JOURNAL OF DATA SCIENCE | Vol.2024:56
eISSN:2805-5160

Comparison of SVM, Naive Bayes, and ELM Models in Plant Growth Classification

M. Muflih[1], Silvia Ratna[2], Haldi Budiman[3], Usman Syapotro[4], Muhammad Hamdani[5]

[1,2,3,4,5,6]Faculty of Information Technology, Islamic University of Kalimantan Muhammad Arsyad Al-Banjari, Indonesia

Email: mmuflihfti@uniska-bjm.ac.id[1], silvia.ratna@uniska-bjm.ac.id[2], haldibudiman@uniska-bjm.ac.id[3], 05usman.syapotro.study@gmail.com[4], mhdhamdani.formal@gmail.com[5]

Abstract

This study investigates the application of machine learning models to predict plant growth milestones based on environmental and treatment data. The dataset comprises categorical variables such as soil type, water frequency, and fertilizer type, alongside numerical variables including sunlight hours, temperature, and humidity. Preprocessing involved one-hot encoding for categorical variables and standard scaling for numerical features. The models employed were Support Vector Machine (SVM), Naive Bayes, and Extreme Learning Machine (ELM). The baseline SVM model achieved an accuracy of 58.97%, and hyperparameter tuning using GridSearchCV did not improve this performance, maintaining the accuracy at 58.97%. The Naive Bayes model achieved an accuracy of 51.28%, while the ELM model had an accuracy of 43.85%. Among the models, the SVM demonstrated the highest accuracy, though further improvement is required for practical implementation. The findings underscore the importance of selecting appropriate machine learning models and optimizing their parameters to enhance prediction accuracy in agricultural applications. Despite the SVM's superior performance in this context, continued refinement is essential to address the challenges posed by predicting plant growth milestones accurately.

Keywords

Plant Growth Prediction, Support Vector Machine, Naive Bayes, Extreme Learning Machine, Hyperparameter Tuning

Introduction

Predicting plant growth is crucial for optimizing agricultural practices, improving crop yields, and reducing resource wastage(Gajula et al., 2021). Plant growth milestones are influenced by a number of variables, including soil type, frequency of watering, fertilizer type, sunshine hours, temperature, and humidity. Planning and management of farming activities can be improved with accurate forecasts of these phases. The intricacy of these treatment- and environmental-related variables, which interact in non-linear ways, makes it difficult to forecast plant development. By examining big datasets and spotting patterns that conventional approaches might miss, machine learning presents a viable remedy.

In previous studies, various machine learning techniques, including linear regression, decision trees, and Support Vector Machines (SVM), have been used to predict plant growth

Submission: 22 October 2024; **Acceptance:** 23 November 2024

based on environmental data. While SVM has shown promise in handling complex, nonlinear relationships, challenges such as overfitting and difficulty in model tuning often limit its effectiveness. Similarly, models like Naive Bayes have been explored, but they tend to struggle with more intricate datasets that require nuanced predictions. Thus, there is a need for exploring alternative models and approaches to further improve prediction accuracy in plant growth applications.

This research introduces the use of Extreme Learning Machine (ELM) as a potential alternative to traditional models. ELM offers faster learning speeds and requires less parameter tuning, making it an attractive option for agricultural data analysis. Additionally, we apply hyperparameter tuning to optimize the SVM model, aiming to assess whether this approach can enhance its performance. The novelty of this study lies in comparing these machine learning models to determine which yields better accuracy for predicting plant growth milestones, providing new insights for agricultural data science.

This paper is organized as follows: The approach is covered in section Methodology , along with the processes used to preprocess the data, apply the model, and employ assessment metrics. The results and their analysis, which compares the models' performances, are shown in section Results and Discusion. Section conclusion wraps up the study by providing a summary of the main conclusions and recommendations for additional research.

Methodology

This study employed a Support Vector Machine (SVM) model to predict plant growth milestones using various environmental and agricultural features. The dataset included six key variables: soil type, water frequency, fertilizer type, sunlight hours, temperature, and humidity, with the target variable being plant growth milestones. The methodology was divided into several key steps: data preprocessing, model development, hyperparameter tuning, and performance evaluation.

Data Preprocessing

The dataset was preprocessed to ensure it was suitable for training the machine learning model. First, the target variable, Growth Milestone, was separated from the feature variables, with the features divided into categorical and numerical types. The categorical features included Soil Type, Water Frequency, and Fertilizer Type, while the numerical features included Sunlight Hours, Temperature, and Humidity.

One-hot encoding, which transforms categorical values into binary vectors so the SVM model can handle them appropriately, was used to change categorical features. In order to reduce the model's sensitivity to scale differences between features, numerical features were simultaneously normalized using the StandardScaler. A ColumnTransformer was used to do both transformations, and the resultant dataset was prepared for modeling.

Pfob et al. (2022) focused on comprehensive data cleaning and feature engineering, including normalization, transformation, encoding, feature selection, and dimension reduction. Lima et al. (2024) emphasized data cleaning by removing or replacing inappropriate data, normalization, and feature selection using correlation filters. Farhangi (2022) highlighted data cleaning by removing outliers, normalization, and dimension reduction by removing features with low variance.

Data Splitting

After preprocessing, the dataset was split into training and testing sets using an 80/20 split ratio. The training set was used to fit the model, while the testing set was reserved for evaluating the model's performance. A random state of 42 was set to ensure the reproducibility of the data split.

(Pfob et al., 2022) emphasized the importance of using a validation set that is ideally independent, or if not possible, randomly dividing the dataset into 80% for model development and 20% for validation. Meanwhile, (Gupta & Goel, 2022) explicitly state that they follow the same split proportion, which is 80% for training and 20% for testing their model.

Support Vector Machine Model

The first model used was a baseline SVM model. This model was trained using the training set, with no hyperparameter tuning applied at this stage. The SVM model used a default radial basis function (RBF) kernel, which is commonly used for non-linear classification problems. After training, predictions were made on the test set, and the model's accuracy was calculated using the accuracy_score function, which compares the predicted values with the actual values in the test set.

(Tahosin et al., 2023) emphasized the efficiency of SVM in handling high-quality data and high-margin hyperplane adaptation, making it suitable for detecting complex relationships in tumor images. Meanwhile, (Farhangi, 2022) explained the basic concept of SVM as an algorithm that discriminates samples in an n-dimensional space by finding the best hyperplane to minimize the error.

Naive Bayes Model

A Naive Bayes model assumes that all attributes are conditionally independent given the class, which rarely holds true in real-world scenarios. To address this limitation, we can extend Naive Bayes by explicitly modeling dependencies between the attributes. This leads to the Augmented Naive Bayes (ANB), where the class node still connects directly to all attribute nodes, but additional links are added between the attribute nodes to capture dependencies. This approach allows the model to better represent the actual relationships between attributes(Zhang, 2004).

Extreme Learning Machine (ELM) Model

Extreme Learning Machines (ELMs) is a machine learning technique used for feedforward neural networks. Unlike traditional neural networks, ELMs do not require the tuning of hidden neurons. ELMs can be used for various tasks, including feature learning, clustering, regression, and classification. They aim to model biological learning mechanisms more efficiently by eliminating the need for iterative tuning of hidden nodes. ELMs work with single-hidden-layer and multi-hidden-layer feedforward networks, where hidden nodes are randomly generated, and they possess universal approximation capabilities(Huang, 2015).

Hyperparameter Tuning

GridSearchCV was used for hyperparameter tuning in order to enhance the performance of the base SVM model. A grid search was performed across a variety of values for the kernel

coefficient (gamma) and regularization parameter (C), both of which have a major effect on the model's performance. The grid search tested the following parameter ranges :

C: [0.1, 1, 10, 100]
gamma: [1, 0.1, 0.01, 0.001]

The RBF kernel was retained during this process, as it is well-suited for complex data patterns. A 5-fold cross-validation was used during the grid search to evaluate the model's performance across different subsets of the training data, ensuring robust model selection.

Hyperparameters are parameters in machine learning models that cannot be directly estimated from the data learning process and must be set before training the model. They define the model architecture and influence how the model learns(Yang & Shami, 2020).

Model Evaluation

After identifying the best hyperparameters, the tuned SVM model was trained again using the optimal parameters obtained from the grid search. Predictions were made on the test set, and the accuracy of the tuned model was calculated using the accuracy_score function. The results were compared with the baseline SVM model to determine whether hyperparameter tuning provided any significant improvement in model performance.

Results and Discusion

This research eploys experimental approach where each model is trained and tested individually on the same data and same splitting ratio, this includes the two version of SVM model. The dataset was acquired form Kaggle, which contains plant growth-related data such as soil type, sunlight exposure duration, water frequency, temperature, and humidity. Model performance will be evaluated using accuracy metric since the dataset classes is balanced.

Table 1. Clasification Result

Algorithm	Splitting Dataset	Accuracy
Naïve Bayes	80% Train, 20% Test	51.28%
ELM	80% Train, 20% Test	53.85%
Base SVM	80% Train, 20% Test	58.97%
SVM Tuned	**80% Train, 20% Test**	**58.97%**

The Naïve Bayes model applied using GaussianNB classifier show an accuracy of 51.28%. While ELM model are applied using MPLClassifier with the hidden layer size of 1000 and a max iteration of 1000 resultted in an accuracy score of 53.85%. These two models can serve as a baseline for the SVM model, both the base an the tuned version. For the SVM mode, the first model, Base SVM, used default parameters and achieved an accuracy of 58.97%, indicating it couldn't capture the complex relationships between the variables. Even after hyperparameter tuning using GridSearchCV to optimize C, gamma, and the kernel type (with the best values being C: 1, gamma: 0.1, and RBF kernel), the accuracy remained the same at 58.97%, suggesting that tuning did not improve the model's performance.

In conclusion, the SVM model outperformed both Naïve Bayes and ELM. Although SVM with tuning was expected to enhance the model's performance through hyperparameter optimization, the experimental results show no improvement in accuracy compared to the Base SVM. The accuracy remained at 58.97%, even after tuning parameters such as C, gamma, and

the kernel. This suggests that hyperparameter tuning alone is insufficient to improve the model's performance, likely due to dataset limitations or the complexity of the data that SVM cannot capture, even with optimized parameters. Therefore, other approaches, such as further feature engineering or using different models, may be necessary to achieve more significant improvements in accuracy.

Conclusion

The results of this study show despite hyperparameter tuning to improve the performance of the Support Vector Machine (SVM) model, both the basic SVM model and the tuned model produce the same accuracy, which is 58.97%. This indicates that hyperparameter tuning does not always guarantee improved performance, especially if the dataset or data complexity is not suitable. Even so, both the base SVM and tuned SVM still outperformed Naïve Bayes and ELM by by 14.99% and 9.50% respectively.

Recommendation

Future research should try other approaches, such as using a larger dataset, feature engineering, using different models, or leverage stacking and boosting, in order achieve more optimal results.

References

Farhangi, F. (2022). Investigating the role of data preprocessing, hyperparameters tuning, and type of machine learning algorithm in the improvement of drowsy EEG signal modeling. *Intelligent Systems with Applications*, *15*(February), 200100. https://doi.org/10.1016/j.iswa.2022.200100

Gajula, A. kumar, Singamsetty, J., Dodda, V. C., & Kuruguntla, L. (2021). Prediction of crop and yield in agriculture using machine learning technique. *2021 12th International Conference on Computing Communication and Networking Technologies, ICCCNT 2021*, *July*. https://doi.org/10.1109/ICCCNT51525.2021.9579843

Gupta, S. C., & Goel, N. (2022). Predictive Modeling and Analytics for Diabetes using Hyperparameter tuned Machine Learning Techniques. *Procedia Computer Science*, *218*(2022), 1257–1269. https://doi.org/10.1016/j.procs.2023.01.104

Huang, G. Bin. (2015). What are Extreme Learning Machines? Filling the Gap Between Frank Rosenblatt's Dream and John von Neumann's Puzzle. *Cognitive Computation*, *7*(3), 263–278. https://doi.org/10.1007/s12559-015-9333-0

Pfob, A., Lu, S. C., & Sidey-Gibbons, C. (2022). Machine learning in medicine: a practical introduction to techniques for data pre-processing, hyperparameter tuning, and model comparison. *BMC Medical Research Methodology*, *22*(1), 1–15. https://doi.org/10.1186/s12874-022-01758-8

Tahosin, M. S., Sheakh, M. A., Islam, T., Lima, R. J., & Begum, M. (2023). Optimizing brain tumor classification through feature selection and hyperparameter tuning in machine learning models. *Informatics in Medicine Unlocked*, *43*(November), 101414. https://doi.org/10.1016/j.imu.2023.101414

Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, *415*, 295–316. https://doi.org/10.1016/j.neucom.2020.07.061

Zhang, H. (2004). The optimality of Naive Bayes. *Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2004*, *2*, 562–567. https://aaai.org/papers/flairs-2004-097/