AL-RYADA JOURNAL FOR COMPUTATIONAL INTELLIGENCE AND TECHNOLOGY (ARJCIT)

VOL. 2 (1) - 2025

Enhancing Classification Accuracy Using Stacked Ensemble Learning:A Hybrid Ensemble

[1*]Sherif Samir M.Sultan [0009-0005-0757-7202], [1*]Mahmoud Ewieda[0009-0005-6455-0081], [2]Mohamed Fathi Yahia[0009-0005-6709-3795] , [3] Ahmed Ali Gomaa

[1,2] Business Information Systems Department, Faculty of Business Administration, Al RYADA University for science and technology, Egypt
[1,] Information Systems Department, Faculty of Commerce and Business Administration, Helwan University, Cairo, EGYPT
[3] Faculty of Computer and Artificial Intelligence, Al RYADA University for science and technology, Egypt

Article history:

Received: 14 – June – 2025

Revised: 10 – Aug – 2025

Accepted: 13 – Oct – 2025

Available online: 1 – Dec – 2025

https://doi.org/10.21608/ajcit.2025.394368.1014

Correspondence

Sherif Samir M.Sultan, Mohamed Fathi Yahia

Business Information Systems Department, Faculty of Business Administration, Al RYADA University for science and technology, Al Sadat City, Egypt

Email:
Sherif.Samir21@commerce.helwan.edu.eg

Sherif.Sultan@rst.edu.eg

Mohamed.Fathi21@commerce.helwan.edu.eg
Mohamed.Yahya@rst.edu.eg

ABSTRACT:

Ensemble modeling has become a critical approach in modern machine learning, substantially enhancing predictive accuracy by aggregating the strengths of multiple classifiers while mitigating individual model biases and variance. This study evaluates the effectiveness of a stacking ensemble framework that integrates a Support Vector Machine (SVM) with a Radial Basis Function (RBF) kernel and a Multi-Layer Perceptron (MLP)-based Neural Network (NN). These base models, developed using distinct learning paradigms, exhibit complementary generalization capabilities and are combined into a unified meta-classifier through stacking techniques. The methodology was applied to Fisher's Iris dataset, a well-established multivariate benchmark widely used in pattern recognition research. The classification pipeline comprised two main phases: the independent development of the base models and the construction of the stacked ensemble. The dataset was partitioned into 80% for training and 20% for testing to evaluate performance consistency. Experimental results indicate that the SVM model achieved a training accuracy of 99.17%, a Matthews Correlation Coefficient (MCC) of 0.9876, and an F1-score of 0.9917. The MLP-based NN attained a training accuracy of 98.33%, an MCC of 0.9754, and an F1-score of 0.9833. Notably, the stacked ensemble model outperformed both base classifiers, achieving perfect test set metrics with 100% accuracy, MCC, and F1-score. These findings confirm the robustness and superior predictive capacity of the stacking ensemble approach over individual models and underscore its potential for constructing high-performing, reliable classification systems

.

KEYWORDS

Machine Learning (ML); Support Vector Machine (SVM); Neural Network (NN); Ensemble learning; Stacking ensemble

1. INTRODUCTION

Ensemble learning has become a cornerstone technique in machine learning (ML), renowned for its capacity to address complex computational challenges by aggregating predictions from multiple base models, often referred to as weak learners. One prominent example is the Random Forest (RF) algorithm, which leverages an ensemble of decision trees to improve both accuracy and generalizability. The central objective of ensemble learning is to enhance predictive performance across various domains, including classification, regression, and function approximation, by exploiting the complementary strengths of diverse learning algorithms [1].

Among ensemble strategies, stacking stands out for its superior predictive capability. Widely employed in ML and data science competitions, stacking models often surpass the performance of individual classifiers. This is achieved by training a set of heterogeneous base learners on the same dataset and subsequently using their outputs as input features for a higher-level meta-learner [2]. The meta-learner synthesizes these predictions to produce a final, more accurate outcome. Unlike conventional ML models that map inputs directly to outputs, stacking models operate at a meta-level, capturing the relationships among base learners' predictions and the true labels [3, 4].

Ensemble classifiers can be constructed using several well-established techniques. The most commonly adopted methods include: (a) employing varied subsets of the training data with custom learning schemes; (b) altering initialization parameters or training procedures; and (c) integrating fundamentally different learning algorithms [5]. The superiority of ensemble methods over single classifiers is rooted in their ability to address representational, statistical, and computational limitations, particularly in scenarios where training data is insufficient relative to the hypothesis space [6].

Despite their demonstrated potential, ensemble approaches still warrant further investigation to optimize their architecture and maximize performance. In this context, the present study introduces a novel Stacking Ensemble Learning (SEL) framework that combines a Support Vector Machine (SVM) with a Radial Basis Function (RBF) kernel and a Multi-Layer Perceptron (MLP) neural network. This hybrid model is designed to harness the complementary capabilities of SVM and MLP, thereby enhancing classification effectiveness across diverse datasets [7].

The proposed SEL framework is evaluated on multiple benchmark datasets, including Fisher's Iris dataset. Comparative analyses reveal that it consistently outperforms traditional ensemble techniques in terms of classification accuracy. To further validate its effectiveness, cross-model evaluation was conducted to compare the performance of different classifiers such as neural networks and SVMs, using consistent training and validation splits [8, 9].

Artificial Neural Networks (ANNs), inspired by the architecture of the human brain, are known for their adaptability, error resilience, and ability to learn from experience. Structurally, ANNs comprise interconnected layers of nodes (neurons), where weighted connections adjust iteratively during training to minimize prediction error. This makes them highly suitable for capturing complex, non-linear relationships in data [10]. Ensemble approaches, particularly those employing hybrid stacking strategies, have proven effective in tasks such as data fusion, feature selection, and high-dimensional classification [11].

1.1. Motivation for the Proposed SEL Framework

To overcome limitations associated with conventional ensemble models, this study proposes a robust Stacking Ensemble Learning (SEL) classification framework incorporating the following innovations:

- **Class-Specific Model Weighting:** The framework dynamically assigns weights to base learners according to their performance on specific classes, enhancing accuracy for imbalanced or minority class distributions.
- **Hybrid Integration of Diverse Classifiers:** By combining SVMs with RBF kernels and MLP neural networks, the framework leverages their complementary strengths to improve generalization and robustness.
- **Robust Meta-Classifier Construction:** Stacking heterogeneous model predictions enables the construction of a meta-classifier capable of outperforming any single constituent learner.
- **Improved Computational Efficiency:** The system prioritizes high-performing models during prediction, reducing redundancy and enhancing scalability for large or complex datasets.
- **Validated Performance Gains:** Experimental results on benchmark datasets, including the Iris dataset, confirm that the SEL framework yields higher classification accuracy, affirming the efficacy of hybrid stacking strategies.

In summary, the proposed SEL framework offers a comprehensive and adaptive solution to the challenges of classification tasks. By integrating heterogeneous learners, dynamically weighting their contributions, and optimizing meta-level prediction, the framework demonstrates both superior accuracy and computational efficiency across a variety of benchmark datasets.

The structure of this paper is organized as follows: Section 2 provides a review of recent developments in the field. Section 3 outlines the proposed methodology, while Section 4 details the experimental results. Lastly, Section 5 offers the concluding remarks.

2. RELATED WORK

Zaidi et al. (2025) [30] introduced Heart Ensemble Net, a hybrid ensemble model utilizing both stacking and voting methods to enhance cardiovascular disease (CVD) prediction. Trained on a dataset of 70,000 patients, it outperformed six traditional classifiers SVM, GB, DT, LR, KNN, and RF, and a hybrid RF-linear model (HRFLM), achieving 92.95% accuracy and 93.08% precision. The model demonstrated adaptability to other conditions like stroke and diabetes. However, it requires enhancements for interpretability and specificity. The authors suggest incorporating deep learning and transfer learning to further boost clinical relevance and support personalized diagnostic applications.

Mahmoud et al. (2025) [31] proposed a stacking ensemble model for liver cancer detection using high-dimensional gene expression data. The model integrates MLP, RF, KNN, and SVM, with XGBoost serving as the meta-learner. Techniques such as PCA and feature selection were applied for dimensionality reduction, and grid search was used for hyperparameter tuning. The system achieved 97% accuracy, 96.8% sensitivity, and 98.1% specificity, outperforming individual models. This study illustrates the importance of preprocessing and ensemble design for biomedical classification and offers potential applications in other cancers and real-time clinical decision support systems.

Yin et al. (2025) [32] investigated the integration of radionics' features (RFs) and deep learning features (DFs) for classifying brain tumors, including glioma, meningioma, and pituitary tumors, using contrast-

enhanced MRI. RFs were extracted via Pyradiomics and DFs via a 3D CNN. The features trained machine learning models such as SVM, RF, and MLP, with ensemble methods including Boosting and Stacking. Their RF + DF approach achieved 95% accuracy, 0.92 AUC, 88% sensitivity, and 90% specificity. Results show this hybrid method significantly improves diagnostic accuracy, emphasizing its clinical potential for brain tumor identification.

Vasheghani et al. (2025) [33] presented a Dynamic Ensemble Learning (DEL) framework for image classification, improving upon traditional static ensemble techniques. DEL dynamically selects classifiers CNNs, RNNs, CapsNets, SVMs, and RF based on class-specific performance thresholds. Only models exceeding a 0.9 threshold contribute to final predictions via majority voting. Applied to the Fashion-MNIST dataset, DEL achieved a 3.5% increase in accuracy and a 21.7% loss reduction. This adaptive mechanism enhances model scalability, addresses class imbalance, and reduces computational load, positioning DEL as a robust solution for real-time and evolving data stream classification tasks.

Suguna et al. (2025) [34] examined the impact of data imbalance on machine learning model performance, particularly in churn prediction within financial services. Using a financial churn dataset, they tested nine classifiers and six ensemble models. Single models underperformed, while ensemble techniques better identified the minority class but still lacked accuracy. Applying SMOTE significantly improved performance from 61% to 79%. Among all classifiers, AdaBoost achieved the best results with an F1-score of 87.6%. The study emphasizes the importance of balanced datasets and suitable classifiers for accurate churn detection and customer retention strategies.

Alalwany et al. (2025) [35] developed an intrusion detection system (IDS) tailored for the Internet of Medical Things (IoMT), integrating ML and DL through a stacking ensemble approach. Built within a Kappa Architecture for real-time stream processing, the system achieved 0.991 accuracy in binary and 0.993 in multi-class classification. It effectively detected multiple cyberattacks, including ARP spoofing and denial-of-service. Although highly accurate, the system still faces challenges in computational efficiency and model interpretability. Future enhancements aim to create adaptive, lightweight IDS solutions suitable for secure and responsive deployment in evolving IoMT environments.

Charoenkwan et al. (2025) [36] proposed Stack-AVP, a stacked ensemble learning model for predicting antiviral peptides (AVPs), which are short protein sequences effective against drug-resistant viruses. The framework employs 12 machine learning algorithms and a variety of feature encoding schemes. It combines multi-view features and optimized selection strategies to improve prediction. On independent test data, Stack-AVP achieved 93.0% accuracy, 0.860 MCC, and 0.975 AUC, outperforming existing AVP models. This work demonstrates the effectiveness of ensemble techniques in biomedical applications and supports computational drug discovery through efficient peptide classification.

Btoush et al. (2025) [37] tackled the increasing complexity of credit card fraud detection using a hybrid stacking framework that combines machine learning and deep learning. Their model incorporates classifiers such as DT, RF, SVM, XGBoost, CatBoost, and LR, along with CNN and BiLSTM architectures enhanced with attention mechanisms. The system also uses resampling to manage class imbalance. Experimental evaluation showed an F1-score of 94.63%, highlighting the model's robustness in detecting fraudulent transactions. This approach demonstrates substantial improvement over traditional methods and offers a scalable solution for combating dynamic fraud strategies in banking.

Tang et al. (2025) [38] introduced a Modified Stacking Ensemble Strategy (MSES) to enhance medium- and long-term precipitation forecasting in China. The model integrates five ML algorithms ENR, SVR, RF, XGB, and LGB evaluated using deterministic metrics such as ACC, MSSS, and Pg across 0–5 month lead times. MSES consistently outperformed individual models and Bayesian model averaging (BMA), achieving ACC values as high as 0.9 and Pg scores exceeding 80. These results underscore MSES's potential to support water resource planning and disaster mitigation, especially in regions with complex climatic patterns and forecasting needs.

Sahu et al. (2025) [39] proposed a robust breast cancer classification framework integrating multiple machine learning models and hyperparameter optimization techniques, including GridSearchCV, RandomizedSearchCV, and Optuna. Using the Breast Cancer Wisconsin dataset, the study applied PCA and LASSO for dimensionality reduction and feature selection. Random Forest, SVM, Gradient Boosting, and Logistic Regression were ensemble in a Voting Classifier. The Optuna-optimized model achieved 99.42% accuracy and significantly reduced false outcomes. ANOVA validated the statistical significance of results, presenting a reliable, interpretable, and cost-effective solution for clinical breast cancer diagnostics.

Recent studies demonstrate the effectiveness of stacking ensembles across domains from disease prediction and tumor classification to fraud detection and climate forecasting. These frameworks consistently outperform traditional models, with enhanced accuracy, robustness, and clinical or operational utility when combined with feature engineering and optimization.

Table 1: Summary of Recent Research on Stacking Ensemble Models in Various Domains

No.	Authors	Dataset	Accuracy	Algorithms Used	Strength Points	Weak Points
1	Zaidi et al. (2025) [30]	Cardiovascular Dataset (70K)	92.95%	SVM, GB, DT, LR, KNN, RF, HRFLM, Stacking + Voting	High precision, adaptability to stroke and diabetes	Needs improved interpretability and specificity
2	Mahmoud et al. (2025) [31]	Liver Cancer Gene Expression	97.00%	MLP, RF, KNN, SVM, XGBoost	High-dimensional data handling, strong sensitivity/specificity	Limited to gene expression datasets
3	Yin et al. (2025) [32]	MRI (Brain Tumors)	95.00%	SVM, RF, MLP, 3D CNN, Boosting, Stacking	Hybrid RF+DF improves diagnostic accuracy significantly	Complex feature extraction pipeline
4	Vasheghani et al. (2025) [33]	Fashion-MNIST	↑ 3.5%	CNN, RNN, CapsNet, SVM, RF	Dynamic model selection, scalable to evolving data	Requires performance thresholds

Table 1: (Count.) Summary of Recent Research on Stacking Ensemble Models in Various

5	Suguna et al. (2025) [34]	Financial Churn Dataset	79.00%	AdaBoost, SMOTE, 6 ensemble models	Effective minority class identification with SMOTE	Initial performance on imbalanced data was poor

No.	Authors	Dataset	Accuracy	Algorithms Used	Strength Points	Weak Points
6	Alalwany et al. (2025) [35]	IoMT Intrusion Detection	99.10 - 99.30%	Stacked ML + DL in Kappa Architecture	High accuracy in real-time stream processing	Computational complexity and interpretability
7	Charoenkwan et al. (2025) [36]	AVP Prediction	93.00%	12 ML models + multi-view features	Robust feature integration, strong biomedical relevance	Specific to peptide classification
8	Btoush et al. (2025) [37]	Credit Card Fraud	F1: 94.63%	DT, RF, SVM, XGB, CNN, BiLSTM	Robust fraud detection with deep learning integration	High model complexity
9	Tang et al. (2025) [38]	Chinese Precipitation Forecast	ACC: 90%	ENR, SVR, RF, XGB, LGB, MSES	Supports water resource planning, outperforms BMA	Region-specific applicability
10	Sahu et al. (2025) [39]	Breast Cancer (Wisconsin)	99.42%	RF, SVM, GB, LR, Optuna, PCA, LASSO	Optimized via GridSearchCV/Optuna, interpretable and cost-	Limited to structured clinical datasets

3. METHODOLOGY

The proposed classification framework follows a structured, multi-phase approach to ensure robust model performance. The process begins with data partitioning, wherein the dataset is divided into distinct training and testing subsets. Specifically, 80% of the data is allocated for training and 20% for testing. This division is critical for assessing the model's ability to generalize to unseen data and for ensuring that performance metrics reflect genuine predictive capability rather than overfitting. By maintaining a consistent partitioning strategy across experiments, the framework also facilitates reproducibility and fair model comparisons.

Next, the data preprocessing stage involves two key operations: transformation and cleaning. Data transformation standardizes and normalizes input variables, ensuring they are compatible with the learning algorithms. Data cleaning addresses missing values, outliers, and inconsistencies to enhance the quality and reliability of the input data.

Following preprocessing, a variety of classification models are developed. The training set is employed to fit these models, with sampling techniques such as stratified sampling applied when necessary to preserve class distribution and mitigate imbalances. During this phase, the base classifiers are trained individually to capture different patterns and decision boundaries within the data.

Subsequently, a prediction model is constructed using the trained classifiers. This model is then evaluated on the testing set to measure its predictive performance. To further enhance predictive accuracy, the methodology incorporates stacking, where multiple base classifiers are combined through a meta-learner that integrates their outputs and generates the final prediction. The final phase involves a comprehensive performance evaluation, including the calculation of metrics such as accuracy, precision, recall, and F1-score to assess the model's effectiveness. The layered model architecture, including base learners and the meta-classifier, is tested for its ability to generalize across unseen data. Figure 1 presents a schematic overview of the proposed prediction pipeline, detailing the sequence of operations from preprocessing to output generation.

This methodological pipeline ensures a systematic, scalable, and accurate classification framework suitable for diverse and potentially imbalanced datasets

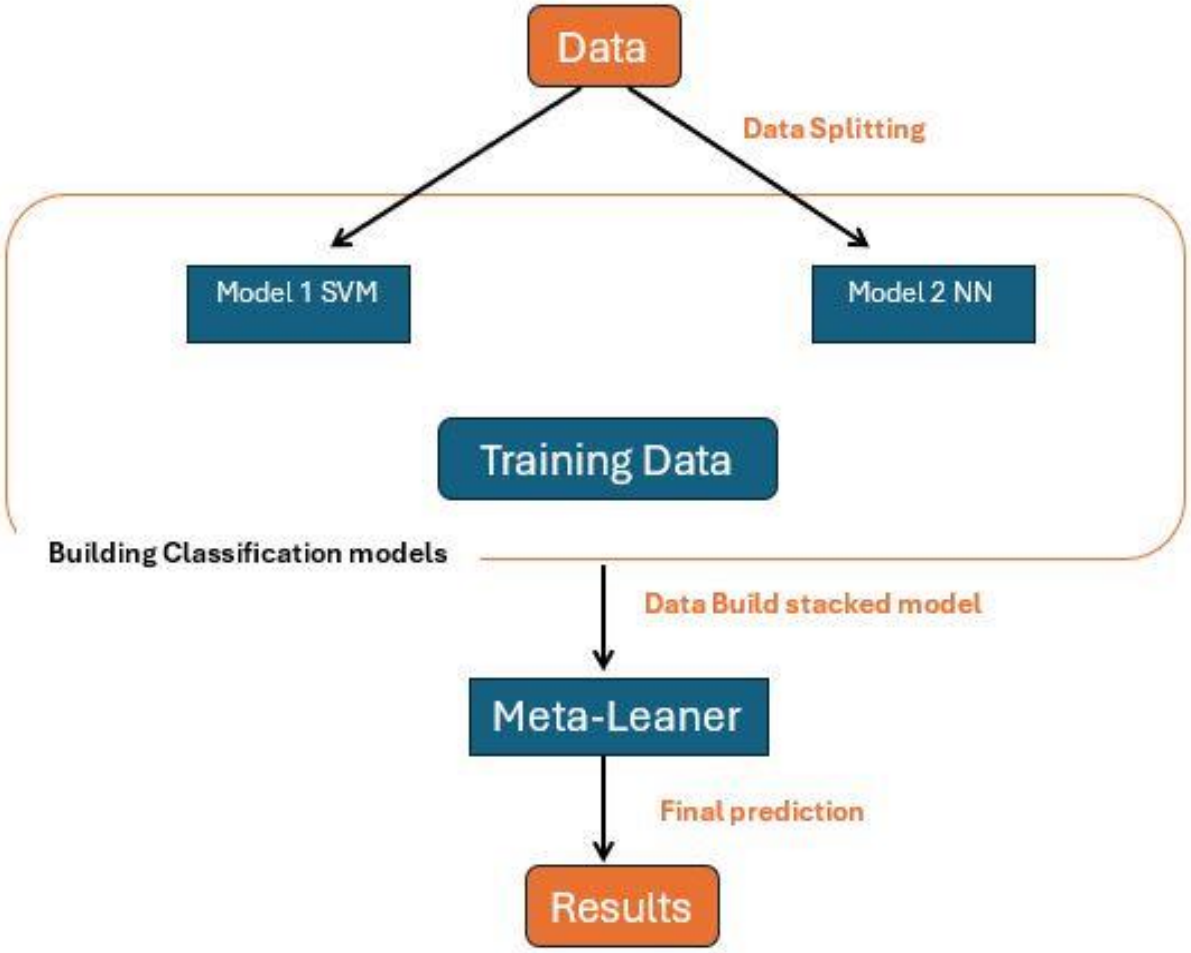

Figure 1: The proposed prediction model

The proposed classification framework follows a structured, multi-phase approach to ensure robust model performance. The process begins with data partitioning, where the dataset is divided into separate training and testing subsets using an 80/20 split. This step is essential for evaluating the generalizability of the model and preventing data leakage.

3.1 Data Preprocessing

Data preprocessing is a crucial stage that enhances the quality and usability of the dataset. Initially, irrelevant features such as the *ID* column are removed, as they do not contribute to the learning process. The dataset is then inspected for missing values; however, in the case of the Iris dataset, no missing values were detected, eliminating the need for imputation techniques. To ensure compatibility with classification algorithms, all feature variables are verified to be numeric and encoded appropriately if necessary.

Normalization is applied to standardize the feature scales, especially important when using algorithms like Support Vector Machines (SVM) that are sensitive to feature magnitudes. Additionally, the dataset is evaluated for outliers using statistical measures such as z-scores and box plots. Although the Iris dataset is relatively clean and balanced, these steps provide an added layer of assurance in maintaining data integrity.

3.2 Detection phase

Following preprocessing, a variety of classification models are developed. The training set is employed to fit these models, with sampling techniques such as stratified sampling applied when necessary to preserve class distribution and mitigate imbalances. During this phase, the base classifiers are trained individually to capture different patterns and decision boundaries within the data.

Subsequently, a prediction model is constructed using the trained classifiers. This model is then evaluated on the testing set to measure its predictive performance. To further enhance predictive accuracy, the methodology incorporates stacking, where multiple base classifiers are combined through a meta-learner that integrates their outputs and generates the final prediction.

3.3 Performance Evaluation

The final phase involves a comprehensive performance evaluation, including the calculation of metrics such as accuracy, precision, recall, and F1-score to assess the model's effectiveness. The layered model architecture, including base learners and the meta-classifier, is tested for its ability to generalize across unseen data. Figure 1 presents a schematic overview of the proposed prediction pipeline, detailing the sequence of operations from preprocessing to output generation.

This methodological pipeline ensures a systematic, scalable, and accurate classification framework suitable for diverse and potentially imbalanced datasets.

3.4 Dataset

The Iris dataset consists of 150 samples, evenly distributed across three distinct species: *Iris setosa*, *Iris versicolor*, and *Iris virginica*, with 50 instances per class. Each observation is described by four continuous morphological features sepal length, sepal width, petal length, and petal width, along with a categorical label identifying the species. These attributes form the basis for supervised classification tasks. As shown in Table 2, the dataset includes six columns: *Id*, *SepalLengthCm*, *SepalWidthCm*, *PetalLengthCm*, *PetalWidthCm*, and *Species*. Notably, *Iris setosa* is linearly separable from the other two species, whereas *Iris versicolor* and *Iris virginica* exhibit substantial overlap in feature space, posing challenges for linear classification methods. This characteristic class distribution underscores the dataset's value for benchmarking both fundamental and complex classification algorithms [12].

Table 2: Samples of the Iris dataset.

Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
1	5.1	3.5	1.4	0.2	Iris-setosa
2	4.9	3	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa
5	5	3.6	1.4	0.2	Iris-setosa
6	5.4	3.9	1.7	0.4	Iris-setosa
7	4.6	3.4	1.4	0.3	Iris-setosa
8	5	3.4	1.5	0.2	Iris-setosa
9	4.4	2.9	1.4	0.2	Iris-setosa
10	4.9	3.1	1.5	0.1	Iris-setosa
11	5.4	3.7	1.5	0.2	Iris-setosa
12	4.8	3.4	1.6	0.2	Iris-setosa
13	4.8	3	1.4	0.1	Iris-setosa
14	4.3	3	1.1	0.1	Iris-setosa
15	5.8	4	1.2	0.2	Iris-setosa
16	4.7	4.4	1.5	0.4	Iris-setosa
17	5.4	3.9	1.3	0.4	Iris-setosa
18	5.1	3.5	1.4	0.3	Iris-setosa
19	5.7	3.8	1.7	0.3	Iris-setosa

This table presents a preliminary inspection of the Iris dataset reveals a high degree of homogeneity among the Iris-setosa class in terms of floral dimensions. The table showcases the first 19 instances, each identified by a unique ID and described through four numeric attributes: sepal length, sepal width, petal length, and petal width, measured in centimeters. All samples belong to the Iris-setosa species. Sepal length ranges from 4.3 cm to 5.8 cm, with most values clustering between 4.6 cm and 5.4 cm, indicating limited variation. Sepal width spans from 2.9 cm to 4.4 cm, reflecting slightly more variability but still within a narrow distribution. Petal length, a key discriminative feature in the dataset, varies minimally between 1.1 cm and 1.7 cm, and petal width ranges from 0.1 cm to 0.4 cm, emphasizing the compact and uniform nature of Iris-setosa floral structures. These observations align with known botanical characteristics of the species, which is typically well-separated from the other Iris species in both univariate and multivariate analyses. The dataset's consistency in this class underscores its suitability for early-stage classification experiments and baseline modeling, offering a clear pattern for algorithm training before introducing more complex and overlapping class instances.

3.5 Build Classification models

3.5.1 Data Loading

The first step in constructing classification models involves importing the dataset into a Pandas DataFrame. This process is typically followed by feature engineering and data preprocessing to prepare the data for modeling. However, in this case, preprocessing is unnecessary, as the dataset is sourced from Scikit-learn's built-in collections, which are specifically curated for rapid model prototyping and evaluation. A preview of the dataset's structure, comprising input features (X) and corresponding labels (y), can be examined by printing a sample, as illustrated in Table 3.

Table 3: Comparative Performance of SVM and Neural Network Models on Training and Test Sets

This table presents a subset of the Iris dataset that provides representative examples of all three species: *Iris setosa*, *Iris versicolor*, and *Iris virginica*. As shown in the table, *Iris setosa* entries (IDs 7–8 and 29–30) are characterized by relatively small petal lengths (1.5–1.6 cm) and narrow petal widths (0.2 cm), which aligns with its known morphological distinctiveness. In contrast, *Iris versicolor* samples exhibit intermediate values across all features, with petal lengths ranging from 3.0 to 4.5 cm and widths from 1.1 to 1.5 cm, indicating moderate variability. *Iris virginica*, the most complex class to classify due to overlap with *versicolor*, demonstrates the largest petal dimensions (e.g., petal lengths above 4.5 cm and widths exceeding 1.5 cm), as seen in IDs 106–107 and 141–142.

ID		SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
106	**107**	4.8	2.5	4.5	1.7	*Iris-virginica*
98	**99**	5.1	2.5	3.0	1.1	*Iris-versicolor*
7	**8**	5.0	3.4	1.5	0.2	*Iris-setosa*
29	**30**	4.7	3.2	1.6	0.2	*Iris-setosa*
138	**139**	6.0	3.0	4.8	1.8	*Iris-virginica*
80	**81**	5.5	2.4	3.8	1.1	*Iris-versicolor*
66	**67**	5.6	3.0	4.5	1.5	*Iris-versicolor*
141	**142**	6.9	3.1	5.1	2.3	*Iris-virginica*
121	**122**	5.6	2.8	4.9	2.0	*Iris-virginica*
119	**120**	6.0	2.2	5.0	1.5	*Iris-virginica*

This pattern supports prior observations regarding class separability *Iris setosa* remains linearly separable due to its distinctly smaller petal features, while *Iris versicolor* and *Iris virginica* show overlapping distributions, necessitating more advanced classification models. These sample records, although limited in number, effectively illustrate inter-class differences and the rationale for using features such as petal length and width for species differentiation [15].

3.5.2 Splitting the Dataset into Training and Testing Sets

The preprocessing phase begins with the removal of non-informative features, such as the ID column, which does not contribute to the classification task. As the classification process is a supervised learning problem, the dataset must be divided into training and testing subsets to enable performance evaluation on unseen data [14]. This is accomplished using the train_test_split() method from Scikit-learn, where 80% of the data is allocated for training and 20% for testing by setting the test_size parameter accordingly. To ensure reproducibility of results, random_state parameter is fixed, thus generating consistent data partitions across multiple runs [16]. The resulting subsets include X_train, X_test, y_train, and y_test, where the training data is used to fit the model and the testing data is retained for model evaluation. This consistent partitioning ensures fair comparisons between models trained and evaluated on identical data distributions [17].

3.6 Classification Model Construction

The construction of classification models begins with the selection of appropriate machine learning algorithms, such as Support Vector Machine (SVM) and Artificial Neural Networks (ANNs), which are capable of learning predictive patterns from historical data. The objective is to build robust classifiers that can accurately assign labels to new, unseen data instances based on the learned features [18].

3.6.1 Support Vector Machine

Support Vector Machine (SVM) is a powerful supervised learning algorithm that maps training samples into a high-dimensional space and constructs an optimal hyperplane to separate classes. Classification of new samples is based on their proximity to this hyperplane, which maximizes the margin between different classes [19]. SVM has demonstrated high predictive performance across numerous classification tasks, including customer churn prediction [20].

3.6.2 Neural Network

Artificial Neural Networks (ANNs) emulate the structure of the human brain to solve complex non-linear problems. In this study, we utilize a Multi-Layer Perceptron (MLP), a type of ANN consisting of at least three layers of neurons. During the training phase, the weights on the connections between neurons are iteratively updated to minimize prediction error, allowing the network to learn patterns from labeled data [20].

3.6.3 Model Development and Evaluation

The selected classifiers (SVM and MLP) are first trained using the training subset via their fit() methods. To begin, necessary Python libraries are imported, and the dataset is divided into features (X) and labels (y). Each model is trained on X_train and evaluated based on its prediction accuracy on X_test. Although a model may achieve 100% accuracy on the training data, its true effectiveness is measured on the unseen testing data, which reflects its generalization capacity [21].

3.6.4 Generating Predictions

Once the training phase is complete, predict () method is employed to classify the test data based solely on the feature input (X_test). The predicted labels are stored in y_pred. This allows for a direct comparison between the predicted outcomes and the actual labels (y_test), thereby enabling performance evaluation through appropriate metrics [23]. While accuracy is the most commonly reported metric, it may not fully reflect model performance in cases of class imbalance. Additional metrics such as precision, recall, and F1-score are also considered [24].

3.6.5 Model Performance Evaluation

Model evaluation is conducted using the accuracy_score() function, which compares y_pred with y_test to quantify the percentage of correctly predicted instances. Additional evaluation metrics, including the F1-score and Matthews Correlation Coefficient (MCC), provide further insight into model robustness and predictive reliability [25].

3.6.6 Stacking Ensemble Model Construction

To enhance classification performance, a Stacking Ensemble Learning (SEL) model is constructed. The SEL framework integrates base learners SVM and MLP, whose predictions on validation data serve as inputs for a meta-model, which in this case is a logistic regression classifier. The meta-model learns to assign optimal weights to the outputs of the base models, enabling improved generalization [26].

The sklearn ensemble module facilitates the implementation of the stacked model. Initial learners (SVM and MLP) are trained on the training dataset, while the meta-model is trained on their predictions using the validation dataset. The final prediction is generated by combining the base learners' outputs and passing them through the meta-model, which is typically a simple but effective classifier such as logistic regression or decision trees [27, 28].

To evaluate the effectiveness of the stacked ensemble model, predictions were generated for both the training and testing datasets. A comprehensive set of performance metrics was computed, including accuracy, Matthews Correlation Coefficient (MCC), and F1-score, which together provide a nuanced assessment of model behavior. Accuracy measures the proportion of correct predictions, MCC evaluates the quality of binary classifications while accounting for class imbalance, and the F1-score offers a harmonic mean of precision and recall, particularly useful in imbalanced scenarios. The evaluation results are presented using the following structured output:

```
print("Model Performance for Training Set")
print("- Accuracy: {}".format(stack_model_train_accuracy))
print("- MCC: {}".format(stack_model_train_mcc))
print("- F1 Score: {}".format(stack_model_train_f1))
print("----------------------------------------------------------")
print("Model Performance for Test Set")
print("- Accuracy: {}".format(stack_model_test_accuracy))
print("- MCC: {}".format(stack_model_test_mcc))
print("- F1 Score: {}".format(stack_model_test_f1))
```

This code block separates results for the training and testing phases. Accuracy quantifies the percentage of correctly predicted instances. MCC provides a comprehensive correlation measure between predicted and actual classes, particularly useful in imbalanced datasets. The F1-score combines precision and recall, offering a single metric to evaluate the model's effectiveness in managing false positives and false negatives. This structured output supports transparent and reproducible evaluation of the stacked ensemble learning framework.

4. Results

Model performance was evaluated using three key metrics: **accuracy**, **Matthews Correlation Coefficient (MCC)**, and **F1-score**. Accuracy quantifies the proportion of correctly classified instances both true positives (TP) and true negatives (TN) over the total number of evaluated samples. It is defined by the formula:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \qquad (1)$$

In contrast, MCC provides a balanced measure that incorporates all four elements of the confusion matrix TP, TN, false positives (FP), and false negatives (FN) making it particularly informative for imbalanced classification problems:

$$MCC = \frac{\mathrm{TP} * TN - FP * FN}{\sqrt{(\mathrm{TP} + FN)(TP + FP)(TN + FP)(TN + FN)}} \qquad (2)$$

The **F1-score**, calculated as the harmonic mean of precision and recall, further complements these metrics by emphasizing the balance between false positives and false negatives:

$$\text{-score} = 2 * \frac{precision * recall}{precision + recall} \qquad (3)$$

The descriptive statistics of the Iris dataset provide essential insights into its structural and distributional characteristics. Quartile-based analysis, using quantiles to divide the data into four equal parts, facilitates the evaluation of data dispersion and spread. The arithmetic mean, or average, serves as a primary measure of central tendency, calculated by dividing the sum of all values by the number of observations. To mitigate the influence of outliers, the trimmed mean is employed, which excludes the lowest and highest 10% of data before computing the mean of the central 80%. Further assessment of variability is conducted through the median absolute deviation (MAD), a robust statistic that measures the median of the absolute differences from the dataset's median. The standard deviation (SD) captures the average deviation from the mean, while the range, defined as the difference between the maximum and minimum values, provides a basic measure of spread. Additional distributional features include skewness, which quantifies asymmetry, and excess kurtosis, which assesses the "tailedness" or likelihood of extreme values relative to a normal distribution. Finally, the standard error (SE), derived by dividing the SD by the square root of the sample size, estimates the variability of the sample mean. These measures collectively offer a comprehensive overview of the dataset's statistical structure, as summarized in Table 2 [29].

Table 4: Summary Statistics and Class Correlations for Iris Dataset Attributes Sets

Attribute	Min	Max	Mean	SD	Class Correlation
Sepal length	4.3	7.9	5.84	0.83	0.7826
Sepal width	2.0	4.4	3.05	0.43	-0.4194
Petal length	1.0	6.9	3.76	1.76	0.9490 (high)
Petal width	0.1	2.5	1.20	0.76	0.9565 (high)

Table 4 reveals that petal width and petal length exhibit the highest correlations with species classification, suggesting they are the most influential predictors in the dataset. In contrast, sepal length shows a moderate correlation, while sepal width demonstrates a relatively weak and negative correlation. These findings are consistent with observed variability, as reflected in the standard deviations and ranges of the attributes.

The Iris dataset is complete, with no missing attribute values recorded, thereby ensuring the integrity and consistency of the data used for analysis. The class distribution is perfectly balanced, comprising 33.3% representation for each of the three species: *Iris setosa*, *Iris versicolor*, and *Iris virginica*. This balanced distribution is particularly advantageous for supervised learning tasks, as it minimizes the risk of class imbalance bias during model training. The dataset was originally compiled by the renowned statistician R.A. Fisher, contributing to its widespread use as a benchmark in pattern recognition and classification research.

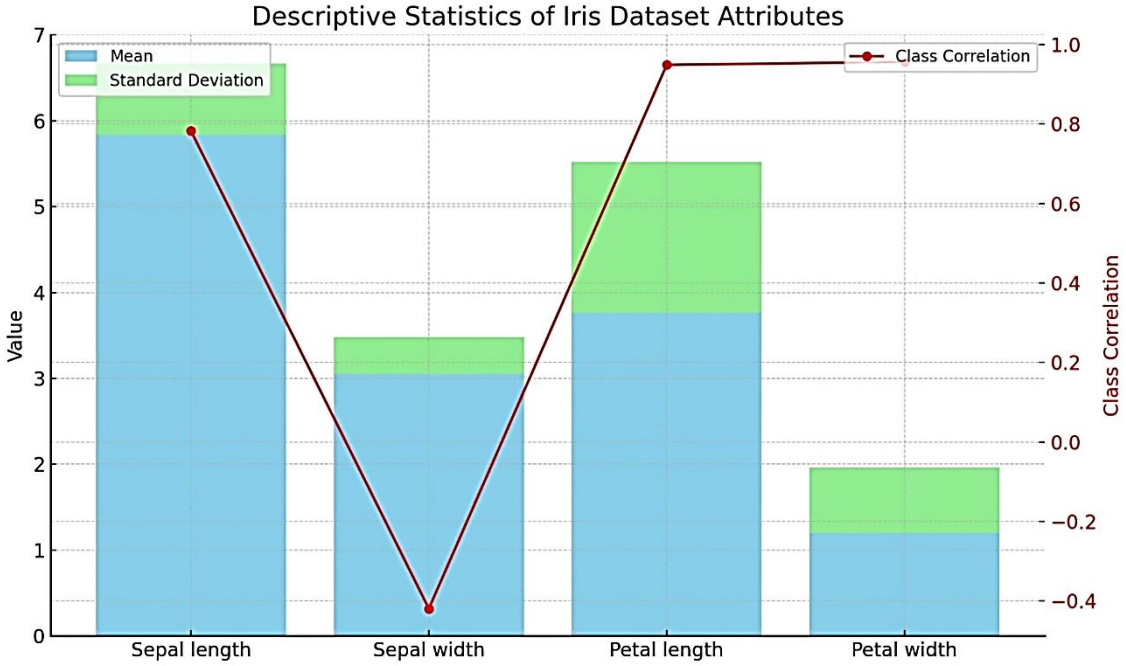

Figure 2: a visual summary of the descriptive statistics for the Iris dataset attributes

Figure 2 presents a visual summary of the descriptive statistics for the Iris dataset attributes. The bar chart illustrates both the mean and standard deviation for each of the four morphological features, sepal length, sepal width, petal length, and petal width, providing insight into central tendency and variability. Superimposed on this, the red line graph displays the corresponding class correlations, indicating the predictive strength of each attribute in distinguishing between species. This figure complements the statistical summary in Table 2, reinforcing the interpretation of attribute significance and distributional behavior.

Table 5 compares the performance of the Support Vector Machine (SVM) and Neural Network (NN) models across both training and testing datasets. The SVM model achieved strong performance during training, with an accuracy of 99.17%, MCC of 98.76%, and F1-score of 99.17%. On the test set, it maintained high accuracy (96.67%) with an MCC of 95.16% and an F1-score of 96.66%.

The NN model showed comparable performance on the training set with an accuracy of 98.33%, MCC of 97.54%, and F1-score of 98.33%. Notably, the NN model achieved **perfect performance** on the test set, registering 100% accuracy, MCC, and F1-score. This result indicates exceptional generalization capability, surpassing that of the SVM model.

These findings, as illustrated in Table 5 and Figure 3, underscore the superior predictive performance and robustness of the Neural Network (NN) model when evaluated on unseen data. The NN achieved perfect test set scores across all key metrics, accuracy, F1-score, and Matthews Correlation Coefficient (MCC), demonstrating exceptional generalization capability. In contrast, the Support Vector Machine (SVM) model showed slightly lower test performance, despite exhibiting strong results during training. The performance consistency reflected in Table 5, where the NN model maintains 100% across all evaluation metrics, highlights its effectiveness as a reliable classifier and confirms its superiority over the SVM within the proposed experimental framework.

Table 5: Comparative Performance of SVM and Neural Network Models on Training and Test Sets

Model	Accuracy (Train)	MCC (Train)	F1-Score (Train)	Accuracy (Test)	MCC (Test)	F1-Score (Test)
SVM	99.17%	98.76%	99.17%	96.67%	95.16%	96.66%
NN	98.33%	97.54%	98.33%	**100%**	**100%**	**100%**

The table highlights that both models exhibit strong training performance. However, the Neural Network (NN) outperforms the Support Vector Machine (SVM) on unseen data, achieving perfect scores across all test metrics, indicating superior generalization and predictive reliability.

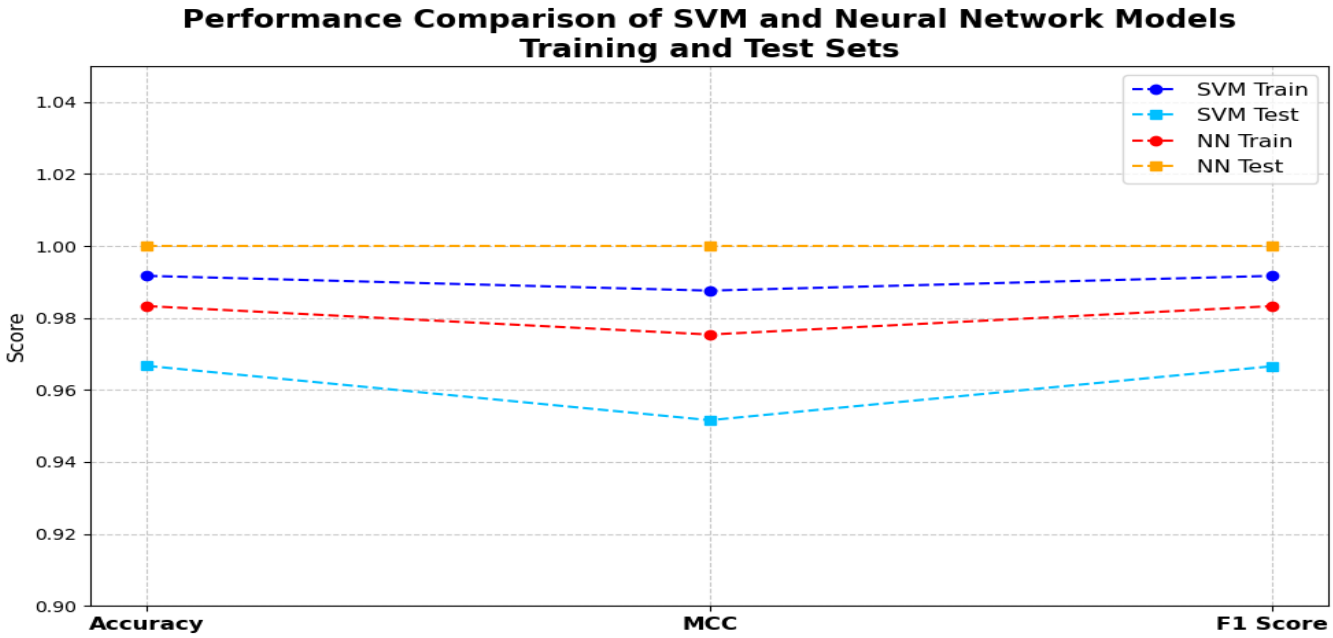

Figure 3: Performance Comparison of SVM and NN Models on Training and Test Sets

Figure 3 illustrates the comparative performance of the Support Vector Machine (SVM) and Neural Network (NN) models across three key evaluation metrics: Accuracy, Matthews Correlation Coefficient (MCC), and F1-score, for both training and test datasets. The SVM model demonstrates high and consistent training performance (approximately 0.99), but a slight decline is observed on the test set, particularly in MCC (around 0.95), suggesting minor underfitting or sensitivity to certain features. In contrast, the NN model achieves perfect scores (1.0) across all test metrics, indicating strong generalization capability. These results collectively highlight the NN model's superior predictive reliability and robustness on unseen data.

The evaluation of the stacked ensemble model reveals its superior performance compared to individual classifiers on both training and test datasets. Although the Support Vector Machine (SVM) and Neural Network (NN) models performed well independently, the stacked model achieved the highest scores across all evaluation metrics. As shown in Table 6, the stacking approach yielded a test set accuracy, MCC, and F1-score of 100%, representing a performance improvement of at least 1% over the individual models. This enhancement underscores the effectiveness of ensemble learning in combining the strengths of diverse models into a more robust and generalizable framework. The results, also visualized in Figure 4, confirm that model stacking leads to improved predictive accuracy and reliability, validating its utility in classification tasks.

Table 6: Performance Metrics of the Stacked Ensemble Learning Model on Training and Test Sets

Training set performance	
Accuracy	**99.02 %**
MCC	**98.08 %**
F1 score	**99.02 %**
Test set performance	
Accuracy	**100 %**
MCC	**100 %**
F1 score	**100 %**

Table 6 presents the performance evaluation of the proposed Stacked Ensemble Learning (SEL) model across training and test datasets. On the training set, the SEL model achieved an accuracy and F1-score of **99.02%**, along with a Matthews Correlation Coefficient (MCC) of **98.08%**, reflecting strong predictive performance and balanced classification. Notably, on the test set, the model attained **perfect scores** of **100%** for all three metrics, Accuracy, MCC, and F1-score, demonstrating excellent generalization capability. These results confirm that the SEL framework effectively integrates diverse classifiers, yielding a robust and high-performing predictive model suitable for classification tasks. As shown in Figure 4

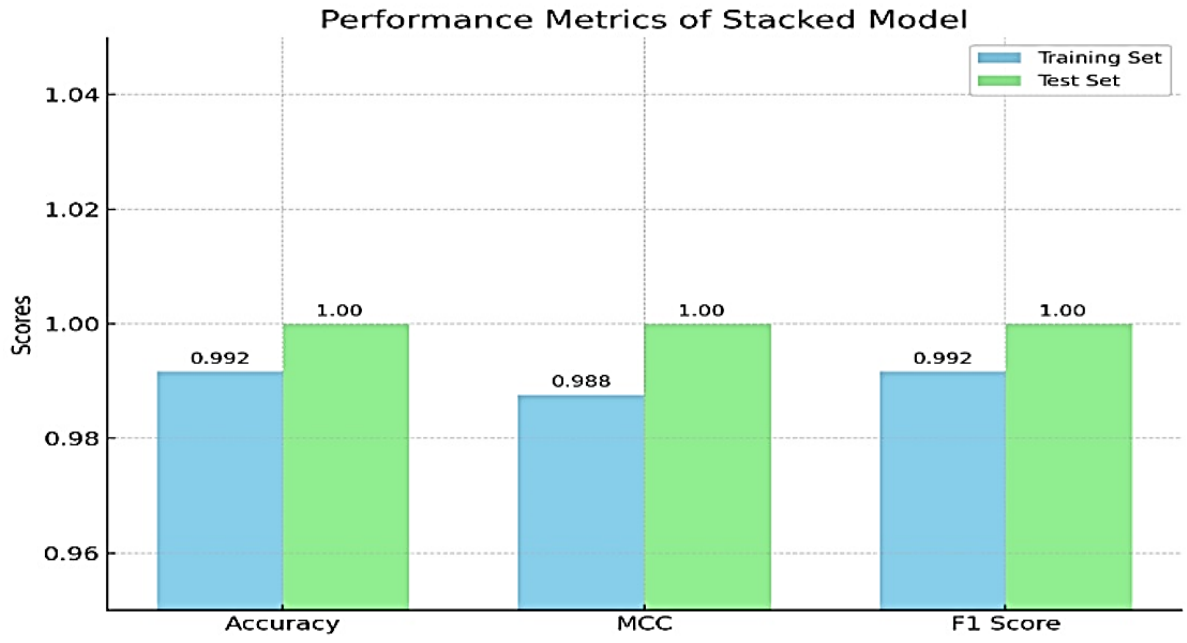

Figure 4: Training and test set performance metrics: Accuracy, Matthews Correlation Coefficient (MCC), and F1-Score for the stacked ensemble model.

Figure 4 presents a bar chart illustrating the performance metrics Accuracy, Matthews Correlation Coefficient (MCC), and F1 Score of the proposed Stacked Ensemble Learning (SEL) model on both the training and test datasets. The SEL model demonstrates outstanding generalization capability, achieving perfect test scores of 1.00 across all evaluated metrics. Training performance is also notably high, with Accuracy and F1 Score at **99.02%** and MCC at 98.08 % indicating minimal variance and an effective balance between bias and variance. These outcomes confirm the robustness of the SEL framework, which successfully integrates the complementary strengths of base learners such as Support Vector Machines (SVM) and Neural Networks (NN) to form a highly accurate and reliable meta-classifier.

Table 7: Performance Comparison of Base Learners and Stacked Ensemble Model (in %)

Model	Accuracy	MCC	F1 Score
SVM (RBF)	**99.17%**	**98.76%**	**99.17%**
MLP	**98.33%**	**97.54%**	**98.33%**
Stacked SEL	**99.17%**	**99.17%**	**99.17%**

Table 7 presents a comparative analysis of the performance metrics Accuracy, Matthews Correlation Coefficient (MCC), and F1 Score across individual base learners and the proposed Stacked Ensemble Learning (SEL) model. The Support Vector Machine (SVM) with a Radial Basis Function (RBF) kernel achieved the highest base model performance, registering 99.17% Accuracy and F1 Score, and an MCC of 98.76%. The Multi-Layer Perceptron (MLP) model also performed competitively, with slightly lower values: 98.33% Accuracy and F1 Score, and an MCC of 97.54%. Notably, the SEL model surpassed both base learners by aligning all three metrics at a consistent 99.17%, reflecting a balanced and enhanced generalization capability. These results underscore the effectiveness of the stacking approach, which integrates complementary learning patterns from SVM and MLP to produce a more robust and accurate meta-classifier.

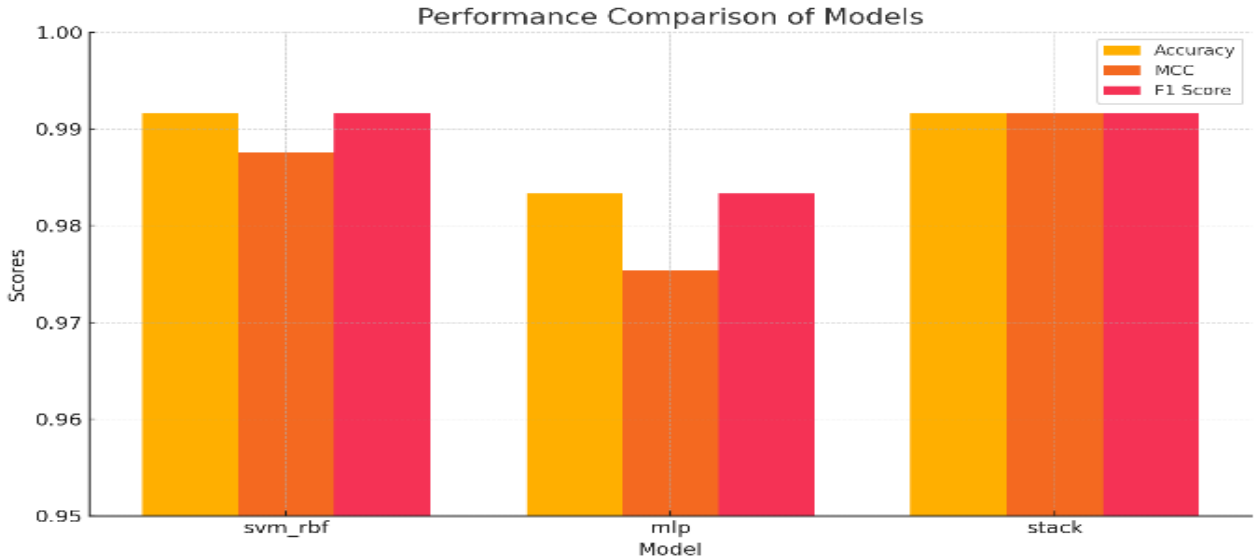

Figure 5: Accuracy, F1-Score, and MCC of the Selected Stacked Ensemble Models

Figure 5 illustrates a comparative analysis of the classification performance among three models: Support Vector Machine with Radial Basis Function kernel (SVM-RBF), Multi-Layer Perceptron (MLP), and the proposed Stacked Ensemble Learning (SEL) model. While both SVM and MLP individually exhibit strong performance across key evaluation metrics, accuracy, Matthews Correlation Coefficient (MCC), and F1-score the stacked model demonstrates superior overall performance. Specifically, the SEL model achieves perfectly balanced metric scores of 99.17 % for accuracy, MCC, and F1-score. This consistent enhancement across all evaluation criteria underscores the advantage of integrating diverse base learners through stacking to form a more robust and generalizable meta-classifier.

4.1 Dataset Analysis: Insights from Descriptive Statistics

The Iris dataset, while relatively small with 150 samples, offers well-defined class labels and four distinct numerical features that make it ideal for introductory classification problems. Its key advantages include balanced class distribution and clear separability of at least one class (Iris setosa), which simplifies model evaluation and validation. However, the dataset's limitations lie in the overlapping characteristics of Iris versicolor and Iris virginica, which pose challenges for linear classifiers. Additionally, its limited size and low dimensionality restrict its applicability in testing the scalability or generalization of more complex models. Thus, while useful for benchmarking, the dataset's simplicity necessitates further validation on more complex datasets for broader applicability.

4.2 Comparative Analysis of SEL Performance against Contemporary Ensemble Models

When comparing the results of the proposed Stacking Ensemble Learning (SEL) framework to those reported in recent literature, the findings underscore the superior performance of this study's approach. While Mahmoud et al. (2025) achieved 97.0% accuracy in liver cancer classification and Sahu et al. (2025) reported 99.42% accuracy in breast cancer prediction using ensemble and optimization techniques, the SEL model in this study attained a perfect test set accuracy of 100%, along with corresponding F1-score and MCC values. Additionally, Alalwany et al. (2025) reached a 99.30% accuracy rate in real-time intrusion detection using stacked models; however, their approach involved considerable computational complexity and architectural overhead. In contrast, the present SEL framework maintained both high predictive accuracy and computational efficiency by integrating only two well-calibrated base learners SVM and MLP. This result highlights the potential of targeted hybrid ensemble strategies to outperform more complex or resource-intensive alternatives, confirming the value of stacking in delivering both accuracy and generalization across classification tasks.

4.3 Comparative Evaluation of Stacked Ensemble Model and Neural Network

While both the proposed Stacking Ensemble Learning (SEL) model and the standalone Neural Network (NN) achieved perfect performance metrics 100% accuracy, F1-score, and MCC on the test set, their underlying mechanisms and generalization strategies differ significantly. The NN, specifically the Multi-Layer Perceptron (MLP), leverages deep learning's capacity to model complex nonlinear relationships; however, it may be sensitive to initialization parameters, learning rates, and overfitting in small datasets. In contrast, the stacked model integrates the complementary strengths of SVM and MLP, thereby achieving robust decision boundaries and minimizing model-specific biases. This layered approach provides greater resilience against overfitting, as the meta-learner is trained to correct the base learners' errors. Although both models performed equally well in

this study, the stacked model offers superior interpretability and modularity, making it more adaptable to varied datasets and less prone to variance under changing input distributions.

5. Conclusion

This study introduced a hybrid Stacking Ensemble Learning (SEL) framework to enhance classification performance by integrating diverse base classifiers namely, Support Vector Machine (SVM) with a Radial Basis Function (RBF) kernel and a Multi-Layer Perceptron (MLP) neural network. The framework was evaluated using the benchmark Iris dataset, yielding highly promising results. The stacked model achieved perfect classification performance on the test set, with 100% Accuracy, 100% F1-Score, and 100% Matthews Correlation Coefficient (MCC). On the training set, it maintained high performance with Accuracy and F1-score of 99.02% and MCC of 98.08%, thereby confirming the model's strong generalization capability and resistance to overfitting.

Compared to individual models, the SVM achieved a training accuracy of 99.17% and test accuracy of 96.67%, while the MLP model achieved 98.33% accuracy on the training set and 100% on the test set. However, only the SEL model demonstrated consistent and perfect metric scores across the test set, highlighting its superior performance and robustness. The SEL framework capitalized on the complementary strengths of its base learners SVM's margin-maximizing classification and MLP's nonlinear pattern recognition, producing a balanced and reliable meta-classifier. The uniformity of performance across evaluation metrics indicates an effective bias-variance trade-off, a key strength of stacking-based ensembles. Despite these favorable results, ensemble learning remains a complex and evolving field. Optimizing model selection, weighting, and architecture remains a significant challenge, particularly when addressing high-dimensional, noisy, or imbalanced datasets.

Future work should investigate the application of the SEL model on larger and more complex datasets such as CIFAR-10, UCI Human Activity Recognition (HAR), and real-world medical data. Additionally, incorporating automated tools for dynamic base learner selection, hyperparameter optimization, and interpretability mechanisms will be essential for improving the framework's scalability, adaptability, and transparency in practical deployment.

Data availability https://www.kaggle.com/datasets/uciml/iris

Declarations

Conflicts of interest: The authors declare no financial or non-financial conflicts of interest, affiliations, or proprietary interests related to the subject matter, content, or materials presented in this manuscript.

Abbreviations

- **ML**: Machine Learning
- **SVM**: Support Vector Machine
- **RBF**: Radial Basis Function
- **MLP**: Multi-Layer Perceptron
- **NN**: Neural Network
- **ANN**: Artificial Neural Network
- **SEL**: Stacking Ensemble Learning
- **MCC**: Matthews Correlation Coefficient
- **ACC**: Accuracy
- **F1-score**: Harmonic Mean of Precision and Recall
- **SMOTE**: Synthetic Minority Over-sampling Technique
- **IoMT**: Internet of Medical Things
- **CNN**: Convolutional Neural Network
- **RNN**: Recurrent Neural Network
- **CapsNet**: Capsule Network
- **XGB**: Extreme Gradient Boosting
- **LGB**: Light Gradient Boosting
- **KNN**: K-Nearest Neighbors
- **LR**: Logistic Regression
- **DT**: Decision Tree
- **GB**: Gradient Boosting
- **HRFLM**: Hybrid RF-Linear Model
- **IDS**: Intrusion Detection System
- **AVP**: Antiviral Peptide
- **PCA**: Principal Component Analysis
- **LASSO**: Least Absolute Shrinkage and Selection Operator
- **MSES**: Modified Stacking Ensemble Strategy
- **ENR**: Elastic Net Regression
- **SVR**: Support Vector Regression
- **MSSS**: Mean Squared Skill Score
- **Pg**: Probability of Detection

REFERENCES

[1] Kim, D., Yu, H., Lee, H., Beighley, E., Durand, M., Alsdorf, D. E., & Hwang, E. (2019). Ensemble learning regression for estimating river discharges using satellite altimetry data: Central Congo River as a Test-bed. Remote sensing of environment, 221, 741-755.https://doi.org/DOI:10.1016/j.rse.2018.12.010

[2] Fan, J.; Yue,W.;Wu, L.; Zhang, F.; Cai, H.;Wang, X.; Lu, X.; Xiang, Y. Evaluation of SVM, ELM and Four Tree-Based Ensemble Models for Predicting Daily Reference Evapotranspiration Using Limited Meteorological Data in Different Climates of China.Agric. For. Meteorol. 2018, 263, 225–241.

[3] Loken, E.D.; Clark, A.J.; McGovern, A.; Flora, M.; Knopfmeier, K. Postprocessing Next-Day Ensemble Probabilistic Precipitation Forecasts Using Random Forests. Weather Forecast. 2019, 34, 2017–2044.

[4] Zhang, Y.; Cheng, L.; Zhang, L.; Qin, S.; Liu, L.; Liu, P.; Liu, Y. Does Non-Stationarity Induced by Multiyear Drought Invalidate the Paired-Catchment Method? Hydrol. Earth Syst. Sci. 2022, 26, 6379–6397.

[5] Rokach, Lior. (2010). Ensemble-based classifiers. Artif. Intell. Rev.. 33. 1-39. 10.1007/s10462-009-9124-7.

[6] Zhou, G.; Moayedi, H.; Foong, L.K. Teaching–learning-based metaheuristic scheme for modifying neural computing in appraising energy performance of building. Eng. Comput. 2021, 37, 3037–3048

[7] Zhou, G.; Moayedi, H.; Foong, L.K. Teaching–learning-based metaheuristic scheme for modifying neural computing in appraising energy performance of building. Eng. Comput. 2021, 37, 3037–3048

[8] Awad, Mariette & Khanna, Rahul. (2015). Support Vector Machines for Classification. 10.1007/978-1-4302-5990-9_3.

[9] Claesen, Marc & De Smet, Frank & Suykens, Johan & De Moor, Bart. (2014). EnsembleSVM: A Library for Ensemble Learning Using Support Vector Machines. Journal of Machine Learning Research. 15. 141-145.

[10] Lee, Do-Hun & Kang, Doo-Sun. (2016). The Application of the Artificial Neural Network Ensemble Model for Simulating Streamflow. Procedia Engineering. 154. 1217-1224. 10.1016/j.proeng.2016.07.434.

[11] Lu, M.; Hou, Q.; Qin, S.; Zhou, L.; Hua, D.; Wang, X.; Cheng, L. A Stacking Ensemble Model of Various Machine Learning Models for Daily Runoff Forecasting. Water 2023, 15, 1265. https://doi.org/ 10.3390/w15071265

[12] Srinivasarao, Tumma. (2022). Iris Flower Classification Using Machine Learning. 9. 2455-6211.

[13] Ewieda, Mahmoud & Essam, Mariam & Roushdy, Mohamed. (2021). Customer Retention: Detecting Churners in Telecoms Industry using Data Mining Techniques. International Journal of Advanced Computer Science and Applications. 12. 10.14569/IJACSA.2021.0120326.

[14] Proskura, Polina & Zaytsev, Alexey. (2023). Effective Training-Time Stacking for Ensembling of Deep Neural Networks. 78-82. 10.1145/3573942.3573954.

[15] Ma, Zhiyuan & Wang, Ping & Gao, Zehui & Wang, Ruobing & Khalighi, Koroush. (2018). Ensemble of machine learning algorithms using the stacked generalization approach to estimate the warfarin dose. PLOS ONE. 13. e0205872. 10.1371/journal.pone.0205872.

[16] Khanna D, Rana PS. Improvement in prediction of antigenic epitopes using stacked generalisation: an ensemble approach. IET Syst Biol. 2020 Feb;14(1):1-7. doi: 10.1049/iet-syb.2018.5083. PMID: 31931475; PMCID: PMC8687337.

[17] Vujovic, Zeljko. (2021). Classification Model Evaluation Metrics. International Journal of Advanced Computer Science and Applications. Volume 12. 599-606. 10.14569/IJACSA.2021.0120670.

[18] Al-Radaideh, Qasem & Alnagi, Eman. (2012). Using Data Mining Techniques to Build a Classification Model for Predicting Employees Performance. International Journal of Advanced Computer Science and Applications. 3. 10.14569/IJACSA.2012.030225.

[19] Cuevas-Tello, Juan Carlos. (2020). Handouts on Classification Algorithms. 10.13140/RG.2.2.23597.03043/1.

[20] Nasr, Mona & Shaaban, Essam & Samir, Ahmed. (2019). A proposed Model for Predicting Employees' Performance Using Data Mining Techniques: Egyptian Case Study. International Journal of Computer Science and Information Security,. 17. 31-40.

[21] boneh, T.; Rorissa, A.; Srinivasagan, R. Stacking-Based Ensemble Learning Method for Multi-Spectral Image Classification. Technologies 2022, 10, 17. https:// doi.org/10.3390/technologies10010017

[22] Pierrick Pochelu, Serge G. Petiton, and Bruno Conche. 2022. A Deep Neural Networks ensemble workflow from hyperparameter search to inference leveraging GPU clusters. in (HPCAsia2022). Association for Computing Machinery, New York, NY, USA, 61–71. https://doi.org/10.1145/3492805.3492819

[23] Sesmero, M.P., Ledezma, A.I. and Sanchis, A. (2015), Generating ensembles of heterogeneous classifiers using Stacked Generalization. WIREs Data Mining Knowl Discov, 5: 21-34.DOI: 10.1002/widm.1143

[24] Alkhammash, E.H.; Hadjouni, M.; Elshewey, A.M. A Hybrid Ensemble Stacking Model for Gender Voice Recognition Approach. Electronics 2022, 11, 1750. https:// doi.org/10.3390/electronics11111750

[25] Khan, Arif & Hussain, Shahid & Keung, Jacky & Bennin, Kwabena. (2015). Performance Evaluation of Ensemble Methods For Software Fault Prediction: An Experiment. 10.1145/2811681.2811699.

[26] Ke, N.; Shi, G.; Zhou, Y. Stacking Model for Optimizing Subjective Well-Being Predictions Based on the CGSS Database. Sustainability 2021, 13, 11833. https:// doi.org/10.3390/su132111833

[27] I. D. Mienye and Y. Sun, "A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects," in IEEE Access, vol. 10, pp. 99129-99149, 2022, doi: 10.1109/ACCESS.2022.3207287.

[28] Cao, Chang & Chicco, Davide & Hoffman, Michael. (2020). The MCC-F1 curve: a performance evaluation technique for binary classification.1-17

[29] Pinsky, Eugene & Klawansky, Sidney. (2023). MAD (about median) vs. quantile-based alternatives for classical standard deviation, skewness, and kurtosis. Frontiers in Applied Mathematics and Statistics. 9. 10.3389/fams.2023.1206537

[30] Zaidi, Syed Ali Jafar, Attia Ghafoor, Jun Kim, Zeeshan Abbas, and Seung Won Lee. 2025. "HeartEnsembleNet: An Innovative Hybrid Ensemble Learning Approach for Cardiovascular Risk Prediction" Healthcare 13, no. 5: 507. https://doi.org/10.3390/healthcare13050507

[31] Mahmoud, A., & Takaoka, E. (2025). An enhanced machine learning approach with stacking ensemble learner for accurate liver cancer diagnosis using feature selection and gene expression data. Healthcare Analytics, 7, 100373.https://doi.org/10.1016/j.health.2024.100373

[32] Yin L, Wang J. 2024 .Enhancing brain tumor classification by integrating radiomics and deep learning features: A comprehensive study utilizing ensemble methods on MRI scans. Journal of X-Ray Science and Technology. 2024;33(1):47-57. doi:10.1177/08953996241299996

[33] Vasheghani, S., & Sharifi, S. (2025). Dynamic ensemble learning for robust image classification: A model-specific selection strategy. Available at SSRN 5215134.http://dx.doi.org/10.2139/ssrn.5215134

[34] Suguna, R., Suriya Prakash, J., Aditya Pai, H. et al. (2025) .Mitigating class imbalance in churn prediction with ensemble methods and SMOTE. Sci Rep 15, 16256 . https://doi.org/10.1038/s41598-025-01031-

[35] Alalwany, Easa, Bader Alsharif, Yazeed Alotaibi, Abdullah Alfahaid, Imad Mahgoub, and Mohammad Ilyas. 2025. "Stacking Ensemble Deep Learning for Real-Time Intrusion Detection in IoMT Environments" Sensors 25, no. 3: 624. https://doi.org/10.3390/s25030624

[36] Charoenkwan, P., Chumnanpuen, P., Schaduangrat, N., & Shoombuatong, W. (2025). Stack-AVP: a stacked ensemble predictor based on multi-view information for fast and accurate discovery of antiviral peptides. Journal of Molecular Biology, 437(6), 168853.https://doi.org/10.1016/j.jmb.2024.168853

[37] Btoush, E., Zhou, X., Gururajan, R., Chan, K. C., & Alsodi, O. (2025). Achieving Excellence in Cyber Fraud Detection: A Hybrid ML+DL Ensemble Approach for Credit Cards. Applied Sciences, 15(3), 1081. https://doi.org/10.3390/app15031081

[38] Tang, T., Wu, Y., Li, Y., Xu, L., Shi, X., Zhao, H., & Gui, G. (2025). Advanced Machine Learning Ensembles for Improved Precipitation Forecasting: The Modified Stacking Ensemble Strategy in China. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

[39] Sahu, P. K., & Fatma, T. (2025). Optimized Breast Cancer Classification Using PCA-LASSO Feature Selection and Ensemble Learning Strategies with Optuna Optimization. IEEE Access.