IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 1 June 2024, accepted 19 June 2024, date of publication 24 June 2024, date of current version 2 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3418499

RESEARCH ARTICLE

Evolving Feature Selection: Synergistic Backward and Forward Deletion Method Utilizing Global Feature Importance

TAKAFUMI NAKANISHI[1], (Member, IEEE), PONLAWAT CHOPHUK[2], (Member, IEEE), AND KRISANA CHINNASARN[2], (Senior Member, IEEE)

[1]Department of Data Science, Musashino University, Tokyo 135-8181, Japan
[2]Faculty of Informatics, Burapha University, Chonburi 20131, Thailand

Corresponding authors: Ponlawat Chophuk (ponlawat.ch@go.buu.ac.th) and Krisana Chinnasarn (krisana@buu.ac.th)

This work was supported in part by the Faculty of Informatics, Burapha University, Chonburi, Thailand; and in part by the Department of Data Science, Musashino University, Tokyo, Japan.

ABSTRACT Explainable artificial intelligence (XAI) techniques are used to understand the rationale behind the decision-making of machine learning models. In addition to the need for model explainability, the demand for an ever-growing number of multimodal features has dramatically increased model complexity. This underscores the importance of precise feature selection to ensure high model accuracy. Using our Approximate Inverse Model Explanations (AIME) technique, which currently presents the best XAI capability in the field, this study incorporated a novel backward and forward deletion process. This pre-assesses global feature importance by calculating and ordering their AIME-reported global importance. Through the backward deletion process, it assesses model accuracy by progressively eliminating less important features, resulting in a feature set configuration that guarantees the highest model accuracy. Then, the forward deletion process further refines the feature set by discarding the least important features until the he model's accuracy declines, which reduces the computational burden and ensures optimal performance. We applied our method to the detailed and expansive Multimodal Emotion Line dataset and leveraged 4,870 facial, voice, and spoken language features in the Google Colab Pro+ environment to demonstrate AIME's efficacy in enabling researchers to maximize both model explainability and performance: the holy grail of XAI.

INDEX TERMS Approximate inverse model explanations (AIME), backward and forward deletion, explainable artificial intelligence (XAI), feature selection, global feature importance.

I. INTRODUCTION

In the digital age, artificial intelligence (AI)-driven machine learning (ML) is pivotal in driving progress in many areas, such as automated driving, medical diagnostics, and financial management. Because ML predictions empower and/or replace modern human decision-making, the use of multi-modal data to address nonlinear problem domains has gained prominence owing to recent technological advancements. As such, there is an increasing need for explainable AI (XAI) methods ([1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20]) that allow users to understand the reasons behind otherwise opaque ML predictions. Explainability is vital in modeling dependable systems for public safety, and social wellbeing.

Our previously proposed approximate inverse model explanation (AIME) method [21] derives an approximate inverse operator that links black-box model predictions to underlying data features. Using the training dataset, this inverse operator is trained to reveal the local and global contributions of features by mathematically analyzing their impact on models' prediction performance. Specifically,

The associate editor coordinating the review of this manuscript and approving it for publication was Pasquale De Meo.

AIME utilizes linear algebra techniques to construct inverse approximation operators between a model's outputs and input features. This enables numerical assessment and visualization of the effects of individual features on prediction results. Furthermore, AIME calculates the contribution of each feature and ranks its importance accordingly. This provides important feature selection indicators that enable interpretability.

Various approaches have sought to improve model accuracy while reducing computer and temporal costs. To succeed, multimodal tasks require a method that can manage large data sources while ensuring interpretability and explainability. Our method provides strong interpretability, even in the presence of multicollinearity, and is more robust than traditional XAI techniques. Furthermore, the feature-importance information derived by AIME can be used to minimize computational costs by selecting only the most necessary features to maximize accuracy. This is crucial for efficient feature selection technologies, particularly for applications with large datasets and complex model structures. AIME draws back the curtain to expose the rationale underlying a model's behavior, allowing for transparent and targeted methods of improvement and efficiency. Moreover, it allows scientists to support AI findings with traceable explainability.

We further confirm this claim by adding a novel Synergistic Backward and Forward Deletion (SBFD) method that is applied to AIME's sorted list of feature importance. To identify and utilize significant features from a complex blend of multimodal data efficiently, AIME applies backward deletion, which progressively eliminates non-contributing features to maximize model accuracy. Forward deletion further refines this list by discarding the least-important remaining features until model accuracy is impacted. Then, it backs up one step. Consequently, the required ML computations are minimized, superior performance is ensured, and full XAI capability is retained.

SBFD advances our understanding of how multimodal features contribute to the performance of ML models through its systematic approach to feature selection and elimination. It addresses the increasing complexity and computational demands posed by large multimodal datasets prevalent in several up-and-coming fields, such as emotion recognition and social media analysis. The SBFD method substantiates the role of XAI in practical applications and sets a powerful precedent for future research in feature selection methodologies.

Using the thoroughly extensive Multimodal Emotion Line Dataset (MELD) [22], [23], we extracted 4,870 features (i.e., 23 facial, 88 speeches, and 4759 utterances) and assessed the global feature importance and accuracy of several ML models using our methods. Our study's key contributions are:

- In addition to AIME's previous superior XAI capability, it can now guide researchers to the most efficient feature selection configuration possible for maximum model accuracy while minimizing computational costs.
- We provide strong empirical evidence of the benefits and shortcomings of conventional ML methods and demonstrate their AIME-guided improvements in real-world tasks.

The remainder of this paper is organized as follows. Section II reviews the previous studies that led us to this advancement, and Section III presents an AIME overview. Section IV then explains the SBFD process, and Section V presents the experimental results of minimizing the number of computations. Section VI follows with a discussion of our interpretations and concludes the paper.

II. RELATED WORKS

This section reviews works we consider important in leading to our research design and its objectives. Speith [17] used ante and post hoc to categorize XAI. The former applies linear regression, decision trees, and k-nearest neighbor methods to introduce a transparent model that provides an ante-hoc XAI. In contrast, the latter method applies model-specific or -agnostic methods to provide XAI for complex and less transparent models.

Model-specific methods have high interpretability but limited applicability. For example, Grad-CAM [24] offers an effective XAI for convolutional neural networks using feature importance scores. Other model-specific interpretability methods [25], [26], [27], [28] have also been used for deep neural networks. Model-agnostic methods are categorized into three types. The first uses a targeted black-box model to understand its behavior by varying its input values and training data using partial dependency plots [29], [30] to visualize the feature impact on prediction results. Other model-agnostic methods use conditional explanations [31], such as sorted feature importance [32], leave-one-feature-out (LOFO) [33] [34], and important features in the forward direction, such as Local Interpretable Model-Agnostic Explanations (LIME) [35] and SHapley Additive exPlanations (SHAP) [36]. The third type solves black-box inverse problems similar to AIME [21]. Table 1 shows a comparison of the classification of the XAI methods.

Several studies on feature selection [37], [38], [39], [40] have been published. For example, Yu and Liu [41] showed that feature relevance alone is insufficient for assessing high-dimensional data. Hence, they proposed a correlation-based method for analyzing relevance and redundancy. Ambarwati and Guyon [42] proposed a method of eliminating features with low variance since they have low information content and do not contribute to improving model accuracy. Guyon and Elisseeff [43] proposed a univariate selection method that evaluates the statistical association between features and target variables and selects the most relevant among them. Tibshirani's Least Absolute Shrinkage and Selection Operator (LASSO) [44] feature-selection method employs L1 regularization to induce sparsity, effectively shrinking some coefficients to zero and thereby selecting relevant features. Battiti's method [45] measures the interdependence between features and target variables using mutual information

TABLE 1. Examples of various XAI methods.

Types of methods	Method description	Types of methods	Method description	Types of methods	Methods
Ante hoc method	Methods that include a mechanism for deriving explanations in the machine learning model in advance, i.e. glass-box models.	Linear models, logistic models, decision tree models, KNN models, etc.			
Post-hoc method	Methods for realizing a mechanism that derives an explanation after creating a machine learning model, i.e., a black box model.	Model-specific method	Explanatory methods that can only be applied to specific models.	Grad-CAM [24], etc.	
		Model-agnostic method	Explanatory methods that can be applied to any model.	Methods of deriving explanations using the machine learning model itself.	Partial dependency plots [30] [31] Individual conditional explanations [32], Permutation feature importance [33] Leave-one-feature-out (LOFO) [34] [35]
				Methods of deriving explanations using forward operations separate from machine learning models.	LIME [36] SHAP [37]
				Methods of deriving explanations using inverse operations separate from machine learning models.	AIME [21] (Our method)

content, and the method proposed by Xiao et al. [46] selects optimal features by aggregating votes from multiple models and feature selection methods.

Kursa and Rudnicki [47] proposed a method that evaluates feature importance using a Random Forest (RF) algorithm and selects those that are clearly more important than random noise. Haq et al. [48] proposed a feature clustering and selection framework that combines multiple feature-ranking methods.

Guyon et al. [49] proposed Recursive Feature Elimination (RFE), which iteratively removes the least important features from a model. Later, Freytes et al. [50] proposed an RFE with cross-validation (RFECV) to determine automatically the optimal number of features, and Liu and Sung [51] provided the Recursive Feature Addition (RFA) model, which adds features in the order of decreasing importance, selecting only those that improve model accuracy. Notably, RFE [49], RFECV [50], and RFA [51] tend to be computationally expensive, as they are evaluated individually for accuracy.

Several forward and backward feature-selection methods [44] have been proposed. For example, Kohavi and John [52] used a wrapper method to evaluate a subset of features with a specific learning algorithm as part of their evaluation function, which is specific to a particular model and selects a combination of features that maximizes its predictive performance. In contrast, SBFD uses XAI techniques and AIME to pre-assess the importance of global features. This differs from Borboudakis and Tsamardinos [53], who temporarily discarded variables that were conditionally independent of the selected variable set, depending on how these variables were reconsidered and reintroduced to realize the intended algorithmic performance. Siebers and Schmid [54] demonstrated similar methods that encountered limitations when using artificial and real-world datasets. Mao [55] proposed general Sequential Forward Selection (SFS), which represents features in an orthogonal space, where feature subset selection is performed by incorporating Gram–Schmidt and Givens orthogonal transformations into the general SFS and sequential backward elimination (SBE) routines. Notably, these routines apply relevance clustering (RC) to locally relevant features, which differ from instance to instance. Domingos [56] proposed a clustering-like approach to select a set of locally relevant features that also differ from instance to instance, and Kamalov et al. [57] applied a forward feature selection algorithm that evaluates each feature in a dataset sequentially and then, progressively selects the features that most improve the model's performance.

More recently, various additional methods have been proposed. For example, Liu et al. [58] applied a feature selection multi-agent reinforcement learning method, which treats each feature as an independent agent, and the state of the environment is represented by a statistical description, auto-encoder, and graph convolutional network to improve

TABLE 2. Examples of different methods of feature selection.

Types of methods	Method descriptions	Examples of the methods
Filter method	Methods of feature selection based on the association between a feature and an objective variable before the model is trained.	Feature selection methods [37][38][39] Unsupervised feature selection [40] Correlation-based feature selection [41] Variance threshold method [42] Univariate feature selection [43] Mutual information feature selection [44] Feature selection using RF [45] Clustering and selection framework [48] Forward-backward selection with early dropping [53] Interleaving forward-backward feature selection [54] Orthogonal feature selection [55] Control-sensitive feature selection [56] Forward feature selection [57] Multi-label feature selection [60] Feature selection for credit risk assessment [61]
Wrapper method	Methods of feature selection using specific machine learning algorithms to assess the impact of a subset of features on model performance, which is computationally expensive but effective in directly improving performance	Recursive feature elimination (RFE/RFECV) [49][50] Recursive feature addition (RFA)[51]
Embedding method	Methods where feature selection is built into the learning algorithm itself.	LASSO [44]
Metaheuristic method	Methods that use algorithms that mimic the behavior of nature to solve combinatorial optimization problems.	Hierarchical voting scheme [46] Multi-agent reinforcement learning for feature selection [58] Feature selection using an optimization algorithm [59]
Methods based on XAI	Methods for analyzing the importance and contribution of features using XAI techniques and feature selection based on this.	Synergistic backward and forward deletion method (SBFD, our proposed method)

efficiency and accuracy. This method accelerates feature selection by improving the reward system and extending the search strategies.

Hamad [59] presented a comprehensive review of the feature selection literature, summarizing strategies such as filters, wrappers, meta-heuristics, and embedding. Notably, nature-inspired algorithms (e.g., particle swarm, grey wolf, bats, genetic, whelk, and ant colony) were assessed. The study confirmed that feature selection approaches were indeed important in reducing ML model complexity. Moreover, these methods often improve simulation performance. Wang and Zhou [60] challenged several of the new perspectives designed to improve the performance of multi-label feature selection in ML data mining applications. Jemai and Zarrad [61] provided a review of credit risk assessment capabilities in the financial industry and proposed a new engineering direction that leverages univariate feature selection, which chooses features based on their relevancy to the target variable. Additionally, they examined Recursive Feature Elimination (RFE), a method that progressively removes the least important features based on the performance of the classifier. Similarly, Feature Importance Decision (FID) trees assess the influence of features on final model decisions, and the information value (IV) method assesses how much information a feature provides for the targeted prediction outcomes using a forward selection algorithm. Kamalov et al. [57] examined the efficiency and effectiveness of a simple forward selection algorithm in the context of linear regression, showing that it requires significantly less computation time than the all-search algorithm. Furthermore, it is versatile when facing heterogeneous datasets.

Our new AIME implementation overcomes the computational cost problems and other limitations of RFE [49], RFECV [50], and others while retaining a superior XAI capability. Table 2 presents a classification comparison of these feature selection methods.

III. AIME OVERVIEW

Here, we provide an overview of AIME, where Section III-A discusses the creation of the core approximate inverse operator, and Section III-B describes how AIME is used to obtain global feature importance.

A. DERIVING THE APPROXIMATE INVERSE OPERATOR

Fig. 1 presents an overview of how to derive an approximate inverse operator, where X is a matrix of training data of arbitrary dimensions, $m \times n$, and m is the number of features. n is the number of samples, and $\hat{Y}$ is a matrix of $k \times n$, where k is the number of classes. Y denotes the training data, and $\hat{Y}$ denotes the estimation results of the ML model for

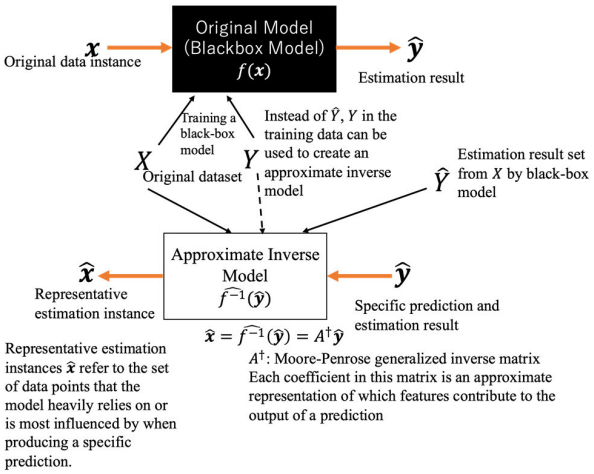

FIGURE 1. AIME used to construct approximate inverse operators [21].

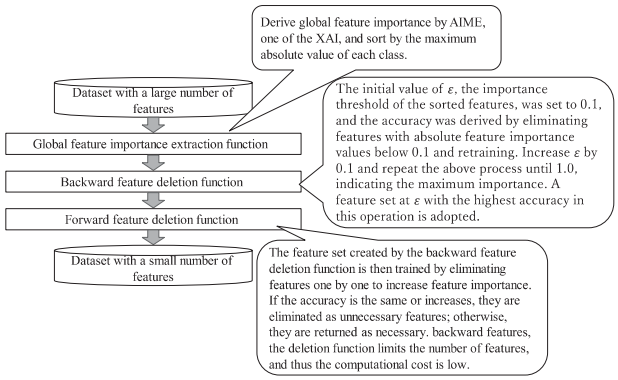

FIGURE 2. Overview of the proposed synergistic backward and forward deletion (SBFD) method.

X. Although it is typical to input $\hat{Y}$ to show the behavior of the ML model in XAI, as shown in Fig. 1, in this study, X and Y (X_train and Y_train) are given as inputs to obtain the pure feature importance of any ML model. Our objective is to extract only the importance of features. Hence, we can use Y as instead of $\hat{Y}$ to derive those that contribute the most.

By learning these data, a black-box model function, $f(\boldsymbol{x})$, is created, which outputs estimate $\boldsymbol{y}$ for input data instance $\boldsymbol{x}$. The X and Y matrices are then viewed as representing the behavior of $f(x)$, where each $\boldsymbol{x}$ must be an $m-dimensional$ vector. A dataset with a similar or resampled distribution to X can be used. X and Y are used to generate the approximate inverse operator, $A^{\dagger}$, of the black-box model, expressed by formulae (1), (2), (3), (4), and (5) in [21]:

$$X = A^{\dagger}Y, \tag{1}$$

$$X\hat{Y}^{T} = A^{\dagger}YY^{T}, \tag{2}$$

$$XY^{T}\left(YY^{T}\right)^{-1} = A^{\dagger}\left(YY^{T}\right)(YY^{T})^{-1}, \tag{3}$$

$$A^{\dagger} = XY\left(YY^{T}\right)^{-1} = XY^{\dagger}, \tag{4}$$

$$A^{\dagger} = XY^{T}\left(YY^{T}\right)^{-1} = XY^{\dagger}, \tag{5}$$

where Y^{T} is the transpose matrix of Y, and $Y^{\dagger}$ is its Moore–Penrose generalized inverse [62], [63]. $A^{\dagger}$ can be used to obtain an approximation $\hat{\boldsymbol{x}}$ of the original data $\boldsymbol{x}$ using $A^{\dagger}\boldsymbol{y}$.

B. GLOBAL FEATURE IMPORTANCE

$A^{\dagger}$ is a matrix of $m \times k$, where m is the number of features, and k is the number of classes. The calculation is explained in Section IV-A. In this case, when the first column is obtained and sorted in the order of increasing absolute values, the top class can easily be identified. When the second column is sorted, the second class is identified, and so forth.

IV. METHOD

In this section, we describe our backward and forward feature deletion processes. Section IV-A presents an overview of the method, Section IV-B presents the global feature importance extraction function, Section IV-C presents the backward feature deletion function, and Section IV-D presents the forward feature deletion function.

A. BACKWARD–FORWARD OVERVIEW

Fig. 2 presents an overview of the proposed method in which matrix X of the explanatory variables and matrix Y of the objective variables are taken as input.

The method consists of a global feature-importance extraction function, a backward feature deletion function, and a forward feature deletion function. The output is matrix X' of the selected explanatory variables representing the selected feature groups.

Matrices X and Y are used to identify important features by using the AIME model, which derives the global feature importance. Backward feature deletion sorts the global features in order of importance and deletes the least important ones until maximum accuracy is reached. Then, the forward feature deletion function eliminates features individually, beginning with the most important, measuring the model accuracy, and eliminating those that do not reduce performance. Notably, the experiments in Section IV show that features may exist with high feature importance, which reduces estimation accuracy. Matrix X' is the final feature selection matrix.

During the backward selection phase, features are removed in the order of decreasing importance based on the importance of the features derived from AIME. Specifically, a threshold value, ε (initial value 0.1) is set, and its threshold value is found at the highest model accuracy where only features with importance (above the threshold value) are retained. This process determines the final set of features with the highest model contributions. The adjustment of ε in this phase can be increased by increments of 0.1, thus limiting the number of validations to a maximum of 10 and significantly reducing the computational cost.

In the forward selection phase, additional accuracy verifications are conducted on the feature set obtained in the backward selection phase. Here, the features are individually removed, and the model's accuracy is evaluated to determine whether it improves as the features are deleted. Features for which the accuracy does not improve or remains the same are deemed unnecessary and are deleted. This ensures that only the minimum necessary features remain in the final model.

The combination of these two phases allows for efficient and effective feature selection at a lower computational cost than traditional feature selection methods; the feature importance information obtained by using AIME further improves the efficiency of this process and provides the advantage that it can be rapidly applied to high-dimensional datasets and provides the advantage of rapid application, even for high-dimensional datasets.

Fig. 3 contains the SBFD method's pseudocode. The inputs include *X_train*, *y_train*, *X_test*, *y_test*, and the outputs are feature-selected *selected_X_train*, and *selected_X_test*. In Step 1, initialization takes place, which checks the data format and converts labels to a single format.

One-hot encoding is used because AIME requires it for categorical variables. In Step 2, AIME calculates the global feature importance and sorts the features accordingly. Step 3 initializes the ML model and calculates the initial accuracy of all features. Step 4 selects the features with importance levels above the given threshold via the backward deletion step to determine the feature set with the highest accuracy threshold, ε In Step 5, forward deletion takes place, where the features are individually removed, and the final feature set is determined based on accuracy. Step 6 creates *selected_X_train* and *selected_X_test* with the final selected feature set. Step 7 returns the final result, *selected_X_train* and *selected_X_test*.

Backward deletion roughly eliminates features of low importance, and forward deletion verifies whether each feature is contributory. Because forward deletion is time-consuming, it is important to delete as many features as possible beforehand using backward deletion. Thus, our method works most efficiently when the difference in the importance of each feature derived from AIME is large. However, if the importance levels are approximately the same, backward deletion does not function as efficiently. Therefore, the feature importance of AIME can be used to decide whether to continue or withdraw from this feature selection method.

```
Algorithm: Synergistic Backward and Forward Deletion
(SBFD)

Input:
    X_train: Training features
    X_test: Testing features
    y_train: Training labels (one-hot encoded)
    y_test: Testing labels (one-hot encoded)
    model: Machine Learning model (e.g.,
RandomForestClassifier)

Output:
    selected_X_train: Training features with selected
features
    selected_X_test: Testing features with selected
features

Begin
    // Step 1: Initialization
    1. Convert input arrays to DataFrame if necessary
    2. Convert one-hot encoded labels to single-label
       format
    3. Normalize column names in X_train and X_test

    // Step 2: Feature Importance Calculation
    4. Initialize AIME explainer
    5. Compute global feature importance
    6. Sort features by absolute maximum importance
       values

    // Step 3: Model Initialization
    7. Fit the model with all features in X_train
    8. Predict and calculate initial accuracy on X_test
    9. Set initial last_X_train and last_X_test with all
       features

    // Step 4: Backward Deletion
    10. For threshold r from 0 to 1.0 with step 0.1:
        a. Filter features with importance >= r
        b. If filtered features are valid:
            i. Fit model with filtered features
            ii. Predict and calculate accuracy on X_test
            iii. If current accuracy > max_accuracy:
                - Update max_accuracy
                - Update last_X_train and last_X_test
        c. Choose the feature set with the highest
            accuracy

    // Step 5: Forward Deletion
    11. Initialize real_features
    12. For each feature f in last_X_train:
        a. Try to remove feature f:
            i. Drop feature f from last_X_train and
               last_X_test
            ii. Fit model with remaining features
            iii. Predict and calculate accuracy on X_test
            iv. If max_accuracy >= accuracy_without_feature:
                - Append f to real_features

    // Step 6: Select Final Features
    13. Create selected_X_train with real_features
    14. Create selected_X_test with real_features

    // Step 7: Return Final Features
    15. Return selected_X_train, selected_X_test
End.
```

FIGURE 3. Pseudo code of synergistic backward and forward deletion (SBFD) method (the proposed method).

B. GLOBAL FEATURE IMPORTANCE EXTRACTION

This function calculates the global feature importance, $A^{\dagger}(m \times k)$, of each feature from input matrices X $(m \times n)$ and $Y(k \times n)$ for classification using the AIME model. Note that "importance" reflects the contribution of a feature to each class in a classification problem. Therefore, to sort them, one contribution per feature must be derived. We offer two methods for this. The first takes the maximum value of each feature, and the second takes the variance. The first method considers a feature to be important if it is included in any class with high importance. The second judges importance based on its variance (high $\rightarrow$ high). The better method is experimentally determined.

C. BACKWARD FEATURE DELETION

Fig. 4 illustrates the backward feature delineation process.

First, it uses matrices $X(m \times n)$ and $Y(k \times n)$ for classification, ordered according to the importance of the features. It then divides them into *X_train* for training

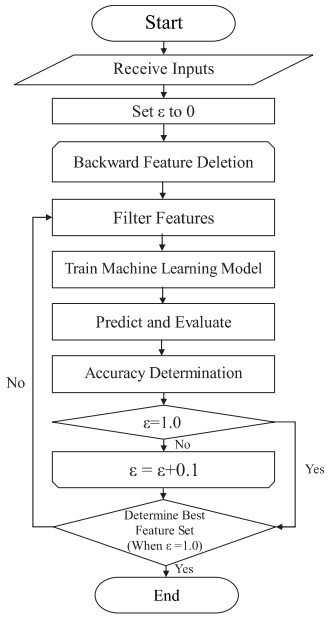

FIGURE 4. Backward feature extraction function flow.

data, *y_train* for training labels, *X_test* for testing data, and *y_test* for testing labels. Next, ε is initialized to zero *tosetthe*global feature importance. Then, features with low global feature importance ($\varepsilon < 0.1$) are eliminated to derive model accuracy. This is repeated at every 0.1 level of accuracy (or any appropriate value based on the given balance between accuracy and computational complexity).

During the feature filtering step, the features in *X_train* and *X_test*, which have a lower absolute importance than ε, are removed. The ML model is then retrained using *X′_train* and *y_train* with the deleted features. In the evaluation step, features with importance less than ε from the *X_test* and *y_test* are removed, and the prediction accuracy is recalculated. This process is repeated as ε incrementally increases, which allows for efficient feature selection from the least to the most important. The training and testing data reconstructed using only the important features obtained here are noted as *X″_train* and *X″_test*.

D. FORWARD FEATURE DELETION

Fig. 5 illustrates the forward feature deletion process. Initially, the training dataset (*X″_train*) and test dataset (*X″_test*) obtained in the previous backward feature detection step are used. This process removes the *i*-th feature at each step and generates a new $X''_{\backslash i}\_train$ and $X''_{\backslash i}\_test$. Next, the ML model is trained using these data, and the model is evaluated with $X''_{\backslash i}\_test$ and *y_test*. If the evaluated accuracy is lower than the highest accuracy thus far, then the feature set is retained. This process is repeated by increasing *i* for all features, and finally, the feature set with the highest accuracy is selected. As a reminder, the number of training and evaluation cycles can be adjusted as needed.

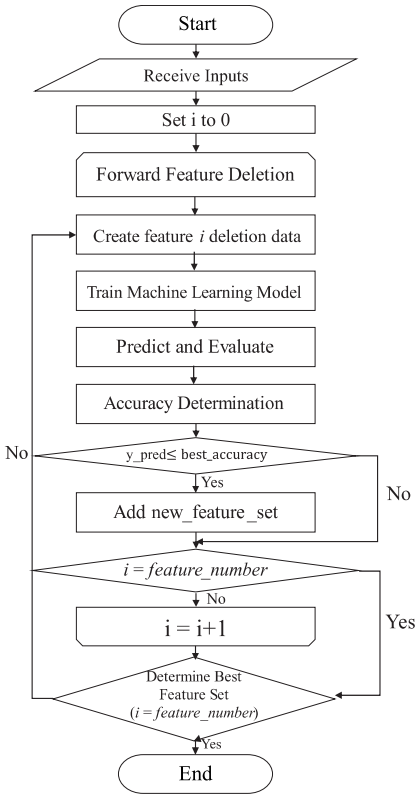

FIGURE 5. Forward feature extraction function flow.

V. EXPERIMENTS

Section V-A describes the experimental environment. Section V-B describes the MELD feature extraction dataset [24], [25], which was used in Experiments 2, 3 and 4. In Section V-C, we explain Experiment 1, which uses 15 of 26 UCI datasets [65] and all 3 ASU feature selection datasets [64]. Although Kılıç et al. [66] used all 26 UCI datasets, they are not publicly accessible. The 11 excluded sets are SonarEW, Krvskp, M-of-n, Penglung, Vote, Exactly, Exactly2, Pendigits, Clean1, Clean2, and WaveformEW. The trends in time, cost, and feature selection between the proposed method and other methods were thus analyzed based on the available data. Table 3 shows the target datasets and their numbers of features and samples. Section V-D describes Experiment 2, in which the model was trained, and the accuracy was verified based on the increasing order of global feature importance from values of 5 to 30. Section V-E describes Experiment 3, which compared the global feature importance of the backward and forward deletion processes. Section V-F then describes Experiment 4, which verified the effectiveness of our method by comparing its accuracy with that of other state-of-the-art feature selection methods.

A. EXPERIMENTAL ENVIRONMENT

This system was implemented in Python 3.10.12 on Google's Colab Pro+. Random forests were used as the target ML model in Experiments 1 and 2, and RF [67] with a Light

TABLE 3. Target dataset (sample size is based on data with missing values removed).

	Number of features	Number of Samples
BreastEW	30	569
Breastcancer	9	699
HeartEW	13	297
Lymphography	18	148
SpectEW	23	267
IonosphereEW	34	351
Zoo	16	101
WineEW	13	178
Vehicle	18	846
Spambase	57	4,601
Tic-tac-toe	9	958
Semeion	256	1,593
CongressEW	16	6,435
Satellite	36	6435
Dermatology	34	358
PCMAC	3,289	1,943
BASEHOCK	4,862	1993
warpPIE10P	2420	210

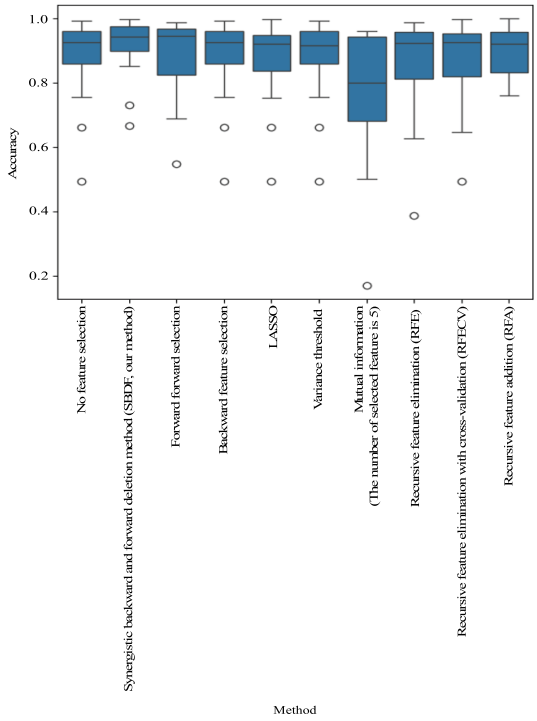

FIGURE 6. Boxplots for each feature selection method.

Gradient Boosting Machine (LightGBM) [68] was used in Experiment 3 because the comparisons are more difficult. Note that any combination can be applied. Conversions to RFs and one hot vectors were performed using scikit-learn-1.2.2. AIME was implemented with numpy-1.25.2, pandas-1.5.3, matplotlib-3.7.1, and seaborn-0.13.1. The results were visualized using pandas-1.5.3 and matplotlib-3.7.1.

B. FEATURE EXTRACTION FROM MELD

The MELD multimodal dataset [24], [25], compiled for emotion and personality recognition tasks was constructed from scenes from the TV series Friends and comprises 1,400 dialog passages and 13,000 utterances. Each utterance was annotated with the speaker's emotion and intensity. It also contains video frames, audio, and textual data. This dataset is particularly suited to studies analyzing complex scenarios in which emotion dynamics and social contexts are intertwined. It is also an important resource for understanding changes in emotions during conversations and for studies combining multiple modes, such as natural language processing, speech analysis, and facial expression recognition.

Using py_feat 0.6.1 (23 features), the facial features' pitch, roll, and yaw values were extracted as well as every action unit (AU). The py_feat can extract only 20 AUs and facial features such as pitch, roll yaw. AUs [69] comprise a system developed by Ekman and Friesen to classify facial muscle movements and are widely used to analyze emotional expressions. Each AU corresponds to a specific facial muscle movement and details subtle differences in emotions. For example, AU1 represents an "eyebrow-raising" movement, which may indicate surprise or questioning. Originally, AUs were numbered from AU1 to AU58, but due to detailed facial expressions or anatomical reasons, such as AU3, AU8, AU13, AU19, AU21, and AU22, specific action units are not defined or are represented by other composite action units, and thus are absent.

We averaged the large features, AU 25 and 26, related to mouth movements. Although we understand that speaker identification and extraction should be performed, we used this method to demonstrate its efficacy, as speaker identification is difficult in most situations. Voice features were extracted from eGeMAPSv02 using opensmile-2.5.0 (88 features) [70], [71], and words were extracted from utterances from the "Utterance" metadata of MELD and weighted using term frequency–inverse document frequency (TF–IDF, 4,759 words/features).

Some MELD videos did not capture facial features and were discarded. Consequently, 9,930 training and 325 test data points were generated for a total of 4,870 features including facial (23), voice (88), and utterance features (4,759). Feature selection was then performed to predict anger, disgust, fear, joy, neutrality, sadness, and surprise emotions, as recommended by Ekman and Friesen [72], which included negative, neutral, and positive states.

Bingöl et al. [73] extracted face and voice features from participants' audio–video data, trained each QoE estimation model separately, and used data fusion techniques to integrate face and voice datasets, thereby enabling improved QoE estimation performance from the integration of the resulting information. Although several of the most important features of the QoE estimation system are discussed next, their comprehensive use will be the subject of future research.

TABLE 4. Computation-time comparison of our method against other typical feature selection methods on 15 UCI datasets [66] and 3 ASU feature selection datasets [67].

	No feature selection	SBFD (our method)	Forward feature selection	Back feature selection	LASSO	Variance threshold	Mutual Information (k = 5)	REF	REF-CV	RFA
BreastEW	0.426	8.659	23.085	7.195	0.231	0.380	0.538	4.605	35.467	16.489
Breastcancer	0.157	4.019	6.275	1.668	0.167	0.167	0.213	1.045	9.450	4.436
HeartEW	0.174	2.614	5.722	2.736	0.192	0.185	0.332	1.570	13.534	7.808
Lymphography	0.292	4.094	15.365	3.025	0.160	0.155	0.311	2.287	17.431	7.931
SpectEW	0.156	6.795	6.038	3.559	0.149	0.160	0.398	3.117	19.124	11.524
IonosphereEW	0.198	8.465	29.498	8.952	0.213	0.206	0.353	3.731	37.797	20.135
Zoo	0.157	3.963	13.577	4.940	0.319	0.289	0.432	1.713	16.708	8.697
WineEW	0.283	4.213	7.775	3.991	0.159	0.172	0.289	1.396	12.847	6.054
Vehicle	0.301	8.684	22.576	7.531	0.277	0.280	0.543	2.897	27.596	15.150
Spambase	0.860	48.035	369.352	51.733	1.200	1.259	2.430	26.025	196.583	64.006
Tic-tac-toe	0.212	8.631	8.792	6.843	0.202	0.196	0.389	2.943	30.575	15.097
Semeion	0.566	101.445	1483.882	154.413	0.682	0.537	14.295	70.556	599.873	142.377
CongressEW	0.147	6.033	10.949	4.876	0.151	0.153	0.305	3.995	30.662	15.626
Satellite	1.700	59.150	225.351	65.869	2.237	1.683	5.098	29.997	216.862	51.288
Dermatology	0.187	7.897	36.249	6.920	0.298	0.274	0.792	3.230	35.998	17.650
PCMAC	1.677	3892.264	26196.212	4651.177	3.488	1.207	51.483	1800.895	12678.033	1883.239
BASEHOCK	5.606	7296.471	41462.764	7457.682	4.564	1.509	78.942	3024.944	21038.068	2746.121
warpPIE10P	0.762	879.209	3883.891	2096.709	1.292	0.773	66.827	888.567	7730.757	1169.394

TABLE 5. Accuracy comparison of our method against other typical feature selection methods on 15 UCI datasets [66] and 3 ASU feature selection datasets [67].

	No feature selection	SBFD (our method)	Forward feature selection	Back feature selection	LASSO	Variance threshold	Mutual Information (k = 5)	REF	REF-CV	RFA
BreastEW	0.937063	0.965035	0.979021	0.937063	0.944056	0.937063	0.944056	0.944056	0.944056	0.937063
Breastcancer	0.964912	0.964912	0.970760	0.964912	0.959064	0.964912	0.959064	0.959064	0.959064	0.959064
HeartEW	0.493333	0.666667	0.546667	0.493333	0.493333	0.493333	0.520000	0.386667	0.493333	0.800000
Lymphography	0.837838	0.891892	0.918919	0.837838	0.837838	0.837838	0.675676	0.810811	0.810811	0.810811
SpectEW	0.850746	0.850746	0.805970	0.850746	0.835821	0.850746	0.776119	0.805970	0.850746	0.791045
IonosphereEW	0.943182	0.943182	0.965909	0.943182	0.943182	0.931818	0.954545	0.920455	0.943182	0.943182
Zoo	0.884615	0.884615	0.961538	0.884615	0.884615	0.884615	0.807692	0.846154	0.846154	1.000000
WineEW	0.977778	0.977778	0.977778	0.977778	0.977778	0.977778	0.955556	0.977778	0.977778	0.844444
Vehicle	0.660377	0.731132	0.688679	0.660377	0.660377	0.660377	0.500000	0.627358	0.646226	0.759434
Spambase	0.940052	0.944396	0.944396	0.940052	0.940921	0.940052	0.940052	0.943527	0.943527	0.906169
Tic-tac-toe	0.991667	0.995833	0.691667	0.991667	0.995833	0.991667	0.741667	0.987500	0.995833	0.920833
Semeion	0.754386	0.922306	0.714286	0.754386	0.751880	0.754386	0.170426	0.756892	0.751880	0.972431
CongressEW	0.948276	0.948276	0.948276	0.948276	0.948276	0.948276	0.948276	0.948276	0.948276	0.948276
Satellite	0.878185	0.916097	0.878807	0.878185	0.877564	0.878185	0.811063	0.881293	0.883157	0.989434
Dermatology	0.966667	0.977778	0.966667	0.966667	0.966667	0.966667	0.688889	0.966667	0.955556	0.922222
PCMAC	0.915638	0.923868	0.962963	0.915638	0.903292	0.901235	0.804527	0.925926	0.905350	0.831276
BASEHOCK	0.977956	0.981964	0.987976	0.977956	0.935872	0.969940	0.793587	0.975952	0.975952	0.833667
warpPIE10P	0.886792	0.981132	0.924528	0.886792	0.792453	0.886792	0.679245	0.811321	0.792453	0.981132

C. EXPERIMENT 1: COMPARISON OF COMPUTATIONAL COST, ACCURACY, AND NUMBER OF FEATURE SELECTION AMONG OUR METHOD AND PREVIOUS METHODS WITH 15 UCI AND 3 ASU FEATURE SELECTION DATASETS

We compared the computation time, accuracy, and number of features after feature selection between our SBFD method and the main feature selection method from Experiment 1 using the dataset in Table 3. The measured computation time, accuracy, and feature number after comparing the methods are shown, respectively in Tables 4, 5, and 6.

The computation times in Table 4 show that the relatively simple LASSO, variance threshold, and mutual information methods did not require much time.

In contrast, the computation times for SBDF, forward selection, backward selection, RFE, RFE-CV, and RFA increased with the number of features and samples. This was particularly true for forward feature selection and RFE-CV, which, with thousands of features, are considered unsuitable for feature selection. Although SBDF requires more computation time due to backward deletion compared with the forward and backward feature-selection filtering methods, SBDF clearly became more efficient with each increase in the number of features. Indeed, for the 10,255 data elements (total of training and test data) and the 4,870 features extracted from the MELD dataset used in Experiments 2 and 3, SBDF's backward feature

TABLE 6. Number of features selected by our method compared against other typical feature selection methods on 15 UCI datasets [66] and 3 ASU feature selection datasets [67].

	No feature selection	SBFD (our method)	Forward feature selection	Back feature selection	LASSO	Variance threshold	Mutual Information (k = 5)	REF	REF-CV	RFA
BreastEW	30	29	3	30	13	30	5	15	14	3
Breastcancer	9	8	4	9	8	9	5	4	8	2
HeartEW	13	5	1	13	13	13	5	6	13	5
Lymphography	18	17	4	18	18	18	5	9	17	2
SpectEW	22	21	1	22	17	22	5	11	22	4
IonosphereEW	34	33	4	34	25	33	5	17	26	5
Zoo	16	15	4	16	15	16	5	8	10	1
WineEW	13	12	3	13	13	13	5	6	11	3
Vehicle	18	17	5	18	15	18	5	9	12	7
Spambase	57	53	12	57	45	57	5	28	47	6
Tic-tac-toe	27	26	1	27	18	27	5	13	18	10
Semeion	256	240	17	256	243	256	5	128	205	9
CongressEW	32	31	1	32	8	32	5	16	15	2
Satellite	36	35	7	36	32	36	5	18	31	3
Dermatology	34	29	5	34	32	34	5	17	21	4
PCMAC	3289	3288	34	3289	636	3288	5	1644	2890	9
BASEHOCK	4862	4861	34	4862	658	4861	5	2431	3786	7
warpPIE10P	2420	1705	6	2420	207	2420	5	1210	78	3

selection worked effectively, and the number of samples was large.

Hence, even after 24 h of running Google Colab Pro+, forward feature selection, backward feature selection, RFE, and RFE-CV could not complete the feature selection task.

Table 5 lists the accuracy of each method depending on the dataset. Among the methods, SBDF showed good overall accuracy, which is discussed in more detail later in this paper. Table 6 lists the number of features after feature selection for each method. The variance threshold and backward feature selection were unable to select features from this dataset. In contrast, the RFA method selected a very small number of features. Forward feature selection and RFA were deemed suitable for removing large numbers of features. Each of the other methods also extracted the necessary features.

Figure 6 presents a boxplot to validate the accuracy of each method further. After comparing the effectiveness of our proposed SBDF and other methods, we can see that the median accuracy of SBDF was very high and exceeded that of many of the other methods. In particular, compared with the variance threshold and mutual information ($k = 5$), the SBDF consistently exhibited high accuracy. Furthermore, the SBDF boxplot is narrower than that of many other methods, indicating less variation in accuracy across datasets. This implies that the SBDF performs consistently well. Furthermore, our model produced fewer outliers and fewer cases with extremely low accuracy compared to the other methods. This suggests that SBDF provides stable performances on many datasets. This indicates that our method is capable of relatively stable feature selection from any dataset.

Table 7 shows the results of pairwise t-testing SBDF and the other methods' accuracy. The results show no statistically significant differences, apart from mutual information ($k = 5$). This implies that the behavior of each method differs depending on the dataset. Table 8 shows the general linear model (GLM) results for dataset, method, and precision. The HeartEW, Lymphography, Semeion, SpectEW, and Vehicle datasets had a significant negative impact on accuracy. No significant methods were found apart from mutual information, which was negatively and significantly correlated. However, the effect of SBDF was positive, although not significant, relative to the other methods; SBDF performed well overall but may not be significant enough to show a noticeable difference.

Based on our experiments, the results show that the proposed SBFD method is positively correlated with the other feature selection methods regarding accuracy, but the correlations are not statistically significant. This result may

TABLE 7. Comparison of p-values from pairwise t-tests evaluating previous feature selection methods and the proposed synergistic backward and forward deletion (SBFD) method. This table demonstrates the statistical significance of the differences between SBFD and existing feature selection techniques.

Method	t-statistic	p-value
Mutual information (k=5)	2.94855651	0.007196127
RFE	1.33128975	0.194103596
RFECV	1.256397732	0.218759009
LASSO	1.238370528	0.22514556
Variance threshold	1.052994207	0.300679134
No feature selection	0.99673307	0.32682748
Backward feature selection	0.99673307	0.32682748
Forward selection	0.939371348	0.355127885
RFA	0.637595384	0.528081185

TABLE 8. Results of generalized linear model (GLM) analysis showing the significance of differences between various feature selection methods and the proposed synergistic backward and forward deletion (SBFD) method across different datasets. This table highlights the statistical significance and effectiveness of SBFD compared with existing techniques.

Term	Coefficient	Std. Error	z-value	P>|z|	[0.025	0.975]
Intercept	0.9487	0.0250	38.1750	0.0000	0.9970	0.9970
(Dataset)[BreastEW]	0.0126	0.0280	0.4550	0.6490	0.0670	0.0670
(Dataset)[Breastcancer]	0.0297	0.0280	1.0740	0.2830	0.0840	0.0840
(Dataset)[CongressEW]	0.0151	0.0280	0.5440	0.5860	0.0690	0.0690
(Dataset)[Dermatology]	-0.0128	0.0280	-0.4640	0.6430	0.0410	0.0410
(Dataset)[HeartEW]	-0.4032	0.0280	-14.5640	0.0000	-0.3490	-0.3490
(Dataset)[IonosphereEW]	0.0109	0.0280	0.3940	0.6940	0.0650	0.0650
(Dataset)[Lymphography]	-0.1022	0.0280	-3.6900	0.0000	-0.0480	-0.0480
(Dataset)[PCMAC]	-0.0409	0.0280	-1.4780	0.1390	0.0130	0.0130
(Dataset)[Satellite]	-0.0492	0.0280	-1.7780	0.0750	0.0050	0.0050
(Dataset)[Semeion]	-0.2315	0.0280	-8.3600	0.0000	-0.1770	-0.1770
(Dataset)[Spambase]	0.0015	0.0280	0.0540	0.9570	0.0560	0.0560
(Dataset)[SpectEW]	-0.1074	0.0280	-3.8780	0.0000	-0.0530	-0.0530
(Dataset)[Tic-tac-toe]	-0.0075	0.0280	-0.2720	0.7860	0.0470	0.0470
(Dataset)[Vehicle]	-0.2772	0.0280	-10.0110	0.0000	-0.2230	-0.2230
(Dataset)[WineEW]	0.0316	0.0280	1.1420	0.2540	0.0860	0.0860
(Dataset)[Zoo]	-0.0486	0.0280	-1.7560	0.0790	0.0060	0.0060
(Dataset)[warpPIE10P]	-0.0873	0.0280	-3.1540	0.0020	-0.0330	-0.0330
(Method)[Forward feature selection]	0.0014	0.0230	0.0620	0.9500	0.0460	0.0460
(Method)[LASSO]	-0.0089	0.0230	-0.3950	0.6930	0.0350	0.0350
(Method)[Mutual information(k =5)]	-0.1188	0.0230	-5.2570	0.0000	-0.0750	-0.0750
(Method)[No feature selection]	0.0000	0.0230	0.0000	1.0000	0.0440	0.0440
(Method)[RFA]	0.0189	0.0230	0.8380	0.4020	0.0630	0.0630
(Method)[RFE]	-0.0185	0.0230	-0.8200	0.4120	0.0260	0.0260
(Method)[RFECV]	-0.0103	0.0230	-0.4570	0.6470	0.0340	0.0340
(Method)[SBDF (our method)]	0.0366	0.0230	1.6170	0.1060	0.0810	0.0810
(Method)[Variance threshold]	-0.0019	0.0230	-0.0830	0.9340	0.0420	0.0420

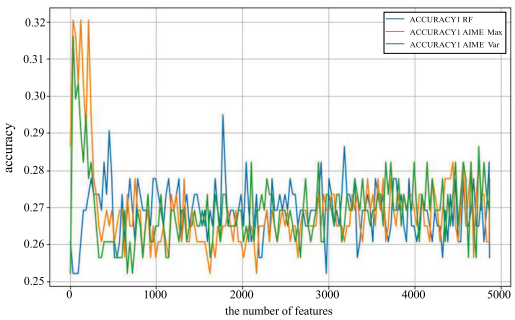

FIGURE 7. Differences in accuracy depend on the number of global feature importance and feature importance from the most to the least important (in the case of seven emotions).

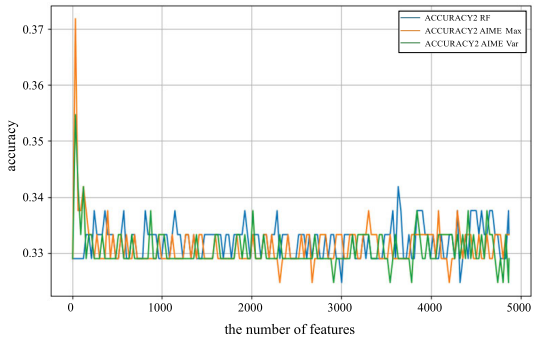

FIGURE 8. Differences in accuracy depend on the number of global feature importance and feature importance from the most important to the least important (in the case of the three states).

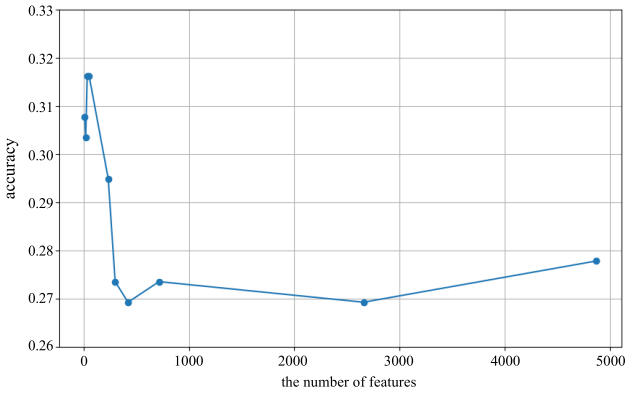

FIGURE 9. Results of backward deletion for seven emotions.

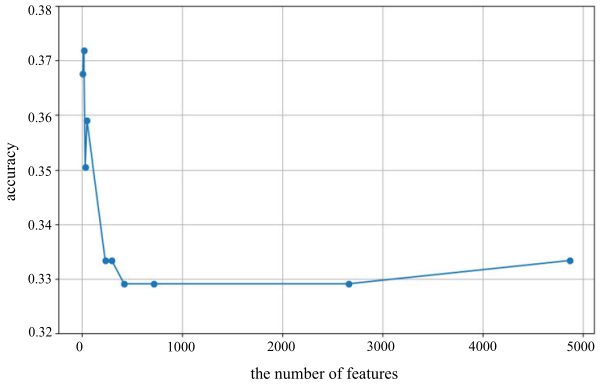

FIGURE 10. Results of backward deletion for three states.

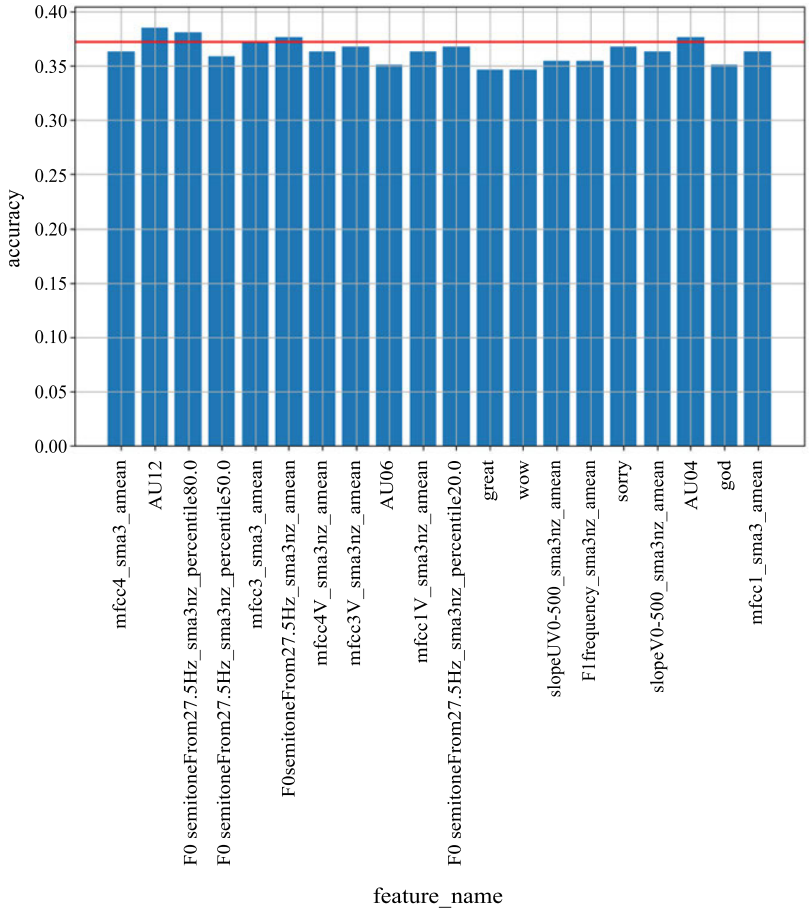

FIGURE 11. Results of forward deletion for seven emotions.

be attributed to the variable performance of the feature selection methods depending on the dataset. In particular, with the exception of mutual information ($k = 5$), no statistically significant differences were identified among the other methods. However, the boxplot results confirm that SBFD maintains a relatively stable and high level of accuracy. This supports the conclusion that SBFD is an effective method that can provide stable accuracy, regardless of the dataset.

D. EXPERIMENT 2: VISUALIZATION OF THE DIFFERENCE IN ACCURACY WHEN INCREASING THE NUMBER OF FEATURES IN ORDER OF GLOBAL FEATURE IMPORTANCE

In this experiment, AIME extracted the global feature importance, and methods, which generate the maximum value, variance, and importance for each feature were applied to the RF model.

Because AIME is model-agnostic, it can be applied to any AI or ML model; therefore, RF and Light GBM were examined in this study. As examined in [21], AIME derives stable explanations compared with LIME [36] and SHAP [37] for CNNs and Light GBM. Therefore, if the predictive value of ML is not low, it does not significantly affect global feature importance. This is because the approximate inverse operator is obtained using X and $\hat{Y}$. Furthermore, as shown in Section III-A, it is possible to derive the global feature importance by assuming an ideal ML result that does not exist when using X and Y [74]. Therefore, when using AIME, one of the XAIs in these cases, it is a model-agnostic method that can derive a more stable global feature importance than LIME [36] and SHAP [37]. Hence, Y instead of $\hat{Y}$ can be used to obtain an ideal global feature importance, as discussed in Section III-A. Because obtaining the global feature importance can also be proposed as one method [74], which ML model is appropriate, is not discussed here.

Training and accuracy measurements were conducted using between 5 and 30 added features. The seven emotions are shown in Fig. 7, and the three states are shown in Fig. 8. *RF* represents the random forest-sorted features, *AIME MAX* represents the sorted features based on their maximum value after deriving the global feature importance, and *AIME Var* represents them based on maximum variance. *AIME MAX* achieved maximum accuracy.

The peaks in Figs. 7 and 8 fall within 100 features, which implies that even if many features are extracted as multimodal data, only a few are truly effective. Therefore, the backward feature deletion process effectively reduces the number of training and accuracy verifications needed for top model performance.

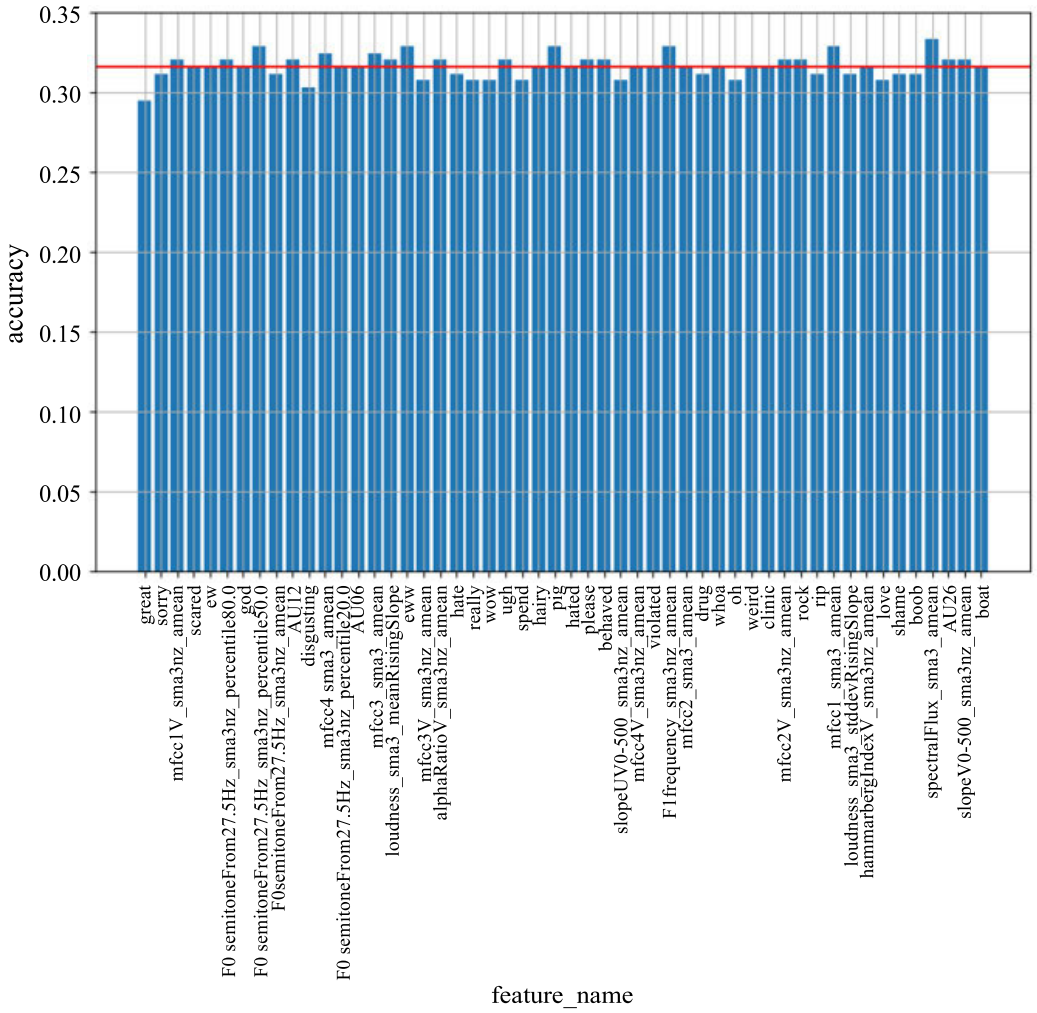

FIGURE 12. Results of backward deletion for three states.

In Fig. 8, the peaks change erratically, indicating that a larger set of features cannot be assumed to improve ML accuracy. Hence, the forward feature deletion process effectively determines the features that increase accuracy the most.

Throughout the experiments, maximum accuracy was found to be low because the extracted features were selected for interpretability. If only accuracy were to be considered, InstructERC [75], which uses a large language model and full embedding, would be the best-performing XAI method at the time of this study. The application of AIME to complex models is a subject for future work.

E. EXPERIMENT 3: VERIFICATION OF THE BEHAVIOR OF SYNERGISTIC BACKWARD AND FORWARD DELETION METHOD UTILIZING GLOBAL FEATURE IMPORTANCE METHOD

The results of the backward feature deletion are shown in Figs. 9 and 10 respectively for the seven emotions and the three states cases. Both show the results of deletion at decrements of 0.1, from 4,870 to 0. As Fig. 9 shows, the highest accuracy (0.316) was obtained for the cases with 52 and 34 features, and Fig. 10 shows the highest accuracy (0.372) was obtained for the cases with 20 features. Hence, most of the 4,870 features did not contribute to discriminating between emotions and states, and only 100 or less contributed to discrimination. This process noticeably reduced the computational complexity of subsequent forward deletion processes.

From these results, we show that real-time decision-making can be achieved, even with the computational costs of discriminating the top features and streamlining the datasets. The red lines in Figs. 11 and 12 indicate the maximum accuracies of the backward deletion process. The bar graph shows the accuracy when the features on the horizontal axis are deleted. If the accuracy is below the red line, the feature can be judged as significantly effective in determining the emotion or state. Conversely, if the accuracy is above the red line, the feature is eliminated, and the accuracy increases; therefore, the feature can be judged as insignificantly effective in determining the emotion or state. Based on the results shown in

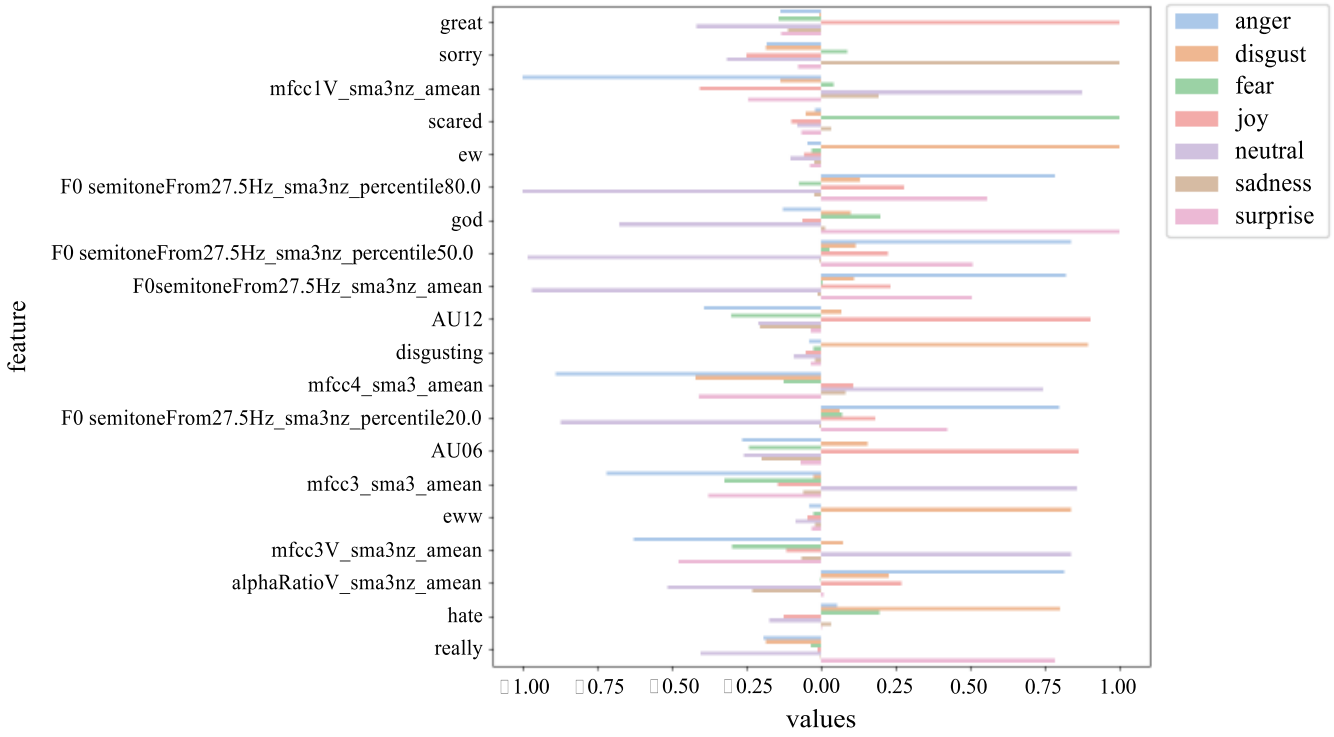

FIGURE 13. Visualization of contributions for the top 20 features and seven emotions after feature selection.

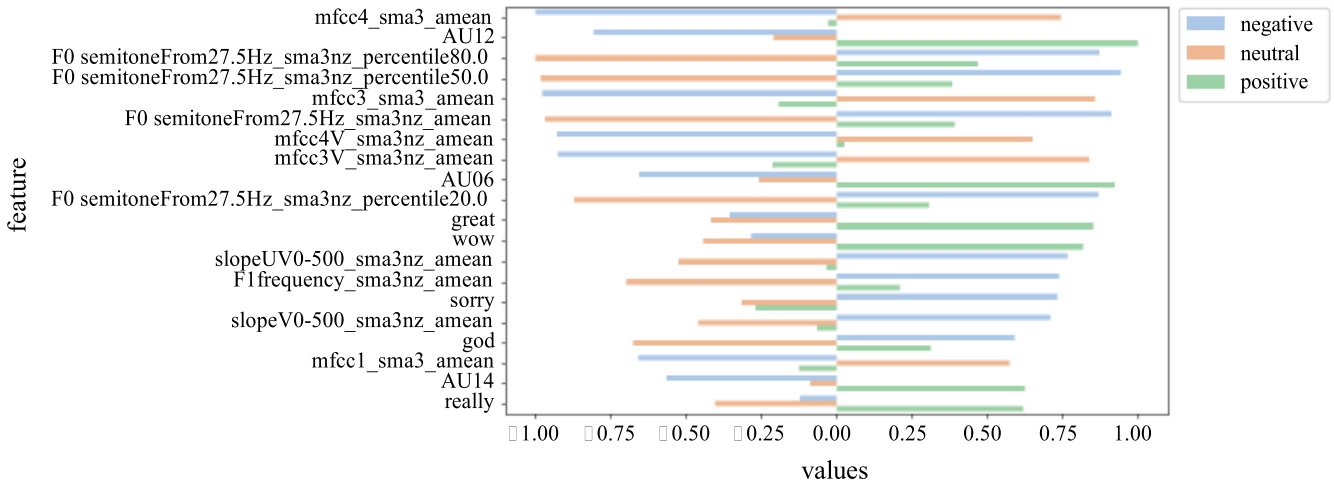

FIGURE 14. Visualization of contributions for the top 20 features and three states after feature selection.

TABLE 9. Our method compared against other typical feature selection methods in seven emotions (random forest).

	No feature selection	SBFD (Our method)	LASSO	Variance Threshold	Mutual Information	Mutual Information	Mutual Information	Recursive Feature Addition
Number of features	4870	32	53	234	1000	500	50	1
Accuracy	0.273504	0.363248	0.282051	0.273504	0.264957	0.252137	0.260684	0.260684

Fig. 11, "mfcc1v_sma3nz_amean" and many other features exceeded the red line, and 32 total features were judged to be valid. As shown in Fig. 12, "AU12" and the others were above the red line, and 16 were selected at the end. These findings indicate that even if the maximum AIME value of global feature importance is high, f features alone probably remain unsuitable for determining the proper prediction result.

TABLE 10. Our method compared against other typical feature selection methods in three states (random forest).

	No feature selection	SBFD (Our method)	LASSO	Variance Threshold	Mutual Information	Mutual Information	Mutual Information	RFA
Number of features	4870	16	39	234	1000	500	50	1
Accuracy	0.337607	0.371795	0.33333	0.329060	0.329060	0.329060	0.329060	0.337607

TABLE 11. Our method compared against other typical feature selection methods in seven emotions (LightGBM).

	No feature selection	SBFD (Our method)	LASSO	Variance Threshold	Mutual Information	Mutual Information	Mutual Information	RFA
Number of features	4870	711	53	234	1000	500	50	1
Accuracy	0.423077	0.423077	0.346154	0.423077	0.388889 7	0.346154	0.320513	0.470085

TABLE 12. Our method compared against other typical feature selection methods in three states (LightGBM).

	No feature selection	SBFD (Our method)	LASSO	Variance Threshold	Mutual Information	Mutual Information	Mutual Information	RFA
Number of features	4870	16	39	234	1000	500	50	1
Accuracy	0.465812	0.521368	0.444444	0.465812	0.303419	0.423080	0.303419	0.465812

TABLE 13. The results of this study show the advantages, disadvantages, and positioning of usage cases of the synergistic backward and forward deletion method (SBFD) and other methods.

Method	Strengths	Weaknesses	Typical Use Cases
SBFD (our method)	Effectively utilizes global feature importance and minimizes computational cost. Uses the model-agnostic XAI technique AIME, applicable across various models.	If feature importance is uniform and does not vary, backward deletion is barely effected, thus not optimizing computational efficiency.	High-dimensional multimodal datasets where interpretation and explanation are crucial.
Forward Feature Selection	Simple and intuitive, adds features individually based on performance. Derives feature importance from glass-box models.	Requires evaluation of each feature, which can be computationally expensive.	Important in small to medium datasets where interpretation is key.
Backward Feature Selection	Starts with a full feature set and iteratively removes the least impactful features. Derives feature importance from glass-box models.	Needs to retrain the model each time a feature is removed, which can be computationally expensive.	Suitable for datasets with a moderate number of features where interpretation is important.
LASSO	Simultaneously performs feature selection and regularization.	May overlook relevant features if the regularization penalty is too strong.	Regression problems where both prediction and interpretation are key.
Variance Threshold	Automatically removes low-variance features.	Uniformly removes features based on variance value without considering the contribution of the features.	Datasets with many non-informative features, such as image data.
Mutual Information	Captures non-linear relationships between features and targets.	Users must specify the number of features, making it difficult to determine the appropriate number of features.	Complex datasets where linear relationships are insufficient, feasible where feature count can be determined.
RFE	Iteratively removes features in order of least importance and derives feature importance from glass-box models.	Can be slow for large datasets as it requires the model to be rebuilt repeatedly.	High-dimensional data where model simplification is required and interpretation is crucial.
RFE-CV	Automatically finds the optimal number of features and derives feature importance from glass-box models.	Excessive computational cost, especially with large datasets.	Applicable in sectors where interpretation and accuracy are a priority.
RFA	Adds features in order of highest importance and derives feature importance from glass-box models.	Similar to RFE, it requires major computational power.	Applicable in cases where interpretation is important, especially with datasets allowing for sequential feature addition.

Fig. 13 shows the feature selection results after the backward deletion process and the forward deletion process for the seven-emotion case, and Fig. 14 shows that of the three-state case.

F. EXPERIMENT 4: COMPARISON OF ACCURACY WITH EXISTING REPRESENTATIVE FEATURE SELECTION METHOD

Table 9 compares the number of features and accuracy values of our method with a typical feature selection method for the problem of estimating seven emotions with an RF model. Table 10 does the same for the three states. Table 11 compares the number of features and accuracy values between our method and the state-of-the-art LightGBM XAI method for the seven emotions, and Table 12 does the same for the three states. RFE and RFECV are not listed in the tables because the LightGBM results could not be derived, even after 24 h of operating in the Google Colab Pro+ environment. The global feature importance provided by AIME and the deletion processes enables our method to greatly reduce computational costs. For cases of mutual information, the number of features cannot be determined; hence, they were set to 1,000, 500, and 50.

Our method was clearly the most accurate, apart from LightGBM's seven emotion estimation performance in Table 11, for which its recursive feature addition (RFA) was more accurate. However, the feature it selected was "F0 semitoneFrom27.5 Hz_sma3nz_percentile80.0," which corresponds to voice pitch. We believe that it is purely coincidental that the emotions were identifiable only by the pitch of the human voice in this case.

In LightGBM, feature selection can be performed indirectly through its feature fraction and the model training process. However, the results in Table 12 confirm that our method improved accuracy more than LightGBM did. Moreover, being based on importance, our feature selection is effective even when feature selection or dimensionality reduction are provided by the baseline ML model.

VI. DISCUSSION AND CONCLUSION

The results of Experiment 1 enabled us to compare our SBFD with previous major feature-selection methods on various datasets. Owing to the different behaviors of the different datasets, no statistically significant differences could be derived; however, the boxplots confirm that our SBFD consistently achieves good accuracy.

The results of Experiment 2 demonstrate that using the maximum value of the global feature importance from AIME is effective for feature selection and that the backward deletion process, in which features are individually deleted in order from the smallest global feature importance, is effective. We also found that the global feature importance reduces accuracy from the top of the list instead of doing so steadily and gradually from the bottom. Therefore, the forward deletion process is essential, and feature discrimination can be complex.

The results of Experiment 3, based on Experiment 2, show that our global feature importance method with backward and forward deletion works well for all features. Because almost features can be eliminated by backward deletion, we also confirmed that removing them individually by forward deletion reduces the computational cost of verification.

The results of Experiment 4 show that, in most cases, our method is more accurate than contemporary feature selection methods. For data of 4,870 dimensions, the RFE and RFECV methods could not complete the job even after 24 hours of processing. Hence, inferentially, AIME can be assumed to contribute to the reduction in computational costs because our method completed the process. Furthermore, AIME continues to provide feature selection with high interpretability, and the cases of seven emotions and three states (Figs. 12 and 13) confirm this.

Mamieve et al. [76] proposed an attention-based fusion framework that combines features from facial expressions and speech signals. This approach involves extracting relevant features from both modalities and applying an attention mechanism to fuse these features effectively. The system was trained and evaluated on standard emotion recognition datasets, and we will support such multimodal features by applying our method in the future.

Table 13 summarizes the advantages, disadvantages, and uses of our method and the results of the experiments. Although our proposed SBFD did not show significant improvements in statistical tests, the boxplots show that the results are stable for all datasets. We believe that the reason for the lack of statistical significance is that different datasets have different writing methods, and we have shown here that the AIME feature results are maximally different after SBFD and that the computational efficiency and accuracy are higher when backward deletion is successful. Compared with other methods, ours' combines backward and forward deletions. As such, it can be considered a filter method. However, the RFE, RFE-CV, and RFA methods use a glass-box model to determine the important feature set by removing and adding features using feature importance. We introduced AIME as a model-agnostic XAI, and we used its feature importance results to realize two phases: backward and forward deletion. Hence, this contributes efficiently to reducing computational complexity. Otherwise stated, AIME functions as both a filter and a wrapper, and as a new feature selection method effectively supports XAI in diverse systems.

Our results further demonstrate that despite the importance of multimodal data, only a small fraction of features contribute notably to prediction accuracy. However, our method still has some limitations. For example, when the most feature in which all features have approximately equal contributions can be regarded as a situation where feature selection does not need to be performed. For example, when backward deletion clearly improves accuracy (see Figs. 7 and 8), it is common to find a high variance in the distribution of importance derived from AIME, but with many features of low importance (see Figs. 5 and 6). This approach is effective when the variance in the feature importance derived from AIME is high, but it is not suitable when the variance is low. When the variance in feature importance is low, feature selection may not contribute to improved

performance, as most features are equally important. From our analysis, it is possible to determine whether backward or forward deletions should be performed based on the feature importance derived by AIME. Backward deletion does not work well with small variances in feature importance because all features are deleted in the same epsilon or none is deleted; in many cases, they are not deleted. Thus, many features must be considered for forward deletion, which is time-consuming. Conversely, however, a small variance in feature importance may mean that all features have the same importance, from which it may be concluded that feature selection is unnecessary.

AIME's computational order of magnitude given by Eq. (5) is $O(mnk + k^2n + k^3 + mk^2)$, where the computational order increases dramatically with the number of Y classes, k. Conversely, in general forward feature selection, the computational complexity increases at $O(nm^3)$ as the number of features, m, increases. This indicates that with a large number of features, the computation time increases. In practice, the importance of forward features is inconsistent because feature selection is often desired when the number of features, m, is high. Therefore, it is critical to assess whether to adopt AIME's determination of feature importance, which is a limitation.

Throughout the experiments, maximum accuracy was found to be low because the extracted features were selected for interpretability. If only accuracy were to be considered, InstructERC [76], which uses a large language model and full embedding, would be the best-performing XAI method at the time of this study. The application of AIME to complex models is a subject for future work.

Our study is the first substantial step in ensuring ML accuracy while simultaneously preserving XAI, which is a critical milestone in the field and paves the way for more transparent and reliable decision-making processes. Given the growing demand for transparent ML in healthcare, finance, public policy-making, and elsewhere, this research makes an essential contribution in many critical areas.

ACKNOWLEDGMENT

The authors would like to thank the Faculty of Informatics, Burapha University, for their support and English proofing, also would like to thank the Department of Data Science, Musashino University for their support, and Editage [www.editage.com] for English language editing, and also would like to thank the role of OpenAI's ChatGPT AI system in facilitating discussion and inspiring innovative ideas during the writing process. Although AI-generated text was not directly used in this study, it was helpful in conceptualizing and refining the content.

REFERENCES

[1] E. Cambria, L. Malandri, F. Mercorio, M. Mezzanzanica, and N. Nobani, "A survey on XAI and natural language explanations," *Inf. Process. Manag.*, vol. 60, no. 1, Jan. 2023, Art. no. 103111, doi: 10.1016/j.ipm.2022.103111.

[2] W. Saeed and C. Omlin, "Explainable AI (XAI): A systematic meta-survey of current challenges and future opportunities," *Knowl.-Based Syst.*, vol. 263, Mar. 2023, Art. no. 110273, doi: 10.1016/j.knosys.2023.110273.

[3] Y.-N. Chuang, G. Wang, F. Yang, Z. Liu, X. Cai, M. Du, and X. Hu, "Efficient XAI techniques: A taxonomic survey," 2023, *arXiv:2302.03225*.

[4] G. Schwalbe and B. Finzel, "A comprehensive taxonomy for explainable artificial intelligence: A systematic survey of surveys on methods and concepts," *Data Mining Knowl. Discovery*, Jan. 2023, doi: 10.1007/s10618-022-00867-8.

[5] R. Dazeley, P. Vamplew, and F. Cruz, "Explainable reinforcement learning for broad-XAI: A conceptual framework and survey," *Neural Comput. Appl.*, vol. 35, no. 23, pp. 16893–16916, Aug. 2023, doi: 10.1007/s00521-023-08423-1.

[6] W. Yang, Y. Wei, H. Wei, Y. Chen, G. Huang, X. Li, R. Li, N. Yao, X. Wang, X. Gu, M. B. Amin, and B. Kang, "Survey on explainable AI: From approaches, limitations and applications aspects," *Hum.-Centric Intell. Syst.*, vol. 3, no. 3, pp. 161–188, Aug. 2023, doi: 10.1007/s44230-023-00038-y.

[7] S. Ali, T. Abuhmed, S. El-Sappagh, K. Muhammad, J. M. Alonso-Moral, R. Confalonieri, R. Guidotti, J. Del Ser, N. Díaz-Rodríguez, and F. Herrera, "Explainable artificial intelligence (XAI): What we know and what is left to attain trustworthy artificial intelligence," *Inf. Fusion*, vol. 99, Nov. 2023, Art. no. 101805, doi: 10.1016/j.inffus.2023.101805.

[8] V. Chamola, V. Hassija, A. Razia Sulthana, D. Ghosh, D. Dhingra, and B. Sikdar, "A review of trustworthy and explainable artificial intelligence (XAI)," *IEEE Access*, vol. 11, pp. 78994–79015, 2023, doi: 10.1109/ACCESS.2023.3294569.

[9] R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan, and R. Ranjan, "Explainable AI (XAI): Core ideas, techniques, and solutions," *ACM Comput. Surv.*, vol. 55, no. 9, pp. 1–33, Sep. 2023, doi: 10.1145/3561048.

[10] C. Molnar. (Apr. 2021). *Interpretable Machine Learning,*. [Online]. Available: https://christophmolnar.com/books/interpretable-machine-learning/

[11] D. Minh, H. X. Wang, Y. F. Li, and T. N. Nguyen, "Explainable artificial intelligence: A comprehensive review," *Artif. Intell. Rev.*, vol. 55, no. 5, pp. 3503–3568, Jun. 2022, doi: 10.1007/s10462-021-10088-y.

[12] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138–52160, 2018, doi: 10.1109/ACCESS.2018.2870052.

[13] Q. Zhang, Y. N. Wu, and S.-C. Zhu, "Interpretable convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8827–8836, doi: 10.1109/CVPR.2018.00920.

[14] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM Comput. Surv.*, vol. 51, no. 5, pp. 1–42, Sep. 2019, doi: 10.1145/3236009.

[15] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso, "Machine learning interpretability: A survey on methods and metrics," *Electronics*, vol. 8, no. 8, p. 832, Jul. 2019, doi: 10.3390/electronics8080832.

[16] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Inf. Fusion*, vol. 58, pp. 82–115, Jun. 2020, doi: 10.1016/j.inffus.2019.12.012.

[17] T. Speith, "A review of taxonomies of explainable artificial intelligence (XAI) methods," in *Proc. ACM Conf. Fairness, Accountability, Transparency*, New York, NY, USA, Jun. 2022, pp. 2239–2250, doi: 10.1145/3531146.3534639.

[18] W. Samek and K. R. Müller, "Towards explainable artificial intelligence," in *Explainable AI: Interpreting, Explaining, and Visualizing Deep Learning*, W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, and K. R. Müller, Eds. Cham, Switzerland: Springer, 2019, pp. 5–22, doi: 10.1007/978-3-030-28954-6_1.

[19] M. Van Lent, W. Fisher, and M. Mancuso, "An explainable artificial intelligence system for small-unit tactical behavior," in *Proc. Natl. Conf. Artif. Intell.*, Cambridge, MA, USA, 2004, pp. 900–907.

[20] D. Gunning and D. Aha, "DARPA's explainable artificial intelligence (XAI) program," *AI Mag.*, vol. 40, no. 2, pp. 44–58, Jun. 2019, doi: 10.1145/3301275.3308446.

[21] T. Nakanishi, "Approximate inverse model explanations (AIME): Unveiling local and global insights in machine learning models," *IEEE Access*, vol. 11, pp. 101020–101044, 2023, doi: 10.1109/ACCESS.2023.3314336.

[22] S. Poria, D. Hazarika, N. Majumder, G. Naik, E. Cambria, and R. Mihalcea, "MELD: A multimodal multi-party dataset for emotion recognition in conversations," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 527–536, doi: 10.18653/v1/p19-1050.

[23] S. Y. Chen, C. C. Hsu, C. C. Kuo, T. H. K. Huang, and L. W. Ku, "EmotionLines: An emotion corpus of multi-party conversations," in *Proc. 11th Int. Conf. Lang. Res. Eval.*, 2019, pp. 1597–1601.

[24] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626, doi: 10.1109/ICCV.2017.74.

[25] A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje, "Not just a black box: Learning important features through propagating activation differences," 2016, *arXiv:1605.01713*.

[26] V. Petsiuk, A. Das, and K. Saenko, "RISE: Randomized input sampling for explanation of black-box models," 2018, *arXiv:1806.07421*.

[27] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, D. Precup and Y. W. Teh, Eds. 2017, pp. 3319–3328. [Online]. Available: http://proceedings.mlr.press/v70/sundararajan17a/sundararajan17a.pdf

[28] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833, doi: 10.1007/978-3-319-10590-1_53.

[29] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001, doi: 10.1214/aos/1013203451.

[30] Q. Zhao and T. Hastie, "Causal interpretations of black-box models," *J. Bus. Econ. Statist.*, vol. 39, no. 1, pp. 272–281, Jan. 2021, doi: 10.1080/07350015.2019.1624293.

[31] A. Goldstein, A. Kapelner, J. Bleich, and E. Pitkin, "Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation," *J. Comput. Graph. Statist.*, vol. 24, no. 1, pp. 44–65, Jan. 2015, doi: 10.1080/10618600.2014.907095.

[32] A. Fisher, C. Rudin, and F. Dominici, "All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously," *J. Mach. Learn. Res.*, vol. 20, no. 177, pp. 1–81, 2019.

[33] J. Liu, N. Danait, S. Hu, and S. Sengupta, "A leave-one-feature-out wrapper method for feature selection in data classification," in *Proc. 6th Int. Conf. Biomed. Eng. Informatics*, 2013, pp. 656–660, doi: 10.1109/BMEI.2013.6747021.

[34] A. Erdem. *LOFO Importance*. Github. Accessed: May 21, 2024. [Online]. Available: https://github.com/aerdem4/lofo-importance

[35] M. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?' Explaining the predictions of any classifier," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Demonstrations*, 2016, pp. 1135–1144, doi: 10.18653/v1/n16-3020.

[36] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–10.

[37] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Comput. Electr. Eng.*, vol. 40, no. 1, pp. 16–28, Jan. 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[38] A. Jovic, K. Brkic, and N. Bogunovic, "A review of feature selection methods with applications," in *Proc. 38th Int. Conv. Inf. Commun. Technol., Electron. Microelectron. (MIPRO)*, May 2015, pp. 1200–1205, doi: 10.1109/MIPRO.2015.7160458.

[39] B. Remeseiro and V. Bolon-Canedo, "A review of feature selection methods in medical applications," *Comput. Biol. Med.*, vol. 112, Sep. 2019, Art. no. 103375, doi: 10.1016/j.compbiomed.2019.103375.

[40] S. Solorio-Fernández, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad, "A review of unsupervised feature selection methods," *Artif. Intell. Rev.*, vol. 53, no. 2, pp. 907–948, Feb. 2020, doi: 10.1007/s10462-019-09682-y.

[41] L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," *J. Mach. Learn. Res.*, vol. 5, pp. 1205–1224, Dec. 2004.

[42] Y. Siti Ambarwati and S. Uyun, "Feature selection on Magelang duck egg candling image using variance threshold method," in *Proc. 3rd Int. Seminar Res. Inf. Technol. Intell. Syst. (ISRITI)*, Yogyakarta, Indonesia, Dec. 2020, pp. 694–699, doi: 10.1109/ISRITI51436.2020.9315486.

[43] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Mar. 2003.

[44] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc. Ser. B, Stat. Methodol.*, vol. 58, no. 1, pp. 267–288, Jan. 1996, doi: 10.1111/j.2517-6161.1996.tb02080.x.

[45] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Trans. Neural Netw.*, vol. 5, no. 4, pp. 537–550, Jul. 1994, doi: 10.1109/72.298224.

[46] F. Xiao, G. Xiao, X. Wang, J. Zheng, Y. Yan, and H. A. Wang, "A hierarchical voting scheme for robust geometric model fitting," in *Proc. 9th Int. Conf. Image Graph.*, vol. 9, Shanghai, China. Cham, Switzerland: Springer, Sep. 2017, pp. 11–22, doi: 10.1007/978-3-319-71607-7_2.

[47] M. B. Kursa and W. R. Rudnicki, "Feature selection with the Boruta package," *J. Stat. Softw.*, vol. 36, no. 11, pp. 1–13, 2010, doi: 10.18637/jss.v036.i11.

[48] A. U. Haq, D. Zhang, H. Peng, and S. U. Rahman, "Combining multiple feature-ranking techniques and clustering of variables for feature selection," *IEEE Access*, vol. 7, pp. 151482–151492, 2019, doi: 10.1109/ACCESS.2019.2947701.

[49] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, no. 1, pp. 389–422, 2002, doi: 10.1023/A:1012487302797.

[50] C. Y. Freytes, R. P. Mayrand, L. O. Sawada, T. Y. Liang, R. E. C. Cid, S. Burke, D. Loewenstein, R. Duara, and M. Adjouadi, "Recursive feature elimination with cross validation for Alzheimer's disease classification using cognitive exam scores," in *IEEE MTT-S Int. Microw. Symp. Dig.*, Jul. 2023, pp. 327–332, doi: 10.1109/IMSA58542.2023.10217660.

[51] Q. Liu and A. H. Sung, "Recursive feature addition for gene selection," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Jul. 2006, pp. 1360–1367, doi: 10.1109/IJCNN.2006.246851.

[52] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, Dec. 1997, doi: 10.1016/s0004-3702(97)00043-x.

[53] G. Borboudakis and I. Tsamardinos, "Forward-backward selection with early dropping," 2017, *arXiv:1705.10770*.

[54] M. Siebers and U. Schmid, "Interleaving forward backward feature selection," in *Proc. Int. Conf. Knowl. Discov. Informat. Retr.*, vol. 2, 2010, pp. 454–457, doi: 10.5220/0003093204540457.

[55] K. Z. Mao, "Orthogonal forward selection and backward elimination algorithms for feature subset selection," *IEEE Trans. Syst., Man, Cybern., B*, vol. 34, no. 1, pp. 629–634, Feb. 2004, doi: 10.1109/TSMCB.2002.804363.

[56] P. M. Domingos, "Control-sensitive feature selection for lazy learners," *Artif. Intell. Rev.*, vol. 11, no. 1, pp. 227–253, 1997, doi: 10.1023/A:1006508722917.

[57] F. Kamalov, S. Elnaffarr, A. Cherukuri, and A. Jonnalagadda, "Forward feature selection: Empirical analysis," *J. Intell. Syst. Internet Things*, vol. 11, no. 1, pp. 44–54, 2024, doi: 10.54216/jisiot.110105.

[58] K. Liu, Y. Fu, L. Wu, X. Li, C. Aggarwal, and H. Xiong, "Automated feature selection: A reinforcement learning perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 3, pp. 2272–2284, Mar. 2023, doi: 10.1109/TKDE.2021.3115477.

[59] Z. O. Hamad, "Review of feature selection methods using optimization algorithm," *Polytech. J.*, vol. 12, no. 2., pp. 203–214, 2023, doi: 10.25156/ptj.v12n2y2022.pp203-214.

[60] X. Wang and Y. Zhou, "A review of multi-label feature selection," *J. Phys., Conf.*, vol. 2504, no. 1, May 2023, Art. no. 012007, doi: 10.1088/1742-6596/2504/1/012007.

[61] J. Jemai and A. Zarrad, "Feature selection engineering for credit risk assessment in retail banking," *Information*, vol. 14, no. 3, p. 200, Mar. 2023, doi: 10.3390/info14030200.

[62] R. Penrose, "A generalized inverse for matrices," *Math. Proc. Cambridge Phil. Soc.*, vol. 51, no. 3, pp. 406–413, Jul. 1955, doi: 10.1017/s0305004100030401.

[63] E. H. Moore, "On the reciprocal of the general algebraic matrix," *Bull. Amer. Math. Soc.*, vol. 26, pp. 294–300, Jan. 1920.

[64] D. Aha, "UCI machine learning repository: Center for machine learning intelligent systems," School Inf. Comput. Sci., Univ. California, Irvine, CA, USA, 2017. [Online]. Available: https://archive.ics.uci.edu/ml

[65] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Comput. Surv.*, vol. 50, no. 6, pp. 1–45, 2018. [Online]. Available: https://jundongl.github.io/scikit-feature/datasets.html

[66] F. Kiliç, Y. Kaya, and S. Yildirim, "A novel multi population based particle swarm optimization for feature selection," *Knowl.-Based Syst.*, vol. 219, May 2021, Art. no. 106894, doi: 10.1016/j.knosys.2021.106894.

[67] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.
[68] H. Zhang, S. Si, and C.-J. Hsieh, "GPU-acceleration for large-scale tree boosting," 2017, *arXiv:1706.08359*.
[69] P. Ekman and W. Friesen. (1978). *Facial Action Coding System*. APA PsycNet Direct. [Online]. Available: https://psycnet.apa.org/doiLanding?doi=10.1037%2Ft27734-000
[70] F. Eyben, M. Wöllmer, and B. Schuller, "Opensmile: The Munich versatile and fast open-source audio feature extractor," in *Proc. 18th ACM Int. Conf. Multimedia*, Florence, Italy, Oct. 2010, pp. 1459–1462, doi: 10.1145/1873951.1874246.
[71] F. Eyben, F. Weninger, F. Gross, and B. Schuller, "Recent developments in openSMILE, the Munich open-source multimedia feature extractor," in *Proc. 21st ACM Int. Conf. Multimedia*, Barcelona, Spain, Oct. 2013, pp. 835–838, doi: 10.1145/2502081.2502224.
[72] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion," *J. Personality Social Psychol.*, vol. 17, no. 2, pp. 124–129, 1971, doi: 10.1037/h0030377.
[73] G. Bingöl, S. Porcu, A. Floris, and L. Atzori, "QoE estimation of WebRTC-based audio-visual conversations from facial and speech features," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 20, no. 5, pp. 1–23, May 2024, doi: 10.1145/3638251.
[74] A. Minematsu and T. Nakanishi, "A factor extraction of preference in musical performances using AIME with focus on time-series derivative features," in *Proc. 15th Int. Congr. Adv. Appl. Informat. Winter*, Dec. 2023, pp. 312–317, doi: 10.1109/IIAI-AAI-Winter61682.2023.00064.
[75] S. Lei, G. Dong, X. Wang, K. Wang, and S. Wang, "InstructERC: Reforming emotion recognition in conversation with a retrieval multi-task LLMs framework," 2023, *arXiv:2309.11911*.
[76] D. Mamieva, A. B. Abdusalomov, A. Kutlimuratov, B. Muminov, and T. K. Whangbo, "Multimodal emotion detection via attention-based fusion of extracted facial and speech features," *Sensors*, vol. 23, no. 12, p. 5475, Jun. 2023, doi: 10.3390/s23125475.

TAKAFUMI NAKANISHI (Member, IEEE) was born in Ise, Mie, Japan, in 1978. He received the Ph.D. degree in engineering from the Graduate School of Systems and Information Engineering, University of Tsukuba, in March 2006. In April 2018, he became an Associate Professor with the Global Communication Center. Since April 2019, he has been the Department of Mathematical Engineering, Faculty of Engineering, Musashino University, as an Associate Professor. He engaged in the research and development of knowledge cluster systems with the National Institute of Information and Communications Technology, International University, where he is engaged in text and data mining methods in April 2006. He was at the Department of Data Science, Faculty of Data Science, Musashino University. His research interests include XAI, data mining, emotional information processing, and media content analyses.

PONLAWAT CHOPHUK (Member, IEEE) received the B.Eng. degree in electronic engineering and the M.Eng. degree in biomedical engineering from the King Mongkut's Institute of Technology Ladkrabang, in 2014 and 2016, respectively, and the Ph.D. degree in electrical and information engineering technology from the King Mongkut's University of Technology Thonburi, Thailand, in 2022. His current research interests include computer vision, image processing, human–computer interaction, and machine learning.

KRISANA CHINNASARN (Senior Member, IEEE) received the B.Sc. degree in statistics from Srinakharinwirot University, Mahasarakham Campus, Thailand, in 1992, the M.Sc. degree in computer science and information technology from the King Mongkut's Institute of Technology Ladkrabang, Thailand, in 1997, and the Ph.D. degree in computer science from Chulalongkorn University, Thailand, in 2004. Since 1997, he has been a Lecturer with the Faculty of Informatics, Burapha University, formerly known as the Department of Computer Science, Faculty of Science. He has been an Associate Professor in computer science, since 2024. His research interests include machine learning and digital image processing and their applications to other science and engineering areas.