An Intelligent Model to combat Soybean Plant Disease based on Random Forest and Support Vector Machine Algorithms

Zainab A. Abdulazeez [1,*], Israa Abdulkadhim Jabbar Al Ali[1], Basma Mustafa M. H.[2], Ghada Kamil Mustafa[1], Refed Adnan Jaleel[3,4]

[1]College Of Education For Human Sciences, University of Kerbala, Kerbala, Iraq

[2]College of Science, University of Kerbala, Kerbala, Iraq

[3]Department of Information And Communication Engineering Al-Nahrain Universit, Baghdad, Iraq

[4]Commission of Rightness for Persons with Disabilities and Special Needs, Ministry of Labor and Social Affairs, Baghdad, Iraq

Emails: zainab.abdulhameed@uokerbala.edu.iq; israa.jabbar@uokerbala.edu.iq; basma.m@uokerbala.edu.iq; ghadakamil.87@gmail.com; iraq_it_2010@yahoo.com

Abstract

Given that plant disease is the primary factor contributing to damage in most plants, decision makers in the agriculture industry are highly interested in enhancing prediction strategies to detect illness in plants at an early stage. This is crucial for ensuring timely and effective plant care. Classifying healthy soybean plants is a dependable and efficient use of noninvasive techniques like machine learning (ML). In this work, we used ML to enhance a smart forecasting model for the prediction of soybean diseases. We utilized two feature selection techniques, namely gain ratio and correlation, two supervised ML algorithms (support vector machine and Random forest) and the cross-validation technique was used for assessing the proposed system, such as accuracy, F-measure, specificity, executing time, and sensitivity. The suggested technique can readily differentiate between soybean plants that are infected and those that are healthy. The suggested approach has undergone testing using a comprehensive collection of soybean characteristics, as well as a subset of attributes. The findings show that performance metrics are impacted when soybean traits are reduced.

Keywords: Support Vector Machine; Random Forest; Plant Disease; Soybean; Feature Selection

1. Introduction

The detection of plant diseases is a crucial component of precision agriculture as it specifically aims to identify diseases during their initial stages [1]. The kind of production that the world needs to create is in agriculture. Although multiple initiatives have been undertaken to enhance various sectors, the agricultural farming industry has made minimal advancements. Thus, the purpose of the plant detection application is to improve the agricultural industry and raise safety standards in order to facilitate the development of plant growth [2].

The use of plant monitoring systems is crucial for agricultural institutions due to the proliferation of plant diseases [3]. Furthermore, the best way to improve prediction is through early identification of plant diseases, which is where machine learning (ML) techniques could be useful in the current study. Combining clustering and classification allows for the development of extremely intelligent frameworks that account for the efficiency and effectiveness of all ML-affiliated institutions in their operations [4] [5] and identify knowledge patterns connected to the characteristics of diseases for plant. If knowledge about the plants is gathered, ML can help with important issues related to plant diseases. It is used to generate diagnostic guidelines and provide decision-makers with an accurate forecasting technique [6] [7].

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

With the help of classification algorithms, putting plants into groups with similar traits helps us learn more about how well different plants are cared for. In addition, by looking at historical data with classifiers, we can make a prediction model for a plant disease [8].

Reference [9] suggested using k-mean to divide input pictures into segments. They then used a classifier of SVM to divide the images into 2 groups, and they improved the evaluating of the SVM classifier by replacing it with classification of k-nearest neighbors (k-NN). In order to identify the affected areas in a leaf image. Reference [10] employed k-means segmentation, whereas SVM was utilized for classification. Reference [11] employed the k-means algorithm and multi-class support vector machine (SVM) to identify leaf diseases based on a dataset of photographs. The achieved accuracy was 98%. Convolutional neural networks (CNNs) were used by reference [12] to study Parkinson's disease and create a dependable platform. In order to identify plant diseases, reference [13] looked at how ML methods have progressed to deep learning. Four steps were utilized by reference [14] to identify the type of plant disease: pre-processing, k-means (leaf segmentation), feature extraction, and k-NN classification. In order to identify leaf diseases, reference [15] employed k-means, principal component analysis, and support vector machines (SVM). They explain how k-means' accuracy (ACC) predicts best from SVM and best from SVM from PCA. Using data that included images of both healthy and sick leaves, reference [16] employed SVM and k-means to forecast sickness. Reference [17] proposed a method for plant disease classification using k-NN, feature selection using random forest (RF), and segmentation using k-means. The experimental findings showed that the suggested technique performed better than SVM.

2. Methods and Materials

A. Methodology

In this section, the paper's methodology was covered. Initially, we imported the data into the WEKA tool and subsequently employed pre-processing and feature selection methodologies such as Relief and mRMR to ascertain pertinent attributes. Classifiers were then used to evaluate how well various machine learning techniques, including SVM and RF, performed on soybean data with complete features, Figure 1. A few characteristics were evaluated to determine their effectiveness. It was also used k folds with cross-validation. Evaluating measurement tools were used to see how well the classifiers worked. All characteristics were standardized and normalized before classifiers were applied. The computations were performed using WEKA. Data on the soybean outbreak can be available in a GitHub repository maintained by John Hopkins University. Compiling and disseminating publicly accessible data from several sources is the responsibility of John Hopkins University's Center for Systems Science and Engineering (JHU CSSE). The data is exclusively accessible to the public for educational and scholarly research purposes. This study will look at the daily data from Louisiana State as well as the variables related to confirmed cases and recoveries. The problem of uneven data is a big problem for systems that use machine learning. This is because different groups are given different treatment when they are categorized. The distribution of data in the result categories is significantly imbalanced in the selected data, with a larger number of samples belonging to the healthy class and a smaller number of samples linked with the infected class. As a result, taught models typically favour the dominant class, and compared to trained models, machine learning models are noticeably better at classifying incoming observations into the majority class. In order to rectify the class imbalance in the dataset of this study, the imbalanced-learn tools implemented the synthetic minority over-sampling approach method.

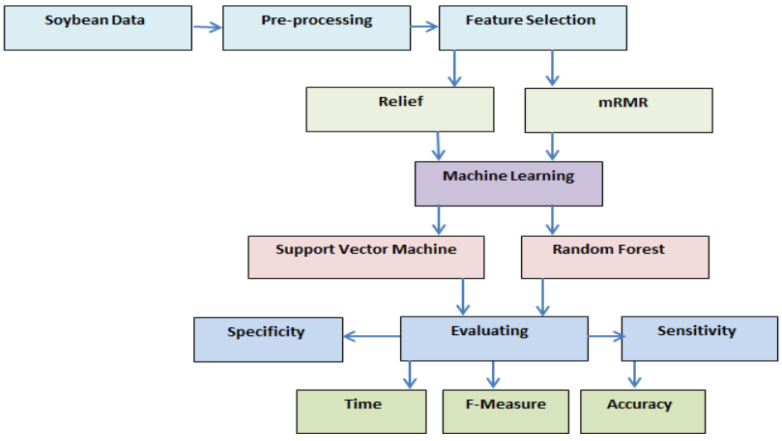

Figure 1: Block Diagram for Proposed System

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

B. Random Forest

Supervised learning methods like random forest are widely used in ML, Figure 2. People often use algorithms like this one because they are easy to understand, work well, and are flexible. Since it is nonlinear, it can adapt to different types of data and situations, and it is effective for both classification and regression issues. Tin Kam Ho first used the name "random choice forest" in 1995.

Ho developed a method for generating forecasts from seemingly unpredictable data. Because of the work done in 2006 by Leo Breiman and Adele Cutler, we are now familiar with random forests, Figure 3. This indicates that both the technology and the underlying science and mathematics are in their early stages [18].

The essential equation hired in Random Forest, a renowned ensemble getting-to-know the technique, is associated with the prediction carried out through aggregating the responses from several selection bushes. Within a Random Forest, the prediction for a singular facts aspect is normally decided by means of both the bulk vote (in class tasks) or the average prediction (in regression tasks) from all of the available trees within the wooded area.

For classification, final prediction y is the class that receives the most votes among all the trees:

$y = mode\ \{h1x, h2x, \ldots, hnx\}$…… (1)

Where:

- $h_i(x)$ is the prediction of the ith decision tree for the input x,
- n is the total number of trees in the forest.

For regression, the final prediction y is the average of all the predictions from the trees:

$y = \frac{1}{n}\sum_{i=1}^{n} h_i(x)$ ……..(2)

Where:

- $h_i(x)$ is the prediction of the i^{th} decision tree for the input x,
- n is the total number of trees in the forest.

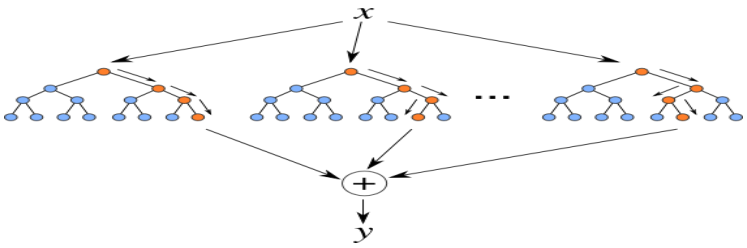

Figure 2: RF

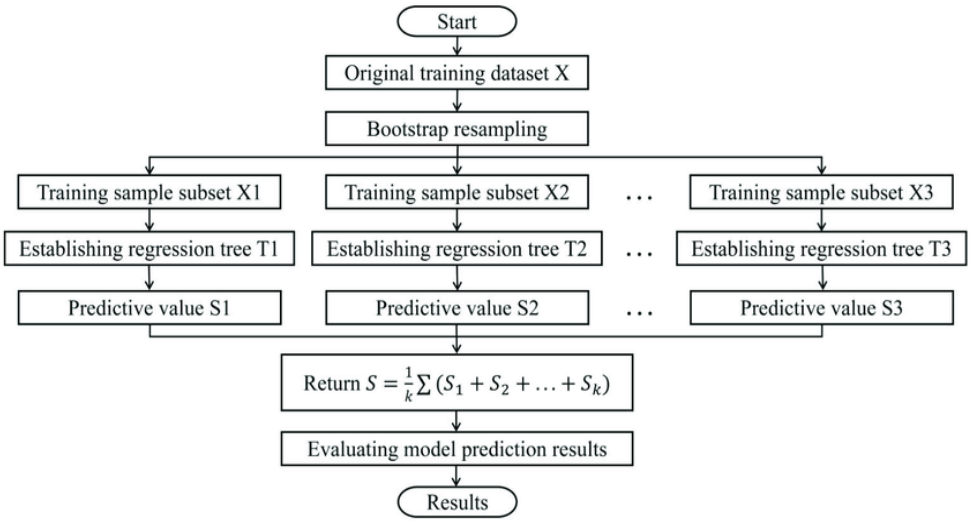

Figure 3: Flowchart of RF

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

C. Suport Vector Machine

Corinna Cortes and Vladimir Vapnik invented SVM algorithm in 1995. This well-known supervised learning method is depend on statistical theory and structural risk reduction. The idea behind the SVM is straightforward: by increasing the distance between the closest instances and the hyperactive plane of each label, it finds a hyperactive plane in space that minimizes the gap between two labels. Figure 4 shows three hyper planes for each label in the two-class problem of predicting healthy and infected attitudes. The blue hyperplane distinguishes healthy sentiments from infected sentiments [19] [20].

Support Vector Machines (SVMs) are the maximum correct classifiers. To simplify the assumed situation, we will count on that the enter facts may be separated linearly. Given this assumption, we are able to distinguish between any training via determining the linear hyperplane defined among them. Figure 3 illustrates the effectiveness of the difference between two instructions.

Let us recollect a binary class task with xi as the source of data, in which i tiers from 1 to n, and identical labels are assigned to each magnificence. We shall now assume that the purpose of decision-making is as tested inside the query as follows:

$G = \{(x_i,\ d_i)\}_i^n$ ……. (3)

Spectral Vector Machine (SVM) is a unique studying algorithm that operates based on minimizing structural risk and has demonstrated notable overall performance in several programs. Nevertheless, Support Vector Machines (SVM) had been to start with developed for binary category obligations, and the growth of SVM to multi-type is currently a topic of persevering with a look at. The method for fixing the multiclass SVM trouble has several cautioned methods. Constructing k (okay-1)/2 classifiers is primarily based on schooling information completely from two distinct lessons. The i[th] Support Vector Machine (SVM) solves the category trouble with education facts(x_1,y_1),...,(x_l,y_l) where x∈R^n,i=1,...,l and y∈1,..., okay is the elegance of xi.

$$\min_{w^{ij},b^{ij},\xi^{ij}} \frac{(w^{ij})'w^{ij}}{2} + C\sum_t \xi_t^{ij}(w^{ij})' \qquad \text{……. (4)}$$

$$\text{Subject to } (w^{ij})'\varphi(x_t) + b^{ij} \geq 1 - \xi_t^{ij}, if \quad y_t = i$$
$$(w^{ij})'\varphi(x_t) + b^{ij} \leq \xi_t^{ij} - 1, if \quad y_t = j, \xi_t^{ij} > 0$$

where the training data x are mapped $\varphi(x)$ to a higher dimensional space, C is the penalty parameter, ξ is a slack value. SVM classifies a new data point if x in the i[th] class voted by the decision function ($sign((w^{ij})'\varphi(x) + b^{ij})$). Otherwise, the j[th] class is voted. The (one versus one) strategy is commonly used to determine the class of a test pattern x [4].

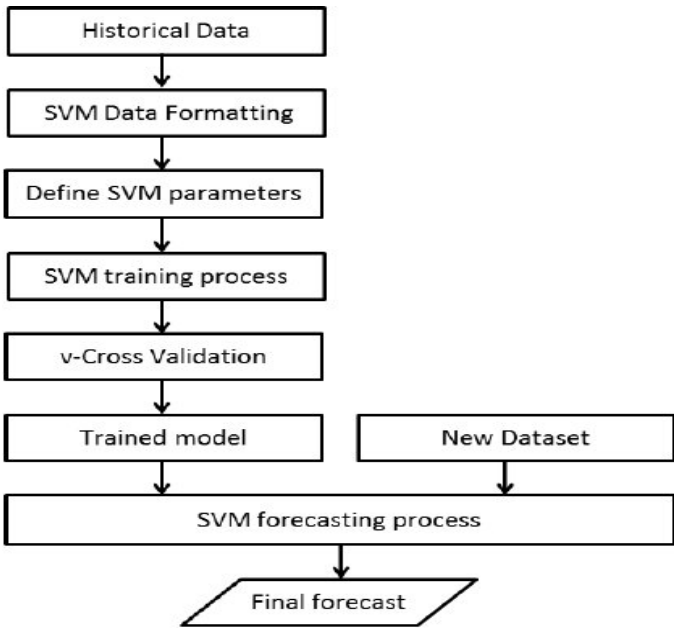

Figure 4: SVM Flowchart

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

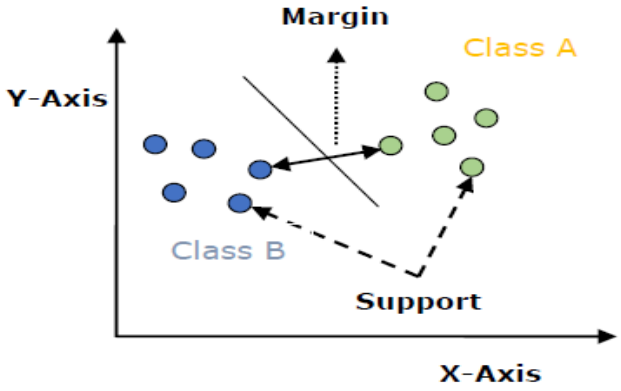

Figure 5: Support Vector Machine

3. Analysis and preparation of soybean datasets

The UCI machine learning repository provided the soybean dataset, which contains thirty-seven observations made from soybean plants across eighteen disease categories, one class for healthy plants, and thirty-five discrete variables. The machine teaching community to analyze their algorithms individually utilizes this data. Google Scholar indicates that at least 52 citations have been made to the archive. The purpose of selecting soybean data based on its raw data and its size is to test the impact of clustering techniques on the efficiency, effectiveness, and scalability of classification algorithms for disease prediction in soybeans. The goal is to demonstrate that applying raw data instead of images for plant disease prediction is possible. The soybean dataset contains errors (noisy), outliers, duplicates, and incomplete values. These issues are resolved through data preparation. Data preprocessing is a method used in both machine learning and data mining to change raw data into a shape that machine learning techniques can understand. ML models are unable to produce any meaningful findings or even provide incorrect responses, which can result in inaccurate conclusions because of the poor quality of the data. Finding out what data the algorithm of ML needs to provide effective results should be done before data preparation begins. Errors, redundant observations, superfluous columns, and irrelevant observations are eliminated. In addition, we handle inconsistent data, outliers, and noise [21] [22].

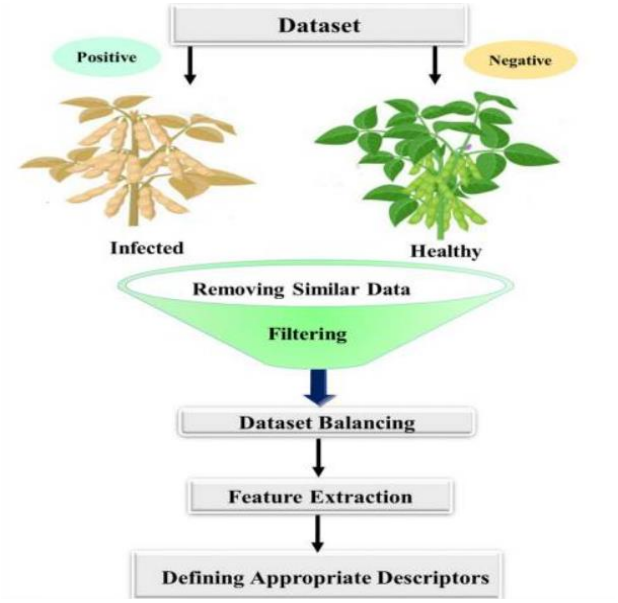

Figure 6: Steps for Pre-processing Data

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

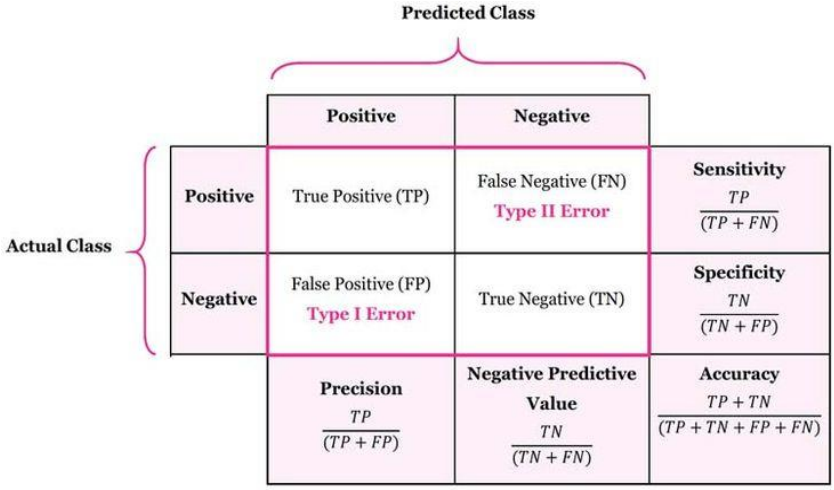

Figure 7: Confusion Matrix

4. Algorithm used for Random Forest and SVM

An elucidation of the used algorithm and its corresponding description is provided below:

a) Algorithm Steps:

1. Input: Training data ^{XXX} with ^{NNN} samples and ^{MMM} features and target values ^{YYY}.
2. Step 1 (Random Forest Feature Selection):
- Train a Random Forest model using the entire dataset.
- Calculate feature importance scores from the Random Forest.
- Select the top ^{kkk} most important features based on the importance scores.
3. Step 2 (Train SVM on Selected Features):
- Use the top ^{kkk} features selected from the Random Forest.
- Train an SVM model using the reduced feature set.
- Optimize the SVM parameters (kernel type, ^{CCC}-value, etc.) during training.
4. Step 3 (Prediction):
- For a new test sample:
- First, reduce its dimensionality using the selected ^{kkk} features.
- Then, use the trained SVM to predict the class label.
5. Output: Predicted class labels for test data.

b) Explanation

1. Random Forest (Feature Selection): A Random Forest is trained on the full dataset. After training, the model provides feature importance scores, which indicate how significant each feature is in making predictions. Based on these important scores, the top K features are selected.
2. Support Vector Machine (Classification): Once the feature space is reduced to the most important ^{kkk} features, the SVM is trained on this subset of features. The SVM will now focus on maximizing the margin between classes using only the most relevant information, which can lead to better performance, especially in high-dimensional datasets.
3. Prediction: When making predictions on new data, we only use the selected features identified in the Random Forest stage. The SVM predicts the class label based on these reduced features.

c) Advantages of Hybrid Approach:

This combined approach can be effective in cases where the data is high dimensional, and you want to improve the efficiency and performance of SVM by leveraging Random Forest's feature selection ability.

1. Feature Selection: Random Forest helps reduce the feature space, which can improve the performance of SVM, especially when the dataset has many irrelevant features.
2. Ensemble Power and Margin Maximization: Random Forest brings the power of ensemble methods, while SVM contributes its strength in maximizing classification margins.
3. Efficiency: Reducing the feature space can significantly speed up SVM training and improve accuracy by reducing overfitting.

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

d) Hybrid RF and SVM Algorithm

This hybrid algorithm integrates Random Forest for feature selection and Support Vector Machine for classification. By first reducing the feature space through Random Forest, the model makes a specialty of the maximum relevant functions, improving SVM's performance and efficiency, particularly in high-dimensional datasets. The final prediction is made by the SVM, which uses these decided on capabilities to maximize the margin between lessons.

Algorithm: HybridRF_SVM(^{XXX}, ^{YYY}, X_test, ^{TTT}, ^{KKK}, ^{CCC}, kernel_function)

Input: Training data ^{XXX}, target labels ^{YYY}, test data X_test, number of trees ^{TTT}, top ^{KKK} features, regularization parameter ^{CCC}, kernel function

Output: Predicted class labels for test data

1. selected_features = RandomForestFeatureSelection(^{XXX}, ^{YYY}, ^{TTT}, ^{KKK})
2. X_selected = ExtractSelectedFeatures(^{XXX}, selected_features)
3. SVM_model = TrainSVM(^{XXX} _selected, ^{YYY}, ^{CCC}, kernel_function)
4. X_test_selected = ExtractSelectedFeatures(^{XXX} _test, selected_features)
5. predictions = PredictSVM(SVM_model, X_test_selected)
6. Return predictions

5. Results and Discussion

The formulas in the previous parts clearly show that RF and SVM have unacceptable ACC when all the data features are used. According to Table and Figure 8 data, RF has an ACC of 62% and a specificity of 65%. F1 measure= 44% , sensitivity=65%, and running time=11 seconds. SVM have ACC=65%, specificity=67%, F- measure = 47%, sensitivity = 68%, and execution time = 9 seconds. Next, we ran an experiment to see which feature subset of the whole dataset would be most useful to the feature selection algorithms.

Gain ratio and correlation were these methods for evaluating attributes. Weka tool supports all of them. Figure 9 displays the percentage of measurements for RF for each feature selection algorithm.

When comparing the two feature selection algorithms, the SVM comes out on top regarding accuracy and speed (6 seconds) and ACC (71%). utilizing the correlation approach in accordance with the experiment's results, which are displayed in Figures 10 and 11. Comparing the experimental results to the dataset results demonstrated that using fewer features for the prediction of target class scores is more precise than the utilization of all data points in the dataset. As in the last experiment, the qualities are closely related. Because of this, the ACC rate of the SVM and RF algorithms is enhanced. The ACC percentages of SVM were higher in this experiment than in the last one, even though different methods for feature selection were used.

Table 1: Evaluating Measures

Algorithms	Specificity	F-Measure	ACC	Time	Sensitivity
SVM	67	47	65	9	68
RF	65	44	62	11	65
SVM with correlation	70	50	71	6	72
SVM with Gain Ratio	74	51	68	4	75
RF with correlation	67	45	64	7	69
RF with Gain Ratio	69	47	68	5	71

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

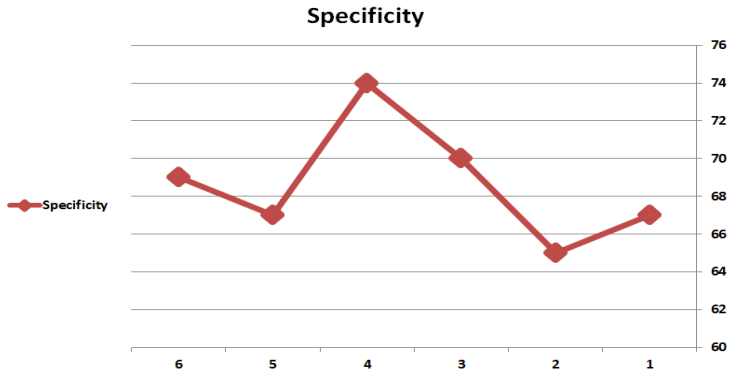

Figure 8: Specificity

Specificity measures the percentage of actual bad cases that have been correctly identified. SVM with Gain Ratio indicates the very best specificity at 74%, observed with the aid of SVM with correlation at 70%. The base RF model has the bottom specificity at 65%. This suggests that SVM-based models, particularly with characteristic choice, are better at efficiently figuring out negative instances.

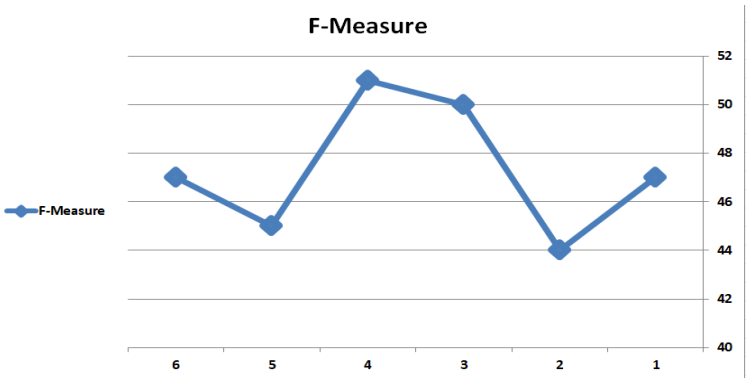

Figure 9: F-Measure

The F-Measure is the harmonic imply of precision and keep in mind, imparting a balanced measure of the model's performance. SVM with Gain Ratio achieves the best F-Measure at 51%, accompanied by SVM with a correlation of 50%. The base RF version has the bottom F-Measure at 44%. This model indicates that SVM-primarily based models, with feature selection; perform better in balancing precision and consideration.

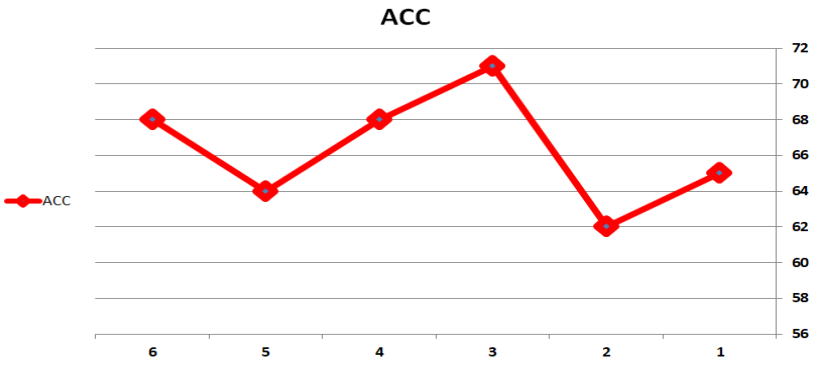

Figure 10: Accuracy

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

Accuracy measures the overall correctness of the model. SVM with correlation shows the very best accuracy at 71%, closely accompanied by the aid of SVM with a Gain Ratio and RF with a Gain Ratio of 68%. The base SVM and RF models display lower accuracy (65% and 62%, respectively), indicating that characteristic selection strategies commonly improve model accuracy.

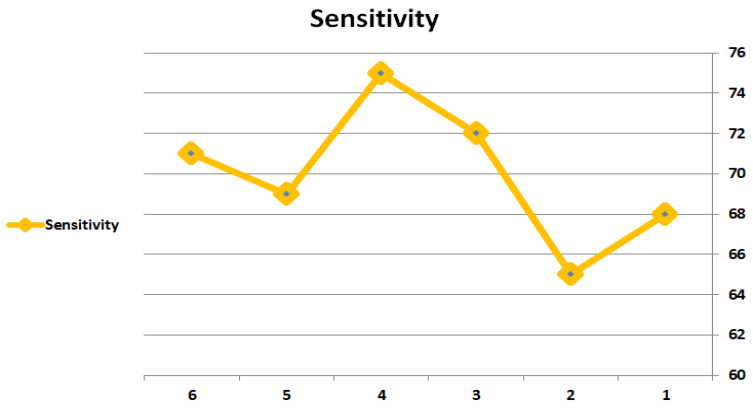

Figure 11: Sensitivity

Sensitivity, referred to as do not forget or actual nice price, measures the proportion of real effective cases successfully recognized. In this dataset, SVM with Gain Ratio suggests the highest sensitivity at s75%, observed by using SVM with correlation at 72%. The base SVM and RF algorithms show decreased sensitivity (68% and 65%, respectively), while characteristic choice techniques (correlation and Gain Ratio) enhance sensitivity for each SVM and RF.

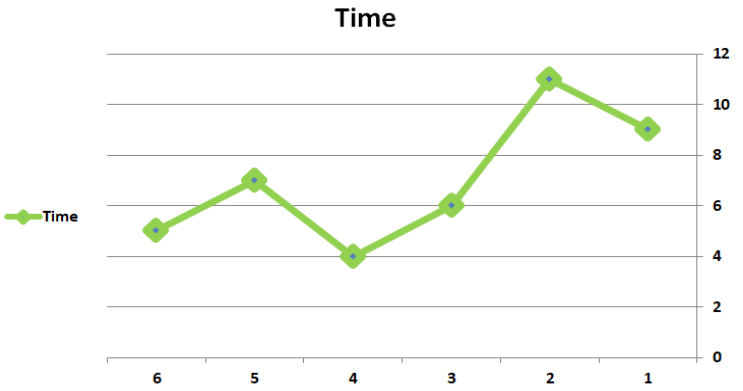

Figure 12: Time

Time likely represents the processing or computation time for each algorithm. RF with Gain Ratio and SVM with Gain Ratio are the fastest (5 and 4 units, respectively), while the base RF algorithm is the slowest (11 units). This suggests that the Gain Ratio feature selection method significantly improves computational efficiency, especially for SVM.

The SVM-based models, particularly those using feature selection methods (correlation and Gain Ratio), consistently outperform the RF-based models across all measures. The SVM with Gain Ratio is the best-performing model, excelling in all measures while being the fastest. Feature selection methods generally improve model performance and computational efficiency for both SVM and RF, with Gain Ratio showing more significant improvements than correlation. The choice between these models would depend on the specific requirements of the task, but based on these metrics, SVM with Gain Ratio appears to be the most promising approach.
The choice between these models would rely on the project's precise necessities, but SVM with Gain Ratio appears to be the most promising approach primarily based on those metrics.

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

6. Conclusion

The suggested approach increases accuracy by aggregating machine learning algorithms with feature selection methods. Feature selection based on soybean data can handle dirty data without jeopardizing the model-development process. The data was cleared using preprocessing methods. The effectiveness of this model was evaluated by contrasting it with other confusion matrix performance metrics, including running time, sensitivity, specificity, and accuracy.

References

[1] A. M. Abdu, M. M. M. Mokji, and U. U. U. Sheikh, "Machine learning for plant disease detection: an investigative comparison between support vector machine and deep learning," *IAES Int. J. Artif. Intell.*, vol. 9, no. 4, pp. 670–683, Dec. 2020, doi: 10.11591/ijai.v9.i4.pp670-683.

[2] S. B. Jadhav, "Convolutional neural networks for leaf image-based plant disease classification," *IAES Int. J. Artif. Intell.*, vol. 8, no. 4, pp. 328–341, Dec. 2019, doi: 10.11591/ijai.v8.i4.pp328-341.

[3] M. A. I. Aquil and W. H. W. Ishak, "Evaluation of scratch and pre-trained convolutional neural networks for the classification of Tomato plant diseases," *IAES Int. J. Artif. Intell.*, vol. 10, no. 2, pp. 467–475, Jun. 2021, doi: 10.11591/ijai.v10.i2.pp467-475.

[4] T. U. Rehman, M. S. Mahmud, Y. K. Chang, J. Jin, and J. Shin, "Current and future applications of statistical machine learning algorithms for agricultural machine vision systems," *Comput. Electron. Agric.*, vol. 156, pp. 585–605, Jan. 2019, doi: 10.1016/j.compag.2018.12.006.

[5] G. Prem, M. Hema, L. Basava, and A. Mathur, "Plant disease prediction using machine learning algorithms," *Int. J. Comput. Appl.*, vol. 182, no. 25, pp. 1–7, Nov. 2018, doi: 10.5120/ijca2018918049.

[6] E. Fujita, Y. Kawasaki, H. Uga, S. Kagiwada, and H. Iyatomi, "Basic investigation on a robust and practical plant diagnostic system," in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Dec. 2016, pp. 989–992, doi: 10.1109/ICMLA.2016.0178.

[7] E. Khalili, S. Kouchaki, S. Ramazi, and F. Ghanati, "Machine learning techniques for soybean charcoal rot disease prediction," *Front. Plant Sci.*, vol. 11, Dec. 2020, doi: 10.3389/fpls.2020.590529.

[8] M. G. Roth et al., "Integrated management of important soybean pathogens of the United States in changing climate," *J. Integr. Pest Manag.*, vol. 11, no. 1, Jan. 2020, doi: 10.1093/jipm/pmaa013.

[9] G. Kaushal and R. Bala, "GLCM and KNN based algorithm for plant disease detection," *Int. J. Adv. Res. Electr. Electron. Instrum. Eng.*, vol. 6, no. 7, pp. 5845–5852, 2017, doi: 10.15662/IJAREEIE.2017.0607036.

[10] R. M. Prakash, G. P. Saraswathy, G. Ramalakshmi, K. H. Mangaleswari, and T. Kaviya, "Detection of leaf diseases and classification using digital image processing," in *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, Mar. 2017, pp. 1–4, doi: 10.1109/ICIIECS.2017.8275915.

[11] S. Bhuvana, B. K. Bharati, P. Kousiga, and S. R. Selvi, "Leaf disease detection using clustering optimization and multi-class classifier," *WSEAS Trans. Comput. Arch.*, vol. 17, 2018.

[12] V. V. Adit, C. V. Rubesh, S. S. Bharathi, G. Santhiya, and R. Anuradha, "A Comparison of Deep Learning Algorithms for Plant Disease Classification," *Adv. Cybern. Cogn. Mach. Learn. Commun. Technol. Lect. Notes Electr. Eng.*, vol. 643, pp. 153–161, 2020.

[13] R. U. Khan, K. Khan, W. Albattah, and A. M. Qamar, "Image-based detection of plant diseases: from classical machine learning to deep learning journey," *Wirel. Commun. Mob. Comput.*, vol. 2021, pp. 1–13, Jun. 2021, doi: 10.1155/2021/5541859.

[14] G. Geetha, S. Samundeswari, G. Saranya, K. Meenakshi, and M. Nithya, "Plant leaf disease classification and detection system using machine learning," *J. Phys. Conf. Ser.*, vol. 1712, no. 1, p. 12012, Dec. 2020, doi: 10.1088/1742-6596/1712/1/012012.

[15] K. S. Sankaran, N. Vasudevan, and V. Nagarajan, "Plant disease detection and recognition using k means clustering," in *2020 International Conference on Communication and Signal Processing (ICCSP)*, Jul. 2020, pp. 1406–1409, doi: 10.1109/ICCSP48568.2020.9182095.

[16] M. A. Hussein, L. E. Kadhim, and A. A. Abdulameer, "Optimum placement of nodes in networks of wireless sensors," *AIP Conf. Proc.*, vol. 2591, Mar. 2023, doi: 10.1063/5.0119640.

[17] A. Yousuf and U. Khan, "Ensemble classifier for plant disease detection," *Int. J. Comput. Sci. Mob. Comput.*, vol. 10, no. 1, pp. 14–22, Jan. 2021, doi: 10.47760/ijcsmc.2021.v10i01.003.

[18] M. Morgan, C. Blank, and R. Seetan, "Plant disease prediction using classification algorithms," *IAES Int. J. Artif. Intell.*, vol. 10, no. 1, pp. 257–264, Mar. 2021, doi: 10.11591/ijai.v10.i1.pp257-264.

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025

[19] A. A. Abdulameer and R. K. Oubida, "The impact of IoT on real-world future decisions," *AIP Conf. Proc.*, vol. 2591, Mar. 2023, doi: 10.1063/5.0119572.
[20] R. S. Michalski and R. L. Chilausky, "Learning by being told and learning from examples: an experimental comparison of the two methods of knowledge acquisition in the context of developing an expert system for soybean disease diagnosis," *Int. J. Policy Anal. Inf. Syst.*, vol. 4, no. 2, 1980.
[21] A. S. Ahmed, Z. K. Obeas, B. A. Alhade, and R. A. Jaleel, "Improving prediction of plant disease using k-efficient clustering and classification algorithms," *IAES Int. J. Artif. Intell.*, vol. 11, no. 3, pp. 939–948, Sep. 2022, doi: 10.11591/ijai.v11.i3.pp939-948.
[22] P. G. Krishna, S. A. Ahamed, K. S. R. Anjaneyulu, and K. S. R. Anjaneyulu, "Artificial Neural Network (ANN) Enabled Weather Monitoring and Prediction System using IoT," in *Proc. 2023 Int. Conf. Intell. Data Commun. Technol. Internet Things (IDCIoT)*, Jan. 2023, pp. 1–5, doi: 10.1109/IDCIoT56793.2023.10053534.

DOI: https://doi.org/10.54216/FPA.180205
Received: July 20, 2024 Revised: November 02, 2024 Accepted: January 14, 2025