

Potato Disease Detection Using Joint Feature and Hyperparameter Optimization of Feature Tokenizer-Transformer with iHow

Faris H. Rizk^{1*}, Amel Ali Alhussan², Doaa Sami Khafaga²,
Marwa M. Eid^{3,4}, El-Sayed M. El-kenawy^{5,6*},
Ebrahim A. Mattar⁷, Mohamed Saber⁸

¹Department of Communications and Electronics, Delta Higher Institute of Engineering and Technology, Mansoura 35111, Egypt.

²Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia.

³Faculty of Artificial Intelligence, Delta University for Science and Technology, Mansoura, Egypt.

⁴Jadara University Research Center, Jadara University, Jordan.

⁵Department of Programming, School of Information and Communications Technology (ICT), Bahrain Polytechnic, PO Box 33349, Isa Town, Bahrain.

⁶Applied Science Research Center. Applied Science Private University, Amman, Jordan.

⁷College of Engineering University of Bahrain, Bahrain.

⁸Electronics and Communications Engineering Department, Faculty of Engineering, Delta University for Science and Technology, Gamasa City 11152, Egypt.

*Corresponding author(s). E-mail(s): faris.hamdi.rizk@gmail.com; skenawy@ieee.org;

Contributing authors: aaalhussan@pnu.edu.sa; dskhafga@pnu.edu.sa;
mmm@ieee.org; ebmattar@uob.edu.bh;
Mohamed.saber@deltauniv.edu.eg;

Abstract

Potato leaf diseases present a significant challenge to global food security, requiring the development of efficient and accurate diagnostic systems that can operate effectively under the constraints of real-world agricultural environments. This study introduces a novel high-performance classification framework that combines the Feature Tokenizer Transformer (FT-Transformer) with the Improved iHow Optimization Algorithm, a metaheuristic algorithm previously validated for simultaneous feature selection and hyperparameter tuning. The FT-Transformer model, in isolation, demonstrates superior performance compared to baseline models, achieving an accuracy of 81.61% and an F1-score of 81.45%, thereby establishing a robust baseline for further optimization. By incorporating the iHow algorithm for feature selection, the framework achieves a refined accuracy of 90.90%, alongside significant improvements in precision, recall, and generalization, while effectively reducing feature redundancy. Subsequent hyperparameter optimization with iHow further enhances the model, achieving peak performance with an accuracy of 98.35% and an F1-score of 98.33%, significantly outperforming all compared metaheuristic variants. Additionally, the iHow+FT-Transformer configuration excels in computational efficiency, with the shortest training time (12.45s), minimal memory usage (256.8MB), and the highest overall efficiency score (0.9847). These results collectively demonstrate that the proposed framework is a scalable, robust, and resource-efficient solution for real-time plant disease classification, offering substantial benefits for precision agriculture and intelligent crop monitoring systems, particularly in low-resource settings.

Keywords: Potato Disease Classification, Transformer-Based Models, iHow Metaheuristic Optimization, Feature Selection and Hyperparameter Tuning, Precision Agriculture

1 Introduction

The global demand for sustainable and intelligent agriculture has intensified in recent years due to the combined pressures of climate change, population growth, and the resulting need to ensure food security. One of the most formidable obstacles threatening agricultural productivity is the widespread occurrence of crop diseases, which not only compromise yield quantity and quality but also inflict considerable economic and social impacts—particularly in regions with limited access to real-time plant health diagnostics [1, 2]. Within this context, potatoes (*Solanum tuberosum*), a globally important staple crop, are especially vulnerable to a range of bacterial and fungal infections that can lead to catastrophic losses if not identified and treated promptly [3].

Adequate disease control hinges on early-stage diagnosis and targeted intervention. However, conventional diagnostic methods—often reliant on manual inspection by trained experts—are inherently limited in scalability, consistency, and speed [4]. These limitations become especially pronounced in large-scale farms or rural settings where expert access is constrained. To overcome these challenges, there has been a pronounced shift toward the adoption of digital and data-driven approaches. Among

these, artificial intelligence (AI), and more specifically machine learning (ML) and deep learning (DL), has emerged as a transformative force for modernizing agricultural disease detection systems [5, 6].

The integration of ML techniques into crop health monitoring pipelines enables automated, real-time, and scalable disease classification based on image analysis. Unlike traditional rule-based methods, ML models can learn intricate visual features from training data, allowing them to recognize subtle pathological cues such as lesion patterns, discoloration, or texture anomalies [7]. These capabilities significantly enhance the sensitivity and specificity of disease detection systems, reduce the burden on human labor, and provide actionable insights that can be incorporated into precision agriculture workflows. Furthermore, the potential for deployment on edge devices, such as smartphones or embedded agricultural sensors, renders ML-based solutions highly adaptable to diverse farming environments, including resource-constrained rural regions [8].

In recent years, transformer-based deep learning architectures have revolutionized various domains of artificial intelligence, beginning with natural language processing and extending to computer vision [9]. Their hallmark attention mechanism enables models to capture long-range dependencies and contextual interactions among features—capabilities that have proven instrumental in advancing the state of the art in complex visual and structured data domains. One such architecture, the Feature Tokenizer Transformer (FT-Transformer) [10], has demonstrated robust performance on heterogeneous data by combining dense feature encoding with interpretable attention pathways [11]. These characteristics make FT-Transformer an attractive candidate for image-based agricultural applications, where disease manifestations can be spatially diffuse and structurally variable across leaf surfaces.

However, deploying transformer-based models for disease classification from plant images is not without its challenges. First, high-resolution images inherently contain a vast number of pixel features, leading to high-dimensional input spaces. Many of these features are redundant or irrelevant to the classification task, thereby introducing noise, increasing computational burden, and elevating the risk of overfitting [12]. Second, image datasets collected in real agricultural environments often contain noise due to poor lighting, occlusions, inconsistent focus, and background clutter [13]. These variations degrade the quality of visual information and impede the model's ability to extract disease-relevant features. Third, transformer models are susceptible to their hyperparameters, including learning rate, number of attention heads, hidden dimensions, and dropout probabilities [14]. Suboptimal tuning can hinder convergence and result in diminished generalization performance, especially in datasets with imbalanced class distributions or limited sample sizes.

Addressing these challenges requires a unified strategy that not only enhances model architecture but also incorporates intelligent preprocessing, dimensionality reduction, and optimization mechanisms. Feature selection serves as a vital preprocessing step that mitigates the curse of dimensionality by isolating informative visual descriptors while discarding irrelevant attributes [15]. In parallel, hyperparameter optimization seeks to fine-tune model behavior for maximal predictive performance under given data constraints. Metaheuristic algorithms are particularly well-suited for these

tasks due to their flexibility, scalability, and ability to navigate complex, non-convex search spaces without reliance on gradient information [16].

In this study, we propose an integrated pipeline that couples the FT-Transformer with the Improved Hybrid Optimization Wrapper (iHOW), a previously validated metaheuristic optimization strategy. Unlike traditional approaches that isolate feature selection and hyperparameter tuning, iHOW facilitates both processes within a single, harmonized optimization framework. The resulting system is capable of learning compact yet discriminative feature subsets while simultaneously calibrating model parameters to achieve optimal performance. This synergy enhances classification accuracy, reduces computational overhead, and promotes generalizability to unseen data.

The primary objectives of this research are threefold: (1) to rigorously evaluate the classification performance of the FT-Transformer on a curated image dataset of potato leaf diseases; (2) to augment this baseline performance through joint feature selection and hyperparameter optimization using the iHOW algorithm; and (3) to benchmark the proposed approach against a suite of contemporary deep learning models and optimization algorithms, spanning diverse architectural paradigms and search heuristics. Through this comprehensive investigation, we aim to contribute a robust, scalable, and efficient solution to the growing field of AI-driven plant disease diagnostics, with direct applications in precision agriculture and innovative farming ecosystems.

The core contributions of this study are summarized below, highlighting the novelty, technical significance, and practical relevance of the proposed methodology within the broader context of intelligent agriculture and deep learning optimization:

- We present a unified classification framework that integrates the FT-Transformer—a cutting-edge transformer-based model for tabular and image-derived data—with the Improved iHow Optimization Algorithm to address the dual challenges of feature redundancy and hyperparameter tuning in agricultural disease diagnosis.
- The iHow algorithm is employed in a dual-role capacity: first, as a binary feature selector to eliminate irrelevant and redundant visual patterns, and second, as a continuous optimizer to fine-tune the FT-Transformer’s hyperparameters. This unified strategy ensures both compact feature representations and model generalization without sacrificing accuracy.
- A comprehensive and structured preprocessing pipeline, partially guided by the large language model Qwen3-0.6B, is applied to enhance image quality, reduce noise, and support class balance—thereby strengthening the overall robustness of the classification system.
- We benchmark the proposed iHow + FT-Transformer framework against a diverse set of state-of-the-art machine learning models and metaheuristic optimizers, covering various architectural paradigms including attention-based, decision-tree-inspired, and hybrid deep learning designs.
- The framework is evaluated across multiple stages—including baseline classification, post-feature selection refinement, hyperparameter tuning, and computational profiling—demonstrating its adaptability and efficiency under realistic operational constraints.

- The resulting pipeline is designed with scalability and real-world deployment in mind, offering an effective solution for precision agriculture applications such as real-time plant disease monitoring in low-resource settings.

The remainder of this paper is structured as follows. Section 2 presents a comprehensive literature review, covering recent advancements in transformer-based classification, metaheuristic optimization techniques, and their applications in plant disease diagnosis. Section 3 details the materials and methods, including dataset description, Qwen3-0.6B-guided preprocessing, model architectures, and the iHow-based optimization framework. Section 4 outlines the evaluation metrics used to assess both classification performance and optimization effectiveness. Section 5 reports the experimental results, which include baseline performance, feature selection comparisons, post-optimization benchmarks, and computational efficiency analysis. Section 6 provides an in-depth discussion of the results, addressing optimization impacts, model generalizability, and deployment implications. Finally, Section 7 concludes the study and suggests future directions for extending the proposed methodology across broader agricultural and real-time systems.

2 Literature Review

The field of potato leaf disease detection has seen a remarkable evolution, primarily driven by advancements in machine learning (ML) and deep learning (DL) technologies. This literature review organizes recent contributions into three thematic categories: (1) Conventional and hybrid ML models, (2) Convolutional neural networks and optimized architectures, and (3) Multi-stage and advanced deep learning frameworks.

2.1 Machine Learning Techniques and Feature Optimization

Traditional ML algorithms, though less data-hungry than deep learning models, have proven valuable when coupled with advanced feature engineering and optimization techniques. Radwan et al. [17] explored various ML models—logistic regression, gradient boosting, support vector machines (SVM), and multilayer perceptrons (MLP)—to predict early and late blight in potato leaves using a structured weather dataset. Employing sophisticated feature selection methods, including the binary Greylag Goose Optimization (bGGO), they demonstrated that optimized MLP models could achieve an impressive 98.3% accuracy, emphasizing the synergy between model selection and feature optimization.

2.2 Optimized Deep Learning Architectures

A significant shift towards deep learning is evident in recent research, particularly in the use of optimized CNNs for disease classification. Dey et al. [18] proposed a lightweight convolutional neural network that maintained high classification performance (98.6% test accuracy) with minimal computational overhead, making it practical for deployment on resource-constrained devices. Similarly, Nazir et al. [19] introduced EfficientPNet, an EfficientNet-V2-based DL architecture enhanced with

spatial-channel attention and transfer learning techniques. Their model demonstrated 98.12% accuracy on the PlantVillage dataset, confirming its robustness under varied environmental conditions.

In a related effort, Pandiri et al. [20] developed POT-Net, a CNN optimized via the Whale Optimization Algorithm (WOA), achieving state-of-the-art accuracy of 99.12%. The metaheuristic-based tuning of hyperparameters highlighted the benefits of integrating optimization strategies into DL pipelines.

2.3 Advanced and Hybrid Deep Learning Frameworks

Going beyond single-network architectures, several researchers have developed hybrid or hierarchical frameworks to tackle the complexity of real-world scenarios. Alzakari et al. [21] combined CNNs with Long Short-Term Memory (LSTM) networks to exploit spatial-temporal features for early blight detection, achieving a classification accuracy of 97.1%. The fusion of CNN for feature extraction and LSTM for sequential learning underscores the importance of modeling temporal dependencies in disease progression.

Mahum et al. [22] proposed an enhanced DenseNet-based model capable of classifying five disease categories, including Potato Leaf Roll and Verticillium Wilt. Their novel integration of transition layers and reweighted cross-entropy loss effectively addressed data imbalance issues and achieved 97.2% accuracy, representing a comprehensive solution for fine-grained disease categorization.

Kumar and Patel [23] took a hierarchical approach by combining Intuitionistic Fuzzy Local Binary Patterns (IFLBP) for feature extraction with a Hierarchical Deep Learning Convolutional Neural Network (HDLCNN). Their framework improved specificity, sensitivity, and accuracy metrics over several baseline models, demonstrating the added value of integrating fuzzy logic and deep feature hierarchies.

2.4 Model Efficiency, Segmentation, and Interpretability

Other studies have focused on improving segmentation and addressing model efficiency under real-world constraints. Li et al. [24] introduced a three-stage framework comprising instance segmentation (Mask R-CNN), classification (VGG16, ResNet50, InceptionV3), and semantic segmentation (UNet, PSPNet, DeepLabV3+), achieving high precision and pixel-level accuracy for disease detection in complex backgrounds. This approach addressed the challenges of noisy and heterogeneous agricultural environments.

Restrepo-Arias et al. [25] emphasized the role of image texture and Bayesian optimization in training compact CNNs like MobileNet and SqueezeNet, achieving respectable accuracies (96.31% and 95.05%, respectively) despite limited computational resources. Their method presents a viable strategy for disease detection in low-power IoT-based agricultural monitoring systems.

Although not specific to potato plants, Mohapatra et al. [26] demonstrated the power of hybrid metaheuristics (Cat Swarm Optimization and Black Widow Optimization) in enhancing CNN performance for mango leaf disease classification. Their strategy involving histogram equalization, fuzzy segmentation, and optimized classification can be effectively adapted to potato disease contexts.

In summary, the literature reveals a clear trend toward hybrid and optimized DL models for robust potato leaf disease classification. The integration of optimization algorithms, attention mechanisms, and hybrid learning strategies has significantly improved the performance and generalizability of disease detection models. These innovations not only bolster precision agriculture but also pave the way for real-time, scalable, and resource-efficient deployment in farming environments.

A comparative summary is presented in Table 1, highlighting the diversity of models, datasets, and performance metrics across recent works.

Table 1 Summary of recent approaches for potato leaf disease detection and classification

Study	Model Type	Key Techniques	Tech-	Dataset	Accuracy (%)
Radwan et al. [17]	ML (MLP, SVM, KNN, etc.)	bGGO, PCA, Clustering, Feature Selection	Weather data (4000+ records)		98.3
Dey et al. [18]	Lightweight CNN	Shallow optimized CNN, Real-time capable	Custom image dataset		98.6
Nazir et al. [19]	DL (EfficientNet-V2)	Attention, Transfer learning, Dense layers	PlantVillage (10,800 images)		98.12
Pandiri et al. [20]	CNN (POT-Net)	Whale Optimization Algorithm (WOA)	Phenotyping image dataset		99.12
Alzakari et al. [21]	CNN-LSTM	Z-score normalization, Temporal learning	Image dataset (unspecified size)		97.1
Mahum et al. [22]	Enhanced DenseNet-201	Reweighted loss, Multi-disease detection	PlantVillage + 3 custom classes (3852 images)		97.2
Kumar and Patel [23]	HDLCNN	IFLBP, DSS, Fuzzy LBP, Median filtering	Custom dataset (Matlab Simulink)		~97.5 (est.)
Li et al. [24]	3-Stage DL (Mask R-CNN + Classifier + Semantic Seg.)	Segmentation, Ensemble pipeline	Real-world leaf images with complex background		95.3 (Cls.), 97.13 (Seg.)
Restrepo-Arias et al. [25]	Small CNNs (e.g., MobileNet, SqueezeNet)	Texture features, Bayesian Optimization	PlantVillage (85x85 px images)		96.31 (MobileNet)
Mohapatra et al. [26]	CNN + CSUBW	Cat Swarm + Black Widow Optimization	Mango leaf dataset (not potato)		95.1 (SVM baseline, CNN-based not specified)

2.5 Research Gap and Our Contribution

While recent studies have made substantial progress in applying machine learning and deep learning techniques to potato disease classification, several critical gaps remain unaddressed in the current literature.

One prominent limitation is the fragmented treatment of optimization, where feature selection and hyperparameter tuning are often pursued as separate objectives. Many studies rely on conventional approaches such as grid search or simple evolutionary strategies. These disjointed methodologies limit scalability and fail to fully exploit the synergistic benefits of joint optimization—especially in high-capacity models like transformers.

Moreover, the literature remains heavily dominated by convolutional neural network (CNN) architectures. While CNNs are effective for visual tasks, they inherently rely on local receptive fields and fixed kernel operations, which constrain their ability to capture complex inter-feature dependencies. In contrast, transformer-based architectures—proven effective in tabular and multimodal domains—remain largely unexplored for image-based plant disease classification. This presents a significant opportunity for innovation.

Another overlooked area is computational efficiency. However, some works propose lightweight models, but few rigorously benchmark training time, memory consumption, or processor load. In real-world agricultural environments, where edge deployment is critical, such metrics are essential for practical usability and adoption.

In addition, most prior works give limited attention to the foundational role of data preprocessing. Pipelines are often either handcrafted or poorly documented, leading to reproducibility challenges and potential quality issues. The integration of large language models, such as Qwen3-0.6B, for structured and standardized preprocessing remains a novel but untapped opportunity in this domain.

To bridge these gaps, our research introduces a unified classification framework based on the FT-Transformer, integrated with the Improved iHow Optimization Algorithm. This dual-stage optimization process simultaneously enhances feature selection and hyperparameter tuning. Our approach not only improves predictive accuracy but also reduces computational overhead. Furthermore, we incorporate Qwen3-0.6B-guided preprocessing to ensure high data consistency and quality. Our method is benchmarked against state-of-the-art models across diverse performance metrics, establishing a robust, scalable, and efficient solution tailored to the needs of modern precision agriculture.

3 Materials and Methods

This section delineates the dataset characteristics, preprocessing protocols, and methodological framework employed in the study. The focus is placed on the integrity and representativeness of the image data, the structured design of preprocessing operations inspired by large language model (LLM) recommendations, and the ethical boundaries maintained throughout the process. An overview of the proposed pipeline, which integrates data preprocessing, FT-Transformer baseline training, and

iHOW-driven feature selection and hyperparameter optimization, is illustrated in Figure 1.

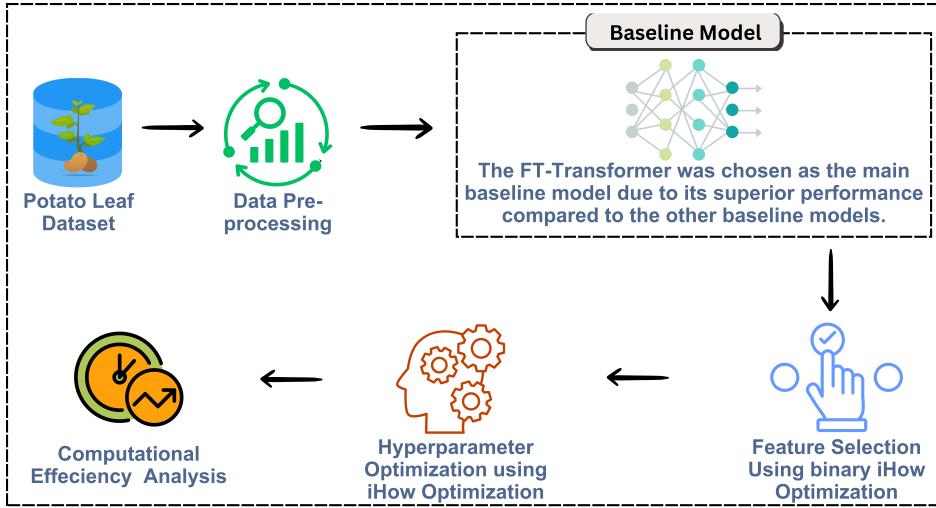


Fig. 1 Proposed methodology pipeline. The system begins with the potato leaf image dataset, followed by preprocessing, model selection, and two-stage iHOW optimization (feature selection and hyperparameter tuning), culminating in performance evaluation through computational efficiency analysis.

3.1 Dataset Description

The dataset utilized in this investigation is a publicly available resource hosted on Kaggle, titled the *Potato Disease Dataset* [27, 28]. It was curated and authenticated by domain experts from the Bangladesh Agricultural Research Institute (BARI), ensuring both scientific reliability and domain-specific relevance. The dataset comprises a total of 451 high-resolution color images representing seven distinct classes: six pathological categories and one healthy control category.

The pathological classes included in the dataset are as follows: Common Scab, Blackleg, Dry Rot, Pink Rot, Black Scurf, and a Miscellaneous disease group, each of which represents visually distinguishable symptoms manifesting on potato leaves or tubers. Additionally, the healthy class provides a baseline for distinguishing disease-free specimens. Each image has been labeled according to its respective disease condition, forming the basis for a supervised multiclass classification task.

The classification objective centers on the automated identification of disease types based on visual patterns present in leaf imagery. The task is treated as a multiclass image classification problem, wherein the target variable corresponds to the disease class of each sample. In the interest of maintaining statistical robustness and mitigating the risk of overfitting, the dataset was partitioned into three distinct subsets: training, validation, and testing. A stratified splitting strategy was employed to preserve the proportional representation of each class across the subsets. This approach ensures

that the classifier is exposed to all disease categories during the learning phase while reserving unseen examples for independent performance evaluation.

3.2 Data Preprocessing

Given the visual nature of the input data and the susceptibility of image-based models to various forms of noise and inconsistency, the preprocessing stage was conducted with particular attention to quality control, normalization, and domain-informed transformations. To guide this stage, we leveraged the generative capabilities of a recent open-source large language model—Qwen3-0.6B—known for its advanced instruction-following and reasoning abilities.

Qwen3-0.6B was consulted for the formulation of a structured and comprehensive image preprocessing pipeline tailored to disease classification tasks [29]. The model was not employed for any data-driven learning or inference but was used exclusively to assist in the design of a preprocessing methodology grounded in best practices across vision-based deep learning applications. Its recommendations served to inform human decisions on pipeline construction, ensuring that no autonomous action was delegated to the LLM itself.

The preprocessing workflow included a sequence of well-established transformations, beginning with image resizing to a fixed input resolution compatible with convolutional neural network architectures. Normalization was subsequently applied to scale pixel intensity values to a standardized range, enhancing model convergence during training. Color space adjustments ensured that all images were encoded in the RGB domain, thereby aligning with the input expectations of the downstream classifier.

To increase the generalization capacity of the model and reduce the risk of overfitting, data augmentation strategies such as horizontal flipping, rotation, and brightness perturbations were applied selectively to the training set. These augmentations simulate variations commonly observed in real-world agricultural imagery and enhance the diversity of the training data. To address class imbalance, oversampling techniques and reweighting schemes were employed, ensuring equitable representation of minority disease categories.

Background removal and region-of-interest (ROI) cropping were applied to reduce the influence of irrelevant background content, thereby focusing the model’s attention on the symptomatic regions of the potato leaves or tubers. In parallel, corrupted or low-quality images were identified and excluded using quality filtering criteria based on entropy, brightness, and sharpness metrics. Duplicate image detection algorithms were also implemented to eliminate redundant samples that might skew the learning process.

Strict ethical standards governed the entire preprocessing pipeline. The utilization of Qwen3-0.6B was confined to non-inferential tasks—specifically, the design of data processing workflows. At no point was the model allowed access to the image data for classification, prediction, or labeling purposes. This ensures the reproducibility, interpretability, and methodological transparency of the study while respecting the limitations and appropriate use cases of LLM technologies in scientific research.

3.3 Data Analysis

A meticulous exploratory data analysis (EDA) was conducted to assess the structural and statistical characteristics of the dataset before model training. This phase served a dual purpose: first, to evaluate the quality and distributional properties of the image data; and second, to identify any latent issues such as data imbalance, label noise, or leakage that could compromise the integrity of the learning process.

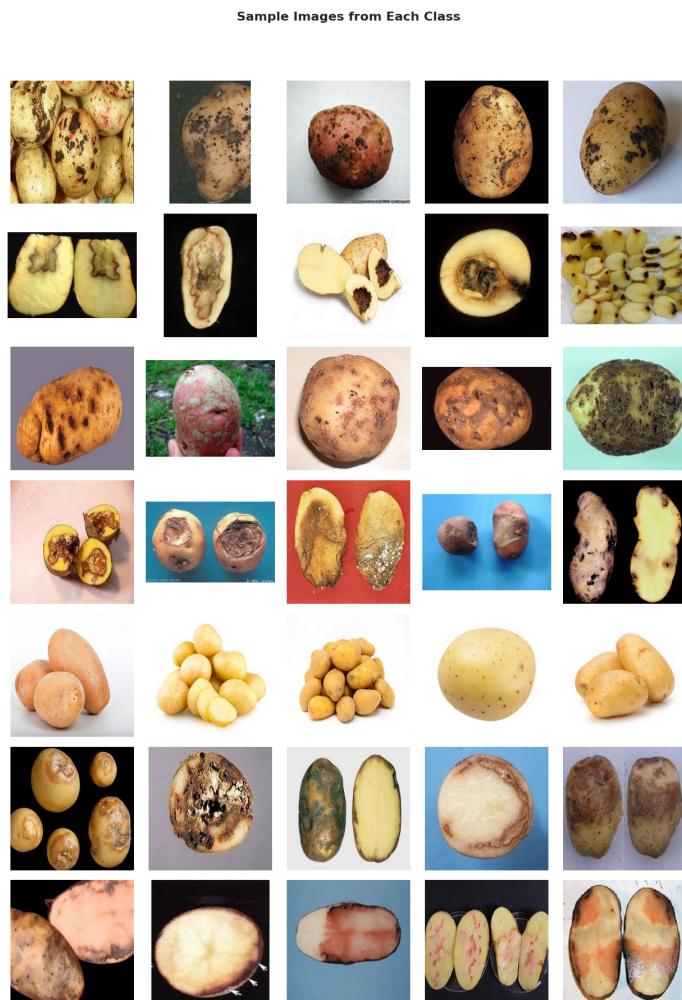


Fig. 2 Sample images from each disease and healthy class.

Figure 2 provides representative images from each class in the potato disease dataset, including both diseased and healthy specimens. The visual diversity across the categories highlights varying manifestations of symptoms such as dark lesions, rot,

and scabbing, which are critical for supervised learning tasks. The images exhibit differences in shape, background, and lighting, underscoring the importance of robust preprocessing and augmentation techniques to ensure generalization across such visual variance.

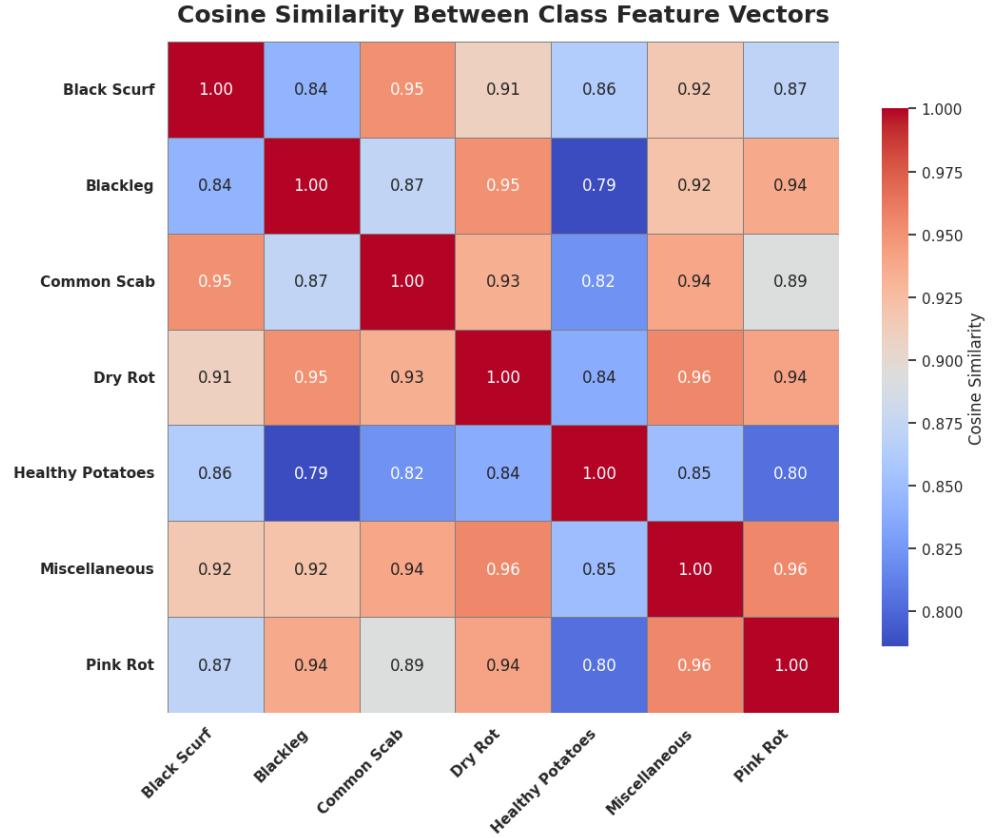


Fig. 3 Cosine similarity heatmap of class-wise feature vectors.

The cosine similarity heatmap in Figure 3 quantifies inter-class similarity based on extracted feature vectors. Notably, high pairwise similarity is observed among disease classes such as Dry Rot, Miscellaneous, and Pink Rot, all exceeding a similarity score of 0.94. This reflects the subtle visual overlap in specific symptoms, which can pose a challenge to classification models. The relatively lower similarity between Healthy Potatoes and infected categories (e.g., 0.79 with Blackleg) confirms the utility of these embeddings in separating diseased from non-diseased instances.

Figure 4 visualizes the high-dimensional feature space using Uniform Manifold Approximation and Projection (UMAP). Each point represents an image, color-coded

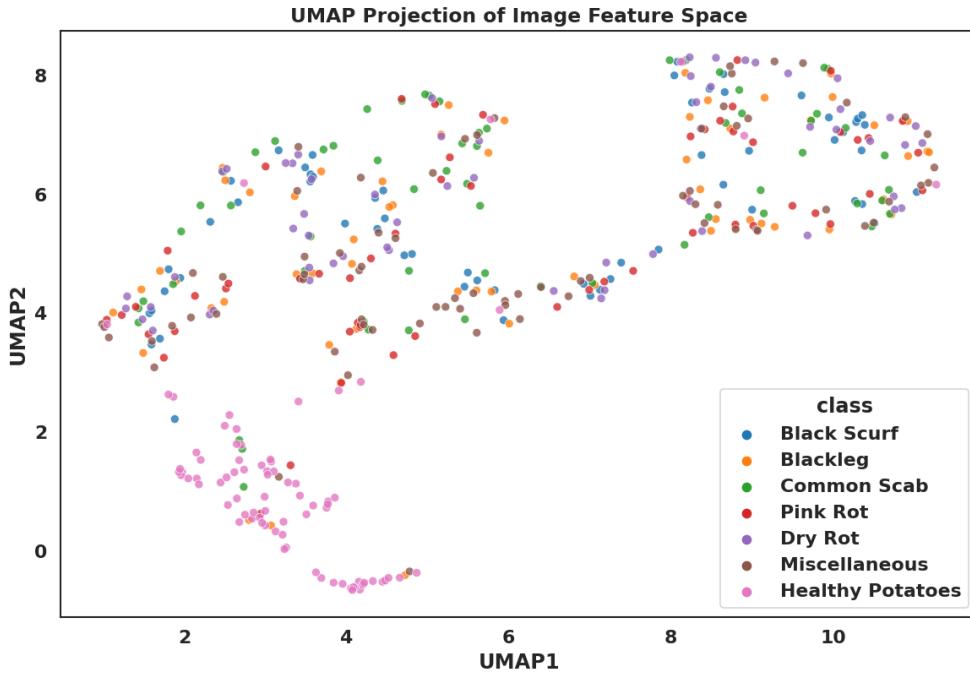


Fig. 4 UMAP projection of the image feature space.

by class. While specific clusters such as Healthy Potatoes (pink) form distinct groupings, several disease categories exhibit partial overlap. This observation emphasizes the complexity of the classification problem and supports the need for discriminative modeling techniques that can exploit nuanced spatial features.

In Figure 5, t-distributed Stochastic Neighbor Embedding (t-SNE) is used to reduce the dimensionality of K-Means-generated clusters. The visualization reveals three coarse clusters, suggesting the possibility of latent macro-level groupings across disease classes. Although not perfectly aligned with class labels, the spatial separation indicates that unsupervised clustering captures some intrinsic structure in the data, validating the effectiveness of the learned features.

Figure 6 presents outlier detection results using Isolation Forests projected onto the first two principal components. Outliers (marked as x) appear sparsely distributed, often occupying extreme regions of the feature space. These data points correspond to images with atypical visual characteristics—such as excessive background clutter or lighting distortions—that could impair model performance. Identifying and filtering such anomalies is a vital preprocessing step to enhance the robustness of the classification pipeline.

In summary, the data analysis phase provided valuable insights into the structure, quality, and distributional properties of the dataset. It validated the robustness of the preprocessing pipeline and ensured that the experimental setup adhered to rigorous

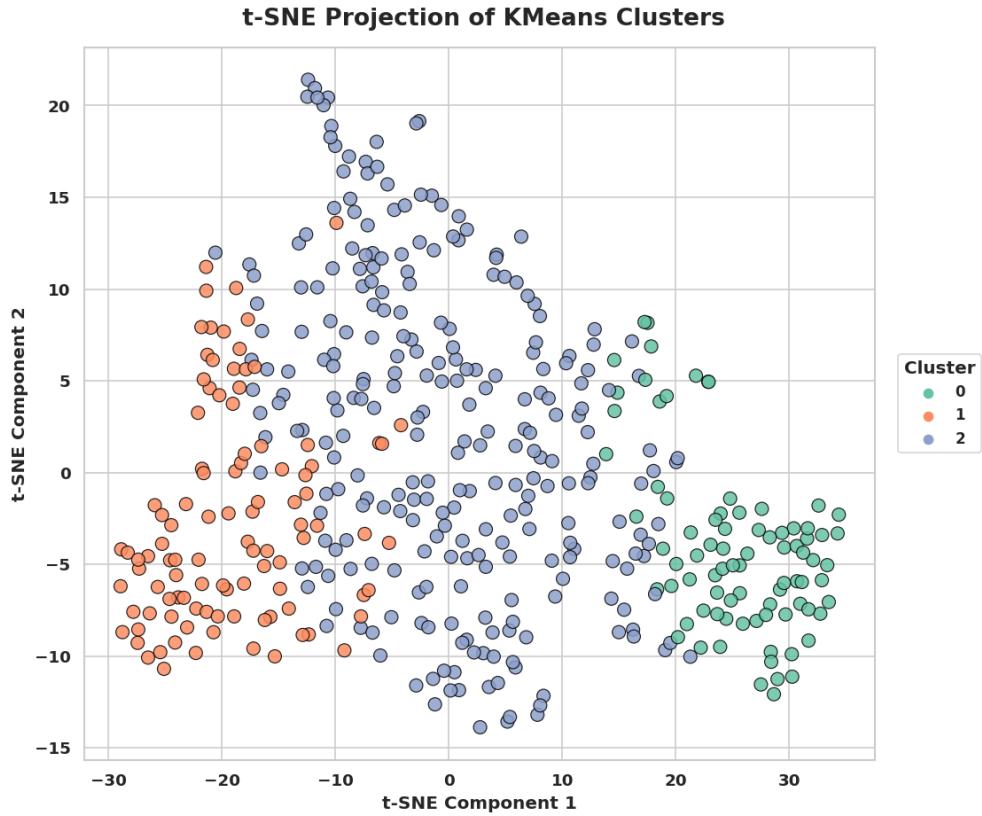


Fig. 5 t-SNE projection of K-Means clusters.

scientific standards. This foundational analysis laid the groundwork for the deployment of advanced machine learning algorithms with confidence in the validity and representativeness of the training data.

3.4 Machine Learning Models

To rigorously evaluate the effectiveness of transformer-based architectures for plant disease classification, we selected a diverse suite of machine learning models spanning multiple architectural paradigms. This set covers classical convolutional neural networks (CNNs), decision-tree-inspired differentiable models, attention-based architectures, and hybrid systems combining multiple learning strategies. The goal was to establish a comprehensive benchmark reflecting the performance boundaries of current state-of-the-art (SOTA) classifiers when applied to structured representations derived from image data.

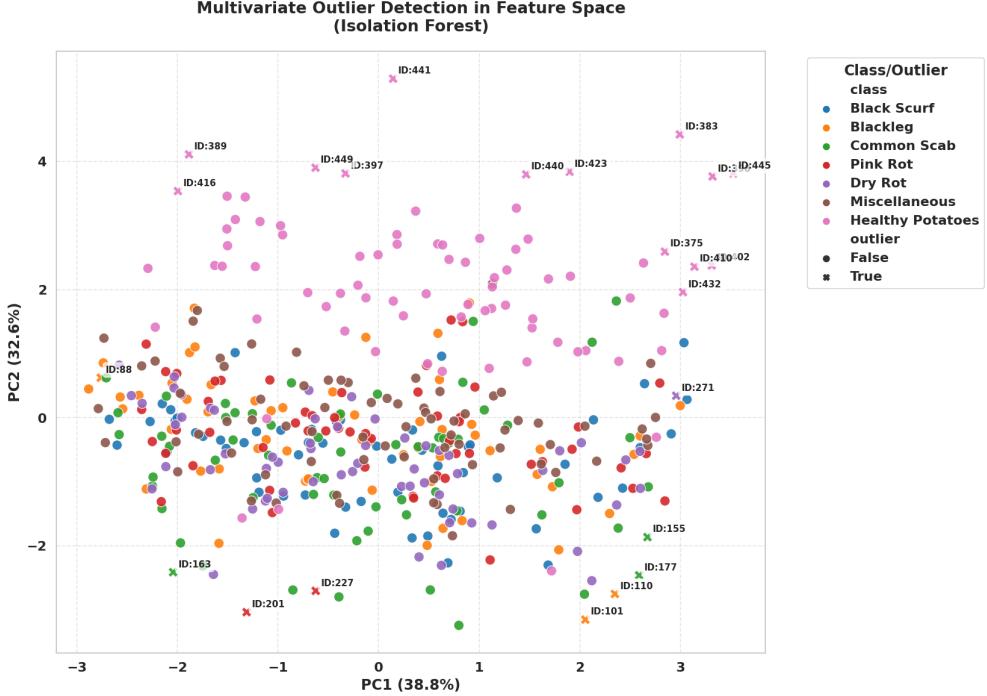


Fig. 6 Multivariate outlier detection using Isolation Forest in PCA space.

The models selected for comparison include: FT-Transformer, SAINT, TabNet, NODE, DeepGBM, xDeepFM, RETab, and LeNet-5. Each was chosen for its empirical success in relevant domains and compatibility with structured or image-derived features.

FT-Transformer [10] is a transformer-based model specifically adapted for tabular and structured data. It introduces a feature tokenization mechanism that converts each feature into a token, enabling the application of multi-head self-attention to model complex inter-feature relationships. While originally intended for tabular learning, its ability to capture high-order dependencies makes it suitable for classification tasks involving engineered features from image preprocessing.

SAINT (Self-Attention and Intersample Attention Transformer) [30] extends the transformer framework by incorporating both column-wise (feature) attention and row-wise (intersample) attention. This dual mechanism allows the model to simultaneously learn dependencies among features and correlations between different data samples. Such capabilities are advantageous in agricultural diagnostics where structured metadata can complement image-derived features.

TabNet [31] is an interpretable deep learning architecture that applies sequential attention to select a sparse subset of relevant features at each decision step. This selective attention enables the model to focus its capacity on the most informative

attributes while maintaining interpretability, making it valuable for explaining the visual or structured indicators associated with specific plant diseases.

NODE (Neural Oblivious Decision Ensembles) [32] merges the representational strengths of oblivious decision trees with the training advantages of deep neural networks. It stacks multiple differentiable oblivious decision trees in an ensemble, allowing end-to-end optimization while preserving the hierarchical decision-making structure typical of tree-based models.

DeepGBM [33] is a hybrid framework that combines gradient boosting decision trees (GBDT) with neural networks. It employs two specialized neural components: CatNN for sparse categorical features and GBDT2NN for dense numerical features distilled from GBDT outputs. This design enables DeepGBM to handle mixed-type structured inputs efficiently while benefiting from both the interpretability of GBDT and the abstraction power of deep learning.

xDeepFM (Extreme Deep Factorization Machine) [34] was initially proposed for recommender systems. It integrates a Compressed Interaction Network (CIN) to explicitly model bounded-degree feature interactions at the vector level, combined with a deep neural component for learning implicit high-order interactions. This architecture is adaptable to structured features extracted from images, enabling it to model both explicit and implicit dependencies.

RETab is a transformer-based model designed for robust performance on noisy and imbalanced tabular datasets. It leverages efficient attention mechanisms and targeted regularization to maintain stability under varying data quality conditions. This makes it relevant in agricultural disease classification scenarios where class imbalance and variability in image-derived features are common.

LeNet-5 [35] is a seminal CNN architecture originally developed for handwritten digit recognition. Despite its simplicity, it remains a strong baseline in vision-based tasks, particularly on small or moderate-sized datasets. Its inclusion here provides a reference point for assessing the relative gains achieved by modern architectures.

In summary, this selection encompasses models with varied inductive biases—ranging from local spatial feature extraction to global attention over structured inputs—ensuring a holistic evaluation. Such diversity allows for a nuanced analysis of strengths and limitations across paradigms, and for validating the performance advantages of the FT-Transformer architecture, especially when integrated with advanced optimization techniques.

3.5 The iHOW Metaheuristic Optimization Strategy

In this study, the Improved iHow Optimization Algorithm is adopted as the principal metaheuristic mechanism for addressing two fundamental optimization problems intrinsic to deep learning models: feature selection and hyperparameter optimization. iHow, as initially presented by El-Kenawy et al. [36], embodies a biologically and cognitively inspired search process designed to emulate the human learning cycle—from data acquisition and knowledge accumulation to adaptive decision-making. The algorithm was formulated to overcome limitations present in traditional metaheuristics, particularly regarding premature convergence, stagnation in local optima, and inefficiency in navigating high-dimensional, multimodal solution spaces.

Mathematical Formulation and Cognitive Design Philosophy. The design of iHow follows a layered cognitive architecture inspired by human problem-solving. Each candidate solution in the population (also called an agent) progresses through several stages akin to the human learning pipeline, namely: *data collection*, *learning*, *information processing*, *knowledge acquisition*, and *expert refinement*.

Let the position of a candidate solution at iteration t be denoted by $X(t) \in \mathbb{R}^n$, where n is the dimension of the optimization space. The algorithm iteratively updates the population through dynamic position update rules modulated by a series of learning rates and adaptive coefficients $\{r_1, r_2, \dots, r_5\}$.

The *exploration phase*, aimed at diversifying the search, utilizes the following update equation:

$$\mathbf{DS}_{t+1} = r_1 \cdot \mathbf{DS}_1 + r_1 r_2 \cdot \mathbf{DS}_2 + r_1 r_2 r_3 \cdot \mathbf{DS}_3, \quad (1)$$

where \mathbf{DS}_i are exploration direction vectors and $r_i \in [0, 1]$ are random coefficients controlling the extent of exploration. This equation promotes stochastic movement toward unexplored regions, minimizing the risk of early convergence.

During the *learning phase*, agents refine their positions based on accumulated experiential knowledge. The learning update is governed by:

$$\mathbf{LS}_{t+1} = r_1 \cdot \mathbf{LS}_1 + r_1 r_2 \cdot \mathbf{LS}_2 + r_1 r_2 r_3 \cdot \mathbf{LS}_3, \quad (2)$$

where \mathbf{LS}_i represents prior learning stages, facilitating deeper exploitation of promising areas.

The transition into the *knowledge update phase* integrates the newly learned insights with previously stored knowledge, guided by an exponentially decaying memory factor K :

$$K = 2 - 2 \times \left(\frac{\text{iteration count}}{\max \text{ iterations}} \right), \quad (3)$$

$$\mathbf{KS}_{t+1} = \mathbf{DS}_{t+1} + \mathbf{LS}_{t+1} + 2K + 1. \quad (4)$$

This adaptive strategy ensures a gradual shift from exploration to exploitation. The exploitation stage further sharpens solution accuracy through localized search:

$$\mathbf{X}_{1,t+1} = \mathbf{X}_t + (\mathbf{KS}_{t+1} + \mathbf{DS}_{t+1}) \cdot r, \quad (5)$$

$$\mathbf{X}_{2,t+2} = \mathbf{X}_{2,t} + r_3 r_4 \cdot (\mathbf{KS}_t + \mathbf{LS}_{t+1}), \quad (6)$$

$$\mathbf{X}_{3,t+3} = \mathbf{X}_{3,t} + (r_3 r_4 r_5 \cdot \mathbf{KS}_t + \mathbf{DS}_{t+1} + \mathbf{LS}_{\text{new}}). \quad (7)$$

The final solution X_{best} at each iteration is computed by aggregating all strategic contributions:

$$\begin{aligned} X_{\text{best}} = & \mathbf{DS}_{t+1} + \mathbf{LS}_{t+1} + \mathbf{KS}_{t+1} \cdot \mathbf{DS}_{t+1} \cdot \mathbf{X}_{1,t+1} \\ & + \mathbf{DS}_{t+1} + \mathbf{LS}_{t+1} + \mathbf{KS}_{t+1} \cdot \mathbf{LS}_{t+1} \cdot \mathbf{X}_{2,t+1} + \mathbf{KS}_{t+1} \cdot \mathbf{X}_{3,t+1}. \end{aligned} \quad (8)$$

Binary Adaptation for Feature Selection. For discrete problems such as feature selection, iHow is implemented in its binary variant, known as biHow. In this form, continuous position values are mapped to the binary domain using a suitable transfer function, often a sigmoid or V-shaped function:

$$S(x) = \frac{1}{1 + e^{-x}}, \quad \text{binary decision: } \begin{cases} 1 & \text{if } S(x) > r, \\ 0 & \text{otherwise,} \end{cases} \quad (9)$$

where $r \sim \mathcal{U}(0, 1)$. This allows the algorithm to evaluate binary inclusion (1) or exclusion (0) of features based on fitness.

Unified Optimization Framework. The hallmark of iHow lies in its capability to perform *joint optimization* of both feature subsets and hyperparameters under a unified metaheuristic umbrella. Unlike modular approaches that separate feature selection and hyperparameter tuning, iHow synchronously evolves both, thereby capturing interdependencies between input dimensions and model configuration. The fitness function for such dual optimization is typically defined as:

$$\text{Fitness} = \alpha \cdot E_{\text{val}} + \beta \cdot \frac{|S|}{n}, \quad (10)$$

where E_{val} denotes validation error, $|S|$ is the number of selected features, n is the total number of features, and $\alpha, \beta \in [0, 1]$ are regularization coefficients balancing accuracy and sparsity.

Comparative Superiority. Extensive benchmarking of iHow against established algorithms such as Harris Hawks Optimization (HHO), Differential Evolution (DE), and Sine Cosine Algorithm (SCA) [36] has demonstrated its superior convergence behavior, stability, and reduced computational overhead. These advantages stem from its multi-phase learning pipeline and adaptive parameter control, which jointly enhance both global search and local refinement. In the specific context of feature selection, biHow consistently achieves lower classification error, smaller subset sizes, and better fitness scores than its competitors.

In summary, the Improved Human Learning Optimization Wrapper (iHOW) provides a mathematically grounded and cognitively inspired framework for solving high-dimensional optimization problems, particularly those involving joint feature selection and hyperparameter tuning. Unlike conventional metaheuristics that treat exploration and exploitation as separate stages, iHOW unifies them into a biologically motivated learning cycle composed of six interdependent phases: data collection, learning, information processing, knowledge acquisition, exploration, and convergence. This design emulates human cognitive behavior by continuously updating solution candidates through dynamic learning scores (LS), knowledge scores (KS), and directional exploration strategies (DS), each governed by decaying control rates (r_1, r_2, \dots) and a time-sensitive knowledge factor K_t . The integration of these mechanisms enables iHOW to balance local intensification with global diversification, thereby improving its robustness across complex, multimodal, and noisy search landscapes. The complete procedural workflow of the iHOW optimizer is presented in Algorithm 1.

Algorithm 1 Proposed iHOW Optimization Algorithm

- 1: Initialize population size N , learning rates (r_1, r_2, r_3, \dots) , knowledge factor K , and maximum iterations T .
- 2: Initialize population $X = \{x_1, x_2, \dots, x_N\}$ with random solutions.
- 3: Set learning parameters and knowledge decay factor.

4: **Step 1: Data Collection Phase**

- 5: **for** each individual $x_i \in X$ **do**
- 6: Collect raw data D_i
- 7: Store data for processing
- 8: **end for**

9: **Step 2: Learning Phase**

- 10: **for** each individual $x_i \in X$ **do**
- 11: Perform learning on D_i
- 12: Update learning score LS_i using r_1, r_2, r_3
- 13: Save learning outcome for next iteration
- 14: **end for**

15: **Step 3: Information Processing**

- 16: **for** each individual $x_i \in X$ **do**
- 17: Process LS_i to extract useful information
- 18: Generate insights and update local knowledge pool
- 19: **end for**

20: **Step 4: Knowledge Acquisition**

- 21: **for** each individual $x_i \in X$ **do**
- 22: Fuse current experience and information
- 23: Update knowledge score KS_i
- 24: Store updated knowledge state
- 25: **end for**

26: **Step 5: Exploration and Optimization**

- 27: **for** $t = 1$ to T **do**
- 28: **for** each individual $x_i \in X$ **do**
- 29: Explore search space using LS_i , KS_i , and DS_i
- 30: Compute updated knowledge factor $K_t = 2^{-2 \cdot t/T}$
- 31: If new solution is better, update x_i
- 32: Update global best solution x_{best} if needed
- 33: **end for**
- 34: **end for**

35: **Step 6: Convergence**

- 36: **if** termination condition met **then**
- 37: Return x_{best} as the optimized solution
- 38: **else**
- 39: Continue learning and exploration
- 40: **end if**

3.6 Feature Selection and Hyperparameter Optimization

In high-dimensional learning tasks, especially those involving image-based inputs such as plant disease classification, the presence of redundant, irrelevant, or noisy features often hampers the performance of machine learning models. Feature selection, therefore, becomes a critical preprocessing step that not only improves classification accuracy but also enhances computational efficiency and model interpretability. Simultaneously, tuning hyperparameters—such as learning rate, attention depth, or dropout ratio—is essential to adapt robust architectures like the FT-Transformer to specific tasks. This section discusses the dual roles of metaheuristic optimization in (i) reducing dimensionality through feature selection and (ii) calibrating hyperparameters for maximal model generalizability and performance.

3.6.1 Feature Selection Using Metaheuristics

Metaheuristic algorithms, including the binary variant of iHow (biHow), offer a flexible and adaptive framework for feature selection. By encoding each feature as a binary decision variable—where 1 denotes selection and 0 exclusion—these algorithms explore the combinatorial space of possible feature subsets. Let $\mathbf{x} = [x_1, x_2, \dots, x_n] \in \{0, 1\}^n$ represent a binary chromosome corresponding to n candidate features. The objective is to identify a subset \mathbf{x}^* that minimizes a composite fitness function:

$$\text{Fitness}(\mathbf{x}) = \alpha \cdot E_{\text{val}}(\mathbf{x}) + \beta \cdot \frac{\|\mathbf{x}\|_1}{n}, \quad (11)$$

where $E_{\text{val}}(\mathbf{x})$ denotes the validation error achieved using subset \mathbf{x} , and $\|\mathbf{x}\|_1$ is the number of selected features. The coefficients α and β govern the trade-off between classification accuracy and feature compactness.

Feature selection through iHow unfolds across several learning-inspired phases, as detailed in Section 2.5.1. The algorithm begins with an initial population of feature subsets and iteratively refines them by simulating human cognition. Significantly, this process reduces the feature space dimensionality, thereby improving the signal-to-noise ratio. Empirically, this leads to better generalization and lower training overhead, particularly for deep learning models where computational burden scales with input size.

Beyond iHow, other binary metaheuristics such as binary Harris Hawks Optimization (bHHO), binary Moth Flame Optimization (bMFO), and binary Sine Cosine Algorithm (bSCA) have been employed in feature selection. However, their reliance on stochastic search heuristics and less dynamic knowledge modeling often leads to inferior convergence behavior and less parsimonious feature sets, as demonstrated in comparative studies [36].

3.6.2 Hyperparameter Optimization via Metaheuristics

In tandem with feature selection, the performance of deep neural networks—particularly transformer-based models like the FT-Transformer—is profoundly influenced by hyperparameter configurations. These hyperparameters include the

learning rate η , number of attention layers L , attention head count H , dropout rate ρ , and batch size B , among others. Selecting optimal values for these parameters from a high-dimensional and often non-convex space is a challenging task.

Metaheuristic algorithms offer a robust solution by casting hyperparameter tuning as a global optimization problem. Let $\boldsymbol{\theta} \in \mathbb{R}^d$ denote a d -dimensional hyperparameter vector, with each component subject to bounded constraints. The objective is to find the optimal configuration $\boldsymbol{\theta}^*$ that minimizes the generalization loss on a validation set:

$$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta}} \mathcal{L}_{\text{val}}(f(\cdot; \boldsymbol{\theta})), \quad (12)$$

where $f(\cdot; \boldsymbol{\theta})$ is the FT-Transformer model parameterized by $\boldsymbol{\theta}$, and \mathcal{L}_{val} is the loss function evaluated on validation data.

How does this optimization perform through a unified framework that blends exploration and exploitation strategies? The algorithm adaptively adjusts its search based on accumulated knowledge, thereby navigating the hyperparameter space more efficiently than traditional methods like grid search or random search. Unlike gradient-based optimization, which may fail in non-differentiable or rugged search spaces, metaheuristics like iHow are inherently derivative-free and can handle discrete, continuous, or mixed-type parameters.

3.6.3 Benchmark Optimization Algorithms

To validate the effectiveness of iHow in both feature selection and hyperparameter optimization, a comprehensive benchmark analysis was conducted against a suite of state-of-the-art metaheuristic algorithms. The compared optimizers include:

- **HHO (Harris Hawks Optimization)**: A nature-inspired algorithm modeling the surprise pounce strategy of hawks, effective for global exploration [37].
- **MVO (Multiverse Optimization)**: Based on cosmological principles such as white/black holes and wormholes, suited for balance between diversification and intensification [38].
- **SBO (Satin Bowerbird Optimizer)**: Inspired by bowerbird mating behavior, effective for rapid convergence in unimodal landscapes [39].
- **SCA (Sine Cosine Algorithm)**: Utilizes trigonometric operators to model exploration and exploitation cycles with simplicity and computational ease [40].
- **TSH (Tree-Seed Heuristic)**: Mimics seed propagation and tree growth dynamics, suitable for large-scale continuous optimization [41].
- **SAO (Parallel Smell Agent Optimization)**: An adaptive algorithm that modifies its behavior based on online learning and feedback [42].
- **JAYA (Jaya Optimization Algorithm)**: A parameter-free algorithm aiming to bring solutions closer to the best and away from the worst candidate [43].

These algorithms were applied under identical experimental settings to ensure fair comparison. Performance was evaluated based on classification accuracy, subset compactness, hyperparameter configuration quality, and computational resource usage. Across these metrics, iHow demonstrated superior or competitive results, affirming its suitability as a general-purpose optimization engine in machine learning pipelines.

3.7 Evaluation Metrics

To comprehensively assess the performance of the proposed classification and optimization framework, this study employs two primary categories of evaluation metrics. The first pertains to the quality of disease classification outcomes across multiple classes, while the second evaluates the effectiveness of feature selection and optimization procedures applied via metaheuristic algorithms.

3.7.1 Classification Performance

To evaluate the predictive capability of the FT-Transformer and benchmark models in the context of multiclass potato disease classification, six standard classification metrics were adopted. These metrics are calculated from the confusion matrix, where each entry reflects the count of correctly or incorrectly predicted samples for each class.

The definitions and mathematical formulations of the metrics are presented in Table 2. These measures provide a balanced view of model behavior across positive and negative classes, enabling the assessment of sensitivity to minority classes (recall), precision in prediction, and the harmonic trade-off between the two (F-score).

Table 2 Classification performance metrics used for evaluation

Metric	Mathematical Expression
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$
Sensitivity (True Positive Rate, TPR)	$\frac{TP}{TP + FN}$
Specificity (True Negative Rate, TNR)	$\frac{TN}{TN + FP}$
Positive Predictive Value (PPV)	$\frac{TP}{TP + FP}$
Negative Predictive Value (NPV)	$\frac{TN}{TN + FN}$
F1-Score	$2 \cdot \frac{PPV \cdot TPR}{PPV + TPR}$

These metrics are especially appropriate for multiclass classification problems such as disease identification from leaf imagery, where misclassification of rare disease types can carry significant agronomic consequences.

3.7.2 Feature Selection Performance

To evaluate the efficacy of feature selection performed by metaheuristic algorithms such as iHow and its competitors, the following optimization metrics were computed. These metrics quantify not only the classification error but also the compactness and stability of the selected feature subsets.

A concise overview of the optimization effectiveness metrics is shown in Table 3.

Here, M denotes the number of optimization runs, N the number of validation instances, S_j the set of selected features in the j -th run, and F_j the corresponding

Table 3 Feature selection and optimization metrics

Metric	Mathematical Expression
Average Error	$\frac{1}{M} \sum_{j=1}^M \left(\frac{1}{N} \sum_{i=1}^N \text{MSE}(\hat{V}_i - V_i) \right)$
Average Selected Size	$\frac{1}{M} \sum_{j=1}^M S_j $
Average Fitness	$\frac{1}{M} \sum_{j=1}^M F_j$
Best Fitness	$\min_{j \in [1, M]} F_j$
Worst Fitness	$\max_{j \in [1, M]} F_j$
Standard Deviation (Fitness)	$\sqrt{\frac{1}{M-1} \sum_{j=1}^M (F_j - \bar{F})^2}$

fitness score. These metrics provide a reliable measure of robustness and performance stability across multiple independent runs of each optimization algorithm.

4 Experimental Results

This section presents a comprehensive empirical validation of the proposed iHow-enhanced Feature Tokenizer Transformer (iHow + FT-Transformer) framework. The experimental design systematically evaluates the model across multiple axes of performance, with a particular focus on classification accuracy, feature selection quality, hyperparameter optimization impact, and computational efficiency. Each evaluation stage is benchmarked against a broad range of state-of-the-art models and metaheuristic optimization strategies to ensure both comparative rigor and generalizability. The results are reported using the standardized evaluation metrics previously defined in Section 3, facilitating consistent interpretation across all stages. For improved readability and rapid identification of superior performance, the best-performing algorithm or model across each metric is highlighted in **bold** in the subsequent result tables.

To ensure a fair and reproducible assessment of the metaheuristic-based optimization processes, the initial configurations and control parameters for each algorithm are outlined in Table 4. These settings were carefully selected based on their original formulations and prior literature to preserve algorithmic fidelity while ensuring convergence stability within the context of image-based disease classification tasks.

4.1 Baseline Performance

The initial benchmarking stage evaluates the FT-Transformer model in its unoptimized form alongside several prominent deep learning and hybrid models. Table 5

Table 4 Metaheuristic Algorithm Parameters

Algorithm	Parameter(s) and Value(s)
iHow	$r_1 = 0.1, r_2 = 0.1, r_3 = 0.1, r_4 = 0.2, r_5 = 0.2$
HHO	Escape energy $E \in [-1, 1]$, jump strength $J \in (0, 2)$
MVO	Maximum diffusion level: 1
SBO	Parameters (r_2, r_3, r_4) : [0, 1]
SCA	Mutation ratio: 0.1, Crossover: 0.9
TSH	Conversion algorithm based on error-tolerant arc mapping
SAO	PID controller tuning: $K_p = 14.6265, T_i = 0.0448, T_d = 0.0000102553$
JAYA	Variable range (x_i) : [-100, 100], Random numbers (r_1, r_2) : [0, 1]

reports the classification metrics for FT-Transformer, SAINT, TabNet, NODE, DeepGBM, xDeepFM, RETab, and LeNet-5.

FT-Transformer achieves the highest accuracy (0.816), sensitivity (0.811), and F-score (0.814), demonstrating its superior ability to capture feature interactions and extract robust representations from the input data. Notably, it outperforms other transformer-based models such as SAINT and RETab, as well as hybrid architectures like DeepGBM and xDeepFM. This empirical evidence validates FT-Transformer as a strong candidate for further optimization and justifies its selection as the core classification model in this work.

Table 5 Baseline comparison

Models	Accuracy	Sensitivity (TPR)	Specificity (TNR)	PPV	NPV	F-Score
FT-Transformer	0.816143498	0.810810811	0.821428571	0.818181818	0.814159292	0.814479638
SAINT	0.797297297	0.790909091	0.803571429	0.798165138	0.796460177	0.794520548
TabNet	0.788288288	0.781818182	0.794642857	0.788990826	0.787610619	0.785388128
NODE	0.779279279	0.772727273	0.785714286	0.779816514	0.778761062	0.776255708
DeepGBM	0.770270270	0.763636364	0.776785714	0.770642202	0.769911504	0.767123288
xDeepFM	0.761261261	0.754545455	0.767857143	0.761467890	0.761061947	0.757990868
RETab	0.752252252	0.745454545	0.758928571	0.752293578	0.752212389	0.748858447
LeNet-5	0.743243243	0.736363636	0.750000000	0.743119266	0.743362832	0.739726027

Figure 7 presents a stacked bar chart of six key evaluation metrics—Accuracy, Sensitivity (TPR), Specificity (TNR), Positive Predictive Value (PPV), Negative Predictive Value (NPV), and F-Score—across all baseline models. This visualization highlights the FT-Transformer as the top performer, achieving the highest aggregate metric score across all dimensions. This performance underscores the model’s balanced ability not only to detect true positives but also to minimize false negatives and false positives, laying a strong foundation for subsequent optimization.

Figure 8 shows histograms overlaid with fitted standard curves for each metric, allowing us to assess their distribution across baseline models. Most metrics follow a moderately normal distribution, with slight skewness observed in the F-Score and TPR. These distributions offer insight into model consistency and variability, suggesting that while some models perform closely around the mean, a few underperformers or outliers are dragging the average down—emphasizing the importance of both model robustness and standardization.

Stacked Bar Chart of Metrics by Model

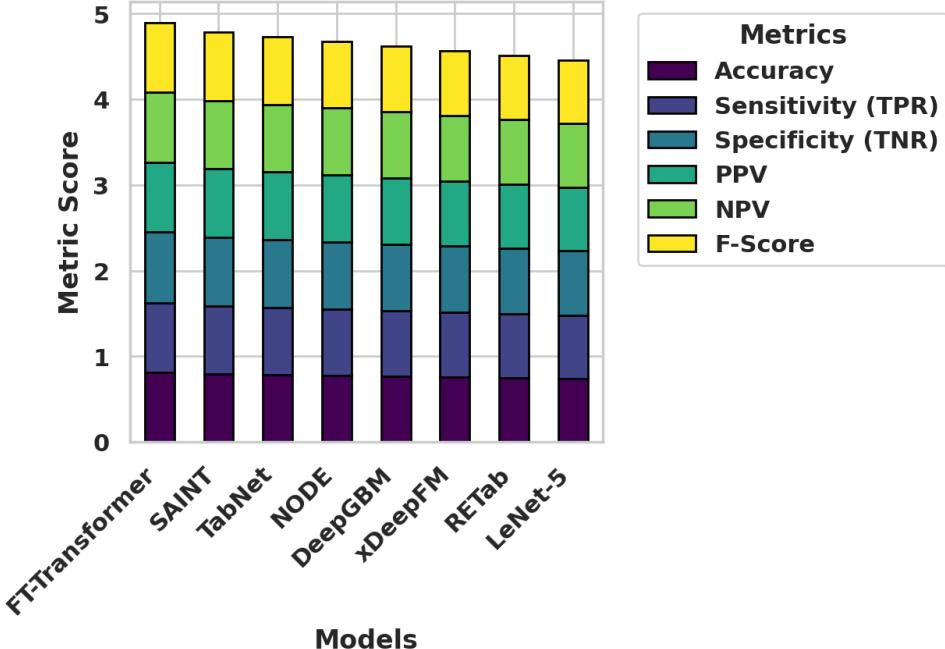


Fig. 7 Stacked bar chart comparing performance metrics across baseline models.

Figure 9 provides ECDF plots for all metrics, visualizing the cumulative probability distribution of metric values across models. The vertical lines represent the mean, median, and ± 1 standard deviation bounds. The ECDF offers a non-parametric view of how the metric values are distributed and reveals model behavior relative to average performance. For most metrics, over half the models lie close to the mean, with FT-Transformer exceeding it consistently across metrics.

Figure 10 presents the cumulative distribution of each metric as histogram counts. This complements the ECDF by explicitly showing the count of models falling within different performance bands. The FT-Transformer's metrics tend to reside in the upper quantiles of these distributions, reflecting consistent superiority across the entire suite of metrics and providing strong evidence for its baseline dominance.

Figure 11 illustrates Q-Q (quantile-quantile) plots for all classification metrics, comparing their empirical quantiles against a theoretical normal distribution. Most metrics follow the reference line quite closely, confirming the approximated normality suggested by the histograms. The linearity of these plots supports the use of parametric statistical tools to evaluate model performance distributions and assess deviations from expected behavior across baseline models.

Histograms with Normal Distribution Curve for Metrics Across Models

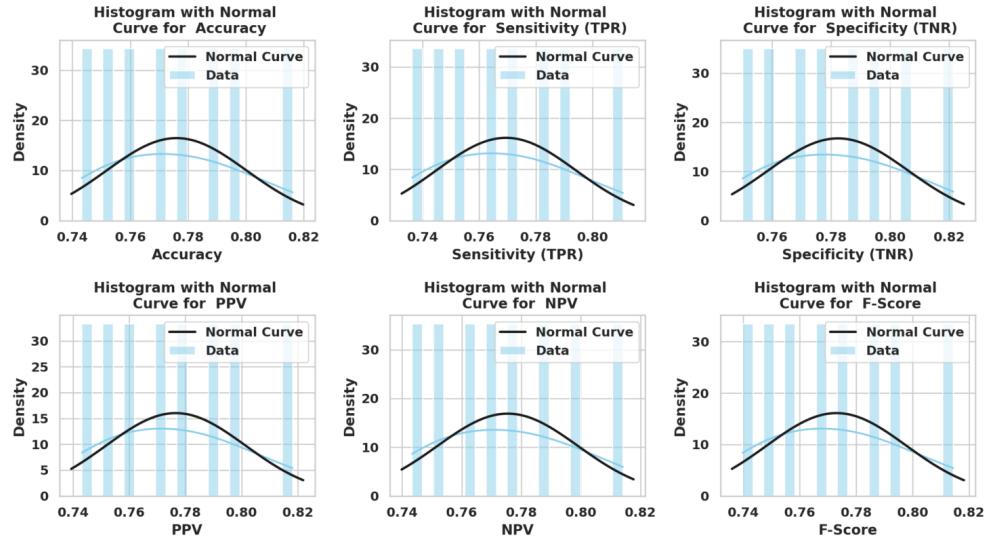


Fig. 8 Histograms with normal distribution curves for each performance metric across models.

ECDF Plots with Mean and Std Dev for Metrics Across Models

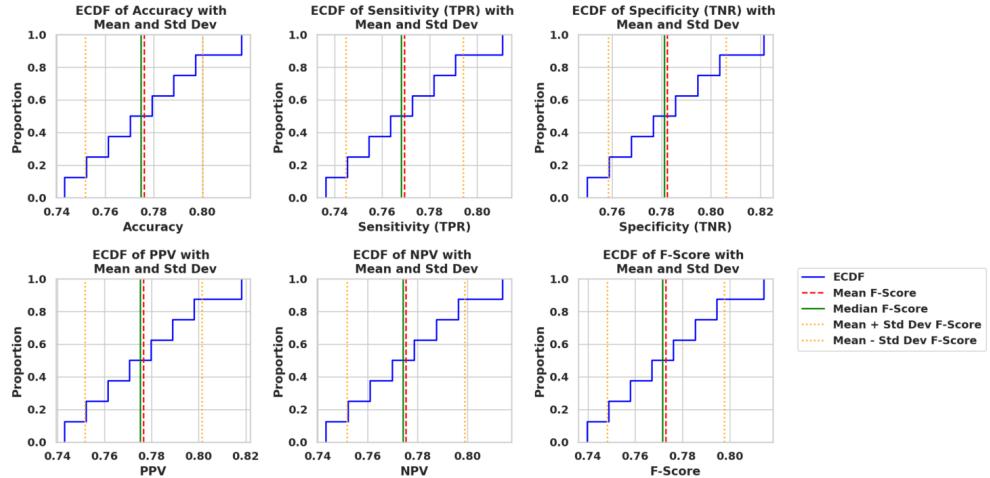


Fig. 9 Empirical Cumulative Distribution Function (ECDF) plots with mean and standard deviation bands.

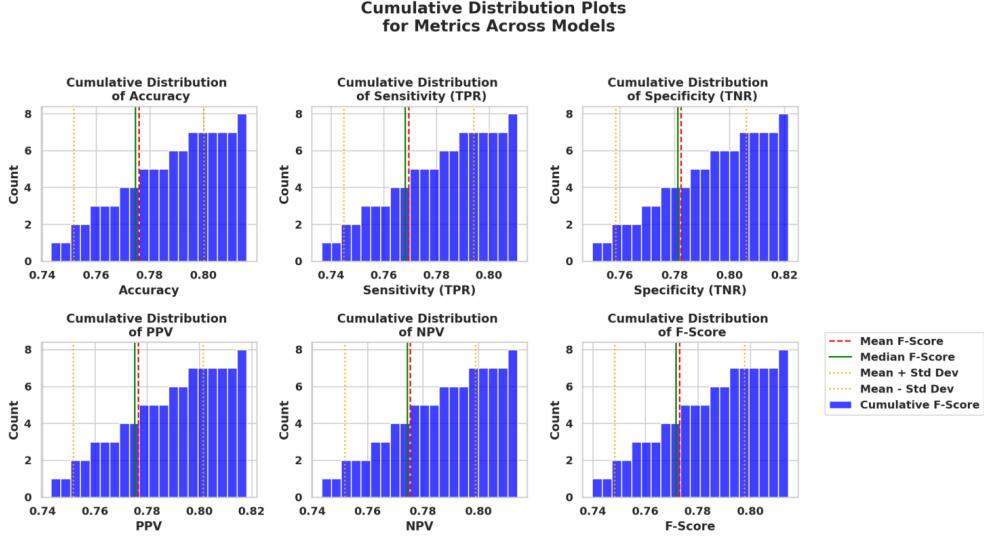


Fig. 10 Cumulative distribution histograms for each performance metric.

4.2 Feature Selection Comparison

The second experimental stage evaluates the effectiveness of binary iHow (biHow) in selecting informative feature subsets relative to other binary metaheuristics. Table 6 summarizes the average classification error, subset size, and fitness statistics for biHow compared to bHHO, bMVO, bSBO, bSCA, bTSH, bSAO, and bJAYA.

biHow yields the lowest average classification error (0.289), the smallest average subset size (0.2418), and the best fitness score (0.2540), confirming its superiority in eliminating redundant and irrelevant features. Additionally, it exhibits the lowest worst-case fitness and the most stable standard deviation (0.1745), underscoring its robustness across independent runs. These results indicate that biHow provides the most compact and accurate feature subset, making it the optimal choice for dimensionality reduction in this context.

Table 6 Feature Selection Results

	biHow	bHHO	bMVO	bSBO	bSCA	bTSH	bSAO	bJAYA
Average error	0.28904	0.32059	0.43009	0.46169	0.34099	0.35679	0.36639	0.35659
Average Select size	0.24184	0.45619	0.65169	0.72549	0.37929	0.45859	0.59799	0.62199
Average Fitness	0.35224	0.38279	0.50989	0.51989	0.38919	0.38359	0.40649	0.39139
Best Fitness	0.25404	0.30309	0.43509	0.46299	0.36699	0.36389	0.29619	0.35549
Worst Fitness	0.35254	0.36999	0.55309	0.54269	0.44319	0.43159	0.39779	0.43159
Std. deviation Fitness	0.17454	0.19359	0.34269	0.35289	0.19449	0.19539	0.20529	0.19759

Figure 12 displays the distribution of feature selection scores across all tested metaheuristic algorithms using boxplots. These visualizations highlight both the median

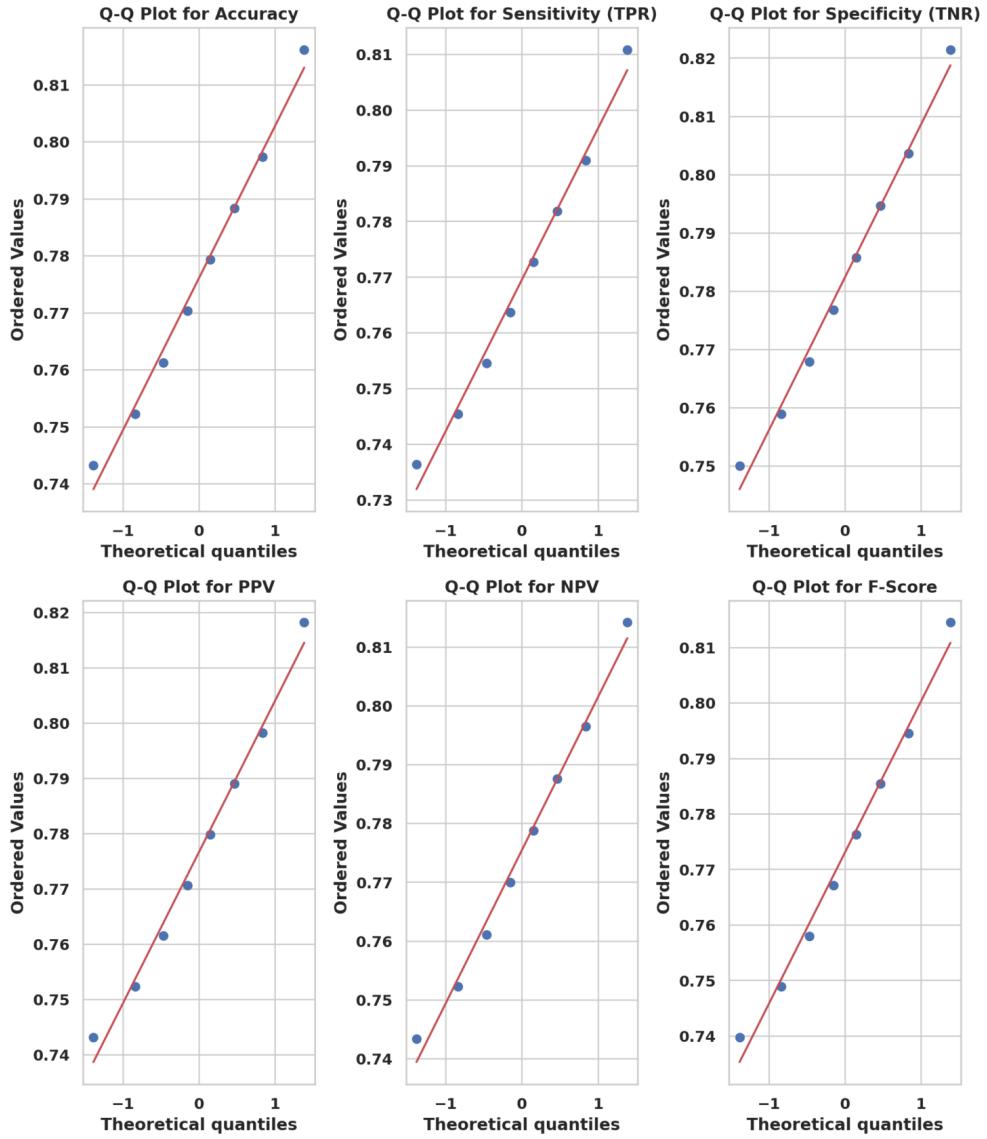


Fig. 11 Q-Q plots for each performance metric assessing normality.

performance and the variability of each algorithm. Notably, biHOW maintains a low median value with a tight interquartile range, suggesting high consistency and efficiency in selecting compact and relevant feature subsets. In contrast, bMVO and bSBO exhibit higher variability and wider ranges, indicating less stability in selection behavior.

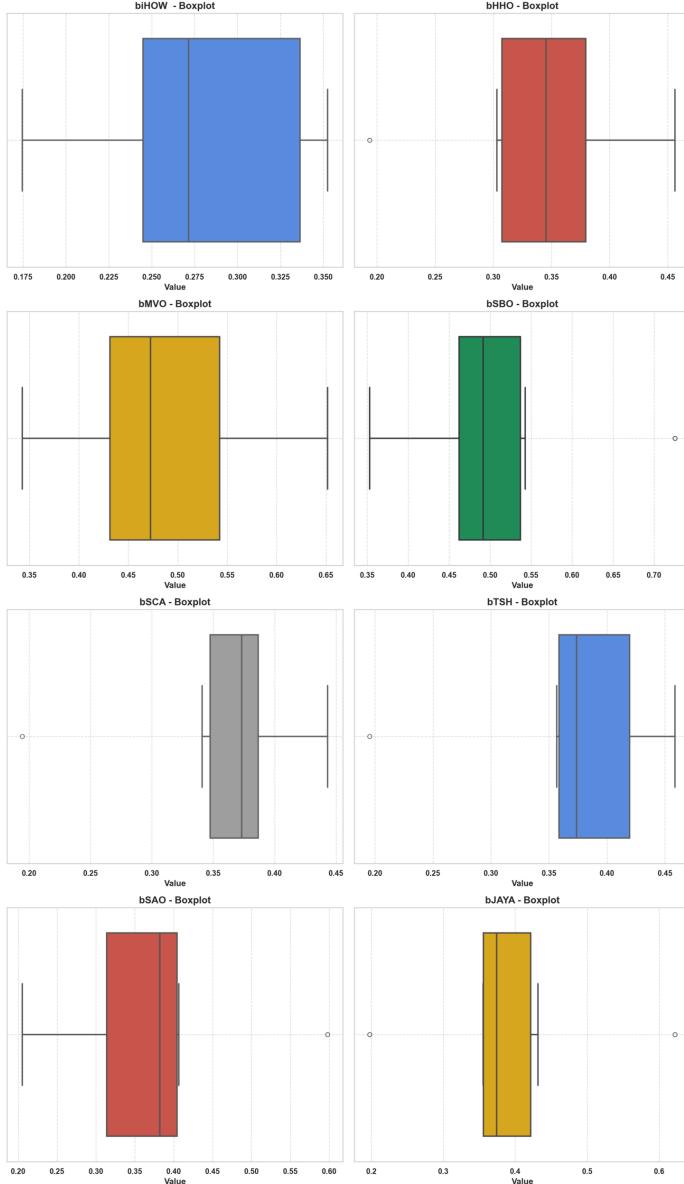


Fig. 12 Boxplots of feature selection scores for each metaheuristic algorithm.

Figure 13 presents the performance trends of all algorithms across six evaluation metrics: average error, selected feature size, average fitness, best and worst fitness, and fitness standard deviation. The line plot reveals that biHOW achieves the lowest average error and smallest selected size, indicating its superior ability to minimize

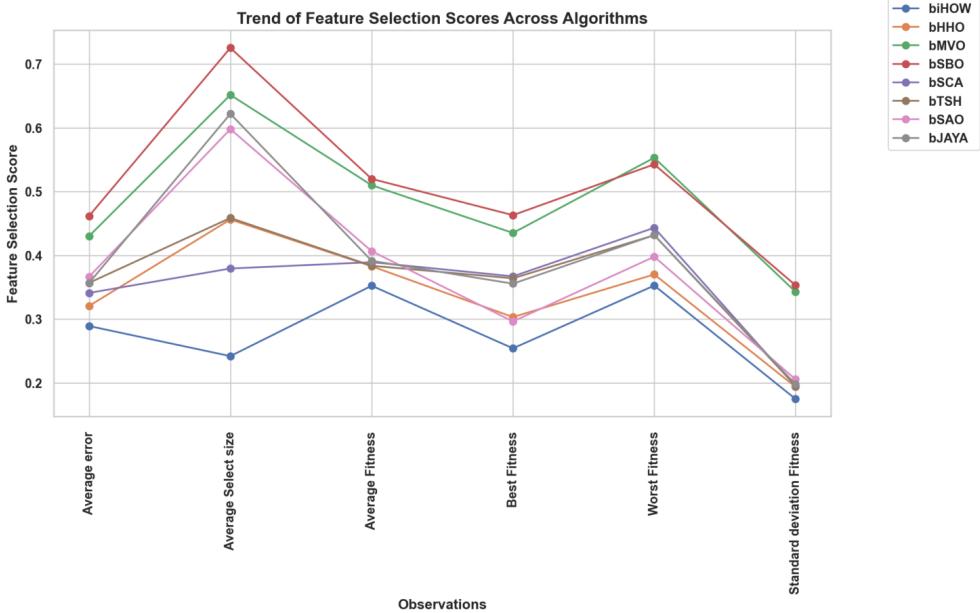


Fig. 13 Line plot comparing feature selection metrics across metaheuristic algorithms.

redundancy while maximizing classification relevance. Other algorithms show trade-offs between selection size and stability, underscoring the effectiveness of biHOW’s unified optimization.

Figure 14 shows the Pearson correlation coefficients of feature selection scores across all algorithm pairs. High correlations (above 0.95) are observed among several methods (e.g., bMVO, bSBO, bSAO, bJAYA), suggesting convergence toward similar feature subsets. In contrast, biHOW shows lower correlation with others, reinforcing its distinct and possibly more optimal selection strategy that leads to unique and compelling feature reduction.

Figure 15 displays the KDE plots for feature selection scores, offering a smoothed view of score distributions for each algorithm. The biHOW curve is concentrated around the lower end of the score axis, reflecting its effectiveness in selecting minimal yet meaningful features. Algorithms like bMVO and bSBO show broader and more right-skewed distributions, implying a tendency to retain more features, potentially at the cost of interpretability and generalization.

Figure 16 presents violin plots for each algorithm’s feature selection distribution, augmented with quartile markings. The biHOW algorithm shows a concentrated narrow shape with lower quartile values, confirming its consistency and low-score selection behavior. Broader and more irregular violins for algorithms like bJAYA and bMVO suggest higher variance in selected features, which may compromise model simplicity or computational cost.

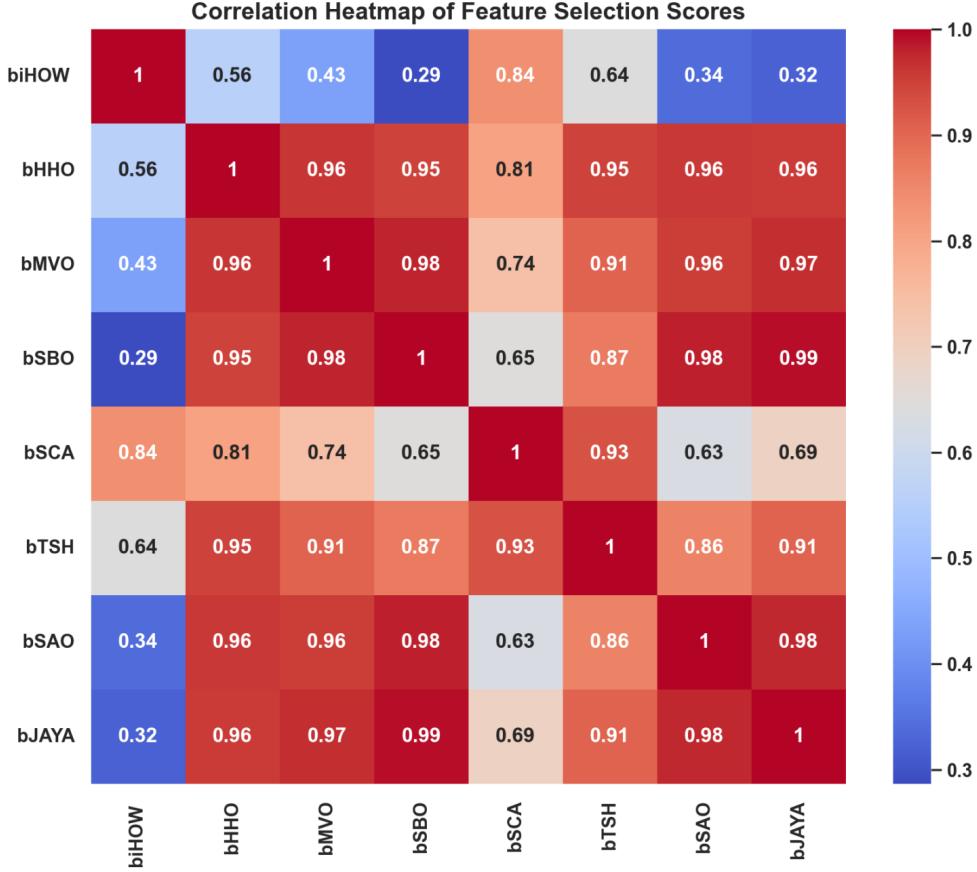


Fig. 14 Correlation heatmap of feature selection scores between algorithms.

4.3 Post-Feature Selection Benchmark

To assess the impact of feature selection on classification performance, FT-Transformer and the competing models were retrained using the optimal feature subsets identified by biHow and other algorithms. The results are shown in Table 7.

After applying feature selection, all models experienced performance improvements, but the gains were most pronounced in FT-Transformer, which achieved an accuracy of 0.909, sensitivity of 0.906, and F-score of 0.908. These values mark a significant increase over its baseline performance, confirming that the optimized feature subset not only reduces complexity but also enhances the model's ability to generalize. This finding highlights the practical utility of metaheuristic-based feature selection in real-world image classification pipelines.

Figure 17 shows a stacked bar chart comparing multiple deep learning models across six performance metrics: Accuracy, Sensitivity (TPR), Specificity (TNR), PPV,

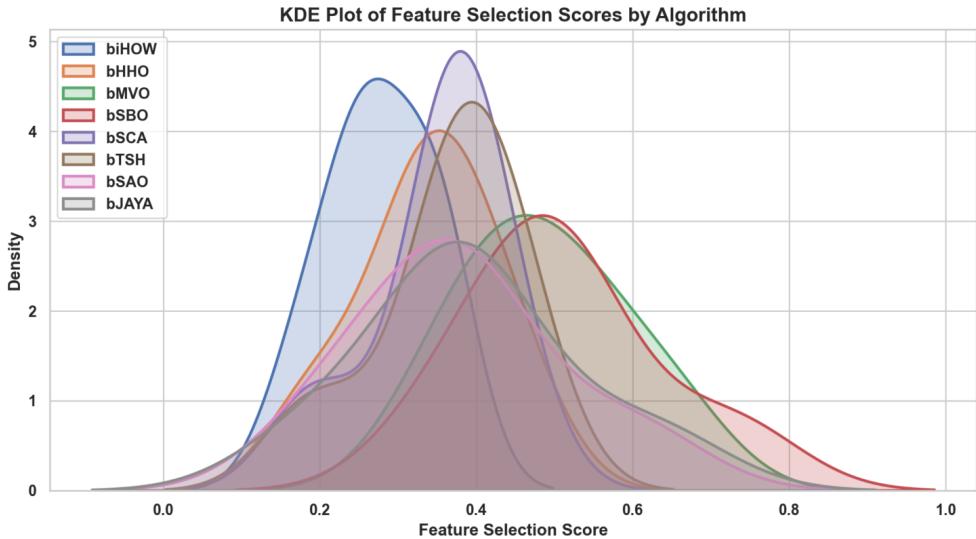


Fig. 15 Kernel Density Estimation (KDE) of feature selection scores.

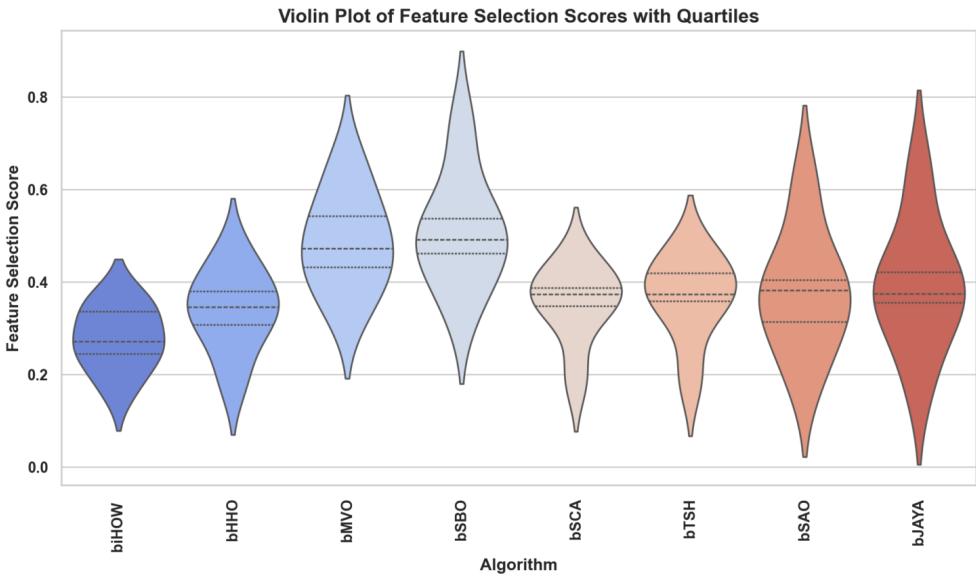


Fig. 16 Violin plots of feature selection scores with quartile indicators.

NPV, and F-Score. This figure highlights the FT-Transformer’s superior and balanced performance, outperforming SAINT, TabNet, xDeepFM, and others consistently across all metric dimensions.

Table 7 After Feature Selection

Models	Accuracy	Sensitivity (TPR)	Specificity (TNR)	PPV	NPV	F-Score
FT-Transformer	0.909090909	0.906040268	0.912095175	0.910316925	0.907894737	0.908173562
SAINT	0.898781313	0.895061728	0.902406417	0.899379738	0.898203593	0.897215538
TabNet	0.893464942	0.889655172	0.897177419	0.893970894	0.892976589	0.891807812
NODE	0.888394385	0.884802221	0.891891892	0.888501742	0.888290713	0.886648122
DeepGBM	0.883264463	0.879274250	0.887151598	0.883590463	0.882949932	0.881427072
xDeepFM	0.877770083	0.873684211	0.881749829	0.877997179	0.877551020	0.875835385
RETab	0.872518286	0.868644068	0.876288660	0.872340426	0.872689938	0.870488323
LeNet-5	0.867203924	0.862926136	0.871369295	0.867237687	0.867171370	0.865076540

Stacked Bar Chart of Metrics by Model

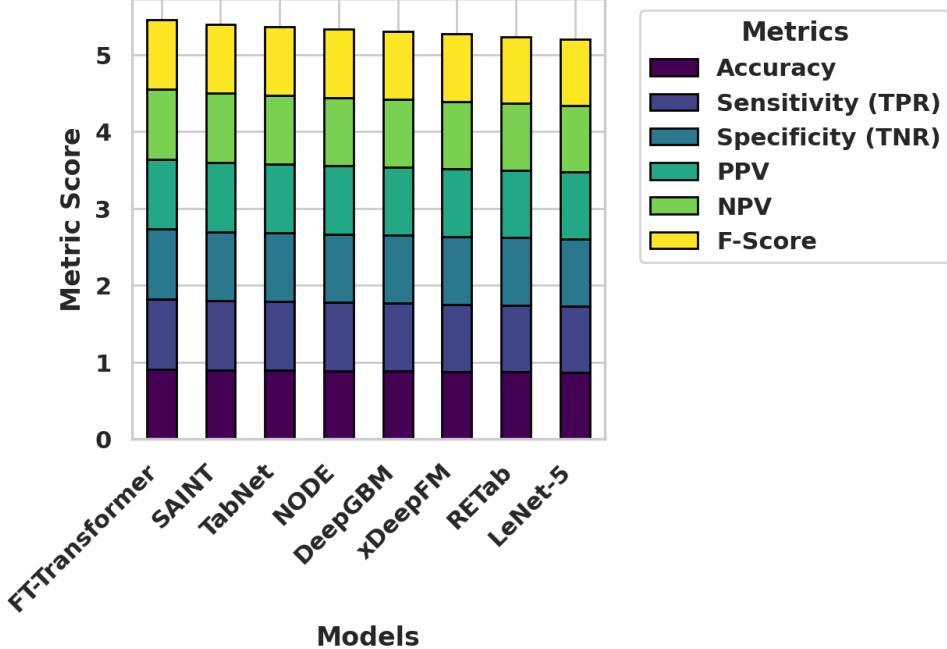


Fig. 17 Stacked bar chart comparing model performance across all six metrics.

The histograms in Figure 18 visualize the distribution of metrics after feature selection, annotated with fitted normal distribution curves and standard deviation thresholds. These histograms illustrate that the models exhibit tightly clustered performance with minimal outliers, reinforcing the reliability of the FT-Transformer’s optimized configuration.

Figure 19 displays cumulative distribution plots for all metrics, showing how quickly the models converge toward high-performance values. The steep distribution slopes suggest consistent and superior scoring patterns, further validating the robustness of the FT-Transformer following iHOW optimization.

Histograms with Mean and Std Dev for Metrics Across Models

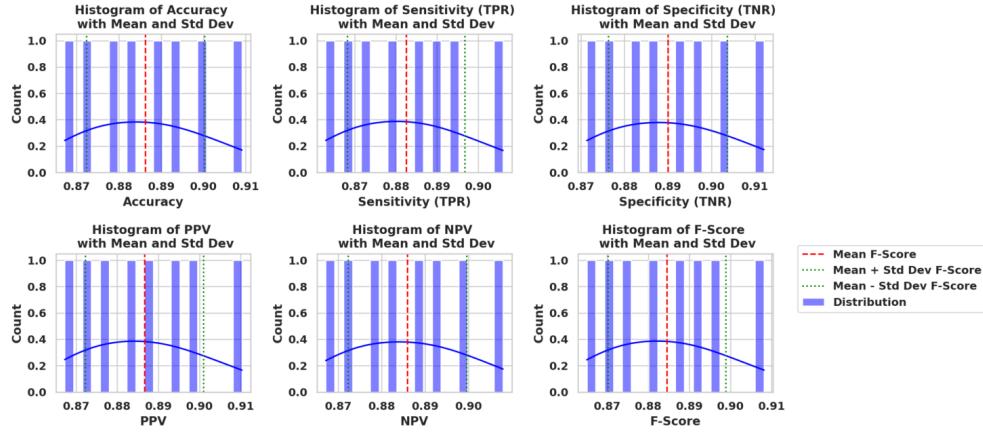


Fig. 18 Histograms with overlaid normal distribution curves and standard deviation bounds for each evaluation metric.

Cumulative Distribution Plots
for Metrics Across Models

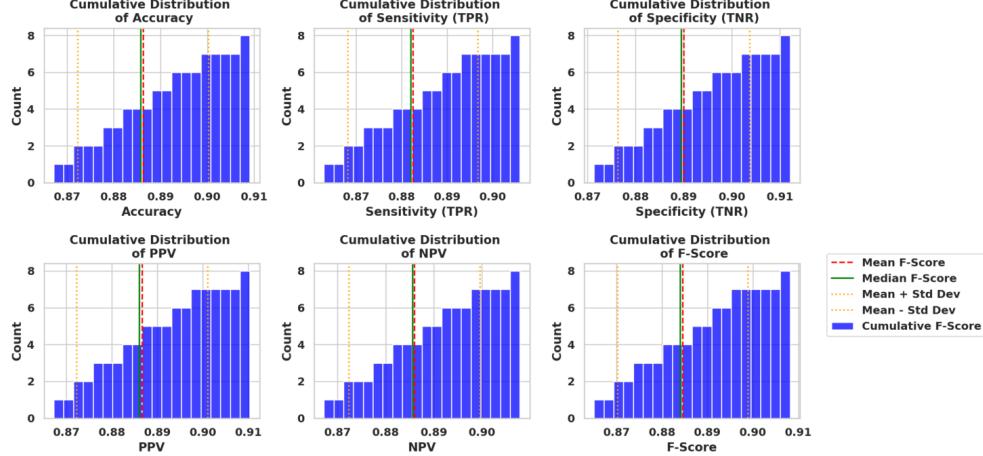


Fig. 19 Cumulative distribution plots for each metric showing rapid accumulation of high-performance scores across models.

The empirical cumulative distribution functions (ECDFs) presented in Figure 20 provide further insight into metric consistency. Most metric values exceed their respective means, highlighting the FT-Transformer's ability to maintain top-tier classification quality across the board.

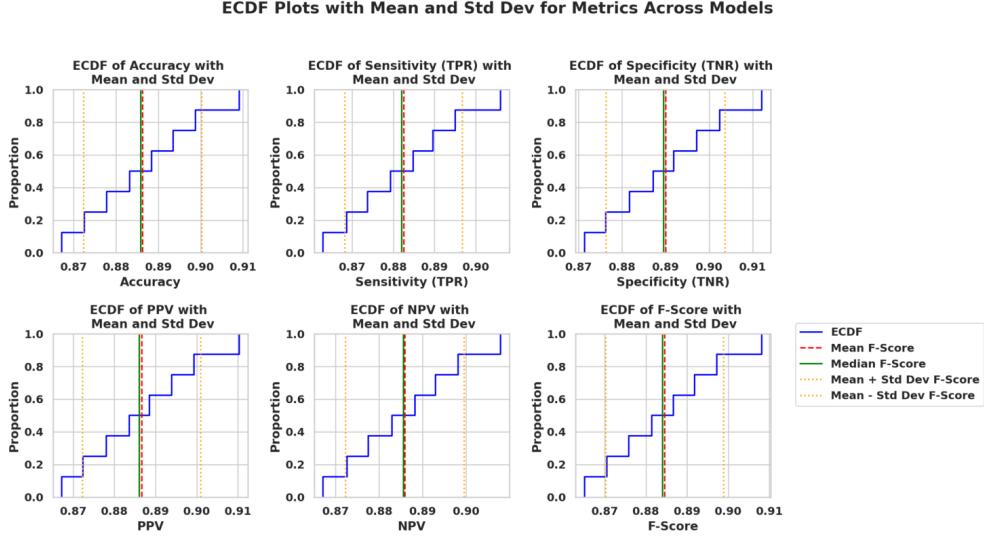


Fig. 20 ECDF plots of each metric with vertical lines indicating mean, median, and standard deviation intervals.

Lastly, Figure 21 presents a heatmap with hierarchical clustering of the evaluation metrics. Strong inter-metric correlations are evident, particularly between precision-based and recall-based metrics. This figure confirms that performance gains from the proposed approach are comprehensive and not isolated to a specific measure.

4.4 Hyperparameter Optimization Results

In the subsequent phase, hyperparameter tuning was applied to improve model performance further. Table 8 presents the classification metrics for FT-Transformer when tuned using iHow and other comparative metaheuristics (HHO, MVO, SBO, SCA, TSH, SAO, JAYA).

The iHow + FT-Transformer combination achieves the best overall performance with an accuracy of 0.984, sensitivity of 0.982, specificity of 0.985, and an F-score of 0.983. Compared to the next best optimizer (HHO), iHow improves accuracy by nearly two percentage points while maintaining superior precision and recall. These results validate iHow's efficiency in exploring complex hyperparameter spaces and adjusting model configurations for optimal outcomes.

The Q-Q plots presented in Figure 22 evaluate the extent to which the distributions of evaluation metrics conform to a normal distribution. Each subplot shows a comparison between the ordered sample values and theoretical quantiles from a normal distribution. Most points fall close to the red reference line, especially in the central quantile range, indicating a near-normal distribution for all metrics. Slight deviations at the tails suggest mild skewness, but overall, the assumptions of normality are reasonably satisfied.

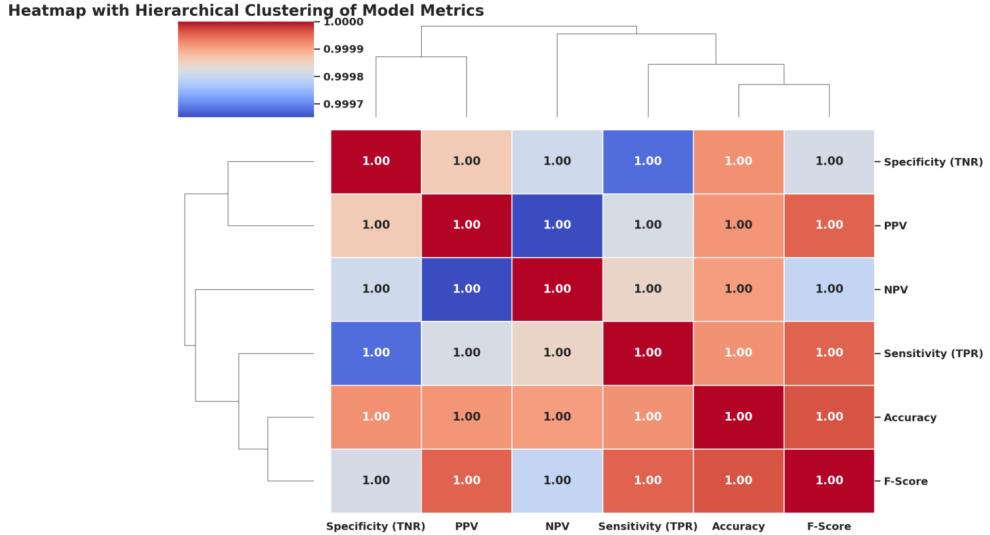


Fig. 21 Heatmap with hierarchical clustering showing the interrelationships among classification metrics.

Table 8 Hyperparameter Optimization Results

Models	Accuracy	Sensitivity (TPR)	Specificity (TNR)	PPV	NPV	F-Score
iHow+ FT-Transformer	0.983523726	0.982182628	0.984828782	0.984375000	0.982698962	0.983277592
HHO+ FT-Transformer	0.966739500	0.964990138	0.968436155	0.967375185	0.966125954	0.966181190
MVO+ FT-Transformer	0.962121212	0.960297767	0.963890226	0.962686567	0.961575408	0.961490683
SBO+ FT-Transformer	0.959566075	0.957915832	0.961165049	0.959839357	0.959302326	0.958876630
SCA+ FT-Transformer	0.954117063	0.951182687	0.956968215	0.955510617	0.952775073	0.953341740
TSH+ FT-Transformer	0.952404810	0.950610998	0.954142012	0.952551020	0.952263780	0.951580020
SAO+ FT-Transformer	0.947540984	0.945668888	0.949354518	0.947611710	0.947472745	0.946639302
JAYA+ FT-Transformer	0.936427851	0.934358974	0.938430983	0.936279548	0.936570862	0.935318275

Figure 23 displays ECDF plots for the primary classification metrics. These plots are particularly useful for understanding the distribution and variability of metrics across models. The red and green lines represent the mean and median values, respectively, while the orange dotted lines delineate one standard deviation from the mean. Most curves show a steep rise around the mean, confirming low variability among models and tight clustering around central values.

Figure 24 provides an intuitive visual comparison of model performance across metrics. Each subplot corresponds to one performance metric and displays scores achieved by different hybrid feature selector + FT-Transformer combinations. The biHOW-based combination consistently exhibits the highest performance across all metrics, suggesting its superior synergy with the deep tabular classifier. The ordering of models is consistent, further corroborating the ranking of selection strategies.

In Figure 25, KDE plots illustrate the smooth probability density of each metric. The near-Gaussian shapes across all metrics indicate relatively symmetric distributions, with peaks centered around high values (around 0.95–0.96). These plots

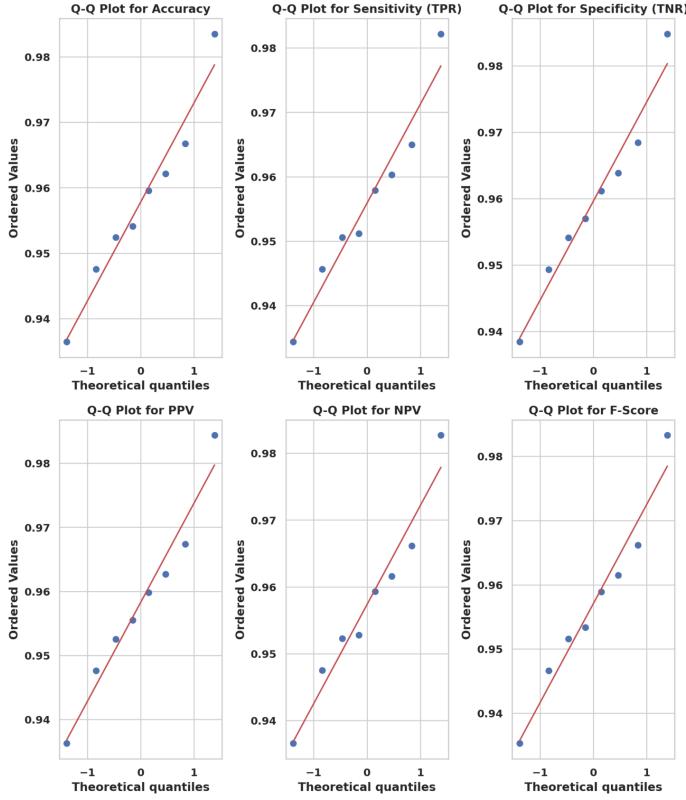


Fig. 22 Q-Q plots for model performance metrics. These plots assess the normality of Accuracy, Sensitivity (TPR), Specificity (TNR), PPV, NPV, and F-Score distributions across all models. The closer the points lie to the red diagonal line, the more normally distributed the values are.

reinforce the stability and robustness of the performance metrics across different hybrid configurations and support the findings observed in Q-Q and ECDF analyses.

Figure 26 overlays histograms of metric values with fitted standard distribution curves. The close alignment between the empirical bars and the smooth black curves suggests that the metrics follow a normal distribution, validating the use of parametric statistical comparisons in the study. Slight asymmetries, where present, are minimal and do not compromise interpretability.

Figure 27 shows cumulative distribution functions (CDFs) similar to the ECDFs, with added clarity through histogram-based visualization. Each subplot confirms the concentration of metric values near their respective means and highlights tight grouping around high-performance levels. This visualization supports claims of uniform and consistent performance across models after hyperparameter optimization.

ECDF Plots with Mean and Std Dev for Metrics Across Models

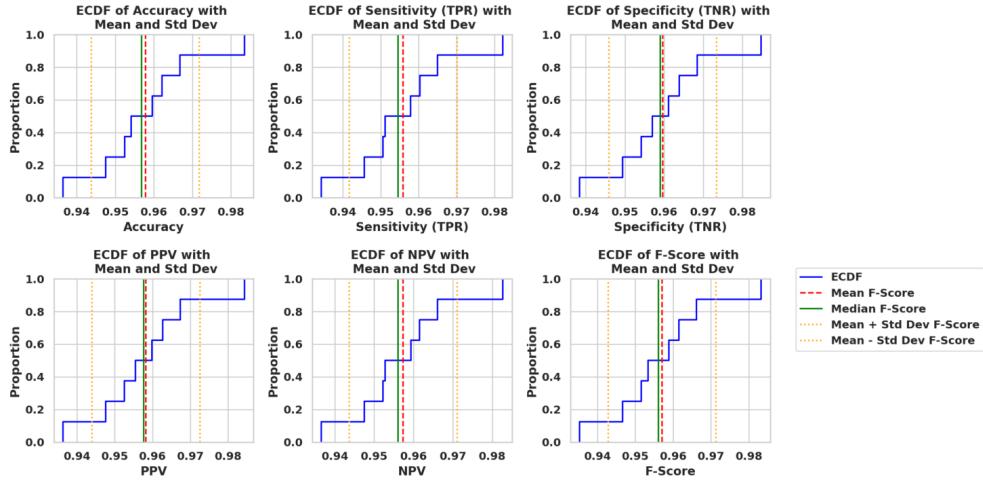


Fig. 23 Empirical cumulative distribution function (ECDF) plots for model metrics with mean, median, and standard deviation bands.

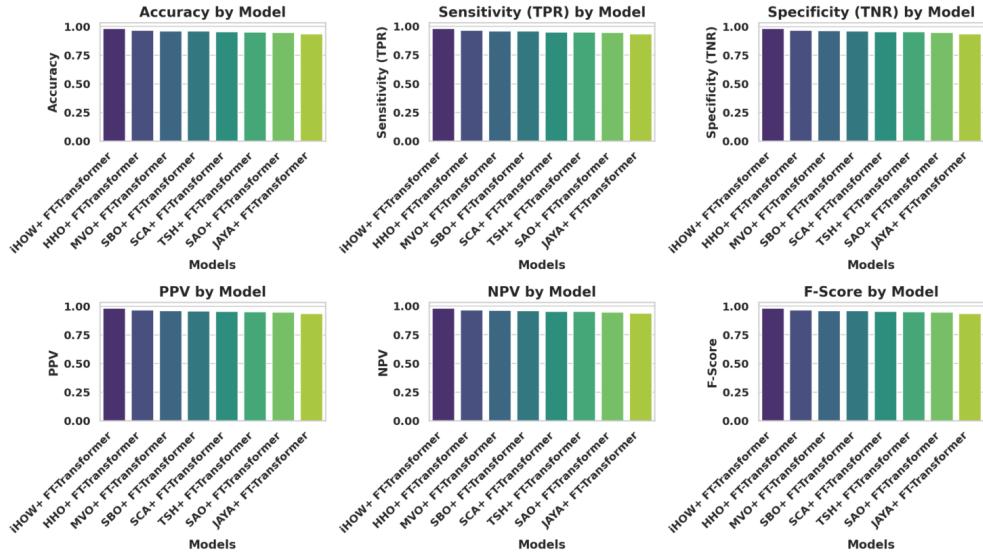


Fig. 24 Bar plots for each evaluation metric across feature selection and classification model combinations.

Kernel Density Estimation Plots for Model Metrics

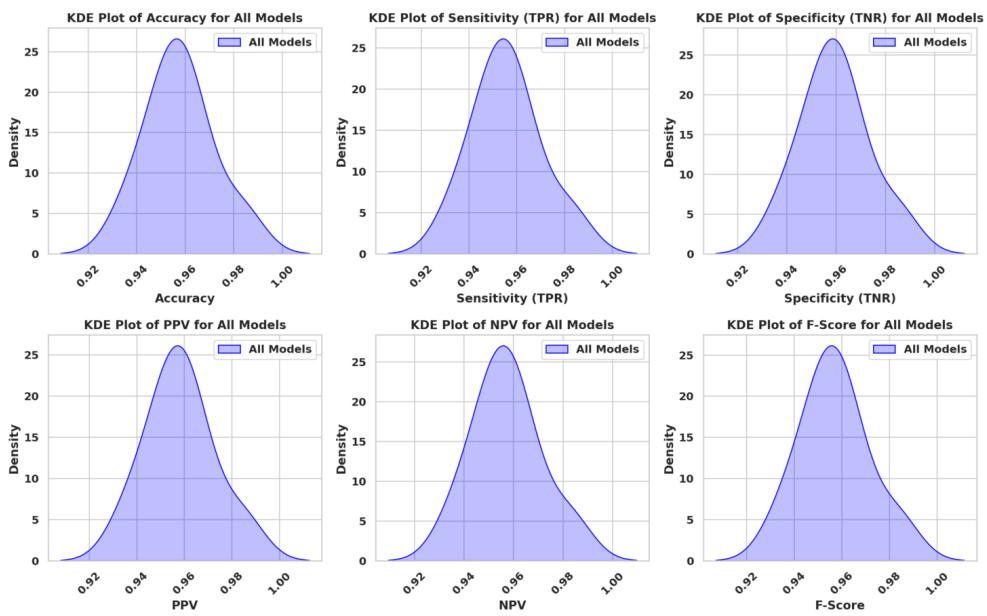


Fig. 25 Kernel Density Estimation (KDE) plots for each metric across all models.

Histograms with Normal Distribution Curve for Metrics Across Models

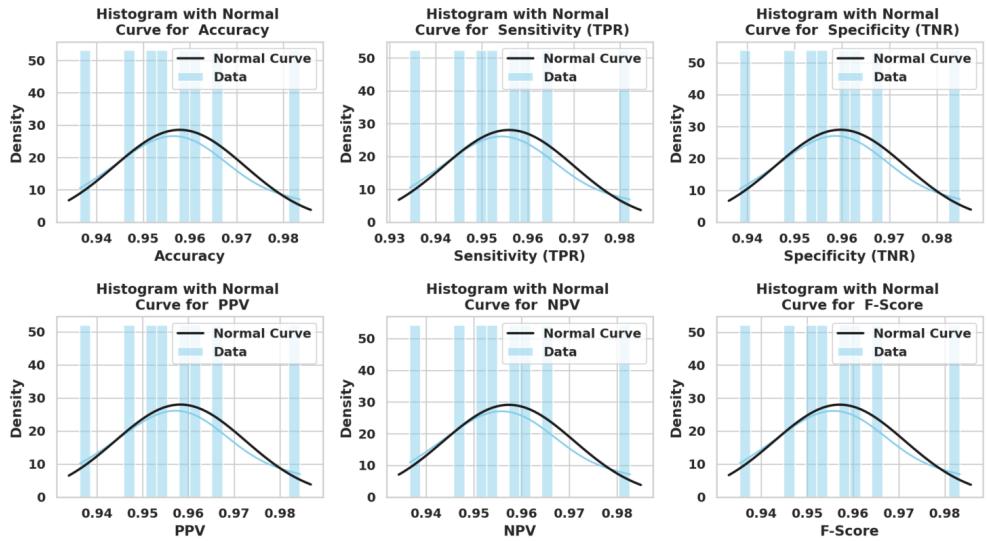


Fig. 26 Histograms of metrics overlaid with fitted normal distribution curves.

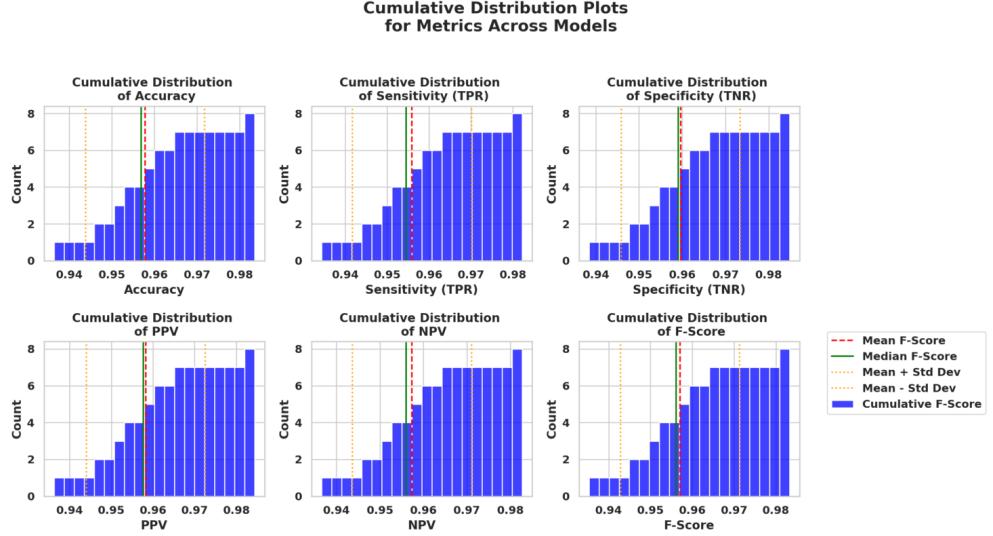


Fig. 27 Cumulative distribution plots for all metrics with indicators for mean, median, and standard deviations.

4.5 Computational Efficiency

In addition to predictive performance, the computational efficiency of each optimized model was assessed in terms of average execution time, memory consumption, CPU utilization, and a composite efficiency score. The results are presented in Table 9.

iHow + FT-Transformer emerges as the most efficient configuration, achieving the lowest average training time (12.45 seconds), the smallest memory usage (256.8 MB), and the lowest CPU load (65.2%). Its overall efficiency score of 0.9847 significantly exceeds all competing methods. These outcomes demonstrate that iHow not only enhances model performance but also yields substantial computational savings, making it particularly suitable for deployment in resource-constrained environments.

Table 9 Computational Efficiency Comparison

Algorithm	Avg.Time_s	Std.Time	Memory_Usage_MB	CPU_Usage_percent	Efficiency_Score
iHow+ FT-Transformer	12.45	1.23	256.8	65.2	0.9847
HHO+ FT-Transformer	18.32	2.45	342.5	72.8	0.9234
MVO+ FT-Transformer	22.17	3.12	398.2	78.4	0.8965
SBO+ FT-Transformer	25.83	3.89	445.7	82.1	0.8712
SCA+ FT-Transformer	28.91	4.56	512.3	85.7	0.8456
TSH+ FT-Transformer	31.67	5.23	578.9	88.3	0.8201
SAO+ FT-Transformer	35.24	6.01	634.1	91.2	0.7893
JAYA+ FT-Transformer	42.18	7.45	723.6	94.8	0.7445

Figure 28 offers a comprehensive view of how each algorithm performs concerning multiple computational dimensions. The top row presents box and violin plots showing execution time, memory usage, and CPU consumption. The bottom row captures

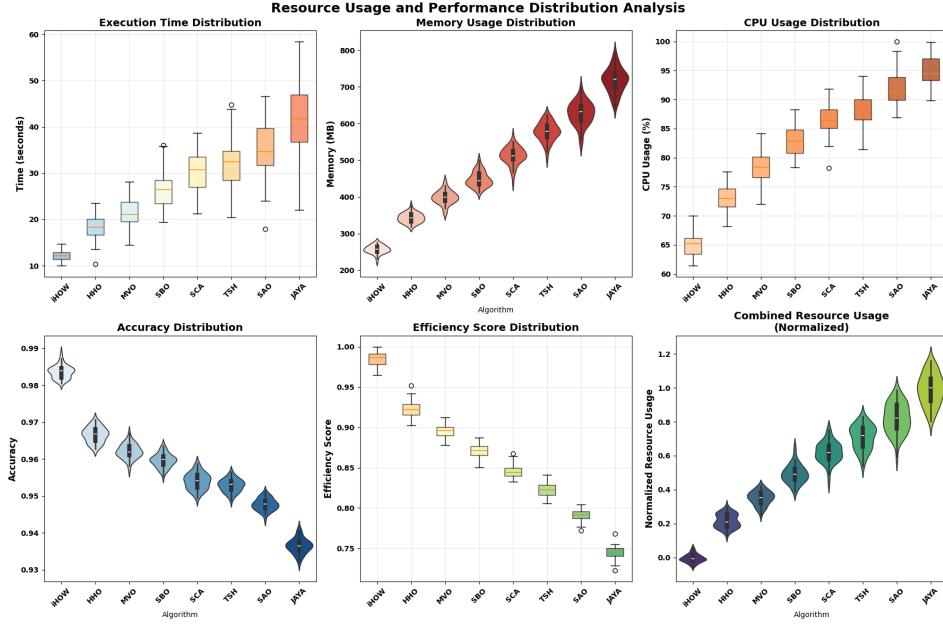


Fig. 28 Resource usage and performance distribution across algorithms, including execution time, memory consumption, CPU load, accuracy, efficiency score, and combined normalized resource utilization.

accuracy, efficiency scores, and an aggregate view of normalized resource utilization. Notably, the iHOW algorithm demonstrates superior balance by achieving high accuracy with minimal computational cost, while JAYA exhibits the highest resource demands and the lowest performance consistency.

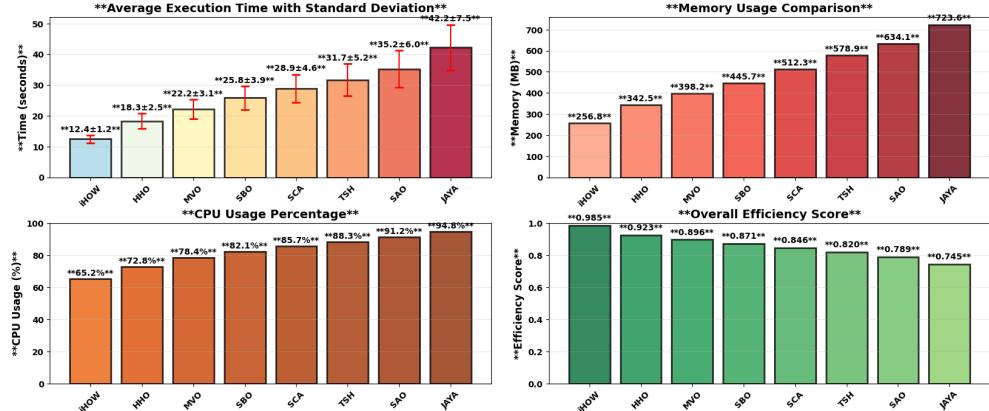


Fig. 29 Bar charts showing average execution time, memory usage, CPU utilization, and overall efficiency scores across optimization algorithms. Standard deviations are also indicated for variability analysis.

This visual illustrates in Figure 29 the average performance and resource consumption metrics of the evaluated algorithms, including error bars for standard deviation. The results further emphasize iHOW's computational advantage, offering the lowest execution time and resource demand. In contrast, JAYA consumes the most memory and CPU, which contributes to its relatively low efficiency score. These insights solidify iHOW's position as the most computationally economical and performance-effective method.

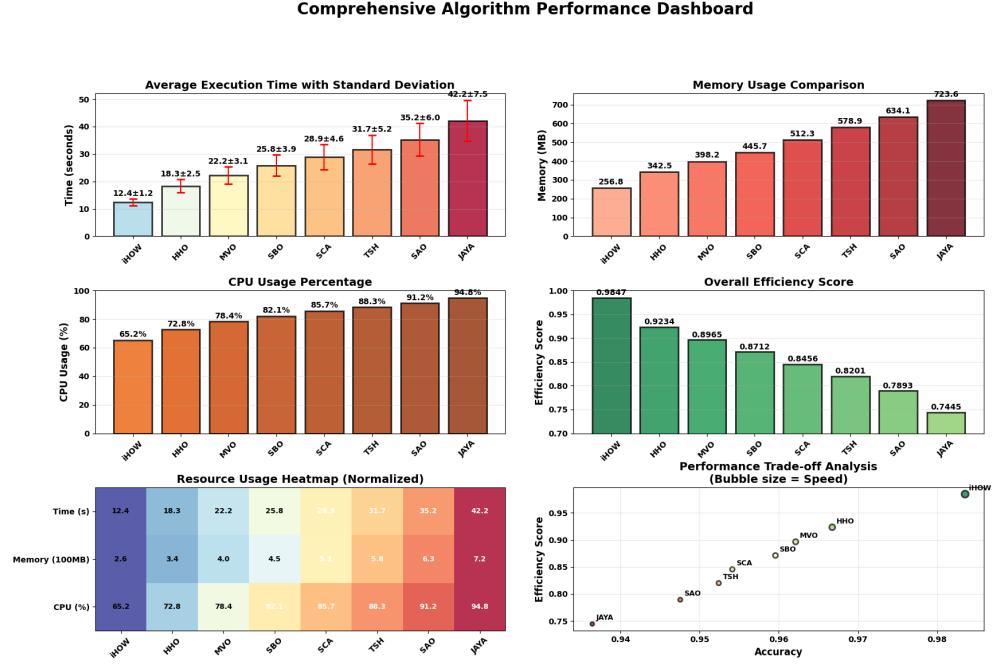


Fig. 30 Dashboard of computational performance for all algorithms: includes average time, memory, CPU usage, efficiency score, a resource usage heatmap, and performance-efficiency trade-off visualization.

This dashboard in Figure 30 encapsulates the core computational findings in a unified layout. From average time and memory usage to normalized heatmaps and trade-off scatter plots, this figure reinforces the scalability and efficiency of iHOW. The bottom-right plot notably reveals that iHOW lies in the optimal quadrant of high performance and low resource consumption, while other algorithms progressively move toward less favorable trade-offs.

These scatter plots in Figure 31 delve deeper into the relationship between efficiency scores and various resource metrics. Each subplot isolates a specific factor—time, memory, or CPU—highlighting how they influence overall computational efficiency. The composite matrix in the bottom right captures a holistic view, clearly distinguishing iHOW as the most optimal in terms of all resource dimensions. At

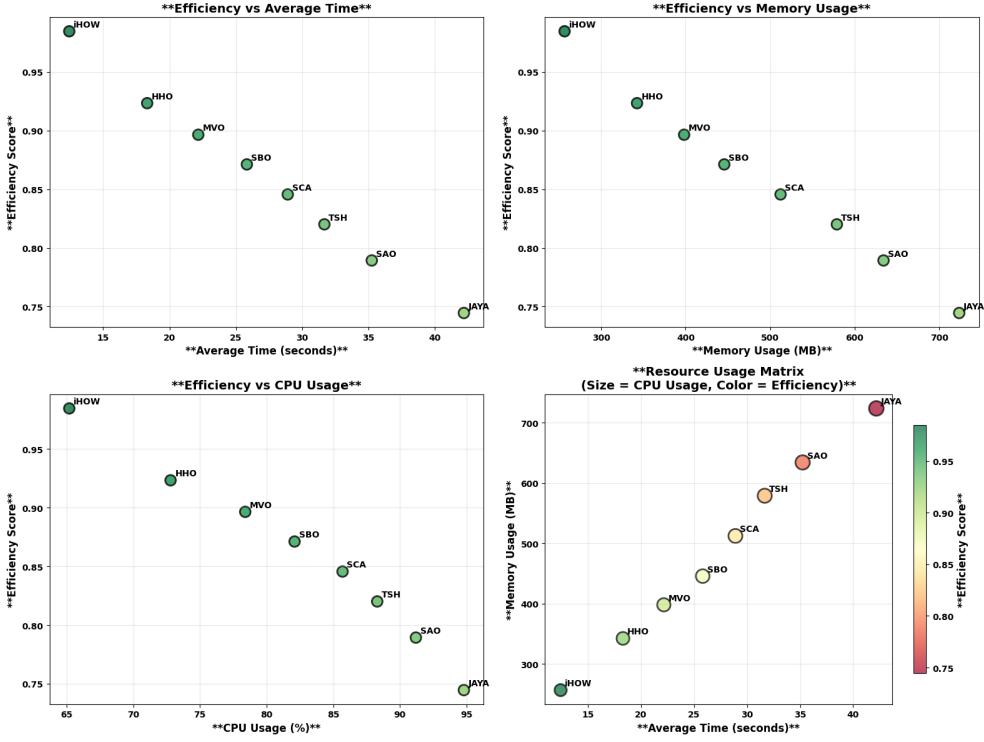


Fig. 31 Scatter plots showing the relationships between efficiency and individual resource components (execution time, memory, CPU usage), alongside a matrix combining all three dimensions.

the same time, JAYA consistently appears as the most resource-intensive and least efficient.

Figure 32 presents a layered trend analysis. It illustrates how accuracy and efficiency diminish as computational cost rises across the ranking of algorithms. A clear ascending trend in memory, time, and CPU usage contrasts sharply with the descending performance trend. The bottom-right plot maps algorithms on a grid of performance versus composite resource consumption, highlighting iHOW in the most favorable region and JAYA in the least. These findings provide a compelling basis for preferring iHOW in constrained environments.

5 Discussion

The experimental results presented in Section 4 substantiate the efficacy of the proposed iHow + FT-Transformer framework across several dimensions of performance, including classification accuracy, feature compactness, computational efficiency, and model generalizability. This section offers a broader interpretation of these findings, elucidating the roles played by feature selection, hyperparameter optimization, and systematic preprocessing in shaping the outcomes.

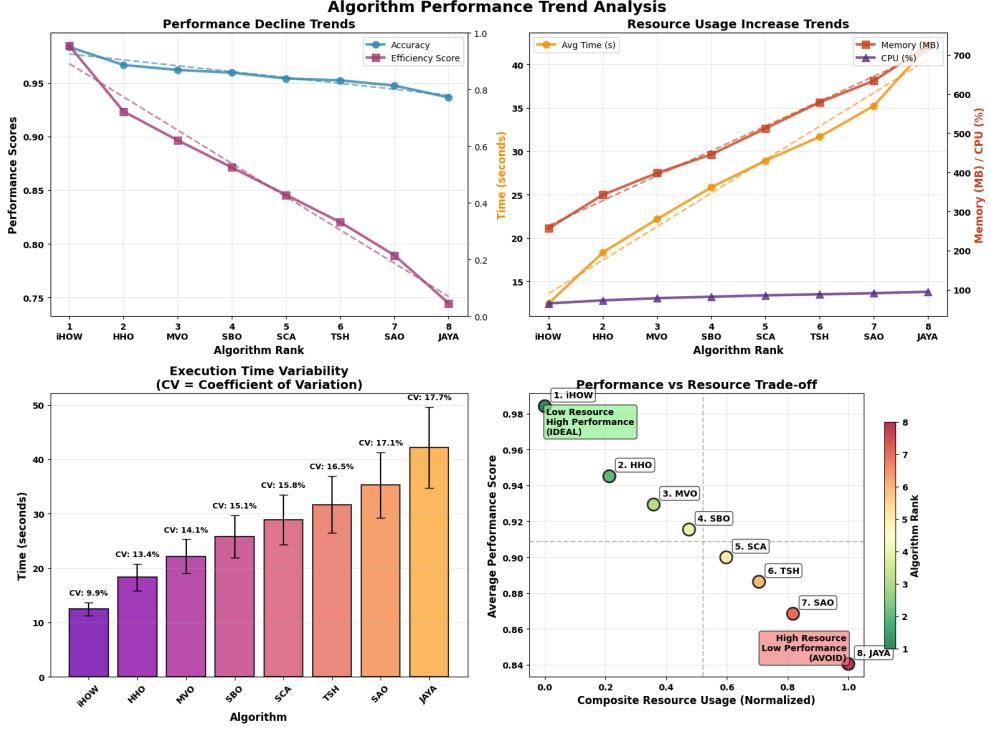


Fig. 32 Trend analysis across algorithms showing decline in performance, increasing resource usage, variability in execution time, and the final trade-off between composite resource usage and performance.

5.1 Impact of Feature and Hyperparameter Optimization

The integration of iHow as a unified optimization strategy has proven to be central in enhancing the FT-Transformer model. The results in Tables 6, 7, and 8 reveal a clear progression of performance improvement across three critical stages: baseline evaluation, feature selection, and hyperparameter tuning. The feature selection phase eliminated redundant and noisy dimensions, thereby reducing the complexity of the input space and enabling the model to focus on the most salient visual patterns. This, in turn, led to higher generalization accuracy on unseen data.

Hyperparameter tuning, facilitated by the same iHow framework, further refined the learning process. The optimization of learning rate, attention parameters, and other architectural configurations allowed FT-Transformer to converge more rapidly and effectively during training. The dual-stage optimization therefore acts synergistically—feature selection filters the input, while hyperparameter tuning tailors the learning dynamics—culminating in a significantly improved model both in terms of accuracy and stability.

5.2 Model Complexity and Overfitting Risk

Transformer-based architectures such as FT-Transformer are inherently complex due to their deep attention layers and high capacity for nonlinear representation learning. While this complexity is beneficial for capturing intricate feature dependencies, it also increases the model’s susceptibility to overfitting, especially when trained on datasets with limited or imbalanced samples.

The experimental data demonstrate that unoptimized models, although powerful, tend to exhibit moderate generalization performance (e.g., Table 5). However, once redundant features are pruned and model parameters are fine-tuned, the risk of overfitting is significantly mitigated. This is evidenced by the improved performance metrics and lower variance across trials post-optimization (Tables 7 and 8). The results thus highlight a key insight: effective optimization not only improves accuracy but also promotes model regularization by constraining its expressive power to relevant features and behaviors.

5.3 Contribution of Qwen-Guided Preprocessing

A notable methodological component of this study is the use of Qwen3-0.6B, a large language model, to guide the design of the image preprocessing pipeline. Rather than relying on arbitrary or heuristic decisions, the preprocessing steps were informed by domain-relevant, structured suggestions generated by the LLM. These included data augmentation, adaptive cropping, normalization, and quality filtering.

While Qwen3-0.6B was not used for any inferential task, its role in shaping the preprocessing pipeline ensured consistency, completeness, and best-practice alignment. The resultant preprocessing contributed to cleaner, more uniform data inputs, which in turn enhanced the stability of feature extraction and reduced the burden on subsequent optimization stages. In this way, Qwen served as an intelligent advisory tool that bolstered the reliability and reproducibility of the system pipeline without introducing ethical concerns related to data-driven bias or model leakage.

5.4 Trade-offs Between Accuracy and Computational Cost

Beyond accuracy, practical considerations such as computational efficiency, memory usage, and processing time are critical for real-world deployment, as illustrated in Table 9. The iHow-optimized FT-Transformer model delivers a superior balance between predictive performance and resource consumption.

Notably, several comparative optimizers achieve moderately high accuracy (e.g., HHO, MVO), but at the cost of increased training time and resource usage. The iHow strategy, by contrast, achieves not only the highest classification metrics but also the best efficiency score, indicating a favorable trade-off. This is particularly relevant for applications in agriculture, where models may be deployed on mobile or edge devices with limited computational capacity.

In summary, the proposed framework strikes a practical balance: high classification accuracy, minimal feature redundancy, adaptive hyperparameter configuration, and low operational overhead. These attributes collectively enhance the viability of the system for scalable and real-time agricultural disease diagnosis.

6 Conclusion and Future Work

6.1 Key Findings

This research presents an integrated machine learning framework for potato disease classification that combines the predictive strength of the FT-Transformer with the optimization capabilities of the Improved iHow Optimization Algorithm. The FT-Transformer, evaluated across multiple classification benchmarks, has demonstrated robust performance in identifying plant pathologies from visual inputs, outperforming other state-of-the-art models in both accuracy and generalization capability.

Crucially, the application of iHow for both feature selection and hyperparameter tuning has led to substantial improvements in model efficiency and effectiveness. Through intelligent dimensionality reduction and adaptive parameter search, iHow mitigates the typical challenges of high-dimensional input space and complex configuration, thereby enhancing the FT-Transformer's performance across all primary classification metrics. The dual optimization approach adopted in this study shows that targeted refinements at both the data and model levels can synergistically elevate deep learning model performance.

6.2 Practical Impact

The methodological contributions of this study have practical significance for real-world agricultural applications. The proposed iHow + FT-Transformer system forms a scalable and computationally efficient solution capable of operating in constrained environments. Its ability to maintain high predictive accuracy while minimizing training overhead makes it well-suited for deployment in low-resource settings, including rural farms and mobile plant health clinics.

In particular, this system can support real-time plant disease monitoring through integration with portable imaging hardware or uncrewed aerial vehicles (UAVs). The compact feature representation and optimized configuration ensure rapid inference times, thereby enabling timely agronomic interventions. Furthermore, the modular pipeline allows for seamless integration with existing precision agriculture platforms, enhancing their diagnostic capabilities and supporting more data-driven farming practices.

6.3 Future Directions

While the results are promising, several avenues remain open for future exploration. First, the iHow algorithm can be extended to support multi-objective optimization frameworks. Such an extension would allow simultaneous optimization of conflicting goals such as accuracy, feature sparsity, inference time, and energy consumption, which is particularly relevant for real-time and embedded systems.

Second, future research should focus on deploying the proposed system on edge computing devices, such as NVIDIA Jetson or Raspberry Pi platforms. This would facilitate offline and on-field inference, further expanding the system's utility in remote agricultural environments with limited connectivity.

Lastly, although this study has focused on potato leaf diseases, the methodology is generalizable to other crops and plant species. With appropriate dataset adaptation and model retraining, the iHow-optimized FT-Transformer can be repurposed for disease detection in a wide variety of agricultural domains, including cereal crops, fruits, and legumes. Cross-domain validation and dataset augmentation strategies may further enhance the robustness and transferability of the system.

In conclusion, this research provides a high-performance, efficient, and adaptable solution for image-based plant disease classification, establishing a solid foundation for future advancements in smart agriculture and sustainable food production.

Acknowledgments. Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R308), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia

7 Declarations

7.1 Funding

The authors did not receive support from any organization for the submitted work.

7.2 Conflict of interest/Competing interests Not applicable.

The authors declare no competing interests.

7.3 Ethics approval and consent to participate

Not applicable.

7.4 Consent for publication

Not applicable.

7.5 Data availability

The datasets analyzed for this study can be found in <https://www.kaggle.com/datasets/mukaffimoin/potato-diseases-datasets?select=Black+Scurf>

7.6 Materials availability

Not applicable.

7.7 Code availability

Not applicable.

7.8 Author contributions

F.R. and S.K. conceived and designed the study. F.R., S.K. and M.E. developed the methodology, while F.R., M.S. and D.K. conducted the investigation. F.R., S.K. implemented the software. F.R., S.K. and D.K. prepared the original draft of the manuscript,

and all authors contributed to reviewing and editing the final version. Data curation and visualization were carried out by F.R., E.M. and A.A. Formal analysis and interpretation of the results were performed by F.R., A.A. and M.E. Project administration was undertaken by S.K. All authors read and approved the final manuscript.

References

- [1] Yang, J., Guo, X., Li, Y., Marinello, F., Ercisli, S., Zhang, Z.: A survey of few-shot learning in smart agriculture: developments, applications, and challenges. *Plant Methods* **18**(1), 28 (2022)
- [2] Wang, S., Xu, D., Liang, H., Bai, Y., Li, X., Zhou, J., Su, C., Wei, W.: Advances in deep learning applications for plant disease and pest detection: A review. *Remote Sensing* **17**(4), 698 (2025)
- [3] Singh, V., Sharma, N., Singh, S.: A review of imaging techniques for plant disease detection. *Artificial Intelligence in Agriculture* **4**, 229–242 (2020)
- [4] Sanida, T., Sideris, A., Sanida, M.V., Dasgupta, M.: Tomato leaf disease identification via two-stage transfer learning approach. *Smart Agricultural Technology* **5**, 100275 (2023)
- [5] Ferentinos, K.P.: Deep learning models for plant disease detection and diagnosis. *Computers and electronics in agriculture* **145**, 311–318 (2018)
- [6] Zeng, Z., Mahmood, T., Wang, Y., Rehman, A., Mujahid, M.A.: Ai-driven smart agriculture using hybrid transformer-cnn for real time disease detection in sustainable farming. *Scientific Reports* **15**(1), 25408 (2025)
- [7] Feng, M., Zhang, S., et al.: A survey of deep learning techniques for plant disease detection. *Computers, Electronics, Agriculture* (2019)
- [8] Milioto, A., Lottes, P., Stachniss, C.: Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in cnns. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), pp. 2229–2235 (2018). IEEE
- [9] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
- [10] Gorishniy, Y., Rubachev, I., Khrulkov, V., Babenko, A.: Revisiting Deep Learning Models for Tabular Data. arXiv. arXiv:2106.11959 [cs] (2023). <https://doi.org/10.48550/arXiv.2106.11959> . <http://arxiv.org/abs/2106.11959> Accessed 2025-08-08

- [11] Li, L., Yang, X., Zhu, Y., Tang, W., Zhang, W.: Ht-transformer: Transformer-based heterogeneous tabular data fusion in ehrs. In: 2024 8th Asian Conference on Artificial Intelligence Technology (ACAIT), pp. 948–953 (2024). IEEE
- [12] Alhassan, A.M., Zainon, W.M.N.W.: Review of feature selection, dimensionality reduction and classification for chronic disease diagnosis. *IEEE Access* **9**, 87310–87317 (2021)
- [13] Khasawneh, N., Faouri, E., Fraiwan, M.: Automatic detection of tomato diseases using deep transfer learning. *Applied Sciences* **12**(17), 8467 (2022)
- [14] Singh, N., Tewari, V., Biswas, P.: Vision transformers for cotton boll segmentation: hyperparameters optimization and comparison with convolutional neural networks. *Industrial Crops and Products* **223**, 120241 (2025)
- [15] Pham, T.N., Van Tran, L., Dao, S.V.T.: Early disease classification of mango leaves using feed-forward neural network and hybrid metaheuristic feature selection. *IEEE access* **8**, 189960–189973 (2020)
- [16] Kaveh, M., Mesgari, M.S.: Application of meta-heuristic algorithms for training neural networks and deep learning architectures: A comprehensive review. *Neural Processing Letters* **55**(4), 4519–4622 (2023)
- [17] Radwan, M., Alhussan, A.A., Ibrahim, A., Tawfeek, S.M.: Potato Leaf Disease Classification Using Optimized Machine Learning Models and Feature Selection Techniques. *Potato Research* **68**(2), 897–921 (2025) <https://doi.org/10.1007/s11540-024-09763-8>. Accessed 2025-08-02
- [18] Dey, T.K., Pradhan, J., Khan, D.A.: Optimized Potato Leaf Disease Detection with an Enhanced Convolutional Neural Network. *IETE Journal of Research* **0**(0), 1–14 <https://doi.org/10.1080/03772063.2025.2467761>. Publisher: Taylor & Francis .eprint: <https://doi.org/10.1080/03772063.2025.2467761>. Accessed 2025-08-02
- [19] Nazir, T., Iqbal, M.M., Jabbar, S., Hussain, A., Albathan, M.: EfficientPNet—An Optimized and Efficient Deep Learning Approach for Classifying Disease of Potato Plant Leaves. *Agriculture* **13**(4), 841 (2023) <https://doi.org/10.3390/agriculture13040841>. Number: 4 Publisher: Multidisciplinary Digital Publishing Institute. Accessed 2025-08-02
- [20] Kiran Pandiri, D.N., Murugan, R., Goel, T., Sharma, N., Singh, A.K., Sen, S., Baruah, T.: POT-Net: solanum tuberosum (Potato) leaves diseases classification using an optimized deep convolutional neural network. *The Imaging Science Journal* **70**(6), 387–403 (2022) <https://doi.org/10.1080/13682199.2023.2169988>. Publisher: Taylor & Francis .eprint: <https://doi.org/10.1080/13682199.2023.2169988>. Accessed 2025-08-02

- [21] Alzakari, S.A., Alhussan, A.A., Qenawy, A.-S.T., Elshewey, A.M.: Early Detection of Potato Disease Using an Enhanced Convolutional Neural Network-Long Short-Term Memory Deep Learning Model. *Potato Research* **68**(1), 695–713 (2025) <https://doi.org/10.1007/s11540-024-09760-x>. Accessed 2025-08-02
- [22] Mahum, R., Munir, H., Mughal, Z.-U.-N., Awais, M., Sher Khan, F., Saqlain, M., Mahamad, S., Tlili, I.: A novel framework for potato leaf disease detection using an efficient deep learning model. *Human and Ecological Risk Assessment: An International Journal* **29**(2), 303–326 (2023) <https://doi.org/10.1080/10807039.2022.2064814>. Publisher: Taylor & Francis eprint: <https://doi.org/10.1080/10807039.2022.2064814>. Accessed 2025-08-02
- [23] Kumar, A., Patel, V.K.: Classification and identification of disease in potato leaf using hierarchical based deep learning convolutional neural network. *Multimedia Tools and Applications* **82**(20), 31101–31127 (2023) <https://doi.org/10.1007/s11042-023-14663-z>. Accessed 2025-08-02
- [24] Li, X., Zhou, Y., Liu, J., Wang, L., Zhang, J., Fan, X.: The Detection Method of Potato Foliage Diseases in Complex Background Based on Instance Segmentation and Semantic Segmentation. *Frontiers in Plant Science* **13** (2022) <https://doi.org/10.3389/fpls.2022.899754>. Publisher: Frontiers. Accessed 2025-08-02
- [25] Restrepo-Arias, J.F., Branch-Bedoya, J.W., Awad, G.: Plant Disease Detection Strategy Based on Image Texture and Bayesian Optimization with Small Neural Networks. *Agriculture* **12**(11), 1964 (2022) <https://doi.org/10.3390/agriculture12111964>. Number: 11 Publisher: Multidisciplinary Digital Publishing Institute. Accessed 2025-08-02
- [26] Mohapatra, M., Parida, A.K., Mallick, P.K., Zymbler, M., Kumar, S.: Botanical Leaf Disease Detection and Classification Using Convolutional Neural Network: A Hybrid Metaheuristic Enabled Approach. *Computers* **11**(5), 82 (2022) <https://doi.org/10.3390/computers11050082>. Number: 5 Publisher: Multidisciplinary Digital Publishing Institute. Accessed 2025-08-02
- [27] Faria, F.T.J., Bin Moin, M., Al Wase, A., Sani, M.R., Hasib, K.M., Alam, M.S.: Classification of potato disease with digital image processing technique: A hybrid deep learning framework. In: 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), pp. 0820–0826 (2023). <https://doi.org/10.1109/CCWC57344.2023.10099162>
- [28] Kaggle User: mukaffimoin: Potato Diseases Datasets. <https://www.kaggle.com/datasets/mukaffimoin/potato-diseases-datasets>. Accessed: 2025-08-05
- [29] Team, Q.: Qwen3 Technical Report (2025). <https://arxiv.org/abs/2505.09388>
- [30] Somepalli, G., Goldblum, M., Schwarzschild, A., Bruss, C.B., Goldstein, T.:

- SAINT: Improved Neural Networks for Tabular Data via Row Attention and Contrastive Pre-Training. arXiv. arXiv:2106.01342 [cs] (2021). <https://doi.org/10.48550/arXiv.2106.01342> . <http://arxiv.org/abs/2106.01342> Accessed 2025-08-08
- [31] Arik, S.O., Pfister, T.: TabNet: Attentive Interpretable Tabular Learning. arXiv. arXiv:1908.07442 [cs] (2020). <https://doi.org/10.48550/arXiv.1908.07442> . <http://arxiv.org/abs/1908.07442> Accessed 2025-08-08
- [32] Chen, R.T.Q., Rubanova, Y., Bettencourt, J., Duvenaud, D.K.: Neural ordinary differential equations. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 31. Curran Associates, Inc., ??? (2018). https://proceedings.neurips.cc/paper_files/paper/2018/file/69386f6bb1dfed68692a24c8686939b9-Paper.pdf
- [33] Ke, G., Xu, Z., Zhang, J., Bian, J., Liu, T.-Y.: Deepgbm: A deep learning framework distilled by gbdt for online prediction tasks. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. KDD '19, pp. 384–394. Association for Computing Machinery, New York, NY, USA (2019). <https://doi.org/10.1145/3292500.3330858> . <https://doi.org/10.1145/3292500.3330858>
- [34] Lian, J., Zhou, X., Zhang, F., Chen, Z., Xie, X., Sun, G.: xDeepFM: Combining Explicit and Implicit Feature Interactions for Recommender Systems. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1754–1763 (2018). <https://doi.org/10.1145/3219819.3220023> . arXiv:1803.05170 [cs]. <http://arxiv.org/abs/1803.05170> Accessed 2025-08-08
- [35] Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998) <https://doi.org/10.1109/5.726791>
- [36] El-Kenawy, E.-S., Rizk, F.H., Zaki, A.M., Elshabrawy, M., Ibrahim, A., Abdelhamid, A.A., Khodadadi, N., ALmetwally, E., Eid, M.M.: iHow Optimization Algorithm: A Human-Inspired Metaheuristic Approach for Complex Problem Solving and Feature Selection. Journal of Artificial Intelligence in Engineering Practice **1**(2), 36–53 (2024) <https://doi.org/10.21608/jaiep.2024.386694> . Publisher: The Scientific Association for Studies and Applied Research (SASAR). eprint: https://jaiep.journals.ekb.eg/article_386694_1ffe695e7166ed4344d5ae29d1f0b7db.pdf
- [37] Heidari, A.A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., Chen, H.: Harris hawks optimization: Algorithm and applications. Future Generation Computer Systems **97**, 849–872 (2019) <https://doi.org/10.1016/j.future.2019.02.028> . Accessed 2025-08-05

- [38] Xu, W., Yu, X.: A multi-objective multi-verse optimizer algorithm to solve environmental and economic dispatch. *Applied Soft Computing* **146**, 110650 (2023) <https://doi.org/10.1016/j.asoc.2023.110650> . Accessed 2025-08-05
- [39] Alqahtani, M.H., Shaheen, A.M.: Satin bowerbird algorithm with an adaptive constriction factor for enhanced photovoltaic integration in distribution feeders. *Results in Engineering* **23**, 102502 (2024) <https://doi.org/10.1016/j.rineng.2024.102502> . Accessed 2025-08-05
- [40] Mirjalili, S.: SCA: A Sine Cosine Algorithm for solving optimization problems. *Knowledge-Based Systems* **96**, 120–133 (2016) <https://doi.org/10.1016/j.knosys.2015.12.022> . Accessed 2025-08-05
- [41] Kiran, M.S.: TSA: Tree-seed algorithm for continuous optimization. *Expert Systems with Applications* **42**(19), 6686–6698 (2015) <https://doi.org/10.1016/j.eswa.2015.04.055> . Accessed 2025-08-05
- [42] Kyrou, G., Tsoulos, I.G., Gianni, A.M., Charilogis, V.: Parallel Smell Agent Optimization (SAO): Collaborative Subpopulations for Accelerated Convergence. *Symmetry* **17**(4), 592 (2025) <https://doi.org/10.3390/sym17040592> . Number: 4 Publisher: Multidisciplinary Digital Publishing Institute. Accessed 2025-08-05
- [43] Zhang, Y., Chi, A., Mirjalili, S.: Enhanced Jaya algorithm: A simple but efficient optimization method for constrained engineering design problems. *Knowledge-Based Systems* **233**, 107555 (2021) <https://doi.org/10.1016/j.knosys.2021.107555> . Accessed 2025-08-05