliyon*, vol. 10, no. 9, May 2024, Art. no. e30607, doi: 10.1016/j.heliyon.2024.e30607.

[9] S. Matharaarachchi, M. Domaratzki, and S. Muthukumarana, "Assessing feature selection method performance with class imbalance data," *Mach. Learn. Appl.*, vol. 6, pp. 1–18, Dec. 2021, doi: 10.1016/j.mlwa.2021.100170.

[10] A. Z. Mustaqim, S. Adi, Y. Pristyanto, and Y. Astuti, "The effect of recursive feature elimination with cross-validation (RFECV) feature selection algorithm toward classifier performance on credit card fraud detection," in *Proc. Int. Conf. Artif. Intell. Comput. Sci. Technol. (ICAICST)*, Yogyakarta, Indonesia, Jun. 2021, pp. 270–275, doi: 10.1109/ICAICST53116.2021.9497842.

[11] I. Abubakar, S. N. Khalid, M. W. Mustafa, H. Shareef, and M. Mustapha, "Application of load monitoring in appliances' energy management—A review," *Renew. Sustain. Energy Rev.*, vol. 67, pp. 235–245, Jan. 2017, doi: 10.1016/j.rser.2016.09.064.

[12] P. A. Schirmer and I. Mporas, "Double Fourier integral analysis based convolutional neural network regression for high-frequency energy disaggregation," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 6, no. 3, pp. 439–449, Jun. 2022, doi: 10.1109/TETCI.2021.3086226.

[13] L. Yan, R. Xu, M. Sheikholeslami, Y. Li, and Z. Li, "State identification of home appliance with transient features in residential buildings," *Frontiers Energy*, vol. 16, no. 1, pp. 130–143, Feb. 2022, doi: 10.1007/s11708-022-0822-z.

[14] S. Liu, Z. Xie, and Z. Hu, "Optimizing smart home appliance energy monitoring using factorial hidden Markov models for Internet of Behavior," *J. Building Eng.*, vol. 97, pp. 1–13, Nov. 2024, doi: 10.1016/j.jobe.2024.110732.

[15] P. Huber, A. Calatroni, A. Rumsch, and A. Paice, "Review on deep neural networks applied to low-frequency NILM," *Energies*, vol. 14, no. 9, p. 2390, Apr. 2021, doi: 10.3390/en14092390.

[16] R. Gopinath, M. Kumar, C. Prakash Chandra Joshua, and K. Srinivas, "Energy management using non-intrusive load monitoring techniques–state-of-the-art and future research directions," *Sustain. Cities Soc.*, vol. 62, pp. 1–19, Nov. 2020, doi: 10.1016/j.scs.2020.102411.

[17] L. Deng, C. Pang, X. Zeng, J. Zhang, and C. Huang, "Real-time energy disaggregation algorithm based on multi-channels DCNN and autoregressive model," *IEEE Access*, vol. 10, pp. 110835–110848, 2022, doi: 10.1109/ACCESS.2022.3211427.

[18] H. Hwang and S. Kang, "Nonintrusive load monitoring using an LSTM with feedback structure," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–11, 2022, doi: 10.1109/TIM.2022.3169536.

[19] M. Ayub and E.-S. M. El-Alfy, "Contextual sequence-to-point deep learning for household energy disaggregation," *IEEE Access*, vol. 11, pp. 75599–75616, 2023, doi: 10.1109/ACCESS.2023.3297552.

[20] D. García-Pérez, D. Pérez-López, I. Díaz-Blanco, A. González-Muñiz, M. Domínguez-González, and A. A. Cuadrado Vega, "Fully-convolutional denoising auto-encoders for NILM in large non-residential buildings," *IEEE Trans. Smart Grid*, vol. 12, no. 3, pp. 2722–2731, May 2021, doi: 10.1109/TSG.2020.3047712.

[21] G. F. Angelis, C. Timplalexis, A. I. Salamanis, S. Krinidis, D. Ioannidis, D. Kehagias, and D. Tzovaras, "Energformer: A new transformer model for energy disaggregation," *IEEE Trans. Consum. Electron.*, vol. 69, no. 3, pp. 308–320, Aug. 2023, doi: 10.1109/TCE.2023.3237862.

[22] Y. Xuan, C. Pang, H. Yu, X. Zeng, and Y. Chen, "Load energy decomposition algorithm based on improved bidirectional transformer combined with time-sensing self-attention," *IEEE Access*, vol. 12, pp. 75625–75639, 2024, doi: 10.1109/ACCESS.2024.3373801.

[23] D. Murray, L. Stankovic, and V. Stankovic, "Explainable NILM networks," in *Proc. 5th Int. Workshop Non-Intrusive Load Monitor.*, Virtual Event, Japan, Nov. 2020, pp. 64–69, doi: 10.1145/3427771.3427855.

[24] K. Basu, V. Debusschere, S. Bacha, U. Maulik, and S. Bondyopadhyay, "Nonintrusive load monitoring: A temporal multilabel classification approach," *IEEE Trans. Ind. Informat.*, vol. 11, no. 1, pp. 262–270, Feb. 2015, doi: 10.1109/TII.2014.2361288.

[25] B. Buddhahai, S. K. Korkua, P. Rakkwamsuk, and S. Makonin, "A design and comparative analysis of a home energy disaggregation system based on a multi-target learning framework," *Buildings*, vol. 13, no. 4, pp. 1–16, Mar. 2023, doi: 10.3390/buildings13040911.

[26] B. Buddhahai and S. Makonin, "A nonintrusive load monitoring based on multi-target regression approach," *IEEE Access*, vol. 9, pp. 163033–163042, 2021, doi: 10.1109/ACCESS.2021.3133292.

[27] J. Bogatinovski, L. Todorovski, S. Džeroski, and D. Kocev, "Comprehensive comparative study of multi-label classification methods," *Expert Syst. Appl.*, vol. 203, pp. 1–18, Oct. 2022, doi: 10.1016/j.eswa.2022.117215.

[28] M. Awad and S. Fraihat, "Recursive feature elimination with cross-validation with decision tree: Feature selection method for machine learning-based intrusion detection systems," *J. Sensor Actuator Netw.*, vol. 12, no. 5, p. 67, Sep. 2023, doi: 10.3390/jsan12050067.

[29] S. Makonin, B. Ellert, I. V. Bajić, and F. Popowich, "Electricity, water, and natural gas consumption of a residential house in Canada from 2012 to 2014," *Scientific Data*, vol. 3, no. 1, pp. 1–12, Jun. 2016, doi: 10.1038/sdata.2016.37.

[30] C. Beckel, W. Kleiminger, R. Cicchetti, T. Staake, and S. Santini, "The ECO data set and the performance of non-intrusive load monitoring algorithms," in *Proc. 1st ACM Conf. Embedded Syst. Energy-Efficient Buildings*, Nov. 2014, pp. 80–89, doi: 10.1145/2674061.2674064.

[31] B. Buddhahai, W. Wongseree, and P. Rakkwamsuk, "A non-intrusive load monitoring system using multi-label classification approach," *Sustain. Cities Soc.*, vol. 39, pp. 621–630, May 2018, doi: 10.1016/j.scs.2018.02.002.

[32] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, CA, CA, USA, Aug. 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[33] P. Szymanski and T. Kajdanowicz, "Scikit-multilearn: A scikit-based Python environment for performing multi-label classification," *J. Mach. Learn. Res.*, pp. 209–230, Jan. 2020, doi: 10.5281/zenodo.3670934.

[34] H. Liu, Q. Zou, and Z. Zhang, "Energy disaggregation of appliances consumptions using HAM approach," *IEEE Access*, vol. 7, pp. 185977–185990, 2019, doi: 10.1109/ACCESS.2019.2960465.

[35] S. Shafiee, L. M. Lied, I. Burud, J. A. Dieseth, M. Alsheikh, and M. Lillemo, "Sequential forward selection and support vector regression in comparison to LASSO regression for spring wheat yield prediction based on UAV imagery," *Comput. Electron. Agricult.*, vol. 183, pp. 1–9, Apr. 2021, doi: 10.1016/j.compag.2021.106036.

[36] N. Batra, J. Kelly, O. Parson, H. Dutta, W. Knottenbelt, A. Rogers, A. Singh, and M. Srivastava, "NILMTK: An open source toolkit for non-intrusive load monitoring," in *Proc. 5th Int. Conf. Future energy Syst.*, Jun. 2014, pp. 265–276, doi: 10.1145/2602044.2602051.

BUNDIT BUDDHAHAI received the B.E. degree in electronics engineering from the King Mongkut's Institute of Technology Ladkrabang and the M.E. degree in electrical and information engineering and the Ph.D. degree in energy management technology from the King Mongkut's University of Technology Thonburi. He is currently a Lecturer with the School of Informatics, Walailak University, Thailand. His research interests include energy monitoring, applied machine learning, and smart embedded systems.

TRAN ANH TUAN received the B.Sc. degree in information technology from the Hue University of Education, the M.Sc. degree in information systems from the HCMC University of Sciences, Vietnam, and the Ph.D. degree in computer science from the Prince of Songkla University, Thailand. He is currently a Lecturer with the School of Informatics, Walailak University, Thailand. His research interests include natural language processing, machine learning, and data science.

WARANYU WONGSEREE received the B.E., M.E., and Ph.D. degrees in electrical engineering from the King Mongkut's University of Technology North Bangkok, Thailand. He is currently a Lecturer with the Faculty of Digital Technology, Chitralada Technology Institute, Thailand. His research interests include applied machine learning, bioinformatics, and home energy monitoring.

STEPHEN MAKONIN (Senior Member, IEEE) is currently an Adjunct Professor of engineering science and the Principal Investigator of the Computational Sustainability Laboratory, Simon Fraser University. His research interests include computational sustainability and the understanding of socioeconomic issues. He serves on the Steering Committee of IEEE DataPort and is the Founding Editor-in-Chief of IEEE Data Descriptions.