12th International Conference on Electrical, Electronics and Computer Engineering (IcETRAN), Čačak, 9-12 June 2025

# Machine Learning Techniques for Smart Agriculture

Zoran Stamenkovic
*Institute of Computer Science, University of Potsdam*
*Potsdam, Germany*
*IHP - Leibniz Institute for Innovative Microelectronics*
*Frankfurt (Oder), Germany*
stamenkovic@uni-potsdam.de; https://orcid.org/0000-0002-6078-413X

***Abstract*—We present the research results that are a basis for integration of hardware devices (sensors and actuators with communication devices, computers, and web servers), data-bases, machine learning (ML) techniques, and related application software. We also describe innovative machine learning approaches and their potential impact on plant treatment and protection. The scientific and technological challenges are discussed as well. The developed machine learning techniques assist farmers in making data-driven and rule-based decisions to reduce the pesticide use and environmental harm. We merge information coming from the features, variables, and correlations extracted by the machine learning algorithms together with information generated by the previous knowledge in the existing databases. The final goal is to learn actionable rules that assist the farmer and help the decision-making process. Using a ML framework, we map scenarios and situations (meteorological data, soil characteristics, nutrients, aggregated daily temperature or features extracted by the knowledge module) to actions (for example, irrigation, dose of fertilizers or pesticides to apply). The developed models can be used to simulate the interaction with the environment and estimate rewards and the expected value function. The framework integrates the objective data, ML algorithms and application software, that can generate and provide end-user relevant information (in the case of traditional crop protection) or control information for actuators (in the case of automated crop protection).**

***Keywords—Machine learning, deep learning, feature selection, image classification, plant, pest, disease, potato blight, crop yield***

## I. INTRODUCTION

The use of artificial intelligence techniques in agriculture dates back to the eighties [2], where some basic classification algorithms were applied to identify rules for the diagnosis of soybean diseases. In this early application, a similarity-based learning program was used to analyze data from over 600 questionnaires describing diseased plants. By using a set of environmental and categorical data (temperature, cropping history and precipitation), as well as the condition of leaves (malformation, defoliation, etc.), each plant was assigned to one of 17 disease categories by an expert collaborator, who used a variety of measurements describing the condition of the plant. Since this early work, the application of data processing and machine learning (ML) techniques to the precision agriculture has witnessed an intensive study. Most of the investigation in this area, however, is still restricted to the analysis of environmental and climate data using general-purpose classification and learning techniques available in off-the-shelf software packages. These approaches cannot provide solutions to the sophisticated problems that precision farming poses nowadays, which demand for specific frame-works and algorithms, and which must be able to process a large volume of multi-source multi-modal data in an online fashion.

Regarding the use of state-of-the-art tools and algorithms, the ML field has witnessed an explosion of sophisticated techniques such as, Kernel Methods [3], Gaussian Processes [4], and Deep Neural Networks [5], which are showing impressive results in many applications such as speech and image recognition, to mention just two example fields. These three representative techniques will form the core of the knowledge extraction system. Although they have not found widespread application in the field of precision agriculture yet, some recent works have considered their use for specific problems in agriculture such as soil classification or crop yield prediction. In the following, we will describe the main ideas behind these three machine learning tools, and we will review some of their most recent applications in agriculture.

Gaussian Processes (GPs) [6] are Bayesian state-of-the-art tools for discriminative machine learning (for example, regression, classification, and dimensionality reduction). GPs can be interpreted as a family of Kernel Methods with the additional advantage of providing a full conditional statistical description for the predicted variable (for example, soil moisture or nitrogen concentration), which can be used primarily to establish confidence intervals and to set the method parameters (called hyper-parameters). In short, GPs assume that a GP prior distribution governs the set of possible latent functions (which are unobserved) and the likelihood (of the latent function) and observations shape this prior to produce the posterior probabilistic estimates. Consequently, the joint distribution of training and test data is a multi-dimensional Gaussian and the predicted distribution is estimated by conditioning on the training data. As a couple of representative examples, GPs have been applied for creating detailed soil maps with predictive properties [7], which is an expensive and time-consuming task, if the conventional methods are used. More recently, the authors of [8] conducted a study using GPs to estimate reference evapotranspiration, which refers to the combination evaporation from the soil surface and transpiration from the crop, with application for irrigation water management.

A related set of ML tools of interest for the precision agriculture is provided by Kernel Methods, which are powerful nonlinear techniques built on the framework of reproducing kernel Hilbert spaces (RKHS). They are based on a nonlinear transformation of the data from the input space to a high-dimensional feature space, where it is more likely that the problem can be solved in a linear manner. One of the best-known Kernel Methods is the so-called support vector machine (SVM), which is powerful non-linear classifier, typically used in the agricultural field for classifying crops and plants from remote sensing measurements [9]. Kernel Methods also provide powerful tools to estimate non-linear correlation between variables that may help to build accurate models for future

ISBN 978-86-6200-031-6 ©2025 AM & ETRAN

https://doi.org/10.69994/12Ic25018

precision agriculture applications. Using these tools, it is possible to estimate whether crop increase and decrease are associated with a specific pattern in the fertilizer usage; or to understand the relationships between the soil characteristics and the nutrient status [10].

A major recent breakthrough in artificial intelligence has been the concept of deep learning and especially Deep Neural Networks (DNNs) [11], which have achieved impressive success in solving hard challenges in many fields including speech recognition, natural language processing, computer vision and image processing. Since images taken from cameras and multispectral images taken from drones are key components of our envisioned database, DNNs will be an important component of our knowledge extraction and decision support system. DNNs use a cascade of multiple layers of simple nonlinear processing units (or linear convolutions) for feature extraction and transformation, where each layer uses the output from the previous layer as input. Figure 1 shows an example of a DNN. Although the fundamental principles behind DNNs are still under discussion, there is consensus that DNNs essentially establish a hierarchical and distributed representation of data, by means of which hidden data invariances are revealed [12]. Another important ingredient for the successful application of DNNs is the availability of large amounts of unlabeled data, which allows the unsupervised learning of the multiple representations of data in hidden layers. Although the application of DNNs in precision agriculture is still largely unexplored, some recent work has shown significant improvement in the classification of plant pests/diseases [13]-[26] and the prediction of crop yield [27]-[40].

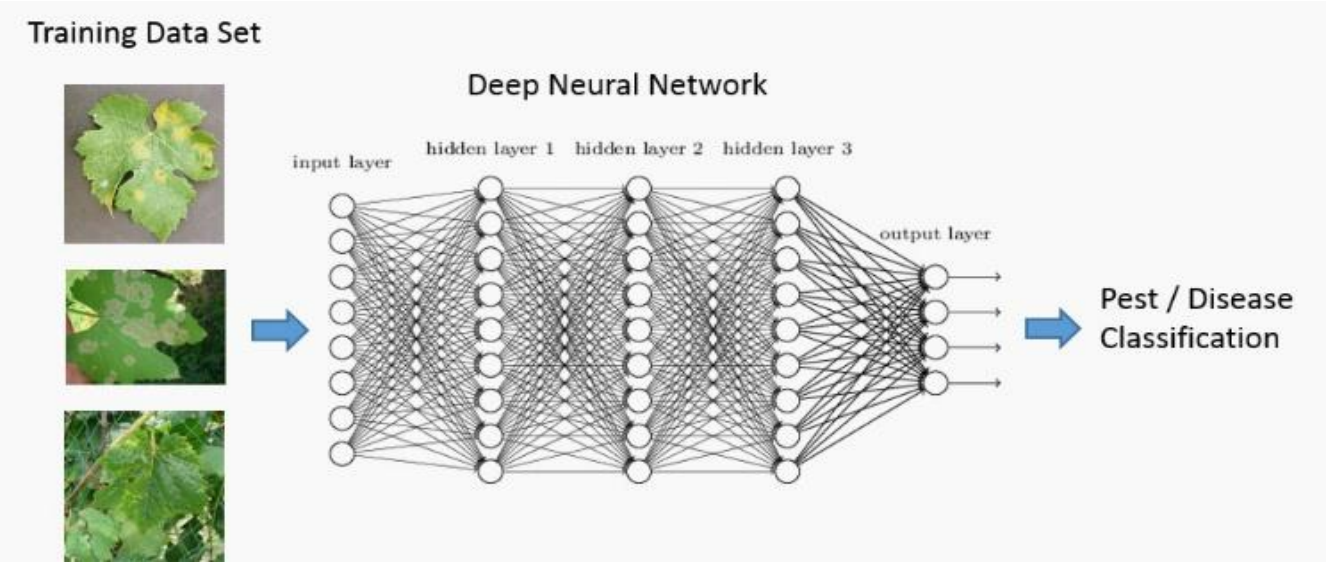


Figure 1. A deep neural network for pest/disease classification.

## II. State-of-Our-Work

Our work has been focused on the following topics:

- Design and development of a decision support system (DSS) able to protect/treat plants and crops considering the temporal and spatial variability of physical, environmental, and agricultural parameters. It is based on remote sensing and the most sophisticated machine learning techniques: Gaussian processes and deep neural networks. Examples of knowledge extraction and actionable rule definition had been presented [1].
- Investigation and development of ML models that can predict the appearance of insects during a season on a daily basis considering the air temperature and relative humidity. Several ML algorithms for classification were applied and their accuracy for the prediction of insect occurrence was presented (up to 76.5%). Since the data used for testing were given in chronological order according to the days when the measurement was performed, an existing method was expanded to consider the periods of three and five days. The proposed method showed better accuracy of prediction and a lower percentage of false detections. In the case of a period of five days, the accuracy of the affected detections was 86.3%, while the percentage of false detections was 11%. The proposed machine learning model can help farmers to detect the occurrence of pests and save the time and resources needed to check the fields [17].
- Development of solutions for early plant disease detection and resource management optimization. The main concept of the proposed solution relies on a direct connection between Cloud and Edge devices, which is achieved through Fog computing. The goal was creation of a deep learning model for image classification that can be optimized and adapted for implementation on devices with limited hardware resources at the Fog level. This increases the importance of image processing for the agricultural operating costs and manual labor. As a result of the off-load data processing at Edge and Fog devices, the system responsiveness is improved, the costs associated with data transmission and storage are reduced, and the overall system reliability and security is increased. The proposed solution can choose classification algorithms to find a trade-off between size and accuracy of the model optimized for devices with limited hardware resources. Testing of our model for the tomato disease classification showed that the decrease in test accuracy is as small as 0.83% (from 96.29% to 95.46%) [22].

- Precise classification of blight infections in potatoes in European countries using the real-time data from 1980 to 2000. To predict the potato blight, we incorporated several hybrid machine learning models, as well as a unique combination of stacking classifier and logistic regression, achieving a high prediction accuracy of 87.22%. Further enhancements of these models and use of new data sources may lead to a higher late blight prediction accuracy and, consequently, a higher efficiency in managing potatoes' health [26].
- Combination of feature selection and classification techniques to predict the yield of plant cultivations. The results depict that an ensemble technique offers a better prediction accuracy than the existing classification techniques. To reduce redundancies and make ML models more accurate, only data features that have a significant degree of relevance in determining the final output of the model must be employed [35].
- Design of a weighted stacked ensemble (WSE) method for the crop yield prediction process. It combines two base learners or classifiers to construct a single predictive ensemble model using weighted instances. The experimental outcomes showed that the proposed WSE outperforms other classification and ensemble techniques in terms of an improved crop yield prediction accuracy [39].
- Yield prediction of various crops such as rice, sorghum, cotton, sugarcane, and wheat using machine learning models. The models were trained using the weather, soil, and crop data to predict future yields of the crops. A dataset of attributes that impact crop production and yield in specific states of India was compiled and the performance of various regressors (linear, gradient boosting, random forest, k-nearest neighbors, decision tree, bagging, extra trees, and ridge) in predicting crop yield was comprehensively studied. The results indicated that the extra trees regressor achieved the highest performance among the models examined [40].

## III. Proposed Research Methods

We focus on integration of software tools and databases into a DSS that can help to gather and analyze critical data to undertake meaningful plant and crop treatment and protection measures in real time. It targets innovative solutions in the area of data processing and knowledge extraction.

Our research generated a base for integration of hardware devices (sensors and actuators with communication devices, computers, and web servers), databases, machine learning techniques, and related application software (Figure 2). This section describes innovative machine learning approaches and their potential impact on plant treatment and protection. The scientific and technological challenges are discussed as well.

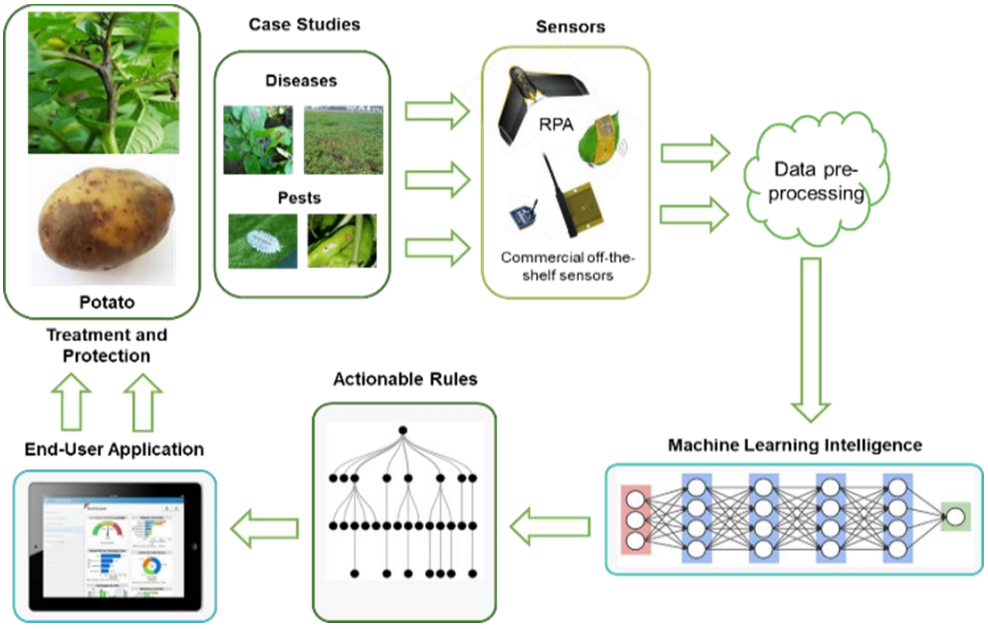


Figure 2. Integration of applications, data, software, and hardware.

The machine learning techniques assist farmers in making data-driven and rule-based decisions to reduce the pesticide use and environmental harm. The foreseen concept is depicted in Figure 3. It merges information coming from the features, variables, and correlations extracted by the machine learning algorithms, together with information generated by the previous knowledge in the existing databases. The final goal is to learn actionable rules that assist the farmer and help the decision-making process.

Using developed ML framework, we will map scenarios and situations (agrometeorological data, soil characteristics, nutrients, aggregated daily temperature or features extracted by the knowledge module) to actions (for example, irrigation, dose of fertilizers or pesticides to apply) that may be learnt using different procedures. The developed models can be used to simulate the interaction with the environment and estimate rewards and the expected value function. Therefore, the framework integrates the objective data, ML algorithms and application software which can generate and provide end-user relevant information (in the case of traditional crop protection) or control information for actuators (in the case of automated crop protection).

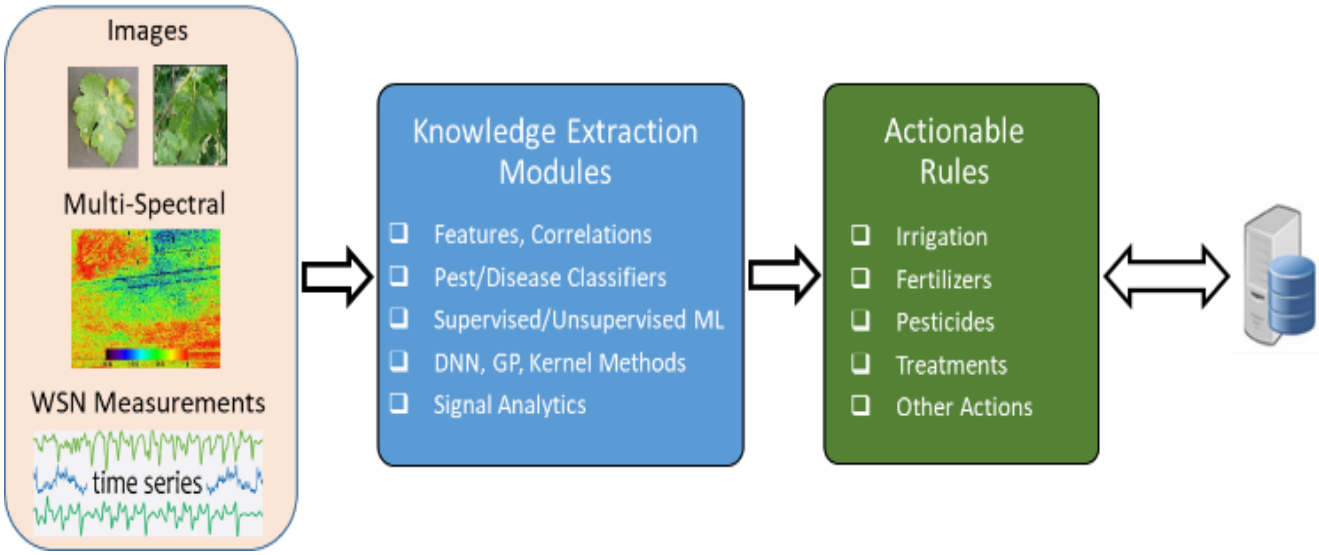


Figure 3. Concept of a machine learning framework.

### A. *Prediction of Pest Insect Appearance*

Our methodology for the prediction of pest insect appearance uses the environmental parameters (temperature and relative humidity) as input data and machine learning for data processing and output generation. According to [41], we initially determine our input parameters: the day of the year, air temperature, and relative humidity. The idea is (based on existing data and without additional investments) to help users (owners of small plantations in different locations) to predict insect appearance and allow them sufficient time to control the dynamics of insect growth. Data on *Helicoverpa armigera* insects caught in traps are collected from different locations. In parallel, the data from meteorological stations located in their immediate vicinity are collected too.

A three to five days observation period is selected depending on the pest development. At the beginning of vegetation, adults (butterflies and moths) emerge for an extended period of approximately three weeks. Oviposition begins two to six days later with the expulsion of a single egg during the night. Female fecundity (the total number of eggs) is around 3000. This species is highly polyphagous. Larval development is harmful (detrimental) for cultivated plants. Larval feeding results in fruits boring, rotting and plant decay. Therefore, it is extremely important to register the appearance of moths in the field and suppress the occurrence of the insects by applying chemicals at two stages (moth and egg). Larvae are usually hidden within stem or fruits and are protected by plant tissue, which makes the spraying impossible. The suppression of moths and eggs will lower the loss and damage in the field. This is the most important step of *Helicoverpa armigera* population control and crop protection.

Our approach predicts the appearance of insects in a period of three to five days when the user can react and prevent the growth of the insect population. It is a customized method guided by real requirements, which estimates the accuracy of the proposed ML model for one, three, and five days in a row. We consider the confusion matrix of our ML model and try predicting the insect appearance (confirmed value '1'), similar to True Positive (TP) values on the given day, in the next three, or the next five, days. If the pest insects appear in these periods, a prediction hit is observed.

Examples of ML models were generated using the Scikit-learn package. Scikit-learn is an open-source Python library that provides support for the implementation of machine learning algorithms. It is based on NumPy and SciPy, which are Python libraries for scientific computation. Due to the wide application of Python, Scikit-learn has gained more popularity. Scikit-learn is characterized by comprehensive coverage of ML models. The implementation of ML model algorithms is optimized for efficient execution on computing resources. Scikit-learn also has great community support for documentation, bug tracking and quality assurance. Within Scikit-learn, the presentation of input and output data is uniform. There is also a fixed procedure for the fitting model so that it is possible to change methods without excessive effort [42].

ML algorithms like the K-Nearest Neighbors, Support Vector Machines with kernel = 'rbf' (RBF SVM), Support Vector Machines with kernel = 'poly' (Poly SVM), Decision Tree, Random Forest, Multi-layer Perceptron classifier (Neural Net), Ada Boost, Gaussian Naive Bayes (G Naive Bayes), and Quadratic Discriminant Analysis (QDA) will be used to form multiple model variants [43].

Based on the data collected from the analyzed locations, an overall dataset was formed, which was used in the first step for the validation and comparison of ML algorithms. The input variables, in addition to the day of the year, represent the values for temperature (T) and relative humidity (RH) in the last ten days in a row.

The accuracy score and confusion matrix are used as outputs to verify the results. The accuracy score is a function that calculates the accuracy of correct predictions and can be represented by:

$$Accuracy = \frac{1}{n}\sum_{i=1}^{n-1} 1(P_i = T_i) \qquad (1)$$

where $P_i$ is the predicted value, $T_i$ is the true value, $n$ is the number of samples and $1(x)$ is the indicator function.

The confusion matrix is one of the metrics used for determining the accuracy of classification when training ML models. In the case of binary problems, the number of true negatives, false positives, false negatives and true positives is obtained. That is the case of classification into two groups [44], where there are two values: true (1) and false (0).

True Negative (TN) is the number of correct negative predictions, False Positive (FP) is the number of incorrect positive predictions, False Negative (FN) is the number of incorrect negative predictions, and True Positive (TP) is the number of correct positive predictions. Having this in mind, the accuracy of prediction can also be calculated using the confusion matrix:

$$Accuracy = \frac{TN + TP}{TN + FP + FN + TP} \quad (2)$$

The confusion matrix was used to represent the validation of the results in the ML model, enabling the observation of how many positive values were actually affected (TP) and how many positive values were incorrectly detected (FP). This way of analyzing the results was convenient because the new correct and false values can be calculated over an extended range that includes three and five days. The obtained values represent the new parameters in Equation (2) for calculating the new accuracy. We also considered the F1 score since the real data represents an imbalanced dataset. The F1 score can be calculated according to

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (3)$$

where *Precision* can be obtained by

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

and *Recall* can be obtained by

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

### *B. Image Processing Using Cloud-Fog Computing*

Fog computing represents a new paradigm that has extended the Cloud to the Edge of the network and is intended to be used with Cloud computing. The Fog computing level has similar characteristics to the Cloud, only with lower performance and closer to the data source, and the same applies to computer, network and storage resources.

They are characterized by being able to perform some data processing near the IoT data source, without sending all the data to the Cloud. The nodes that form the Fog computing layer are connected to the corresponding IoT end devices. These end devices can be cameras, wearable devices, mobile phones, smart glasses, GPS devices, or various types of sensor-based devices. The architecture of Fog computing enables a different execution setting of IoT applications with-out the need to send data to the Cloud enabling the fulfillment of requirements such as dynamic scalability, reduced latency and lower resource consumption in the Cloud [45].

With the Fog layer formed, it is possible to conduct intermediate processing which may include aggregating and preprocessing data from multiple Edge devices before sending it to the Cloud, reducing the load on central servers (Figure 4). It also enables usage of Location-Based Services, since data would be processed within the proximity of the devices. A Fog layer with multiple nodes enables better scalability, enhancing system performance in large-scale deployments by distributing computational tasks across multiple Fog nodes. Fog devices are characterized by their integration with the Cloud and act as intermediaries between Edge devices and Cloud services optimizing data transmission and storage.

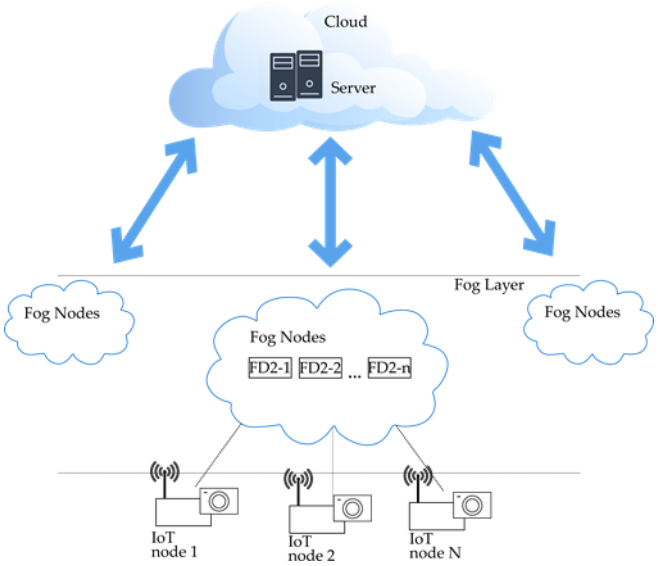


Figure 4. Cloud-Fog computing structure.

Hardware acceleration for deep learning models on Edge devices involves using specialized hardware to improve performance and efficiency. Benefits of using acceleration can be seen through improved performance where hardware accelerators can significantly speed up inference times compared to running models solely on CPUs. Accelerators are designed to perform computations more efficiently, reducing energy consumption on battery-powered devices. Also, accelerator support enables real-time processing of data without relying on Cloud services, enhancing responsiveness and privacy. By using hardware accelerators, powerful deep learning models could be deployed on Edge devices for applications ranging from image and speech recognition to autonomous systems and IoT devices.

The application of artificial neural networks (ANNs) can be found in various business areas and industries, providing a number of services that are supported by high-performance Cloud computing and storage. On the other hand, there are requirements for applications that imply certain intelligence in embedded devices and Edge devices, such as autonomous systems or the Internet of Things (IoT). Although the computing capabilities of Edge devices have increased significantly in recent years, there are still challenges in executing certain neural network algorithms on devices with limited resources. The authors of [46] give a summary of hardware support at the

network Edges for applications of machine learning techniques that require low power consumption.

A summary of research and challenges around Fog computing for IoT are given in [47], while a study on implementation of machine learning models in devices on the network Edge is presented in [48]. This article contains concepts, guidelines and future directions that may facilitate decentralization of machine learning to the network Edge devices, end devices, embedded systems and FPGA.

ANNs involve a large number of parameters and calculations, which require high energy consumption and memory usage. In order to transfer them to devices with limited resources, it is necessary to apply compression techniques in order to optimize the ANN models [49]. As neural networks have become more powerful, their depth and complexity have also made deployment on resource-constrained devices more challenging. The neural network quantization addresses this by reducing network precision, allowing for smaller and simpler networks that fit within hardware constraints. The authors of [50] survey recent quantization techniques and propose some future research directions.

Convolutional neural networks (CNN) progressively extract higher-level features from the input image through a series of convolutional and pooling layers, followed by fully connected layers that perform the final classification or regression. The hierarchical structure of CNNs allows for learning complex patterns and relationships within the data, making them highly effective for various image classification tasks [51].

### *C. Early Detection of Potato Blight Infection*

Potato is the most consumable vegetable all around the world. However, the early or late blight, which is caused by the *P.infestans* pathogen, deteriorates the health of potato plants. Also, bacterial infection in tubers is extremely harmful to human body. So, the infected tubers do not only reduce the nutritional values but are also susceptible to attacks by micro-organisms. Even though the early and late blight can be controlled by fungicides, it is not recommended as they can damage the environmental and human health.

Many machine and deep learning techniques are being used to deal with this issue [52]-[60]. They employ mathematical modeling to predict and monitor the spread of potato disease. These models can analyze many factors which affect the prior prediction of potato blight infection. We use meteorological datasets [26] which are a fusion of different environmental parameters like the air and soil temperature, air and soil humidity, precipitation, solar insolation, wind direction and speed, etc.

In addition to the meteorological conditions, we introduce a vast dataset of potato blight images. The dataset is very unique as it includes real-life images of the two most important potato infections: *P.infestans* and *Alternaria*. Our dataset contains 169 images of plants affected by *P.infestans* and 58 images of plants affected by *Alternaria*. Figure 5 shows a few images (from our own sources) of infected tubers and plants.

Figure 5. Potato affected by *Alternaria* (left) and *P.infestans* (right).

A combination of meteorological data and potato images, as per our views, has never been used before. We apply a unique machine learning technique for categorical classification of meteorological data first. In this manner, a robust way to combat the infection using a uniform blend of stacking classifier with logical regression is provided. Then, a unique deep-learning framework for classification of potato images is used. Finally, we fuse results obtained using the meteorological datasets and potato blight images by employing a maximum probability score technique (Figure 6).

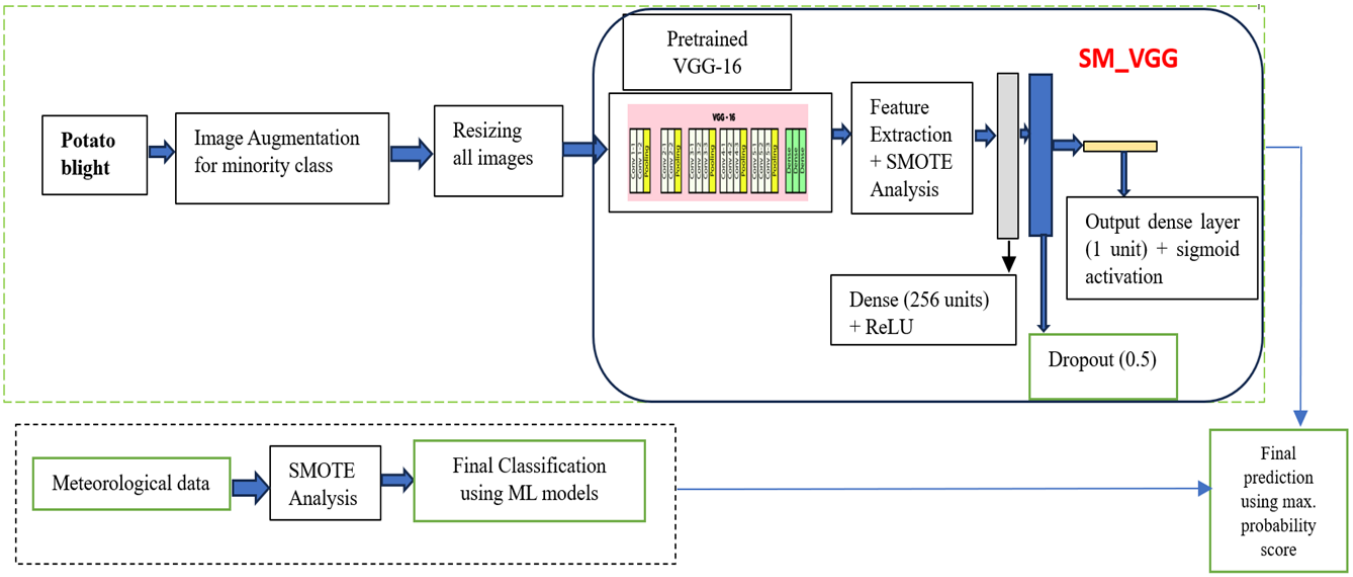


Figure 6. Proposed classification and feature extraction framework.

The following mathematical models are used to predict the potato blight infection:

1. *Stacking classifier* with logistic regression which handles the non-linear relationships between a target variable and features.
2. *Hybrid stacking classifier* which improves the test accuracy combining the best performances of k-nearest neighbor and random forest classifiers.

The key elements of the proposed framework are:

1. *Meteorological data processing layers*: A multi-layer processing of meteorological data such as temperature, humidity, pressure, insolation, precipitation, and wind speed is used for the feature extraction. These features are then combined to represent the atmospheric conditions associated with the development of potato blight.
2. *VGG image module*: Potato images are processed by a convolutional layer (based on VGG-16 model) that is optimized to detect detailed visual features of infection such as color changes and leaf damages.
3. *Output fusion*: Both modules (meteorological and image) are combined in the final layers of the model for joint analysis. The model applies a maximum likelihood aggregation strategy (max-pooling) to generate a console-dated forecast. Once combined, the result is passed to the classification layer that predicts whether the sample is infected and, if yes, which level of infection may occur.

The main advantage of our framework and its architecture is an increased accuracy and precision in potato blight forecasting, which can secure significantly better results in managing potato diseases and preventing their spread.

### *D. Crop Yield Prediction: Variable Identification, Data Collection, and Data Preprocessing*

Variable identification, data collection, and data preprocessing steps are some of the most important steps when it comes to training of a ML model as the performance of the model depends greatly on the quality, consistency, and accuracy of the training dataset. Therefore, it is crucial that these steps are done with diligence (Figure 7).

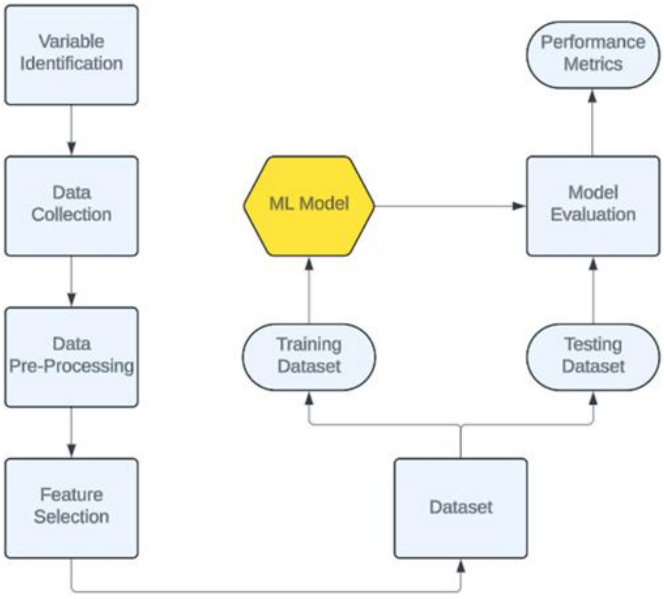


Figure 7. Machine learning methodology flow.

We first identify the variables that are relevant for the crop yield prediction. The most contributing factors are the geographic region of interest, meteorological parameters, soil profile, and crop type [40].

Once the required data is identified, we proceed to identify the data sources and collect the data itself. The data usually comes from the official and verified open access sources [61]-[64]. The data on yield, area, crop type, etc. was obtained from the ICRISAT [65]. The meteorological data such as air temperature, air humidity, etc. was obtained from [62]. The district wise soil data was obtained from [63].

Data preprocessing involves the steps necessary to get them more usable and in the right structure and format for a ML model. This is done by handling missing values, data

discrepancies, and inconsistencies. Firstly, the independent meteorological, soil, and agricultural datasets were merged based on the common district and year attributes. By merging the datasets, a more comprehensive and diverse set of features was created by leveraging the information from multiple sources, which is much more useful for the ML model. After merging all the datasets, the total number of instances is 2550 (34 districts × 15 years × 5 crops).

Secondly, the predictor variables that were redundant and could lead to overfitting were dropped. For example, the product of the area and yield (two predictor variables) as another existing predictor variable was redundant here as it is directly dependent on other two predictor variables. As this variable contributes nothing unique, it was dropped. Absence of redundant predictor variables prevents from construction of an unnecessarily complex model.

Thirdly, after the above two steps some of the categorical variables must be converted to numeric as the models that are used to train accept numeric inputs only. The variables like nitrogen content, phosphorus content, soil type, soil depth, and pH were converted using one hot encoding, and crop type was converted using label encoding.

Finally, the normalization was completed by transforming data into standard ranges. It removes the impact of different units and scales, increases the speed of model training, helps in giving consistent results, and handles outliers and skewed data distributions. We standardized the data by removing the means and scaling it to unit variance. After outliers were removed, the range of values for the target variable was between 1 and 1000. This also included the removal of rows with the zero target variable, indicating that the specific crop was not cultivated in the corresponding district and year. A total of 918 instances were excluded. The final dataset, after preprocessing, comprised 38 attributes (21 attributes without the hot encoding) and consisted of 1632 instances.

### *E. Crop Yield Prediction: Classification and Feature Selection*

Our framework for the selection and subsequent classification of crop features undertaken to predict the crop yield has been described in [35]. While the existing studies have resorted to a single prediction method, our work uses several classification techniques for the crop yield prediction. We used the wrapper techniques to select salient features.

#### *1) Classification techniques*

*Naive Bayes*

The Naive Bayes (NB) is a supervised machine learning algorithm that is mainly used for classification problems. It works under the assumption that the probability of occurrence of any feature is independent of the occurrence of other features and every feature contributes equally to the final outcome. It is based on the Bayes theorem for calculating the probability of events given the occurrence of another event. The Bayes theorem aims to calculate the probability of an event occurring given that another event is true. Therefore, this algorithm is a probabilistic classification technique that predicts the outcome based on probability. Given a labeled dataset and a target variable, the Naive Bayes algorithm calculates the result based on probability. First, the entire dataset is preprocessed and organized into a frequency table by noting the events and their frequencies. Then, a likelihood table is generated using the frequency table. Finally, the Bayes theorem is applied to calculate the posterior probability.

The advantages of the Naive Bayes algorithm include: a) ability to classify binary as well as multi-class variables/data, b) easy implementation comparing to other ML algorithms, c) small amount of training data, d) ability to work with both discrete and continuous data, and e) high scalability.

The main disadvantage of this algorithm is a poor handling of correlated variables since it works on the assumption of independence. In the agriculture, the Naive Bayes method can be used to make lucrative recommendations regarding the crop yield. The method is used to classify weather data, agricultural products, and selling prices and recommend the crop types to farmers. The recommendations can be extremely helpful in an era of the climate change.

*Decision tree*

A decision tree (DT) is a flowchart-like tree structure generally used in supervised ML for classification and prediction. A DT can be turned into a set of rules where each path, heading from the root node to every leaf node, is a rule. In a decision tree, every internal node represents a test/condition or an attribute, every branch is a result of the test, and every leaf node has a class ascribed to it that is reachable if the attribute fulfils the condition of the branch leading to it. There are two types of decision trees (based on categorical and continuous variables) depending on target attribute types.

A decision tree starts with a root node that is compared with other attributes/features in the dataset for a perfect split. A perfect split implies that the number of outputs of one class are on one side of the tree and those of the other class on the other. In this way, every node gets split until it reaches a perfect split, the outcome of which becomes the leaf node of a tree. The real challenge in constructing a DT is attribute selection. Given the large number of attributes available, the most difficult task is to select the ones to be used as root nodes or internal nodes. To this end, we can apply a metrics called Gini Index:

$$\text{Gini Index} = 1 - \boldsymbol{\Sigma}\,(p)^2 = 1 - [(p^+)^2 + (p^-)^2] \qquad (6)$$

where $p^+$ represents the probability of true and $p^-$ the probability of false hypothesis.

*Support vector machine*

Support vector machines (SVM) are supervised learning models that dissect the information for order and relapse the examination. Given a bunch of models, each set apart as possessing a place with one of two classes, the SVM assembles a model that designates new guidelines to one or another type of classification, rendering it a non-probabilistic double-straight classifier (although strategies such as Platt scaling exist to utilize the SVM in a probabilistic order setting). The SVM works to widen the gap between two classifications. New models are planned that fit in and have a place with a class that is dependent on which side of the gap they fall. SVMs help tackle an array of certifiable issues. They are useful in text and hypertext form, as their application decreases the demand for named preparing examples in both conventional inductive and transductive settings. A few techniques for shallow semantic parsing depend on support vector machines.

The SVM algorithms are very effective in analyzing high dimensional datasets. It is of great use in cases where the number of dimensions is greater than the number of samples. SVM consumes less memory for training. However, it is not suitable for large datasets as the time required to train the model rapidly increases. It also gives inaccuracies when the target classes overlap.

*K-nearest neighbor*

This is one of the most used supervised and non-parametric machine learning techniques for classification and regression of labelled data. The labelled data refers to input data that is already tagged with the correct output in supervised learning. Supervised learning algorithms take the data and attempt to make models that predict the output data, given relevant inputs. However, there is no actual learning in the kNN algorithm, which follows the "lazy learning" principle, where all the work happens at the time a prediction is required.

The algorithm depends on distances between points, which can be ascertained using a few methods. The distance must be either zero or positive. This is done by squaring the distance, raising it to a certain power, or taking the absolute values. Distances include the following:

1. Manhattan distance given by

$$|x_2 - x_1| + |y_2 - y_1| \tag{7}$$

2. Euclidean distance given by

$$((x_2 - x_1)^2 + (y_2 - y_1)^2)^{1/2} \tag{8}$$

3. Hamming distance given by

$$|x_1 - y_1| + |x_2 - y_2| \tag{9}$$

If $x$ and $y$ are of the same type, their difference is 0, else is 1.

4. Minkowski distance given by

$$((x_2 - x_1)^p + (y_2 - y_1)^p)^{1/p} \tag{10}$$

The *x* and *y* are coordinates of a point on an x-y plane.

The kNN method can be used for predicting crop yield using a set of known parameters, namely, rainfall, temperature, humidity, and soil moisture. The value of crop yield is calculated by using distances of the nearest neighbors. kNN has yielded suitable accuracy in predicting crop yield. This model can be further enhanced by adding additional features and more data from all the seasons. kNN has also been applied for predictive analysis of the paddy production and has shown better performance compared to the SVM algorithm.

*Bagging*

The bagging, also known as bootstrap aggregation, is used with decision trees, where it significantly raises the stability in terms of advancing accuracy and diminishing variance so eliminates overfitting. In the ensemble machine learning, bagging takes numerous weak models specializing in distinct sections of the feature space and aggregates them to pick the best prediction. An ensemble set of learners is developed and built utilizing the learning algorithm, but with each learner trained on a different set of data. Such a process is termed bootstrap aggregation or bagging. Initially, numerous subsets of the data are created, with each being a subset of the initial data. If an original dataset comprising *n* different instances, *n'* of these are grabbed at random with replacements from the initial data. It is bagged, implying a certain degree of repetition, resulting in some recurrence. Such recurrence, however, creates no problems. Then *m* groups or bags are created altogether, with each holding *n'* different data instances, randomly picked, with replacements. Thus, *n* is the number of training instances in the original data, *n'* is the number of instances in individual bags, and *m* is the number of bags. We always want *n'* to be smaller than *n*, usually by about 60%.

Each of the data groups is used to practice a different model. There are *m* different models, each one practiced on different data, producing an ensemble of different learning algorithms, along with an ensemble of models to be queried identically. The outputs of specific models are taken with their mean to generate the output of the ensemble.

*Random forest*

The random forest (RF) is another supervised ML algorithm. It embodies the essence of ensemble learning since it links multiple classifiers to resolve a complex problem, thereby enhancing the performance of the model. The algorithm builds a "forest" of decision trees. The characteristics are randomly picked in each decision split. The correlation between trees is diminished by randomly picking features that promote the prediction and result in a greater efficiency.

The RF technique provides several advantages. Firstly, it is simple and relatively easy to understand. This technique is also capable of performing both classification and regression tasks. It is suitable for handling large datasets of high dimensionality and, most importantly, it makes the model much more precise and resolves the overfitting issue. The RF cannot be used in case of data extrapolation as it might produce inaccurate results. Also, it does not produce proper results when dealing with sparse data.

The random forest algorithm was used to determine the pH level which in turn helps determine the water supply required by a piece of land. The concept of the random forest is given in Figure 8.

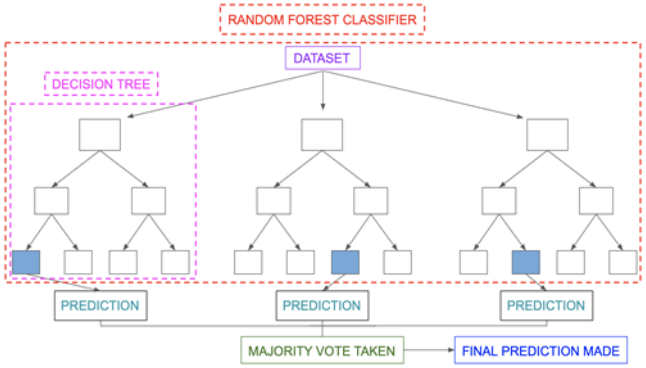


Figure 8. Concept of the random forest

Given that each bagged tree is identically disseminated, the expectation of an average of *B* trees is the equivalent of the expectation of each. Since this accounts for the bias of bagged trees being the same as that of individual trees, a change may only be affected through variance reduction. This contrasts with advancing, where trees are grown adaptively to exclude bias, and hence are not identically distributed. An average of *B* identically

distributed random variables has a variance of $\sigma^2$. If the variables are completely identically distributed with positive pairwise correlation ρ, the variance of the average is given as

$$\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2 \quad (11)$$

It is observed that as *B* increases, the value of the second term shifts negligibly while that of the first term remains unchanged. The RF focuses on bagging variance minimization by cutting the correlation between the trees without increasing the variance excessively. The tree growing process makes this procedure possible through picking input variables randomly.

*2) Feature selection techniques*

*Boruta algorithm*

Boruta is a random forest based algorithm that involves the voting of versatile unbiased indistinct classifiers in decision trees. The importance of a characteristic is estimated by calculating the loss of classification exactness caused by the random permutation of attributes within objects. The average and standard deviation of the accuracy loss are calculated and the average loss is divided by the standard deviation to obtain the Z score to measure fluctuations in the mean accuracy loss.

The importance of every attribute is determined by analyzing all the attributes in the system. Given the random nature of the fluctuations, the shadow attributes are used as a reference to point to the most important ones. As is to be expected, the degree of accuracy depends greatly on the shadow attributes. Consequently, the values are re-shuffled constantly to obtain optimal results.

The algorithm starts with three start-up rounds with a simple criterion for importance. The three rounds help to deal with the tremendous Z score fluctuations when there are large numbers of attributes. In the three start-up rounds, the attributes are compared to the 5th, 3rd, and 2nd best shadows. Rejections occur at the end of each initial round and confirmations, on the contrary, in every round. The time complexity of the algorithm is approximately O (P x N), where P is the number of attributes and N is the number of objects.

*Recursive feature elimination*

This technique is a wrapper feature selection technique that starts with the entire dataset. The ranking method, which is crucial for success, orders the dataset entries from the best to the worst and, consequently, selects the salient features. At each iteration, it eliminates the least important features from the dataset and updates the dataset, continuing the process until the most relevant features are selected and a sophisticated level of optimization is achieved.

The main advantage the recursive feature elimination (RFE) has over other methods is that it categorically verifies every feature's role in processing the output of the model and eliminates features only based on their performance. Thus, it produces better results than filter methods. RFE is also better suited for small sample problems. By using the multiple parameters like soil texture, pH, humidity, topography, gypsum content, etc., the machine learning can be used to assess the land suitability which helps planning suitable use of the agricultural land. Multiple features may be considered to determine the suitability of land but it is not known in advance which features play a key role in determining the final output and which features bring redundancies.

*Modified recursive feature elimination*

This wrapper feature selection technique removes non-salient features from the dataset. Initially, it permutes the dataset by shuffling and combines the shuffled and original datasets. The permutation reduces the computation time and eliminates the need for dataset update. Thereafter, using a random forest classifier, it ranks the features in order from the best to the worst. Based on the ranking result, it selects salient features for the prediction process.

*F. Crop yield prediction: weighted stacked ensemble*

We have proposed a weighted stacked ensemble classifier (WSE) [39] that combines the best heterogeneous base classifiers into a single model (see Table I). For identifying the two diverse characteristic classifiers, the collinearity method is used.

Collinearity refers to a situation where two or more predictor variables are highly correlated, and it can lead to issues such as unstable coefficient estimates and difficulties in interpreting the model. Depending on the correlation results, the two best base classifiers are combined and used for prediction. The algorithm includes four steps:

*Step 1*: Initially, a weight is added to each instance of the crop dataset. Occasionally, different classifiers predict different crops for a given instance during the prediction process. To deal with this problem, a weight is added to each instance of the crop dataset. In this work, the level of the weight factor added to each instance is calculated by breaking the known number of the target class by the number of classes in the sample. The weight range varies from 0 to 1.

*Step 2*: The RFE technique discussed in the section 3.4.2 is applied to the crop dataset to select key features.

*Step 3*: The reduced crop dataset is trained with the kNN, NB, DT, and SVM heterogeneous base classifiers. After the base classifiers are trained with the reduced crop dataset, their collinearity values are determined. Collinearity, or the correlation coefficient, identifies the similarity between the classifiers. Based on the collinearity results, the two best base classifiers are identified.

*Step 4*: The two predicted base classifiers are combined into a single ensemble model called the WSE, which is validated with the testing dataset for predicting a suitable crop to cultivate. The prediction rate of the WSE classifier is higher than that of the base classifiers.

TABLE I. PSEUDO-CODE OF THE WSE ALGORITHM

**Input**: Crop Dataset → D, Training Set $D_T\{wx_i, wy_i\}_{i=1}^{n}$,
Validating Set $D_y = [-D_T, \ ]$,
Base Classifiers $= \{B_1, \dots, B_c\}$
**Output**: Ensemble Classifier E, Suitable Crop
**Begin**:
A ← Swap out missing values (D) using preprocess
F ← Apply RFE technique to select important features
Learn base classifiers
**for** B = 1 to c
Learn $E_B$ based on $D_T$

**end for**
Select base learners
**for** B = 1 to c
    Find collinearity for all base learners
    Select two best base learners to build an ensemble model E based on collinearity results
**end for**
Learn an Ensemble Classifier
    Learn E based on $D_T$
**return** E
Identify the appropriate Crop
    Validate the model E using $D_y$
    Compute the confusion matrix for a model E on the basis of crop prediction
**return** Crop
**End**

The performance of the base classifiers is evaluated using the collinearity metric

$$Collinearity = \frac{\sum(c_i-\bar{c})(d_i-\bar{d})}{\sqrt{\sum(c_i-\bar{c})^2 \sum(d_i-\bar{d})^2}} \quad (12)$$

to construct an ensemble model. The metric parameters are: $c_i$ – values of the c variable in a sample, $\bar{c}$ – mean value of the c variable, $d_i$ – values of the d variable in a sample, and $\bar{d}$ – mean value of the d variable.

The various performance metrics like accuracy (ACC), kappa (K), precision (P), recall (R), specificity (S), F1 score (F1S), mean absolute error (MAE), area under the curve (AUC), and log loss (LL) were used to evaluate the proposed WSE model. ACC helps to identify the overall performance of the model, K is used to find the possibility of correct predictions, P, R, S, F1S, and AUC are used to detect the imbalanced class distribution, and MAE and LL identify the false prediction for multiclass problems.

## IV. Conclusion

This study demonstrates the potential of machine learning and deep learning approaches to predict appearance of pest insects and plant diseases and, consequently, to enable early intervention to protect agricultural crops from severe losses. By introducing unique datasets containing both meteorological data and real-life images, we offer effective prediction of pest and disease appearance and actionable rules that can maximize the crop yield. Currently, our datasets are geographically constrained but our future work will be focused on combination and integration of many different aspects of meteorological data, human genomics, and biotechnology.

## Acknowledgment

The work presented in this paper was supported by the German Federal Ministry for Education and Research in the form of the Brandenburg/Bayern initiative for integration of the artificial intelligence hardware subjects in the university curriculum (BB-KI Chips), project no. 16DHBKIO20.

## References

[1] Z. Stamenkovic, S. Randjic, I. Santamaria, D. Markovic, S. Van Vaerenbergh, and U. Pesovic, "Decision support system for plant and crop treatment and protection based on wireless sensor networks", Proc. 41st IEEE International Spring Seminar on Electronics Technology, Zlatibor (Serbia) 2018, (pp. 1-5)

[2] 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. Info. Systems, vol. 4, pp. 125-161, 1980.

[3] B. Scholkopf and A. J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, Cambridge, MA, MIT Press, 2001.

[4] C. Rasmussen and C. K. Williams, Gaussian Processes for Machine Learning, Cambridge, MA, MIT Press, 2006.

[5] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning", Nature, vol. 521, pp. 436-444, 2015.

[6] F. Perez-Cruz, S. Van Vaerenbergh, J. J. Murillo-Fuentes, M. Lazaro Gredilla, and I. Santamaria, "Gaussian processes for nonlinear signal processing", IEEE Signal Processing Magazine, vol. 30, pp. 40-50, 2013.

[7] J. P. Gonzalez, S. Cook, T. Oberthur, A. Jarvis, J. Bagnell, and B. Dias, "Creating low-cost soil maps for tropical agriculture using Gaussian processes", Proc. Workshop on AI in ICT for Development at the 20th International Joint Conference on Artificial Intelligence, San Francisco (USA) 2007, (pp. 1-6)

[8] D. Holman, M. Sridharan, P. Gowda, D. Porter, T. Marek, T. Howell, and J. Moorhead, "Gaussian processes for reference ET estimation from alternative meteorological data sources", Journal of Hydrology, vol. 571, pp. 28-35, 2014.

[9] G. Camps-Valls, L. Gomez-Chova, J. Calpe-Maravilla, E. Soria-Olivas, J. D. Martin-Guerrero, and J. Moreno, "Support vector machines for crop classification using hyperspectral data", Lecture Notes in Computer Science, vol. 2652, pp. 134-141, 2003.

[10] D. Li and Y. Chen (Eds), Computer and Computing Technologies in Agriculture, Springer Verlag, 2011.

[11] X. Wang, "Deep learning in object recognition, detection, and segmentation", Foundations and Trends in Signal Processing, vol. 8, pp. 217-382, 2014.

[12] S. Mallat, "Understanding deep convolutional networks", Philosophical Transactions of the Royal Society A, vol. 374, pp. 1-16, 2016.

[13] M. Blum, D. Nestel, Y. Cohen, E. Goldshtein, D. Helman, and I. M. Lensky, "Predicting Heliothis (Helicoverpa armigera) pest population dynamics with an age-structured insect population model driven by satellite data", Ecological Modelling, vol. 369, pp. 1-12, 2018.

[14] S. Skawsang, M. Nagai, N. K. Tripathi, P. Soni, "Predicting rice pest population occurrence with satellite-derived crop phenology, ground meteorological observation, and machine learning: A case study for the central plain of Thailand", Applied Sciences, vol. 9, pp. 484601-484619, 2019.

[15] M. C. F. Lima, D. M. E. de Almeida Leandro, C. Valero, L. C. P. Coronel, and C. O. G. Bazzo, "Automatic detection and monitoring of insect pests: A review", Agriculture, vol. 10, pp. 16101-16124, 2020.

[16] J. G. A. Barbedo, "Detecting and classifying pests in crops using proximal images and machine learning: A review", AI, vol. 1, pp. 312-328, 2020.

[17] D. Markovic, D. Vujicic, S. Tanaskovic, B. Djordjevic, S. Randjic, and Z. Stamenkovic, "Prediction of pest insect appearance using sensors and machine learning", Sensors, vol. 21, pp. 484601-484613, 2021.

[18] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification", Computati. Intelligence and Neuroscience, 3289801, pp. 1-11, 2016.

[19] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, "Review of image classification algorithms based on convolutional neural networks", Remote Sensing, vol. 13, 4712, 2021.

[20] S. D. Alwis, Z. Hou, Y. Zhang, M. H. Na, B. Ofoghi, and A. Sajjanhar, "A survey on smart farming data, applications and techniques", Computers in Industry, vol. 138, 103624, 2022.

[21] B. Rokh, A. Azarpeyvand, and A. Khanteymoori, "A comprehensive survey on model quantization for deep neural networks in image classification", ACM Tran. Intelligent Systems and Technology, vol. 14, pp. 1-50, 2023.

[22] D. Markovic, Z. Stamenkovic, B. Djordjevic, and S. Randjic, “Image processing for smart agriculture applications using Cloud-Fog computing”, Sensors, vol. 24, pp. 596501-596526, 2024.

[23] M. Vaidheki, D. S. Gupta, P. Basak, M. K. Debnath, S. Hembram, and S. Ajith, “Prediction of potato late blight disease incidence based on weather variables using statistical and machine learning models: A case study from West Bengal”, J. Agrometeorology, vol. 25, pp. 583-588, 2023.

[24] L. Meno, O. Escuredo, I. K. Abuley, and M. C. Seijo, “Predicting daily aerobiological risk level of potato late blight using C5. 0 and random forest algorithms under field conditions”, Sensors, vol. 23, 3818, 2023.

[25] J. Shi, W. Wang, H. Jin, M. Nie, and S. Ji, “A lightweight Riemannian covariance matrix convolutional network for PolSAR image classification”, IEEE Trans. Geoscien. Remote Sensing, vol. 62, pp. 1-17, 2024.

[26] P. Bagchi, B. Sawicka, Z. Stamenkovic, D. Markovic, and D. Bhattacharjee, “Potato late blight outbreak: A study on advanced classification models based on meteorological data”, Sensors, vol. 24, pp. 786401-786427, 2024.

[27] B. Viviliya and V. Vaidhehi, “The design of hybrid crop recommendation system using machine learning algorithms”, Int. J. Innov. Technol. Exploring Engineering, vol. 9, pp. 4305-4311, 2019.

[28] S. Khaki and L. Wang, “Crop yield prediction using deep neural networks”, Front. Plant Science, vol. 10, pp. 62101-62110, 2019.

[29] T. van Klompenburg, A. Kassahun, and C. Catal, “Crop yield prediction using machine learning: A systematic literature review”, Comput. Electron. Agriculture, vol. 177, 105709, 2020.

[30] K. T. Thomas, S. Varsha, M. M. Saji, L. Varghese, and E. J. Thomas, “Crop prediction using machine learning”, Int. J. Future Gener. Commun. Networks, vol. 13, pp. 1896-1901, 2020.

[31] A. Venugopal, S. Aparna, J. Mani, R. Mathew, and V. Williams, “Crop yield prediction using machine learning algorithms”, Int. J. Eng. Res. Technology, vol. 9, pp. 87-91, 2021.

[32] N. Bali and A. Singla, “Emerging trends in machine learning to predict crop yield and study its influential factors: A survey”, Arch. Computational Methods in Engineering, vol. 29, pp. 95-112, 2021.

[33] L. Gong, M. Yu, S. Jiang, V. Cutsuridis, and S. Pearson, “Deep learning based prediction on greenhouse crop yield combining TCN and RNN”, Sensors, vol. 21, pp. 453701-453716, 2021.

[34] A. K. Srivastava, N. Safaei, S. Khaki, G. Lopez, W. Zeng, F. Ewert, T. Gaiser, and J. Rahimi, “Winter wheat yield prediction using convolutional neural networks from environmental and phenological data”, Scientific Reports, vol. 12, pp. 321501-321514, 2022.

[35] S. P. Raja, B. Sawicka, Z. Stamenkovic, and G. Mariammal, “Crop prediction based on characteristics of the agricultural environment using various feature selection techniques and classifiers”, IEEE Access, vol. 10, pp. 23625-23641, 2022.

[36] M. Sadenova, N. Beisekenov, P. S. Varbanov, and T. Pan, “Application of machine learning and neural networks to predict the yield of cereals, legumes, oilseeds and forage crops in Kazakhstan”, Agriculture, vol. 13, pp. 119501-119527, 2023.

[37] M. Kuradusenge, E. Hitimana, D. Hanyurwimfura, P. Rukundo, K. Mtonga, A. Mukasine, C. Uwitonze, J. Ngabonziza, and A. Uwamahoro, “Crop yield prediction using machine learning models: Case of Irish potato and maize”, Agriculture, vol. 13, pp. 22501-22519, 2023.

[38] A. B. Sarr and B. Sultan, “Predicting crop yields in Senegal using machine learning methods”, Int. J. Climatology, vol. 43, pp. 1817-1838, 2023.

[39] G. Mariammal, A. Suruliandi, Z. Stamenkovic, and S. P. Raja, “A novel ensemble machine learning algorithm for predicting the suitable crop to cultivate based on soil and environment characteristics”, IEEE Canadian Journal of Electrical and Computer Engineering, vol. 47, pp. 127-135, 2024.

[40] U. V. Nikhil, A. M. Pandiyan, S. P. Raja, and Z. Stamenkovic, “Machine learning based crop yield prediction in South India: Performance analysis of various models”, Computers, vol. 13, pp. 13701-13727, 2024.

[41] D. Sagar, S. M. Nebapure, and S. Chander, “Development and validation of weather based prediction model for Helicoverpa armigera in chickpea”, J. Agrometeorology, vol. 19, pp. 328-333, 2017.

[42] J. Hao and T. K. Ho, “Machine learning made easy: A review of Scikit-learn package in Python programming language”, J. Educ. Behav. Statistics, vol. 44, pp. 348-361, 2019.

[43] Scikit-learn, Machine learning in Python, https://scikit-learn.org

[44] S. Visa, B. Ramsay, A. Ralescu, and E. van der Knaap, “Confusion matrix based feature selection”, Proc. Twenty Second Midwest Artificial Intelligence and Cognitive Science Conference, Cincinnati OH (USA) 2011, (pp. 120-127)

[45] R. M. Abdelmoneem, A. Benslimane, and E. Shaaban, “Mobility-aware task scheduling in Cloud-Fog IoT-based healthcare architectures”, Computer Networks, vol. 179, 107348, 2020.

[46] Z. Zou, Y. Jin, P. Nevalainen, Y. Huan, J. Heikkonen, and T. Westerlund, “Edge and Fog computing enabled AI for IoT: An overview”, Proc. IEEE International Conference on Artificial Intelligence Circuits and Systems, Hsinchu (Taiwan) 2019, (pp. 51–56)

[47] A. Hazra, P. Rana, M. Adhikari, and T. Amgoth, “Fog computing for next-generation IoT: Fundamental, state-of-the-art and research challenges”, Computer Science Review, vol. 48, 100549, 2023.

[48] S. Branco, A. G. Ferreira, and J. Cabral, “Machine learning in resource-scarce embedded systems, FPGAs, and End-devices: A survey”, Electronics, vol. 8, 1289, 2019.

[49] B. Rokh, A. Azarpeyvand, and A. Khanteymoori, “A comprehensive survey on model quantization for deep neural networks in image classification”, ACM Trans. Intelligent System Technology, vol. 14, pp. 1-50, 2023.

[50] O. Weng, “Neural network quantization for efficient inference: A survey”, arXiv:2112.06126, 2021.

[51] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, “Review of image classification algorithms based on convolutional neural networks”, Remote Sensing, vol. 13, 4712, 2021.

[52] A. Alvarez-Morezuelas, N. Alor, L. Barandalla, E. Ritter, and J. I. R. de Galarreta, “Virulence of Phytophthora infestans isolates from potato in Spain”, Plant Protection Science, vol. 57, no. 4, pp. 279-288, 2021.

[53] J. Gonzalez-Jimenez, B. Andersson, L. Wiik, and J. Zhan, “Modelling potato yield losses caused by phytophthora infestans: Aspects of disease growth rate, infection time and temperature under climate change”, Field Crops Research, vol. 299, no. 4, 108977, 2023.

[54] M. Javaid, A. Haleem I. Haleem Khan, and R. Suman, “Understanding the potential applications of artificial intelligence in agriculture sector”, Advanced Agrochem, vol. 2, no. 1, pp. 15-30, 2023.

[55] L. P. N. M. Kroon, H. Brouwer, A. W. A. M. de Cock, and F. Govers, “Phytophthora species: Taxonomy, biology, and disease control”, Phytopathology, vol. 102, pp. 16-30, 2012.

[56] C. Qi, M. Sandroni, J. C. Westergaard, E. H. R. Sundmark, M. Bagge, E. Alexandersson, and J. Gao, “In-field classification of the asymptomatic biotrophic phase of potato late blight based on deep learning and proximal hyperspectral imaging”, Computers and Electronics in Agriculture, vol. 205, 107585, 2023.

[57] L. Wang, X. Liu, J. Chen, and Y. Zhang, “A fast and efficient deep learning model for early detection of potato diseases using hyperspectral imagery”, Journal of Agricultural Informatics, vol. 16, no. 1, pp. 24-34, 2023.

[58] G. Maito, M. P. Bortoluzzi, S. Z. Dal Magro, and L. V. Caus Maldaner, “Potato late blight control based on the Blitecast forecast system on the Rio Grande do Sul Plateau, Brazil”, Ciencia Rural, vol. 55, e20230691, 2024.

[59] E. S. Ibrahim, C. Nendel, B. Kamali, E. N. Gajere, and P. Hostert, “Predicting potato diseases in smallholder agricultural areas of Nigeria using machine learning and remote sensing climate data”, PhytoFrontiers, vol. 4, pp. 89-105, 2024.

[60] D. Kumar and R. Mukhopadhyay, “Climate change and plant pathogens: Understanding dynamics, risks and mitigation strategies”, Plant Pathology, vol. 74, pp. 59-68, 2025.

[61] https://agridata.ec.europa.eu/extensions/iacs/iacs.html

[62] https://power.larc.nasa.gov/data-access-viewer

[63] https://geoportal.natmo.gov.in

[64] https://www.fao.org/statistics/en

[65] http://data.icrisat.org/dl