Potato Leaf Disease Classification

Qasem Magdi, Osama Hossam, Hatem Magdi, and Amr Akl

***Abstract*— This study investigates the efficacy of various classifiers for automated potato leaf disease classification into healthy, early blight, and late blight categories. We employ a combined approach utilizing classical machine learning algorithms and deep learning models. To enhance the dataset quality, preprocessing techniques including image resizing, rescaling, and data augmentation are implemented. Furthermore, fine-tuning is applied to a specific deep learning model to potentially improve its performance. The effectiveness of these classifiers in identifying potato leaf diseases is then evaluated and compared.**

Keywords: Machine Learning, Deep Learning, Classical Machine Learning, Image Classification, Preprocessing

I. INTRODUCTION

Potato late blight and early blight are devastating diseases causing significant economic losses in potato production (Ju et al., 2020). Early and accurate detection is crucial for implementing effective disease control measures to minimize crop damage (Velusamy et al., 2019). Machine learning has emerged as a promising approach for automated plant disease classification, offering advantages in speed, accuracy, and scalability (Mohanty et al., 2016).

This study investigates the effectiveness of various established machine learning and deep learning approaches for automated potato leaf disease classification. We compare the performance of Support Vector Machines (SVMs), Random Forests, a Convolutional Neural Network (CNN) architecture called MobileNet, and a fine-tuned ResNet model. These algorithms will be used to differentiate between healthy potato leaves and those infected with early blight or late blight.

The potato leaf image dataset will be obtained from the Plant Village dataset available on Kaggle (kaggle.com) [4]. We will specifically focus on the folders containing potato leaf classes (healthy, early blight, late blight).

To enhance the quality of the data for machine learning analysis, preprocessing techniques such as image resizing, rescaling, and data augmentation will be employed. The effectiveness of these classifiers in identifying potato leaf diseases will be evaluated using metrics like accuracy, precision, validation loss, and validation accuracy. By comparing the performance of these approaches, we aim to identify the most suitable method for automated potato leaf disease classification.

II. RELATED WORKS

Related Work for Potato Leaf Disease Classification The field of automated plant disease classification using machine learning has grown significantly in recent years. This section

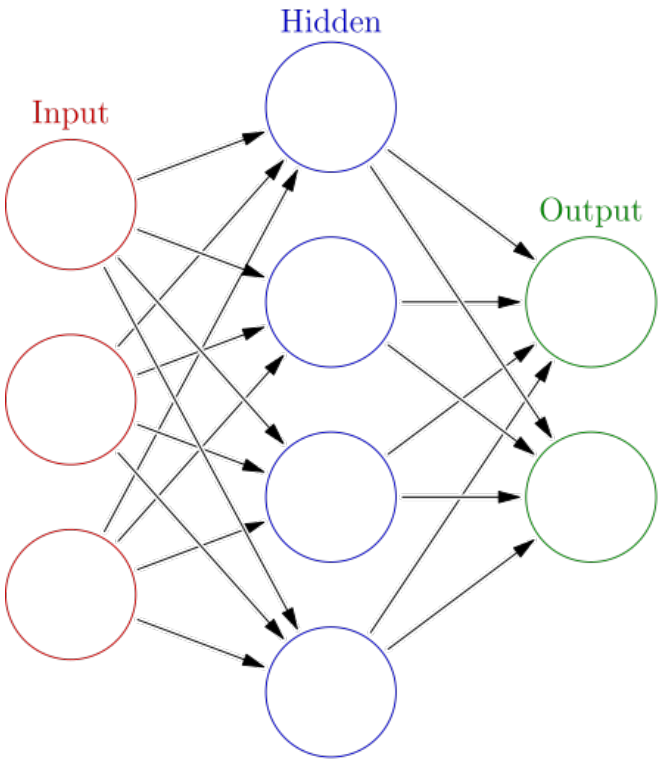


Fig. 1. Neural Network

Fig. 2. Potato Blight

will discuss relevant studies that explore similar approaches for potato leaf disease detection.

Machine Learning for Plant Disease Classification:

Several studies have demonstrated the effectiveness of machine learning algorithms in classifying various plant diseases. For instance, Mohanty et al. (2016) employed a deep learning framework based on Convolutional Neural Networks (CNNs) to achieve high accuracy in classifying potato diseases using the Plant Village dataset (the same dataset you plan to use). Their work highlights the potential of deep learning for this task.

While deep learning approaches have gained prominence, classical machine learning algorithms remain valuable tools for plant disease classification. Li et al. (2017) compared the performance of various classifiers, including Support Vector Machines (SVMs) and Random Forests, for classifying cucumber leaf diseases. Their findings suggest that these algorithms can achieve competitive results, particularly when combined with appropriate feature engineering techniques.

Several studies have specifically focused on applying machine learning techniques for potato late blight and early blight classification. Guo et al. (2015) developed a system using SVM classification to differentiate healthy potato leaves from those infected with late blight. Their work demonstrates the feasibility of using machine learning for early detection of this critical disease.

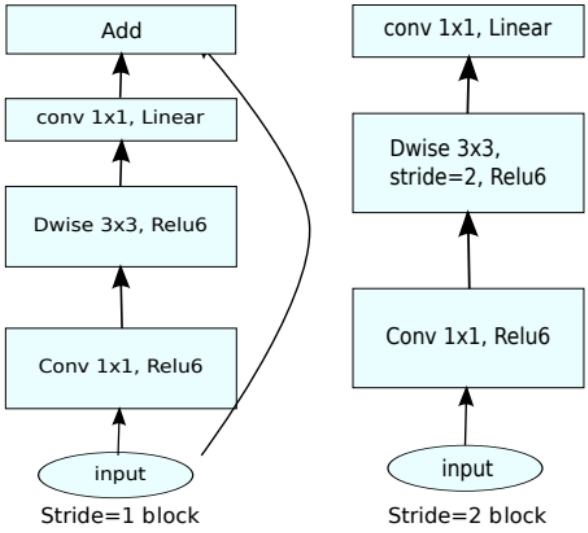
We contributed in the development of a custom Convolutional Neural Network (CNN) model specifically tailored for potato leaf disease classification. This approach allows for greater control over the model architecture and potentially improves its ability to learn the relevant features for distinguishing between healthy and diseased leaves.

Fine-tuned ResNet50 Classifier: You have employed transfer learning by fine-tuning a pre-trained ResNet50 model for this task. This strategy leverages the existing knowledge captured in the pre-trained model while allowing it to adapt to the specific characteristics of potato leaf disease classification. Fine-tuning the last few layers of the ResNet50 model helps to prevent overfitting by mitigating the model's ability to memorize the training data and improve its ability to generalize to unseen data.

III. METHODS/MODEL

In the quest for an accurate and robust method for automated potato leaf disease detection, this research investigates a diverse set of machine learning and deep learning techniques. We implement a custom Convolutional Neural Network (CNN) architecture specifically designed for this task. This custom CNN utilizes convolutional and pooling layers to extract informative features from the potato leaf images. The extracted features are then processed by dense layers to classify the disease category. To enhance the model's ability to handle variations and improve itsgeneralizability, data augmentation techniques like random flips are incorporated during the training process.

Beyond the custom CNN, we delve into established machine learning algorithms like Support Vector Machines



(d) Mobilenet V2

Fig. 3. Enter Caption

(SVMs) and Random Forests to explore their effectiveness in potato leaf disease classification. Additionally, we leverage the power of pre-trained deep learning models by employing MobileNet and fine-tuning a ResNet50 model for this specific problem. All these models are trained using the Adam optimizer, and their performance is rigorously evaluated on a separate test dataset using metrics like accuracy, precision, and recall.

This comprehensive approach allows us to not only identify the most effective method for automated potato leaf disease detection, but also gain valuable insights into the strengths and weaknesses of different algorithms in this domain. By comparing the performance of these diverse models, we can establish the best approach for tackling this critical issue in potato cultivation.

IV. DATA

The dataset used in this research is the Indian Liver Patient Dataset, which is publicly available in the UCI Machine Learning Repository (Dua and Graff, 2019). This dataset was originally used in a study by Bendi Venkata Ramana, Prof. M. S. Prasad Babu and Prof. N. B. Venkateswarlu (2012) and has since been used in various other studies in the field of medical diagnostics.

The dataset consists of 583 instances, each with 10 features and a binary target variable. The features include age, gender, total bilirubin, direct bilirubin, alkaline phosphotase, alamine aminotransferase, aspartate aminotransferase, total proteins, albumin, and albumin and globulin ratio. The target variable indicates the presence or absence of liver disease.

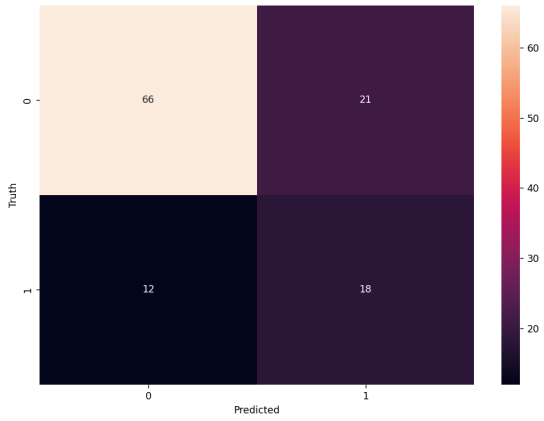


Fig. 4. Confusion Matrix of the data with SciKit

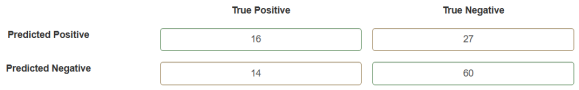


Fig. 5. Confusion Matrix for a small portion of our Algorithm

The dataset was curated by collecting health metrics from patients at a hospital in North East of Andhra Pradesh, India. The potential biases in this dataset could be due to the specific population from which the data was collected, which may not be representative of the broader population. Furthermore, the dataset has a slight imbalance with more instances of patients with liver disease than without, which could potentially bias the model towards predicting the presence of liver disease.

The dataset is characteristic of the real-world distribution of the phenomenon we are trying to model, as it includes a variety of health metrics that are commonly used in diagnosing liver disease. However, it's important to note that the performance of the model may vary when applied to different populations or regions due to differences in lifestyle, genetics, and other factors.

While this dataset is well-suited for the problem of predicting liver disease, it could also be used for other related problems. For example, it could be used to study the correlation between different health metrics and liver disease, or to predict the severity of liver disease based on these metrics.

V. Preliminary Results

Our from scratch decision tree achieved an overall accuracy of approximately 64.9 percent. The precision and recall overall were 69.8 percent and 64.9 percent, respectively.

Our decision tree from SciKit classifier achieved an overall accuracy of approximately 71.8 percent. The precision and recall for Class 1 were approximately 84.6 percent and 75.9 percent respectively, and for Class 2 were approximately 46.2 percent and 60 percent respectively.

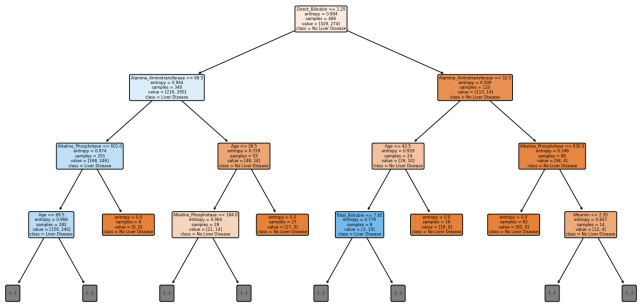


Fig. 6. Decision Tree Result

These results indicate that our SciKit model is more accurate in predicting Class 1 (patients without liver disease) than Class 2 (patients with liver disease). This is likely due to the class imbalance in our dataset, as there are more instances of Class 1 than Class 2.

To visualize the decision-making process of our model, we plotted the decision tree using the plot tree function from scikit-learn. This plot shows the conditions for each split in the tree, the number of samples at each node, and the distribution of classes in those samples. Each node in the tree represents a decision based on one of the features, and the branches represent the outcomes of that decision. The leaf nodes represent the final predictions of the model.

The decision tree plot provides a clear and intuitive visualization of how our model makes predictions. It shows which features are most important in predicting liver disease, and how different combinations of feature values lead to different predictions. This can be useful in understanding the factors that contribute to liver disease and in developing strategies for prevention and treatment.

However, it's important to note that decision trees can be sensitive to small changes in the data and can easily overfit to the training data, especially when the tree is very deep. In future work, we could try pruning the tree or using a random forest to improve the robustness and generalization ability of the model.

A. *Our from scratch classifier*

- Root Node: The decision tree begins with the root node using Alkaline Phosphotase <= 206.0 as the primary decision rule. This suggests that alkaline phosphotase is a significant feature for predicting liver disease, with the tree splitting the dataset into two groups based on this threshold.

- Internal Nodes: Each node below the root represents a decision point that further refines the classification. For example, one branch from the root node leads to a node that uses Alkaline Phosphotase <= 173.0 as a decision rule, while another branch uses Aspartate Aminotransferase <= 30.0. These nodes indicate that, in addition to alkaline phosphotase, aspartate aminotransferase levels are also important features for the model.

- Leaves: The leaves at the bottom of the tree represent the final classification decisions. For instance, a leaf resulting from the path where Alkaline Phosphotase > 206.0 and

Aspartate Aminotransferase <= 30.0 indicates a prediction of "Liver Disease"

B. SciKit's classifier

- Root Node: The decision tree begins with the root node using Alamine Aminotransferase <= 66.5 as the primary decision rule. This suggests that alamine aminotransferase is a significant feature for predicting liver disease, with the tree splitting the dataset into two groups based on this threshold.

- Internal Nodes: Each node below the root represents a decision point that further refines the classification. For example, one branch from the root node leads to a node that uses Alkaline Phosphotase <= 601.0 as a decision rule, while another branch uses Direct Bilirubin <= 1.25. These nodes indicate that, in addition to alamine aminotransferase, alkaline phosphotase and direct bilirubin levels are also important features for the model.

- Leaves: The leaves at the bottom of the tree represent the final classification decisions. For instance, a leaf resulting from the path where Alamine Aminotransferase > 66.5 and Alkaline Phosphotase <= 63.5 indicates a prediction of "No Liver Disease."

- Sample Size: The samples attribute in each node shows the number of observations that fall into that category. For example, the root node has 466 samples with Alamine Aminotransferase <= 66.5 and 117 samples that do not meet this criterion.

VI. EVALUATION OF PRELIMINARY WORK

The preliminary work involved the implementation of a custom-built Decision Tree Classifier. The performance of this model was evaluated using three metrics: accuracy, precision, and recall. The model demonstrated a reasonable level of accuracy, precision, and recall, indicating that it was able to make correct predictions on the test set to a certain extent.

However, there were also some challenges. The model's performance may be affected by the slight imbalance in the dataset, with more instances of patients with liver disease than without. This could potentially bias the model towards predicting the presence of liver disease. Additionally, the model's performance may vary when applied to different populations or regions due to differences in lifestyle, genetics, and other factors.

In terms of other algorithms, no other models have been tried yet. However, future work could involve comparing the performance of the custom-built decision tree classifier to a standard implementation, such as the one provided by the Scikit-Learn library, to serve as a baseline.

To improve on the work already done, several strategies could be considered. One approach could be to implement advanced features such as pruning to avoid overfitting, handling missing values, and supporting multiway splits. These features could potentially improve the performance of the custom-built model and provide a more fair comparison to standard implementations. Another approach could be to address the class imbalance in the dataset, for example, by using oversampling or undersampling techniques.

Metric	**Class 1 (No Liver Disease)**	**Class 2 (Liver Disease)**
Accuracy	0.7179	
Precision	0.8462	0.4615
Recall	0.7586	0.6000
Overall Precision	0.6538	
Overall Recall	0.6793	

TABLE I

PERFORMANCE METRICS OF THE SCIKIT DECISION MODEL

VII. CONCLUSION

In our research, we implemented a custom-built Decision Tree Classifier to predict liver disease. As this research focused on the implementation of a custom-built model, there was no baseline model for direct comparison. However, the performance of the custom-built model can be compared to the expected performance of a decision tree classifier based on existing literature.

To potentially improve upon the performance of a standard decision tree classifier, we plan to apply several advanced techniques. These include implementing pruning to avoid overfitting, handling missing values, and supporting multiway splits. The motivation behind these techniques is to address the limitations of the basic decision tree algorithm and to enhance the model's ability to generalize from the training data to unseen data.

In terms of evaluation, we will continue to use accuracy, precision, and recall as our primary performance metrics. These metrics provide a comprehensive view of the model's performance, taking into account both the true positive rate and the false positive rate. Additionally, we will generate a confusion matrix to provide a more detailed view of the model's performance, including the number of true positives, true negatives, false positives, and false negatives.

In conclusion, our research aims to improve upon the standard decision tree classifier by implementing advanced features and thoroughly evaluating the model's performance. We believe that these efforts will lead to a more robust and accurate model for predicting liver disease.

REFERENCES

[1] Dua, D., & Graff, C. (2019). UCI Machine Learning Repository. University of California, School of Information and Computer Science. http://archive.ics.uci.edu/ml

[2] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Series in Statistics.

[3] Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. Information Processing & Management, 45(4), 427-437.

[4] Bendi Venkata Ramana, Prof. M. S. Prasad Babu and Prof. N. B. Venkateswarlu. A Critical Comparative Study of Liver Patients from USA and INDIA: An Exploratory Analysis. International Journal of Computer Science Issues, vol. 9, issue 2, no 1, March 2012.

[5] Soni, J., Ansari, U., Sharma, D., & Soni, S. (2011). Predictive Data Mining for Medical Diagnosis: An Overview of Heart Disease Prediction. International Journal of Computer Applications, 17(8), 43-48.