4

Deep learning based prediction of disease-resistance proteins in *Oryza sativa*

4.1 Introduction

O. sativa remains highly susceptible to a variety of biotic stresses, including pathogenic bacteria, which significantly contribute to yield loss. Among these, *M. oryzae* is a major pathogen causing RB disease in *O. sativa* through neck or panicle blast infections. This pathogen invades the stem nodes of *O. sativa* facilitating extensive hyphal proliferation, resulting in neck rot and impaired grain filling [Liu and Wang, 2016]. Similarly, *Xoo* infects *O. sativa* through hydathodes, wounds on the roots or leaves and stomata, proliferating in the xylem vessels leading to wilting and causing BB disease [Kumar *et al.*, 2017]. The impact of such biotic stresses is substantial, with global yield reductions in 2019 estimated at approximately 30% [Vo *et al.*, 2021], posing a significant challenge to food security and threatening the achievement of SDG2. While pesticides have been employed to mitigate these effects, their efficacy has been limited due to the rapid development of resistance in pathogens. Furthermore, the widespread use of pesticides has raised serious concerns regarding their adverse effects on human health and the environment. These challenges underscore the critical need for more effective and sustainable strategies to manage biotic stresses. One promising approach involves the identification of novel disease-resistance proteins that can be integrated into breeding programs. By developing genetically disease-resistant varieties of *O. sativa*, holds the potential to improve productivity and global food security [Pal *et al.*, 2016].

Disease-resistance genes and proteins have been found to play a crucial role in enhancing the productivity of *O. sativa* under biotic stress conditions. Recent advancements have identified several key resistance genes. For instance, the resistance gene *Xa7* has demonstrated long-lasting effectiveness in providing resistance in *O. sativa* varieties under field conditions [Zhao *et al.*, 2022b]. Similarly, the novel NLR family gene *Xa47* has been identified as offering broad-spectrum resistance to BB disease [Lu *et al.*, 2022]. These approaches used traditional methods for their identification, which are often time-consuming. These methods also focus on a limited set of known disease-resistance domains or motifs and rely on domain-centric methodologies, which may inadvertently exclude unexplored disease-resistance proteins.

Computational approaches, particularly ML and DL methods, have become invaluable for the rapid and accurate identification of disease-resistance proteins. This advancement has been facilitated by the integration of diverse data types, such as transcriptomic profiles and stress-response datasets. For instance, a study leveraging microarray data from 559 samples across 13 distinct stress scenarios developed a support vector machine (SVM)-based approach to classify biotic and abiotic stress conditions. This integration of stress-specific transcriptomic data with ML techniques enables precise stress classification in *O. sativa* and other related species [Das *et al.*, 2022]. Another study utilized receptor proteins, including RLKs and RLPs, to predict disease-resistance proteins. This led to the development of the RLPredictOme tool, specifically designed to identify LRR-RLP proteins, which are crucial for mediating immune responses in *O. sativa* [Silva *et al.*, 2022]. A comparative study analyzed the genomic sequence of *O. sativa*, evaluating eight ML methods to identify the most effective approach for predicting disease resistance. Among these, the "three plus kinship (K)" method demonstrated the highest prediction accuracy for disease resistance in *O. sativa* and wheat, significantly advancing genomics-based crop breeding efforts [Liu *et al.*, 2024a]. Expanding the scope to causal gene identification, another study developed QTG-Finder, an ML-based tool designed to rank and identify causal genes within QTLs. This algorithm achieved a prediction accuracy of 79% in *O.*

sativa and *A. thaliana*, successfully identifying genes associated with biotic and abiotic stress responses and key developmental processes [Lin *et al.*, 2019a]. Further extending these advancements, a study employed resistance protein sequences to develop a DL-based model, prPred-DRLF, for predicting plant resistance mediated by R-proteins [Wang *et al.*, 2022b]. Similarly, a DL-based Faster Region-Based Convolutional Neural Network (Faster R-CNN) was implemented to diagnose rice leaf diseases [Bari *et al.*, 2021]. To complement these efforts, a combined DL and SVM-based model was developed to classify four rice leaf diseases using image data, enhancing the accuracy and efficiency of disease diagnosis [Jiang *et al.*, 2020].

Despite the significant progress demonstrated by these ML and DL models, several limitations persist. Most of the reported studies rely heavily on microarray data, genomic sequences, or receptor protein datasets, which may not comprehensively capture the complexity of disease-resistance mechanisms. The reliance on genomic and transcriptomic data alone often overlooks critical proteomic-level interactions and post-translational modifications, which play a pivotal role in plant immune responses. Furthermore, many existing models face challenges in integrating diverse data types, limiting their applicability to broader stress scenarios and related species.

To address these challenges, a robust DL-based model was employed to identify disease-resistance proteins in *O. sativa* and its related species by integrating proteomics data with advanced ML and DL algorithms. An internal dataset, consisting of known disease-resistance proteins and the complete proteome from 13 species closely related to *O. sativa*, was retrieved from the the RiceRelativesGD V4.0 database http://ibi.zju.edu.cn/ricerelativesgd/. Key features, such as amino acid composition, autocorrelation, and proteochemometric properties, were extracted from the protein sequences in the dataset using computational methods. These features were then scaled, normalized, and divided into training (80%) and testing (20%) subsets to ensure robust model evaluation. Six different ML and DL models were trained using this dataset, and their performances were systematically compared. Among these, the MLP model exhibited superior predictive capabilities and was further tested and validated using an external dataset. This external dataset included proteins associated with disease resistance, morphology, and flowering processes in *O. sativa*. To ensure the robustness of the model, its performance was further tested on novel datasets not included during training. This study highlights the potential of DL-based approaches to overcome the limitations of experimental methods and image-based analyses, offering novel insights into disease-resistance mechanisms aiding breeders in improving *O. sativa* varieties.

4.2 Materials and Methods

The computational framework of the DL model highlights the key steps for predicting disease-resistance proteins in *O. sativa* and its related species, including data retrieval, preprocessing, feature extraction and selection, model building, and validation.

4.2.1 Data collection

A total of 98 experimentally validated disease-resistance proteins from 13 plant species closely related to *O. sativa* were retrieved from the RiceRelativesGD V4.0 database. These species included *O. nivara*, *O. rufipogon*, *O. glaberrima*, *O. barthii*, *O. glumaepatula*, *O. meridionalis*, *O. punctata*, *O. brachyantha*, *Zizania latifolia*, *Echinochloa crus-galli*, *O. sativa* f. *spontanea* (*Indica* type), *Leersia perrieri*, and *O. sativa Indica*. These 98 proteins were designated as the positive dataset. Additionally, the complete proteomes of the same 13 species were retrieved from the RiceRelativesGD V4.0 database, yielding a total of 651,535 proteins, which served as the negative dataset. Together, the positive and negative datasets constituted the internal dataset used in this study. In addition, the models were further evaluated and validated through an external dataset consisting of proteins linked to disease resistance, morphological traits, and flowering processes in *O. sativa*. Genes responsible for the *O. sativa* plant morphology refers to the physical traits like leaf shape, root structure, stem, tiller, panicle, spikelets, and grain size. There are several well-known morphological genes, such as *RENL1*, *SRL1*, *SP1* and *GS3*, which govern various morphological traits of *O.*

sativa. In contrast, the flowering genes are the genes that regulate the transition of a plant from the vegetative to the flowering stage, thus controlling the timing and development of flowering. Further, these genes are coordinated by photoperiod, temperature and internal signals such as hormones. *Hd3a* (Heading date 3a), *RFT1* (rice flowering locus T1), *Hd1* (HEADING DATE 1), and *OsGI* (*O. sativa* GIGANTEA) are some of the flowering genes in *O. sativa*. Thus, the morphological and flowering genes are the genes responsible for the overall growth and development of *O. sativa*. On the other hand, the disease-resistance genes specifically focus on *O. sativa* defense system against disease-causing pathogens, providing immunity to *O. sativa* to fight against various pathogenic diseases and maintain its overall health. Moreover, details regarding the databases and computational tools employed are provided in Table B.2.

4.2.2 Data preprocessing

The retrieved dataset exhibited a substantial imbalance, with only 98 proteins in the positive dataset compared to 6,51,535 proteins in the negative dataset. Addressing such imbalance is critical for building unbiased predictive models. While previous studies have addressed this issue by randomly sampling negative datasets from the entire proteome [Pal *et al.*, 2016], this approach carries the risk of introducing biases. Unannotated disease-resistance proteins may exist within the proteome, and the model's performance could become overly reliant on arbitrary criteria for selecting negative samples. To address this, a systematic approach was implemented to balance the negative dataset with the positive dataset. First, proteins containing disease-resistance domains, such as LRR, kinase, NBS, and related domains, were excluded from the negative dataset using the Disease Resistance Analysis and Gene Orthology (DRAGO 2) tool from Plant Resistance Genes Database (PRGdb) [Calle Garcia *et al.*, 2022]. Second, incomplete protein sequences containing unknown residues ('X' sequences) were removed to ensure data quality. Third, the dimension of the negative dataset was reduced using NCBI-BLASTP [Johnson *et al.*, 2008], applying stringent criteria including percent identity ($> 95\%$), query coverage ($> 75\%$), and e-value threshold of 1×10^{-6} to filter redundant and highly similar sequences. Additionally, the Cluster Database at High Identity with Tolerance (CD-HIT) toolkit [Fu *et al.*, 2012] was employed to cluster DNA and protein sequences, further minimizing redundancy and enhancing dataset quality. This structured approach ensured a balanced and reliable dataset, facilitating robust model training and evaluation.

4.2.3 Feature extraction

The processed datasets were subsequently utilized for feature extraction. Essential features of the protein sequences were extracted using the protr package [Xiao *et al.*, 2015] in the R programming environment. The protr package provides various numerical representation methods for protein and peptide sequences, facilitating the generation of diverse descriptors. These descriptors include amino acid composition, autocorrelation, composition-transition-distribution (CTD), conjoint triad, quasi-sequence order, pseudo amino acid composition, and profile-based descriptors derived from the Position-Specific Scoring Matrix (PSSM). Further, the proteochemometric modeling (PCM) is a scale-based descriptor which consists of principal components analysis, factor analysis, multidimensional scaling, amino acid properties (AAindex), more than 20 classes of 2D and 3D molecular descriptors, and BLOSUM/PAM matrix-derived descriptors. A detailed description of each descriptor is described below:

1. **Amino acid composition (AAC)**

   The amino acid composition represents the fraction or composition of each amino acid in a protein sequence. It can be depicted through the following mathematical formula, i.e.,

$$f(r) = \frac{N_r}{N} \quad r = 1, 2, \ldots, 20.$$

   Where, N_r is the number of the amino acid type r, and N is the sequence length. In the present study,

the amino acid composition feature was studied on 98 proteins as positive and 198 proteins as negative datasets, respectively. The function “readFASTA()” of the descriptor to load the protein sequences in the FASTA format file was used. Further, the amino acid composition descriptor was extracted through the "extractAAC()” function, resulting in a vector with elements tagged with the name of each amino acid present in the internal dataset. The descriptor also calculated dipeptide and tripeptide composition with 400 and 8000-dimensional descriptors, respectively.

2. **Autocorrelation**

Autocorrelation describes the distribution of various amino acid properties along the sequence. The properties of amino acids consist of different physicochemical and biochemical properties and pairs of amino acids. In addition, a few amino acid properties are hydrophobicity index, residue volume, molecular weight, polarity, partition coefficient, etc. There are three types of autocorrelation descriptors, i.e., Normalized Moreau-Broto, Moran and Geary autocorrelation. All the autocorrelation types include similar parameters such as polarizability and steric parameters, residue volume, normalized average hydrophobicity scales, etc. In the current study, all eight parameters of Normalized Moreau-Broto, Moran and Geary autocorrelation were calculated with the default functions. Moreover, autocorrelation can be implicated by:

$$P_r = \frac{P_r - \bar{P}}{\sigma}$$

Where, $\bar{P}$ is the average property of the twenty amino acids and can be depicted as:

$$\bar{P} = \frac{\sum_{r=1}^{20} P_r}{20} \quad \text{and} \quad \sigma = \sqrt{\frac{1}{2}\sum_{r=1}^{20}(P_r - \bar{P})^2}$$

3. **Composition, Transition, Distribution (CTD)**

Composition (C) describes the overall distribution of amino acids in a protein sequence and transition (T) represents the frequency with which different amino acids are adjacent to each other. In addition, distribution (D) describes the distribution of amino acids along the length of the protein sequence. CTD includes two steps, i.e., sequence encoding and calculating composition, transition and distribution for a given attribute. In sequence encoding, the amino acids are classified into three classes based on their attributes, and each amino acid residue is encoded by one of the indices 1, 2, and 3 according to their classes. The attributes used in this step include hydrophobicity, normalized van der Waals volume, polarity, charge, secondary structure, solvent accessibility and polarizability. Further, the composition is calculated as the global percentage for each encoded class in the protein sequence. The mathematical formula for the composition is defined as:

$$C_r = \frac{n_r}{n} \quad r = 1,2,3$$

Where, n_r is the number of amino acid type r in the encoded sequence; n is the sequence length.

In addition, the transition from class 1 to 2 is the percent frequency with which class 1 is followed by class 2 or class 2 is followed by class 1 in the encoded sequences. The transition can be computed by:

$$T_{rs} = \frac{n_{rs} + n_{sr}}{N-1} \quad rs = 12,13,23$$

Where, n_{rs} and n_{sr} are the numbers of dipeptides encoded as "rs" and "sr" in the sequence, and N is the sequence length. Finally, the distribution of each attribute in the protein sequence, known as the distributor descriptor, is calculated. There are five distribution descriptors for each attribute, which have position percentages in the whole protein sequence for the first residue, 25% residues, 50% residues, 75% residues, and 100% residues for a certain encoded class. For this study, all the attributes were calculated using the sequence encoding and CTD steps on the internal dataset.

4. **Conjoint Triad descriptor**

The conjoint triad descriptor classifies the amino acids into seven groups, representing the frequency of each triad, i.e., sequence of three amino acids in a protein sequence. The descriptor is used to identify patterns in protein sequences that are associated with protein-protein interactions. Further, each protein sequence is illustrated by a vector space composed of descriptors of amino acids. To reduce the dimensionality of vector space, all 20 amino acids cluster into several classes based on their dipoles and volumes of the side chains. The descriptor is calculated in two steps, i.e., the classification of amino acids and conjoint triad calculation. In the amino acid classification, the dipoles and volumes of the side chains of amino acids are calculated. The classification is based on the density-functional theory method B3LYP/6-31G and the molecular modeling approach. Finally, the 20 amino acids are clustered into seven classes. The conjoint triad descriptor calculation includes the properties of one amino acid and its vicinal amino acids and considers any three continuous amino acids as one unit. The triads can be differentiated based on the classes of amino acids, i.e., triads with three amino acids belonging to the same classes are treated as identical. The conjoint triad descriptor can be computed by:

$$d_i = \frac{f_i - \min\{f_1, f_2, \ldots, f_{343}\}}{\max\{f_1, f_2, \ldots, f_{343}\}}$$

The numerical value of d_i of each protein ranges from 0 to 1, enabling the comparison between proteins, and f_i is the i^{th} value of frequency of the triads of the protein sequences. The conjoint triad descriptor on the internal dataset through the two-step approach with default parameters, thus effectively calculating the protein features was calculated.

5. **Quasi-sequence-order descriptor**

The quasi-sequence-order descriptor is derived from the distance matrix between 20 amino acids. For a particular amino acid, the quasi-sequence order is calculated by adding the distances between one amino acid and all other amino acids in the protein sequence. The calculation consists of two steps, i.e., sequence-order-coupling number (SOCN) followed by quasi-sequence-order (QSO) descriptor. Based on the physicochemical distance between amino acids, a set of sequence-order coupling numbers is calculated to showcase the quasi-sequence-order effect in the first step. Further, the distance matrix between the 20 amino acids is calculated by:

$$X_r = \frac{f_r}{\sum_{r=1}^{20} f_r + w\sum_{d=1}^{\text{maxlag}} \tau_d} \quad r = 1, 2, \ldots, 20$$

Here, f_r is the normalized occurrence for amino acid type i, and w is a weighting factor (w = 0.1). The d-th rank sequence-order-coupling number is defined as:

$$\tau_d = \sum_{i=1}^{N-d} (d_{i,i+d})^2 \quad d = 1, 2, \ldots, \text{maxlag}$$

Where d $_{i,i+d}$ is the distance between the two amino acids at position i and i+d. SOCN and QSO on all the proteins in the internal dataset through the above-mentioned mathematical formulae were calculated.

6. **Pseudo-amino acid composition (PseAAC)**

Pseudo-amino acid composition varies from the classical amino acid composition as it includes a set of discrete numbers derived from the amino acid sequence of the protein. It is based on classifying amino acids into 20 groups, representing each group's frequency in the protein sequence. 20+λ discrete numbers generate PseAAC to express a protein and can be defined as:

$$H_1(i) = \frac{H_1^o(i) - \frac{1}{20}\sum_{i=1}^{20} H_1^o(i)}{\sqrt{\frac{\sum_{i=1}^{20}[H_1^o(i) - \frac{1}{20}\sum_{i=1}^{20} H_1^o(i)]^2}{20}}}$$

where $H_1^o(i)$ is the original hydrophobicity value. The pseudo-amino acid composition represents protein sequences in predicting protein structural and functional attributes. Thus, the structural and functional attributes of the internal dataset were calculated, resulting in hydrophobicity values of the proteins present in the internal dataset.

7. **Profile-based and Proteochemometric modeling descriptors**

The feature vectors of profile-based methods are based on the PSSM by running PSI-BLAST ([Altschul *et al.*, 1997]), resulting in a good performance. Further, protein alignment includes substitution scores for matching amino acids and gap penalties for aligning sequences with gaps. Moreover, substitution scoring for profile-sequence alignments is well understood, and optimal methods for profile-profile substitution scoring are still lacking. Thus, the profile-based descriptor includes different profile-profile substitution scoring systems by calculating the probability that a profile column scores higher when aligned with a related column. In the study, the profile-based descriptors through three functions, i.e., "extractPSSM()", "extractPSSMAcc()", and "extractPSSMFeature()" were generated. On the other hand, proteochemometric modeling includes statistical modeling techniques to model ligand-target interaction space. It includes various descriptors, such as Scales-based descriptors derived from principal component analysis (PCA), factor analysis, multidimensional scaling and BLOSUM and PAM matrix-derived descriptors. Further, the proteochemometric modeling descriptor through the scale-based descriptor, which combined more than 20 classes of 2D and 3D molecular descriptors was calculated. However, these functions are user-friendly and can provide self-defined property matrices to construct scale-based descriptors. The matrix based on the default properties of the internal dataset was calculated. In addition, Table B.2 depicts a list of features of the protr package used in the study.

4.2.4 Feature selection

The features extracted using the protr package resulted in large matrices for both linear and nonlinear descriptors. To address this, feature selection was performed to reduce the dimensionality of descriptor vectors, ensuring that only relevant descriptors generating meaningful data without missing values were retained. This step was essential to minimize computational complexity and processing time.

The matrices generated by feature extraction were examined manually. Few descriptors resulted in null values for a specific feature vector calculation. Descriptors with a significant number of null values within their feature components were discarded after a thorough evaluation. Additionally, the F-score for each descriptor was calculated to assess its feature characteristics, and these scores were compared against a benchmark control dataset using the F-select protocol provided by the libSVM tool [Chang and Lin, 2011].

Based on the calculated F-scores and the alignment of each descriptor's functional properties with the study objectives, the features were ranked and selected accordingly. Descriptors such as AAC, autocorrelation, CTD, conjoint triad, quasi-sequence order, and PseAAC were selected for further analysis. Among the profile-based descriptors, extractPSSM and extractPSSFeature were chosen. From the proteochemometric modeling descriptors, those based on BLOSUM (AABLOSUM62) and PAM (AAPAM120) matrices, amino acid property scales (AAindex), molecular topology-based scales (AAMOE2D, AAMOE3D, AATopo, AAGeom, AAInfo, and AAmolProp), and weighted holistic invariant molecular (WHIM) descriptor scales (AAWHIM) were also selected. The remaining 17 descriptors from this category were excluded to optimize the model.

4.2.5 Modeling of disease-resistance protein on the internal dataset

The 19,503 protein features from the internal dataset through the protr package were extracted and selected. In addition, the internal dataset consists of a high number of features or dimensions which can result in the overfitting and reduced performance of ML and DL models and more computation time. Further, this can also affect the classification between disease and non-disease-resistance proteins by the ML and DL models as the algorithms require larger feature space. Therefore, to reduce the dimensionality of the internal dataset, PCA was performed with 50 as the total number of components [Jordan *et al.*, 1998]. Further, splitting the positive and pre-processed negative datasets into 80% as train and 20% as test sets was performed by a Python program using Scikit-learn [FABIANPEDREGOSA *et al.*, 2011]. This package includes advanced ML and DL algorithms for supervised and unsupervised problems. For the current study, both statistics, i.e., logistic regression (LR), ML, and DL classifiers, to predict disease-resistance proteins were used. These classifiers include Random Forest (RF), SVM, Naïve Bayes (NB), KNN, and MLP. The ML and DL models used for the model building in the study are mentioned below:

1. **Logistic Regression**

   LR is the statistical algorithm for classifying categorical class variables [Kirola *et al.*, 2022]. It analyses the relationship between a set of independent and dependent binary variables. It is based on modelling the probabilities of the output classes while ensuring that output probabilities sum to one and remain between zero and one. Thus, in the study, the LR model classifies the internal dataset into two categorial class variables, i.e., disease-resistance and non-disease-resistance proteins. As the internal dataset consists of binary target variables thus, the binomial type of LR model, classifying the dataset into 0 and 1 was considered. Here, 0 is for disease resistance, and 1 is for non-disease-resistance proteins. The model predicted the two classes based on its logistic function, i.e., if the input was greater than 0.5, which was a threshold value, then it belonged to 1; otherwise, it belonged to 0. In addition, the sigmoid function of the LR model was used to map the predicted values to probabilities.

2. **Random Forest**

   RF is a collection of predictable trees in which each tree is based on independently sampled random vector values with the same distribution of all the trees in the forest [Chauhan *et al.*, 2021]. It consists of a collection of tree-structured classifiers i.e.,

   $$h(x, \Theta_k), k = 1, \ldots$$

   where Θ_k is an independent identically distributed random vector, and x is the input where each tree casts a unit vote for the most popular class (Breiman, 2001). For the study, the RF model performed classification based on the generation of several decision trees on various subsets of the internal dataset and considered the average to improve the predictive accuracy of the dataset. Thus, 100 as 'n_estimators', i.e., the number of decision trees in the forest and 0.01 as 'ccp_alpha,' i.e., cost-complexity pruning were considered. Further, the RF model takes predictions from each tree out of

100, and based on the majority of tree predictions, the final output was predicted. In addition, the minimum cost-complexity pruning to avoid overfitting in the model was maintained by the ccp_alpha, which was a complexity parameter. The RF model first selected random data points from the train set of the internal dataset, followed by the generation of decision trees having selected data points. Then, the 100 decision trees were chosen, and the initial steps were repeated. Finally, the majority of predictions by the trees were selected to classify disease-resistance and non-disease-resistance proteins.

3. **Support Vector Machine**

SVM is a supervised ML technique that can analyse data and recognize patterns. Further, it is extensively used for classification and regression analysis [Pal *et al.*, 2016]. Its working is based on the decision planes that define decision boundaries in the data. It transforms input space into multidimensional space through nonlinear functions known as kernels. The SVM model mapped the proteins in the internal dataset to a high-dimensional feature space to categorise the data points. Then, a separator or hyperplane between the categories, i.e., disease-resistance and non-disease-resistance proteins, was generated, and the data was transformed, classifying the proteins into two categories. In addition, the 'linear' function as a kernel because of the linear internal dataset was used.

4. **Naïve Bayes**

NB is a probabilistic classification method that employs the Bayes theorem and determines the probability of each feature occurring in each class within a dataset using a set of features from that dataset [Morgan *et al.*, 2021a]. Bayes theorem can be represented by the following formula:

$$P(c|x) = (P(x|c)P(c))/P(x) \tag{4.1}$$

Where, P(c|x) is the posterior probability of class (c is the target) given predictor (x is the attribute), P(c) is the prior probability of a class, P(x|c) is the likelihood, which is the probability of the predictor given class, P(x) is the prior probability of the predictor.

The posterior probability for each class within the dataset is calculated by the attribute values for each row of data. The row of data is then assigned to the class with the highest posterior probability. Thus, based on the Bayes theorem, the NB model assumed that the effect of a particular feature, i.e., disease-resistance proteins in a class, is independent of other features, i.e., non-disease-resistance proteins. The NB model first calculated the probability for predicting disease-resistance proteins through the class labels, i.e., 0 and 1 provided to the model. The model then calculated the likelihood probability with each attribute for each class. Further, the probability values were computed using the Bayes formula to calculate posterior probability. Therefore, based on the higher probability of each class, the provided input belonged to the higher probability class.

5. **K-Nearest Neighbor**

KNN performs classification and prediction based on locating the K nearest matches in training data and then using the label of the closest matches [Kartikeyan and Shrivastava, 2021]. It extracts the most similar K data points from the training set and assigns the new data points to the category where K-similar data points occur most frequently. The KNN model by first selecting the number of K of the neighbors, which was '1' for the internal dataset was performed. The Euclidean distance on this K number of neighbors was calculated, and thus, the number of data points in each category was selected. Further, the new data points were assigned to the category, which consisted of a maximum number of neighbors. Finally, the KNN model predicted whether the provided protein of the internal dataset is a disease-resistance or non-disease-resistance protein.

6. **Multilayer Perceptron**

MLP is a non-linear computational method for classifying and regressing complex systems [Yoosefzadeh-Najafabadi *et al.*, 2021]. This algorithm includes several highly interconnected processing neurons that can be used in parallel to detect a solution for a specific problem. It is a feedforward neural network whose nodes are connected to the network, and each node transfers its value to the adjacent node only in the forward direction. The MLP model utilizes a backpropagation algorithm to improve the training model. In the study, the MLP model consisted of an input node that represented the features of the internal dataset. The input node then passes the vector input values to the hidden layer. Further, every edge in the hidden layer consists of some weight multiplied by the input variable, which generates the output. The 'relu' as the activation function in 1000 as the hidden layers were used. Moreover, 'sgd' as a solver, an epoch of 1000, and a learning rate of 0.001 were considered for further analysis. Finally, the output was passed to the output layer, and the difference between predicted and actual output was calculated at the output layer. In the end, the model was backpropagated after calculating the predicted output.

A confusion matrix was constructed for each model using the previously created test dataset to validate the models. The confusion matrix of the models includes information regarding true positive (TP), true negative (TN), false positive (FP), and false negative (FN). TP shows whether the model has correctly predicted a positive outcome or not. TN depicts that the model correctly predicted a negative outcome. FP and FN represent that the model incorrectly predicted a positive and a negative outcome, respectively. In addition, accuracy, Area Under Curve (AUC), F1-score, precision, and recall were calculated through a 10-fold cross-validation method for evaluating the performance of all the models. Detailed descriptions of these metrics are provided below:

(a) **Accuracy**

It is the percentage of correctly classified data points from the total number of samples [Khalili *et al.*, 2020]. It is the number of correct predictions to the total number of predictions. Further, it measures the correct predictions of both TP and TN among all the predictions made by the ML and DL models. The accuracy of the internal dataset of the study was calculated by:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \times 100$$

(b) **AUC-ROC curve**

The Receiver Operating Curve (ROC) curve summarizes the performance of a binary classifier in a single number. It is a visual representation of the performance of the classification model, i.e., LR, RF, SVM, NB, KNN and MLP at various threshold values. The ROC curve for the internal dataset by considering two parameters, i.e., true positive rate (TPR) and false positive rate (FPR) were plotted. Further, the AUC includes calculating the 2D area under the ROC curve. The overall performance across all the thresholds was calculated, providing an aggregate measure of the AUC.

(c) **F1-score**

F1-score depicts the harmonic mean of recall and precision and is computed by:

$$F1score = \frac{2TP}{2TP+FP+FN} \times 100$$

(d) **Precision**

It is also known as positive prediction value (PPV), which compares the number of correctly classified positive samples to the total number of positive samples predicted ([Khalili *et al.*, 2020]). The precision includes calculating TP values that are actually true to the total positive predictions, i.e., TP and FP. The following expression calculated the precision on the internal dataset:

$$Precision = \frac{TP}{TP+FP} \times 100$$

(e) **Recall**

It is also known as specificity, and it measures the ability of the ML and DL models to detect positive samples. The higher the recall, the more positive samples detected. The study included a measure of the prediction of TP values among all the actual positive instances in the internal dataset. The recall was calculated by:

$$Recall = \frac{TP}{TP+FN} \times 100$$

4.2.6 Evaluation of the DL-based model with ML models on an external dataset

The performance of the studied DL-based model was tested on an external dataset comprising a positive dataset of known disease-resistance proteins and a negative dataset of morphological and flowering-related proteins. The positive dataset of 88 disease-resistance proteins and the negative dataset of 121 and 80 proteins for morphology and flowering, respectively, were retrieved from the rice SNP-seek database ([Mansueto *et al.*, 2017]). After removing three incomplete sequences, feature extraction, selection, and size reduction were performed on 286 proteins to test and compare the MLP model with other ML models. The features selected on these proteins include AAC, autocorrelation, CTD, Conjoint Triad Descriptors, Quasi-sequence-order, PseAAC, and Profile-based and Proteochemometric modeling descriptors extracted by the protr package in the R program. Next, among these features, some consist of null outputs in the matrix created by the protr package; thus, these features were discarded from testing the external dataset. Furthermore, features including AAC, autocorrelation, CTD, Conjoint Triad Descriptors, Quasi-sequence-order, and PseAAC were selected. Additionally, profile-based descriptors consisting of extractPSSM and extractPSSFeature, proteochemometric modeling descriptors, BLOSUM (AABLOSUM62) and PAM (AAPAM120) matrix-based scales, amino acid properties-based scales (AAindex), molecular topology-based scales (AAMOE2D, AAMOE3D, AATopo, AAGeom, AAInfo, AAmolProp) and WHIM based descriptor scale AAWHIM were considered for the study.

PCA was performed on the selected features of the external dataset with the number of components set at 50 to reduce the dimension. Further, the external dataset was split into 80% as the train and 20% as the test sets, which was performed by the Scikit-learn package in the Python program. The train set of the internal dataset was utilized as the train set, whereas the test set obtained from splitting the external dataset was considered the test set. The external dataset was then tested by LR, RF, SVM, NB, KNN, and MLP models to analyze their performance in identifying the disease-resistance proteins. The performance matrix, i.e., the confusion matrix having TP, TN, FP and FN values, was studied on the external dataset. Moreover, the model performance, including accuracy, ROC, F1-score, precision, and recall, was evaluated through a 10-fold cross-validation method.

4.2.7 Comparison of performance of the MLP model with well-established models

It is essential to evaluate the suggested approach against other existing state-of-the-art methods. Thus, the performance of the DL-based MLP model against well-established sequence similarity-based approaches on internal and external datasets was compared. These approaches include Pfam, LRRpredictor,

NLR-parser, NBSPred, LRRsearch ([Bej *et al.*, 2014]), LRRfinder ([Offord and Werling, 2013]), Prosite ([Hulo *et al.*, 2006]), SMART ([Letunic *et al.*, 2021]), DeepLRR ([Liu *et al.*, 2022]), RD-RFPDR ([Simón *et al.*, 2022]), PSI-BLAST, RRG Predictor ([Silva and Micheli, 2020]) and DRAGO2 ([Osuna-Cruz *et al.*, 2018]).

4.3 Results

This section describes the findings of the various steps followed in the "Materials and Methods" section, showing the computational framework of the DL model validated and compared with ML models to predict disease-resistance proteins in *O. sativa* and its related species. This section includes outcomes of the data pre-processing and optimization, protein feature extraction and selection, splitting of datasets, validation and comparison of internal and external datasets.

4.3.1 Data optimization for model performance

The study considered experimentally validated disease-resistance proteins of 13 closely related species of *O. sativa* as positive and a whole set of proteins of these related species as negative datasets. The RiceRelativesGD V4.0 database was utilized to retrieve both positive and negative datasets, which were labelled as internal datasets. The positive dataset includes 98 experimentally validated disease-resistance proteins, whereas the negative dataset consists of 6,51,535 proteins. Therefore, to balance the negative dataset with the positive dataset, these underwent preprocessing to enhance the accuracy and consistency of our model. This involved the removal of redundant and noisy data. In a previous study, negative datasets were randomly selected from the full proteome [Pal *et al.*, 2016]. However, this method introduces selection bias since unannotated disease-resistance proteins could be included, making the model reliant on arbitrary selection criteria. To avoid this bias, we applied a three-step approach to balance our dataset.

In step 1, some 3,33,342 proteins containing known disease-resistant domains such as LRR, Kinase, TM, RLK, RLP, coiled-coil, Toll-interleukin-1 receptor, and NBS, were excluded from the negative dataset. These disease-resistant domains were identified by utilizing the DRAGO 2 tool available in PRGdb. Further, after the removal of proteins with disease-resistant domains, a total of 3,18,193 proteins were found. In step 2, some 9,007 protein sequences containing incomplete fragments, indicated by the presence of 'X' characters, were removed. These characters represent unidentified amino acids within protein sequences. After this step, 309,186 proteins were remained. Finally, in step 3, the dimensionality of the negative dataset was further reduced using the NCBI-BLASTP. The BLASTP search was performed with an e-value threshold of 1x10-6, query coverage >75%, and sequence identity >95%, yielding 45,379 proteins. Next, data redundancy of the negative dataset was further minimized using the CD-HIT toolkit. It is a widely used tool in bioinformatics for clustering large protein or nucleotide sequence datasets. It groups sequences that share a high level of identity, reducing redundancy and improving computational efficiency for downstream analyses. In our study, CD-HIT was utilized to cluster sequences from the negative dataset, helping to minimize data redundancy. By clustering sequences with a predefined sequence identity threshold, the tool ensures that highly similar or identical sequences are represented by a single representative sequence. This process enhances the performance of subsequent sequence analyses by reducing computational complexity and potential biases introduced by redundant sequences. CD-HIT operates by sorting sequences by length, selecting the longest sequence as the representative, and comparing shorter sequences to determine if they meet the identity threshold. If they do, they are clustered together; otherwise, they form a new cluster. The parameters were set to balance between maintaining diversity and reducing redundancy. The CD-HIT clustering was performed based on a word size of 5 and a clustering threshold between 0.7 and 1.0, resulting in 33 clusters. Finally, among these 33 clusters, 9 were selected based on the similarity among the protein sequences. In our study, sequences with 95-100% similarity were considered, and in total, these were 198 proteins. Thus, our final dataset contained 98 positive proteins and 198 negative proteins for feature extraction, model training, and validation. Figure 4.2 visually represents the steps involved in performing the study to predict disease-resistance proteins in *O. sativa* and its related species on the internal dataset. Moreover,

similar steps were followed on the external dataset to validate and predict the performance of the DL-based MLP model.

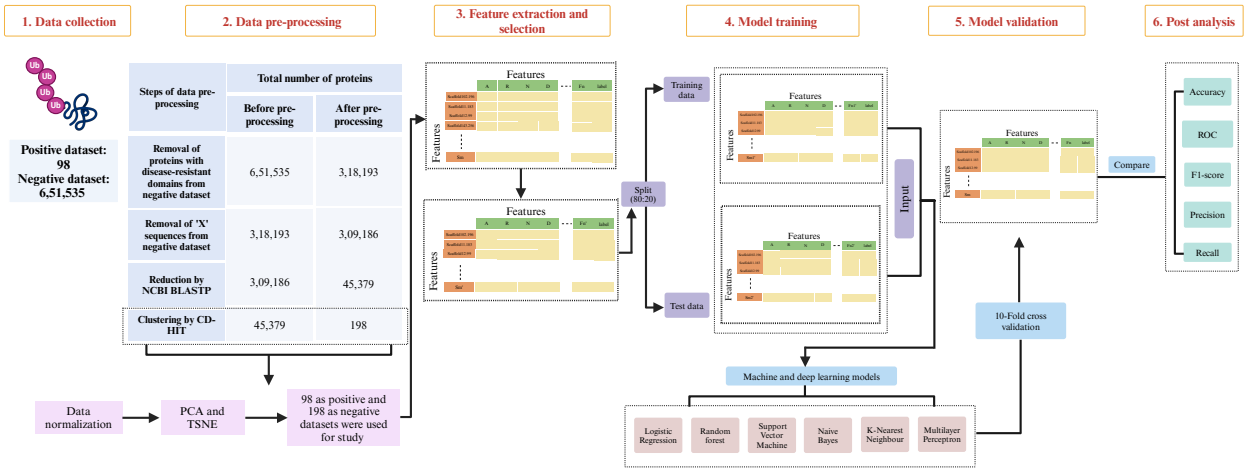


Figure 4.1 : Schematic representation of proposed machine learning and deep learning workflow. It includes proteomics data collection and its pre-processing to ensure data quality. This step involves filtering datasets, normalizing features, and addressing missing or redundant values. Further, relevant features were selected for model building. Finally, the model performance was cross-validated based on various performance metrics in post-analysis.

4.3.2 Protein feature extraction and selection

After pre-processing the data, the protr package of the R program was utilized to extract and select the protein features. Thus, 19,503 protein features were extracted from the internal dataset by the protr package. The features represent the structural, functional, expression, and interaction profiles of the proteins. Further, dimensionality reduction of the extracted features was performed with the top 50 principal components, resulting in an 80.40% variance on the internal dataset. Figure 4.3 (A) depicts the PCA analysis of 50 principal components of the internal dataset, where blue- and orange-coloured dots represent “0” and “1”, respectively, which were non-disease-resistance and resistance proteins in the internal dataset. In addition, the high-dimensional internal dataset through a t-SNE plot, which displayed a two-dimensional map of the selected protein features, was visualized. Figure 4.3 (B) depicts the t-SNE plot that modelled the protein features into a high-dimensional object to three-dimensional space with two components. It showed that similar proteins tend to cluster together, and dissimilar ones were modelled at distant points. Furthermore, the t-SNE analysis on the internal dataset effectively separates proteins into two classes, showing that the protein features in the internal dataset are separable. Subsequently, this dataset was used for building ML and DL model for further analysis.

Furthermore, we have utilized LR, RF, SVM, NB, KNN and MLP models for the identification of the disease-resistance proteins in the internal datasets. However, the ML models, including LR, RF, SVM, NB and KNN, have a few limitations, such as dependence on feature engineering and reduced ability to capture non-linear relationships. The ML models underperform when there are multiple or non-linear decision boundaries. For instance, the SVM model is trickier to tune due to the selection of the right kernel affecting non-linear relationships. In addition, the ML models can effectively predict disease-resistance proteins as they carry straight-line or linear relationships. Moreover, neural networks, including the MLP model, are widely utilized for the linear or real-world relationships, such as the identification of disease-resistance proteins.

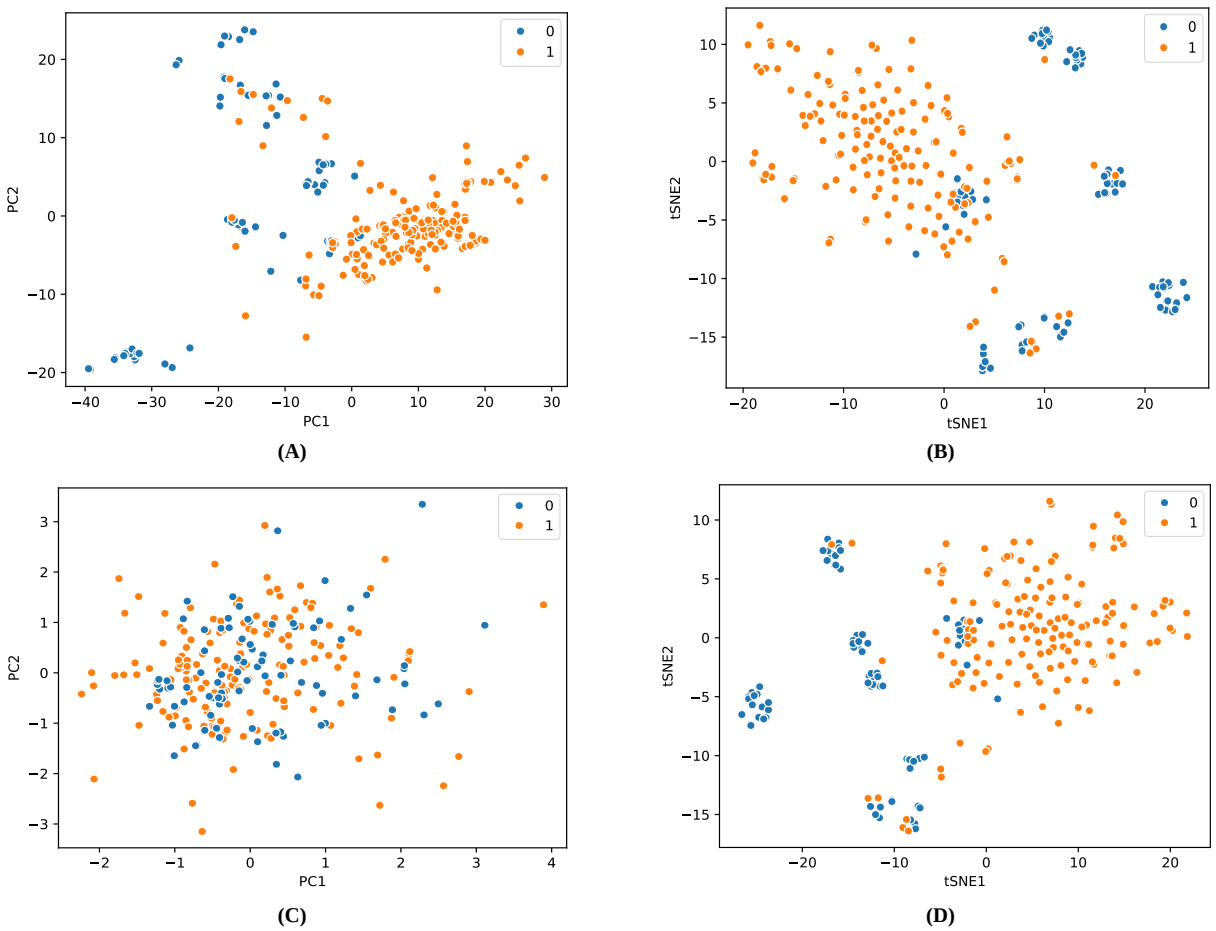


Figure 4.2 : Dimensionality reduction and visualization on both internal and external datasets : (A) and **(B)** display PCA and t-SNE plots for the internal dataset, while **(C)** and **(D)** for the external dataset, respectively. The blue and orange dots correspond to class labels "0" and "1," where "0" represents non-disease-resistance proteins and "1" represents disease-resistance proteins. PCA applied to the 19,503 extracted protein features and showed a clear representation of variance distribution within the dataset. Additionally, t-SNE was employed to project the datasets into a three-dimensional space, enabling a more detailed visualization of the separation between disease-resistance and non-disease-resistance proteins based on the selected features. The t-SNE plots confirmed clustering patterns, indicating the ability of the selected features to discriminate between the two classes effectively.

4.3.3 Evaluation of the ML and DL model performance on the internal dataset

The selected protein features were then used for model building. 80% of the sequences were considered train sets, and the remaining 20% were taken as test sets. Thus, the 98 proteins in the positive dataset were divided into 78 sequences in the train set and 20 in the test set. In addition, the 198 proteins in the negative dataset were split into 157 and 41 in the train and test sets, respectively. The internal dataset was partitioned into train and test sets to receive an equal proportion of the proteins from positive and negative datasets, resulting in no biases. After dividing the internal dataset into train and test sets, well-established ML and DL models were utilized and compared on the test set. The ML models include LR, RF, SVM, NB, and KNN, whereas MLP was the DL model used in the study. Further, the confusion matrix generated by all the models on the test set of the internal dataset is shown in Figure B.1. In addition, detailed information regarding TP, TN, FP, and FN of all the models tested on the test set of the internal dataset is described in Table 4.1.

Evaluation of the overfitting and underfitting status of the ML and DL models is crucial in predicting disease-resistance proteins on the test sets. Therefore, the 10-fold cross-validation method is performed to

analyze the performance of the models in predicting disease-resistance proteins. This method is a commonly used model validation technique where a subset of data is preserved by testing once or repeatedly. Thus, various parameters, including accuracy, AUC, F1-score, precision, and recall, were calculated on the six classifiers. Among all six classifiers, the DL-based MLP model showed the highest mean training and validation accuracies of 91.59% and 96.39%, respectively. In addition, MLP showed the best performance with accuracy, AUC, F1-score, precision, and recall of 94.18%, 97.84%, 96.29%, 98.48% and 94.20%, respectively. This highlighted the robustness and efficiency of the MLP model for identifying disease-resistance proteins in *O. sativa* and related species. Figure 4.4 highlights a comparative analysis of the performance parameters of all the models on the internal dataset. Table 4.1 illustrates the performance evaluation of each model on the internal dataset. Figure B.3 shows the AUC curves for the internal dataset on all the ML and DL models.

We would like to clarify that the results reported in Table 4.1 represent the performance metrics of all six models evaluated on the internal dataset using the 10-fold cross-validation method.

Table 4.1 : Performance matrix comparison of different models on the internal dataset through the 10-fold cross-validation method.

Classifiers	**Performance parameters**									
	TP	**TN**	**FP**	**FN**	**Accuracy (%)**	**AUC (%)**	**F1-score**	**Precision (%)**	**Recall (%)**	**Specificity (%)**
LR	37	18	3	2	84.17	91.05	86.47	92.39	81.86	**86**
RF	40	18	3	2	88.89	95.31	91.39	91.24	92	**88**
SVM	38	17	2	3	79.06	87.36	84.8	90.68	80.39	**86**
NB	38	17	2	3	85.52	92.96	86.84	95.37	80.31	**72**
KNN	36	18	4	2	**90**	**88**	**92.30**	**90**	**94.73**	**89**
MLP	65	16	1	4	**94.18**	**97.03**	**96.29**	**98.48**	**94.20**	**95**

Classifiers: machine and deep learning models including Logistic Regression (LR); Random Forest (RF); Support Vector Machine (SVM); Naive Bayes (NB); K-Nearest Neighbour (KNN); Multilayer perceptron (MLP); Performance parameters: performance metrics of all the models on internal dataset showing their True positive (TP); True negative (TN); False positive (FP) and False negative (FN) values along with their accuracy, Area Under Curve (AUC), F1-score, precision and recall. The metrics were calculated through the 10-fold cross-validation method.

Table 4.2 : Performance matrix comparison of different models on the external dataset by the 10-fold cross-validation method.

Classifiers	**Performance parameters**									
	TP	**TN**	**FP**	**FN**	**Accuracy (%)**	**AUC (%)**	**F1-score**	**Precision (%)**	**Recall (%)**	**Specificity (%)**
LR	**102**	43	96	44	84.18	90.82	92.99	82.89	87.24	**31**
RF	186	7	12	80	88.91	95.82	91.38	92.5	91.67	**37**
SVM	148	16	50	71	79.21	86.1	89.7	78.34	83.15	**24**
NB	112	34	86	53	85.19	93.9	95.51	81.78	87.81	**28**
KNN	184	6	14	81	90.59	90	94.24	92	92.86	**30**
MLP	**181**	**11**	**17**	**76**	**90.91**	**97.03**	**95.15**	**91.49**	**70.42**	**39**

Classifiers: machine and deep learning models including Logistic Regression (LR); Random Forest (RF); Support Vector Machine (SVM); Naive Bayes (NB); K-Nearest Neighbour (KNN); Multilayer perceptron (MLP); Performance parameters: performance metrics of all the models on external dataset showing their True positive (TP); True negative (TN); False positive (FP) and False negative (FN) values along with their accuracy, Area Under Curve (AUC), F1-score, precision and recall and was predicted by the 10-fold cross-validation method.

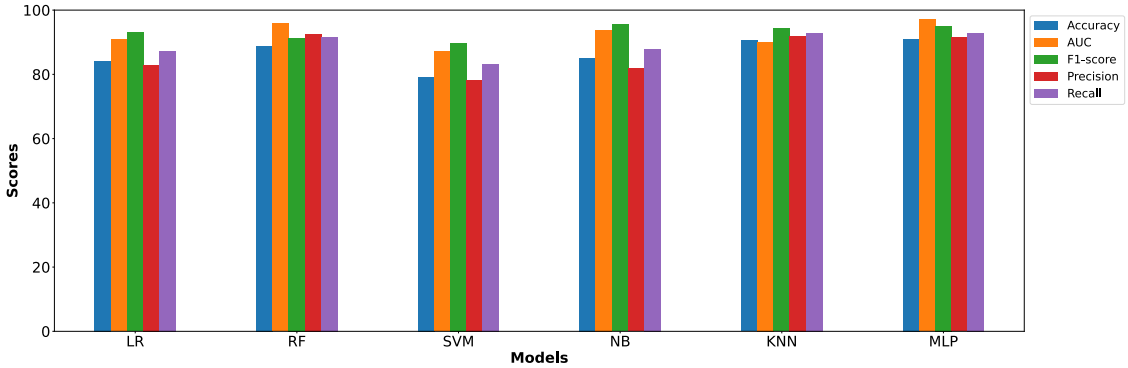


Figure 4.3 : A comparative evaluation of the performance of ML and DL models on the internal dataset. The bar chart provides a visual representation of each model's performance during K-fold cross-validation. The results indicate that the DL-based MLP model outperformed other models, achieving the highest scores across multiple metrics, including accuracy, ROC-AUC, F1-score, precision, and recall. The x-axis represents the model names, while the y-axis corresponds to the range of scores attained by the models.

4.3.4 Evaluation of the ML and DL model performance on the external dataset

To further evaluate and validate the performance of the DL-based MLP model over other ML models, an external dataset, was utilized. The external dataset consists of known disease-resistance proteins as positive and morphological and flowering-related proteins as negative datasets from *O. sativa*. The proteins in the positive and negative datasets were obtained from the rice SNP-Seek database. Further, the data preprocessing on the external dataset only includes the removal of incomplete protein sequences from both the positive and negative datasets. Thus, 286 proteins were considered as an external dataset for feature extraction, selection, normalization and model development. The protr package of the R program was utilized for the protein feature extraction and selection. The normalization of the selected protein features was performed by PCA and t-SNE techniques, with 50 as the principal and 2 as the total number of components, respectively. The variance in PCA analysis was 47.01% with the external dataset. After performing the PCA and t-SNE, the external dataset was divided into 80% train and 20% test sets. However, the train set from the internal dataset was utilized as the train set for the external dataset. The test set obtained from splitting the external dataset was used to predict disease-resistance proteins. In this case, the MLP model will predict the disease-resistance proteins from the external dataset, considering the train set from the internal dataset as its reference. Thus, the performance of the MLP model was tested and validated on the external dataset and, finally, compared with other ML models. Table 4.2 summarizes TP, TN, FP, and FN values for each model on the external dataset. Moreover, with 10-fold cross-validation on the external dataset, the DL-based MLP model outperformed other ML models with the highest accuracy, ROC, F1-score, precision, and recall of 90.91%, 97.03%, 95.5, 91.49%, and 70.42%, respectively. Table 4.2 depicts detailed information on the performance metrics of each model. Thus, the comparative performance metrics obtained by the DL-based MLP model concerning other models reflect that the MLP model robustly and precisely predicts disease-resistance proteins in *O. sativa* and its related species. Moreover, the MLP model has predicted 47 proteins as disease-resistance proteins, summarized in Table A.3. Figure B.2 shows the confusion matrices for each model's external dataset of proteins. A comparative analysis between the DL-based MLP and ML models for the external dataset of the proteins in *O. sativa* is presented in Figure 4.4.

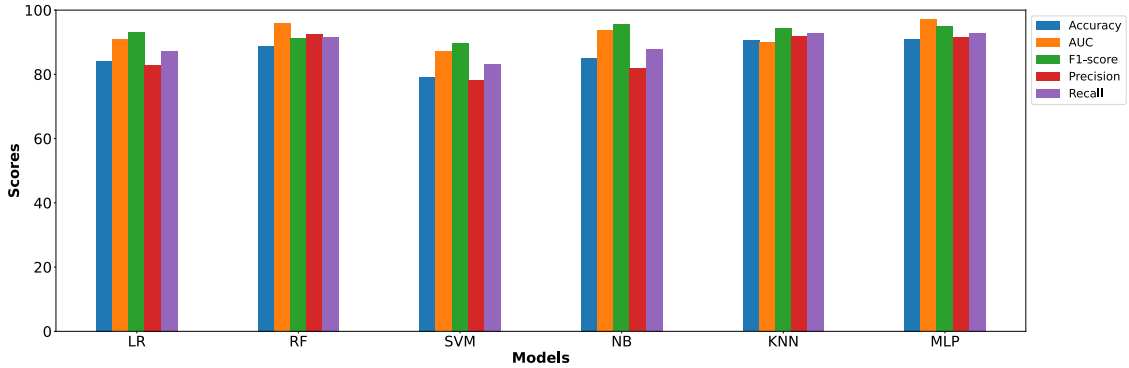


Figure 4.4 : Comparative assessment of the DL-based MLP model with other ML models on the external dataset. Comparison of the performance metrics, i.e., accuracy, ROC, F1-score, precision, and recall for each model predicting disease-resistance proteins from the new dataset. The graphical representation of the performance of the models shows the outperformance of the DL-based MLP model against other models with positive scores even when tested on an entirely new set of proteins.

4.3.5 Comparison of the DL-based MLP model performance with existing models to predict disease-resistance proteins

A computational framework, including integration, testing, validation and comparison of the well-established MLP with ML models to predict disease-resistance proteins in *O. sativa* and its related species were designed. To test, validate, and test the MLP with various ML models, internal and external datasets were used. The DL-based MLP model has robustly and efficiently predicted the disease-resistance proteins. Therefore, to further validate the performance of the DL-based MLP model, it was compared with existing models based on sequence similarity. Table 4.3 shows the performance comparison of the MLP model with the known models. The results showed that on the external dataset, the F1-score of the MLP model was the highest as compared to the LRRpredictor, LRRsearch, LRRfinder, Prosite, DeepLRR, RD-RFPDR, PSI-BLAST, RRG Predictor and Pfam. However, the F1-score of the MLP model was slightly lower than that of the DRAGO tool, which indicates that the MLP model is also a robust model for predicting disease-resistance proteins. In addition, the precision percentage was highest in RD-RFPDR, PSI-BLAST and DRAGO2, in contrast to the MLP model. Despite this, the RD-RFPDR model is based on two parameters, which include model development only through the NLR and PRR proteins and through the RF model. It has been reported that plants have eight classes of disease-resistance genes to provide immunity against various pathogens ([Kumar *et al.*, 2018]). The RFPDR model development only considers NLR and PRR proteins, which do not have a broader scope of predicting disease-resistance genes or proteins. In addition, the PSI-BLAST is an iterated BLAST program that is sequence-specific, derived from a profile of the multiple sequence alignments of the sequences. However, the PSI-BLAST program is more sensitive than the BLAST program and is time-consuming in each iteration as it gives a threshold score using protein-protein BLAST. Therefore, this can generate issues when a large protein sequence is provided to predict the presence of the disease-resistance proteins. Furthermore, the F1-score and precision of LRR-specific methods, including LRRpredictor, LRRsearch, LRRfinder and DeepLRR, were lower than the studied MLP model. This showed that the LRR-specific methods were less efficient in identifying disease-resistance proteins than the MLP model. The RRGPredictor depicted a higher F1-score and precision than MLP. However, it requires a Tab Separated Values File (TSV) from InterProScan as input, which can be a limiting factor in identifying and classifying these PRRs and R proteins. On the other hand, Pfam can identify disease-resistance proteins, but it is less efficient than MLP due to its lower F1-score and precision. The DRAGO2 tool includes predicting disease-resistance proteins from a few classes out of eight. This will provide a limited knowledge of the disease-resistance proteins in *O. sativa* and its related species. Therefore, the MLP model offers a more generalized and scalable approach, making it a powerful tool for disease-resistance protein prediction in *O.*

sativa and related species. These results affirm that MLP outperforms existing sequence similarity-based methods in both predictive accuracy and practical usability.

Table 4.3 : Comparison of performance metrics between MLP model and other existing models on internal and external datasets.

Models	Performance parameters	
	F1-score	**Precision (in %)**
LRRpredictor	69.2	58.2
LRRsearch	73.9	67.6
LRRfinder	72.8	79.8
Prosite	52.2	83.6
DeepLRR	76.3	74.4
RD-RFPDR	95.3	100
PSI-BLAST	78.1	100
RRG Predictor	96.6	98.8
DRAGO2	97.1	100
Pfam	6.2	19.2
DL-based MLP (Internal dataset)	93.71	95.65
DL-based MLP (External dataset)	97.03	91.49

Models: well-established models used for the comparison with the Multilayer perceptron (MLP) model i.e., Leucine-rich-repeats predictor (LRRpredictor); Leucine-rich-repeats search (LRRsearch); Leucine-rich-repeats finder (LRRfinder); Protein database (Prosite); Deep Leucine-rich-repeats (DeepLRR); Random Forest for Plant Disease Resistance (RFPDR); Position-Specific Iterative Basic Local Alignment Search Tool (PSI-BLAST); Resistance Gene predictor (RRG Predictor); Disease Resistance Analysis and Gene Orthology (DRAGO2); and protein families database (Pfam); Performance parameters: performance parameters of all the models, including F1-score and precision.

4.3.6 Discussion

O. sativa is an essential agricultural food source and serves as a nutritional grain for above half of the world's population. Its consumption is expected to increase because of the expanding global population. Additionally, *O. sativa* production is threatened by various biotic stresses, such as bacteria, fungi, viruses, and nematodes, compromising crop yield and quality. However, it consists of a well-established immune system composed of PTI and ETI to counter the invasion of pathogens. Various disease-resistance proteins, such as R-proteins and PRRs, control the immune system. Thus, the identification of disease-resistance proteins not only helps reduce pesticide application to fields but also improves the yield of *O. sativa*. The development of disease resistance in *O. sativa* has always been a primary goal for breeding programs, and early detection of these proteins is a major step toward this goal.

To achieve this goal, computational approaches, including ML and DL, have been extensively used over the traditional methods. Recently, a novel method of diagnosis of BB-infected leaves by brown spot and leaf smut diseases has been developed based on ML and DL-driven methods. It showed that DL-based segmentation with bitwise logical AND operation and DL-based methods were more robust than SVM [Sharma *et al.*, 2023]. Another study analysed different ML algorithms, including RF, SVM, and partial least squares discriminant analysis (PLS-DA), and one DL model, i.e., CNN, was performed on one and two-dimensional spectral images. CNN model outperformed other ML models in terahertz imaging technology, providing great potential in rapidly identifying BB disease-resistant *O. sativa* seeds to develop BB-resistant varieties [Zhang *et al.*, 2020]. Although computational methods can identify disease-resistance proteins, only some consider sequence-derived structural and physicochemical features of proteins. As disease-resistance proteins are crucial in breeding programs, their rapid and accurate prediction is required. Therefore, this study aims to develop a model through a comparative study of six different types of ML and

DL classifiers, i.e., LR, RF, SVM, NB, KNN, and MLP. The best model was selected and validated based on various performance measures. It was identified that the DL-based MLP model outperformed other ML models and was regularised through different model parameters.

By utilizing ML and DL algorithms, researchers can precisely identify the genes or quantitative trait loci underlying the most advantageous traits of plants according to the specific requirements of plant breeders [Najafabadi *et al.*, 2023]. In our study, several ML models including LR, RF, SVM, NB and KNN were utilized for the identification of disease-resistance proteins. In addition, the LR model performs binary classification and predicts the probability of the occurrence of disease-resistance proteins and determines the likelihood of the proteins belonging to a particular class [Che *et al.*, 2025]. The RF model is based on decision trees, and it fits multiple trees to different subsets of the dataset [Liu *et al.*, 2024b]. It predicts disease-resistance proteins from each decision tree and increases prediction accuracy through a voting mechanism. Further, the SVM is an ML model that identifies the best hyperplane to separate the samples into classes [Che *et al.*, 2025]. It classifies and predicts disease-resistance proteins based on a hyperplane and sorts them into classes. Furthermore, the NB is a supervised ML model that utilizes Bayes theorem. It classifies the proteins based on the set of features from the dataset and determines the probability of each feature occurring in each class within the data [Morgan *et al.*, 2021b]. KNN model uses proximity and predicts a new data point based on the average value of its 'K' nearest neighbours [Che *et al.*, 2025]. It predicts disease-resistance proteins by assigning labels on the basis of 'K' values. On the other hand, we have utilized the MLP model, a DL-based model for the prediction of disease-resistance proteins. It consists of several layers that propagate feature data in a feed-forward manner through the layers. It predicted the disease-resistance proteins by processing the inputs from previous layers, which allows the information flow in one direction, i.e., to the output layer. For the current study, firstly, an internal dataset, including known disease-resistance proteins as the positive and the entire proteome as the negative datasets, respectively, was collected for 13 species related to *O. sativa*. Next, the data pre-processing steps on the negative dataset were performed to balance the size of both datasets for further analysis. These steps include the removal of proteins with disease-resistant domains and 'X' sequences. In addition, clustering was performed to reduce the redundancy in the negative dataset. Further, feature extraction and selection were carried out on various features of the proteins. The performance of the models through the confusion matrix showed that the MLP model has the highest TP values. Moreover, the accuracy, AUC, F1 score, precision, and recall scores in the 10-fold cross-validation showed that the DL-based MLP model has the best performance with values of 91.59%, 97.84%, 93.77%, 95.65%, and 92.5%, respectively. These results depict that the MLP model is the most efficient in predicting disease-resistance proteins in *O. sativa* and its related species compared to other ML models. The ML and DL models were also assessed and compared on an external dataset of the 286 proteins. These proteins are involved in disease resistance mechanisms and morphological and floral development in *O. sativa*. Thus, on the external dataset, the MLP model showed the best performance in predicting disease-resistance proteins compared to the applied ML models. Therefore, based on the promising model assessments, the robust MLP model emerged as the best model for identifying disease-resistance proteins, providing insight into these underlying mechanisms. Furthermore, the MLP model consists of several hidden layers underneath it and is referred to as a deep neural network. The model is based on a feed-forward neural network in which each node is linked to the next. Additionally, it has several strengths, resulting in the best performance of the MLP classifier over the ML models. For example, it can acquire a non-linear relationship between inputs and work best on large datasets for faster predictions. Further, it provides high accuracy scores on smaller datasets as well. The DL-based MLP model utilizes a supervised learning technique, i.e., the backpropagation technique, to develop weight values that correctly predict the outcome [Sarkar *et al.*, 2023]. The backpropagation technique is the fundamental principle of the MLP model by which it can efficiently classify disease-resistance proteins and outperform other ML models. Moreover, the MLP model is a superior model over ML models due to its simplicity, short training times and backpropagation property. Moreover, a Densely Convolution Neural Network (DenseNet) with MLP (Dense Net169-MLP) was developed to divide *O. sativa* disease into three classes, i.e., BB, brown spot, and leaf smut. The MLP model showed the highest sensitivity, specificity, precision, accuracy, and F1-score over other methods. The Dense Net169-MLP can be applied for the identification and classification of *O. sativa* diseases [Narmadha *et al.*, 2022]. A recent

study has reviewed the identification of RB and brown spot diseases in *O. sativa* leaves by the MLP model. The results proved that the MLP classifier accurately classified 89% of the images [Manavalan, 2020].

4.3.7 Conclusion

In summary, the current study accurately and precisely identified disease-resistance proteins of *O. sativa* and its related species through the MLP model based on sequence-derived structural and physico-chemical features of proteins. Thus, it predicts the disease-resistance proteins with the highest scores in internal and external datasets with various performance metrics. This model will help breeders, as well as biotechnologists, to detect disease-resistance proteins rapidly. The identified proteins can then be used to develop disease-resistant varieties of *O. sativa* in less time.