IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON FEATURE REPRESENTATION AND LEARNING METHODS WITH APPLICATIONS IN LARGE-SCALE BIOLOGICAL SEQUENCE ANALYSIS

Received February 14, 2020, accepted March 4, 2020, date of publication March 23, 2020, date of current version April 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2982410

FunEffector-Pred: Identification of Fungi Effector by Activate Learning and Genetic Algorithm Sampling of Imbalanced Data

CHAO WANG[1], PINGPING WANG[2], SHUGUANG HAN[3], LIDA WANG[4], YUMING ZHAO[5], AND LIRAN JUAN[2]

[1]Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 610054, China
[2]School of Life Science and Technology, Harbin Institute of Technology, Harbin 150001, China
[3]School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu 610054, China
[4]Scientific Research Department, Heilongjiang Agricultural Reclamation General Hospital, Harbin 150088, China
[5]Information and Computer Engineering College, Northeast Forestry University, Harbin 150040, China

Corresponding authors: Lida Wang (427334@qq.com), Yuming Zhao (zym@nefu.edu.cn), and Liran Juan (lrjuan@hit.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 31601072, Grant 61971117, and Grant 61601110.

ABSTRACT Fungal pathogens have evolved the ability to cause serious plant diseases and threaten the world food security. Fungal effectors are proteins that exploit the host cellular functions to facilitate infection. Effector identification is crucial for disease control in crops and to understand plant-pathogen interactions. However, fungal effector identification has been challenging as most fungal effectors lack of consensus motifs and data imbalance problem. In this study, a fungal effector predictor was designed to effectively learn from an imbalanced dataset. A granular support vector-based under-sampling (GSV-US) strategy combined with a genetic algorithm was used for majority class sampling. When evaluating on an independent test dataset, the FunEffector-Pred significantly outperformed the existing predictors for fungal effector identification. Several informative feature patterns, such as the patterns of Ile, Gly, Val, Leu and Thr, as well as the combination of aromatic amino acids with positively-charged amino acids, are reported for fungal effector identification for the first time.

INDEX TERMS Activate learning, fungal effector, genetic algorithms, imbalanced data.

I. INTRODUCTION

Fungi and oomycete pathogens that have evolved the ability to cause serious plant diseases and threaten the world food security. Grain stocks capable of feeding more than 600 million people have been destroyed by five major fungal species [1]. Effector molecules, usually proteins, are important virulence factors of plant pathogenic fungi. Effectors are secreted from plant-associated organisms to the host apoplast or cytoplasm, which then alter circumvent host defense and modulate host-cell physiology [2]–[4]. Effector identification is crucial for experimental validation, crop resistance development [5] and disease control strategies [6]. Effector prediction mainly relies on bioinformatics; for example, bacterial effectors were predicted based on N-terminal sequence signal information related to specialized secretion pathways, such as the type III secretion system [7]. In oomycetes, consensus N-terminal sequence motifs, such as RXLR, can be used for effector prediction [8]. However, the effector delivery systems of fungal pathogens are diverse; in general, fungal effectors lack consensus motifs and do not share obvious sequence similarity with each other [8].

Given the lack of unifying sequence-based features, fungal effector prediction methods are based on relatively broad criteria, i.e., presence of a secretion signal, small protein size, cysteine residue richness and genome location [3], [6], [8]. However, these generalized properties risk excluding genuine effectors that are large and cysteine-poor as reported or misidentifying proteins with small size and high cysteine content as effectors [9]. Instead of setting predefined criteria, an alternative approach is modelling of protein properties that are crucial for effector prediction from positive and negative samples using machine learning (ML). In this regard, EffectorP [3] was the first ML classifier of fungal effectors

The associate editor coordinating the review of this manuscript and approving it for publication was Leyi Wei.

that employs as an alternative tool to simple criterions such as protein size and signal information. An upgraded version, EffectorP 2.0 [6], was provided.

Although great progress has been made by ML-based methods for fungal effector prediction, the following challenges remain. First, the ML approach relies on fungal effectors and secreted non-effectors as positive and negative training sets, respectively. Where 21,840 proteins were used as a negative sample for EffectorP2.0 [6] model training, however, only 97 fungal effectors that were experimentally validated were served as the positive sample. This resulted in an inevitable data imbalance problem. Numerous studies [10]–[25] have shown that the ML algorithms, such as random forest (RF) and support vector machines (SVMs), tend to be biased in favor of the minority class, while the minority class samples are more likely to be misclassified. For example, the SVM is one of the commonly used ML algorithms [26]–[32] and usually results robust performance on balanced data, but it fails to achieve convincing results for imbalanced data because of ''support-vectors-dependency'' of the SVM [11]. The behave of the SVM-based classifier is determined by the support vectors (SVs) that are near the separating hyperplane [10]. The corresponding hyperplane may be biased toward the minority class (usually the positive class), which tends to predict the minority class as the majority ones [11]. Second, features used for ML in EffectorP are mainly based on physicochemical properties of proteins, such as the amino acid composition, amino acid classes, molecular weight, protein instability index, hydropathicity, etc [3], [6]. A good feature representation algorithm has been demonstrated to be a great aid to the prediction accuracy [33]–[39]. For example, the composition of k-spaced amino acid pairs capture residue correlation information [40]; pseudo amino acid composition (PseAAC) algorithm that merges sequence-order information with compositional features [41]. Moreover, a variety of other feature descriptors have been developed for protein feature representation [26], [42]–[54].

With regard to imbalanced data, many imbalance learning techniques have been proposed in recent years. Among various methods, random resample-based methods represent the basic techniques. The random over-sampling method duplicates the minority classes instance, which may lead to overfitting and computational costs [10]; random under-sampling may discard informative samples and therefore decrease the prediction accuracy of the model. In addition to the above resampling-based method, other learning methods have also shown impressive performance in several studies, such as active learning strategies [10], [12], [55] and granular computing strategies [56]. Intrinsically, the two types of methods are based on the SVs of the SVM. Ertekin *et al.* [10] first proposed a skew-insensitive active learning solution, which starts with a small number of randomly chosen samples (T_{start}) of the training set and then iteratively selects and adds the most relevant SVs from the SVM model trained on the remaining training set to the T_{start}. Zięba and Tomczak [12] proposed a boosted SVM strategy that achieves better estimation of misclassification costs, especially when the center of the majority class is far from the SVM separating hyperplane. Tang *et al.* [56] proposed a granular SVM-repetitive under-sampling (GSVM-RU) algorithm; the negative support vectors (NSVs) of the SVM trained on the original imbalanced dataset are extracted and then these NSVs are removed from the training set to obtain a smaller set for which a new SVM is trained. Then, the related NSVs are iteratively extracted. All of these NSVs are combined and supplied for the next granule fusion procedure. Zhu *et al.* [11] used an ensembled hyperplane-distance-based support vector machines (E-HDSVM) algorithm. This method iteratively deletes a constant number of NSVs from the training set and the remaining training set is used as one of subsets for ensemble analyses.

Although active learning and granular computing strategies have shown excellent results for imbalanced datasets, there still some issues need to be further enhanced and optimized. For instance, in terms of the active learning strategy, the selected SVs are used to reconst the majority class, but they cannot guarantee that the number of the SVs will be balanced with the minority class; granular computing strategies combines the all NSVs or keep the remaining samples after removal of part of the NSVs, which does not always maintain a ratio between positive and negative groups. Genetic algorithms (GAs) [57] are parallel and global search heurists that mimic the natural selection process, where the best individual/groups survive and evolve in the following generations. Global optimization is performed via a number of genetic operators, e.g., reproduction, mutation and crossover. GAs are more commonly used for samples selection [58] as GAs have succeed in many combinational optimization problems [59], [60], including the imbalanced problems related to biological sequences [61], [62].

Keeping these problems in mind, we here developed a ML method for fungal effector prediction. First, eight feature groups containing 25 feature descriptors were selected to encode each protein sequences. Second, an active learning strategy was employed to extract SVs of the majority class. Third, a genetic algorithm strategy was used for majority class sampling. Combining the aforementioned strategies, a novel tool called FunEffector-Pred was developed. In an evaluation using independent test datasets, FunEffector-Pred significantly outperformed the existing predictors for fungal effectors.

II. MATERIALS AND METHODS

A. DATA COLLECTION AND PREPROCESSING

To train models for fungal effector prediction, positive training sets were collected from experimentally validated effectors. Benchmark datasets containing 94 effectors of 23 species used by EffectorP2.0 [6] were collected; furthermore, 87 effectors reported in peer reviewed articles were collected from the pathogen–host interactions database (PHI-base) [9]. The two datasets were combined, and the

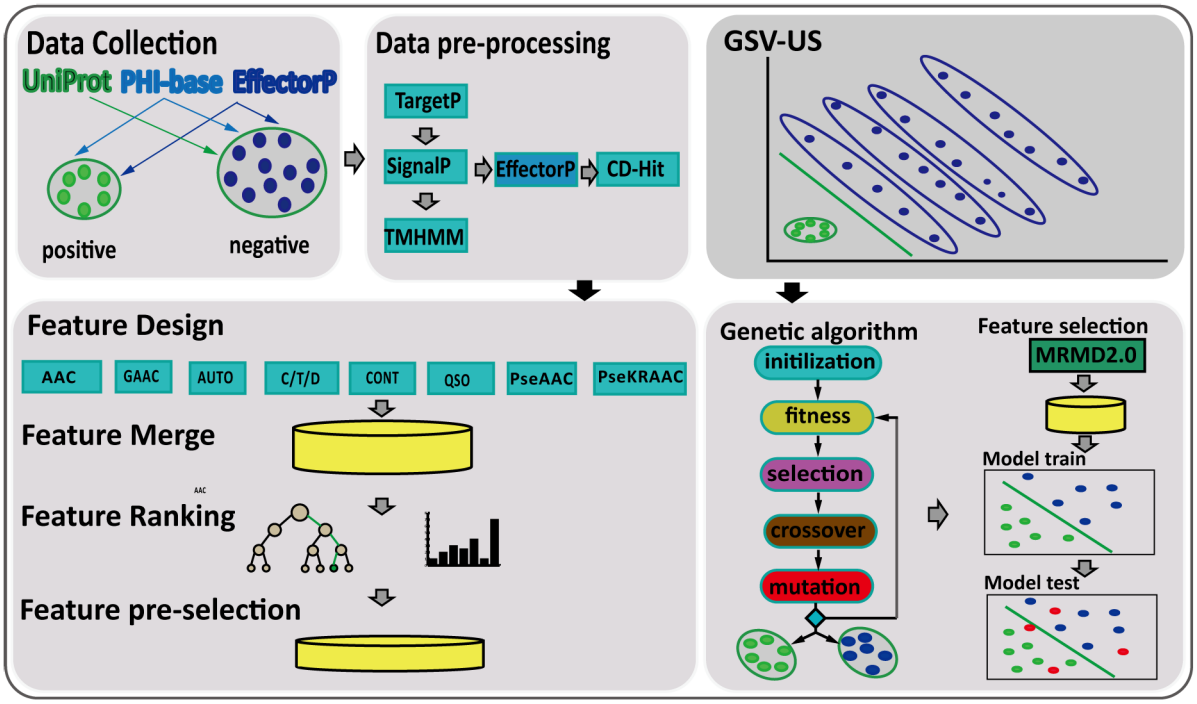

FIGURE 1. Framework of granular support vector-based under-sampling (GSV-US) and genetic algorithm strategy.

sequence homology was reduced by CD-HIT software [63] setting the threshold of 40% similarity [64]. 117 proteins with low sequence similarity were retained as a positive training set. The negative training set was collected from three different subsets, which included 38,988 secreted fungal proteins from UniProt database [65], 142 fungal proteins that were annotated as having unaffected pathogenicity from the PHI-base, and 24,432 secreted proteins of 27 saprophytes that were not pathogenic in plants [6]. To eliminate redundancy, homology was reduced in the three subsets by CH-HIT software using a threshold of 40%; these proteins were further confirmed to be secreted/extracellular by SignaplP 5.0 [66] and TargetP2.0 [67], and TMHMM [68] was used to verify that the proteins had no transmembrane domain, as recommended [3]. To filter out the undiscovered effectors in above three subsets, potential fungal effectors predicted by the EffectorP2.0 software were removed as described previously [6]. Finally, 7,458 proteins were obtained and included in the negative training dataset.

B. FEATURE EXTRACTION STRATEGY

To build an effective prediction tool, sufficient information should be incorporated into the model, especially hidden features in protein sequences [69]–[72]. In this work, eight feature groups containing 25 feature descriptors were used to formulate the protein sequences. The feature groups included amino acid composition groups, grouped amino acid composition groups, binary groups, autocorrelation groups, C/T/D groups, conjoint triad groups, quasi-sequence-order groups, PseAAC groups, and pseudo k-tuple reduced amino acid composition groups (PseKRAAC) (Fig. 1). The 25 feature encoding algorithms are provided in detail in the Supplementary methods; each feature was generated by using a package of comprehensive protein feature generation tools, including 188D [45], Pse-in-One [73], iFeature [42] and pyFeat [43], or custom Python scripts.

C. FRAMEWORK OF ACTIVE LEARNING

1) FEATURE PRE-SELECTION AND SUPPORT VECTOR MACHINES

The 25 feature descriptors generated a feature vector with 35,394 dimensions for each protein sequence. To reduce the computation and simplify the model, the high dimensional feature pool was first filtered by an RF classifier [74]–[77] with a less stringent standard. RF is an ensemble method that consists of a number of decision trees, a voting strategy is utilized for the final outcome prediction. During the decision procedure, the features are ranked based on their importance. This strategy has been widely embedded in feature selection methods and shows excellent performance [78]–[80]. In this study, 117 proteins were randomly selected from negative training dataset consisting of 7,458 proteins and combined with the 117 positive protein samples, and this set was trained using an RF classifier with a 5-fold cross validation (CV). The feature importance value was predicted for each CV, and features with a value > 0.001 were recorded. This procedure was iterated 100 times. Finally, all recorded features were retained for the subsequent steps. Note that this feature selection step was performed to filter out the extremely sparse

features, instead of using the final feature optimization, which is described in section 2.3.4.

2) GRANULAR SUPPORT VECTOR-BASED UNDER-SAMPLING ALGORITHM

Granular computing represents information as aggregates such as subsets or clusters of a universe [56]. Here, a negative granule, denoted as NG, is defined as the SVs of SVM model, which are identified based on their distance from the hyperplane. First, an original SVM, denoted as SVM_0, was trained on the imbalanced dataset (TDS_{t0}) including all negative protein samples and positive dataset. Then, K negative SVs, denoted as NG_1, which have the shortest distance to the separating hyperplane in SVM_0, were predicted and extracted from the TDS_{t0}. Second, the NG_1 was removed from the TDS_{t0}, and the remaining negative training set (TDS_{t1}) was used as a training dataset in the next iteration. This procedure was iterated 700 times. To avoid the ideal-hyperplane-missing phenomenon (refer to [33] for details), we set the numbers of extracted NSVs, namely the value of K, to 10 in each iteration. Finally, a balanced negative granular support vector dataset, denoted as BNGSV, was generated, in which $BNGSV = [NG_1, NG_2, \ldots, NG_{700}]$, and each NG_i ($i = 1, 2, \ldots, 700$) contained 10 negative SVs extracted at the i-th iteration.

The performances of the granular support vector-based under-sampling (GSV-US) was evaluated as follows. We adopted the sliding window techniques to generate subsets from the BNGSV; the number of window size was set to 12, and the sliding length was set to 1. The first negative granular subset was $NGS_1 = [NG_1, NG_2, \ldots, NG_{12}]$, and the second subset was $NGS_2 = [NG_2, NG_3, \ldots, NG_{13}]$ and so forth. Each of the subset, NGS_i, was integrated with the positive training dataset to generate a fused training dataset, denoted as FTD_i. The FTD_i was used to train the SVM model, and then the trained model was evaluated using an independent test dataset (ITD) as described in section 2.3.4. A consistent number of negatives samples were randomly selected from ITD1 and combined it with ITD2 to constitute a balanced ITD dataset, and the performance was evaluated with 10-fold cross-validation. We repeated the procedure 20 times.

3) GENETIC ALGORITHM SAMPLING FOR IMBALANCED DATA BASED ON THE SVS

GAs, inspired by Darwin's evolution theory, mimic biological natural selection and natural genetics. A GA is an iterative process that begins with a constant number of populations (chromosomes) as ancestors. During each iteration (generation), each chromosome is evaluated with a value of fitness that is a user-defined function. Then, three genetic operators, namely selection, crossover and mutation, are used to generate new populations (offspring) [81]–[83]. After the termination condition is meet, better, new populations are formed and kept as part of the desired balanced dataset [58], [60], [84].

In this study, the BNGSV is used as the initial dataset, and the length of a chromosome (the number of genes) is equal to the protein number of the minority class. The gene index in chromosomes is encoded as a binary string. For a pair of parents selected from chromosome populations, a two-point crossover operation is performed to create offspring. The two binary strings bits of parents are divided into three segments by two crossover points that are randomly generated, and the middle segment is exchanged with each other. To prevent the algorithm from prematurely stopping [58], the mutation operator changes a random bit value in a selected string with a low probability, which was set to 0.003 in this study. The selection operator is used to selecting surviving individuals with better fitness from current populations, where chromosomes possessing higher fitness are more likely to be chosen. A stochastic tournament selection [85] operator was used in this study, and the number of tournaments was set to 3.

D. MODEL COMPARISON WITH INDEPENDENT TESTING DATA

The proposed predictor was compared with the 2 other predictors, EffectorP1.0 and EffectorP2.0. As the EffectorP model cannot be re-trained by a customized dataset, ITDs were used for the model comparison. ITD1 was derived from the literature work [6] for the purpose of model comparison and was composed of 2,000 eukaryotic (fungal, plant and mammalian) proteins with signal peptide. ITD2 was composed of 42 fungal effectors that were removed based on homology as described in section 2.1.

In this study, five commonly used metrics were used for model evaluation, including accuracy (ACC), precision (PRE), sensitivity (SEN) and mathew's correlation coefficient (MCC) and the AUC [52], [86]–[93]. They are formulated as follows:

$$\mathrm{ACC} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$\mathrm{SEN} = \frac{TP}{TP + FN} \tag{2}$$

$$\mathrm{PRE} = \frac{TP}{TP + FP} \tag{3}$$

$$\mathrm{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{4}$$

The AUC means the area under the receiver operating characteristic (ROC) curve, which is calculated by the true positive rate (TPR) and the false positive rate (FPR) by varying the threshold, the TPR and the FPT are calculated as follows:

$$\mathrm{TPR} = \frac{TP}{TP + FN} \tag{5}$$

$$\mathrm{FPR} = \frac{FP}{TN + FP} \tag{6}$$

where TP = true positive, FP = false positive, FN = false negative, TN = true negative.

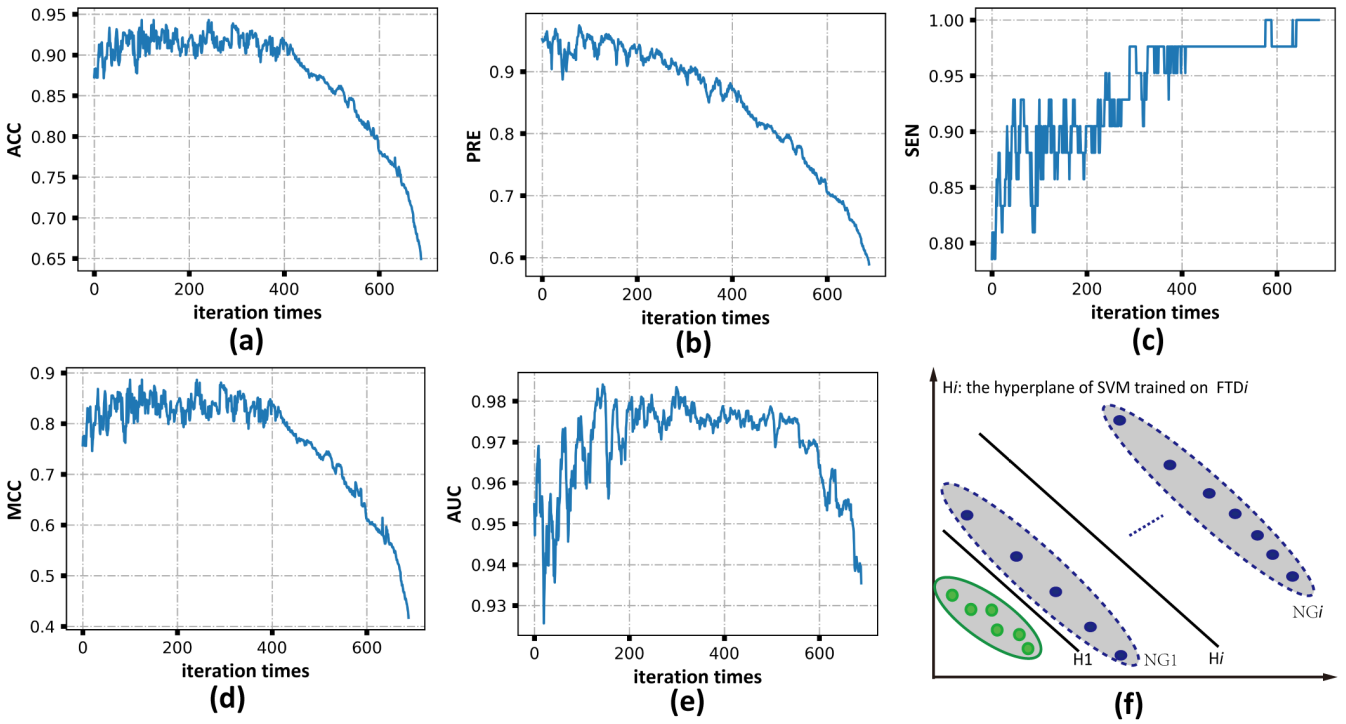

FIGURE 2. The ACC, PRE, MCC and AUC value for GSV-US.

III. RESULTS AND DISCUSSION

A. PROCESSING OF DATASETS, COMBINATION OF FEATURES AND PRE-SELECTION

Positive training samples were collected from the experimentally validated effectors that were used by other prediction tools or from the PHI-base. Negative training samples were collected from three different data sources and subjected to secretion prediction, effector screening and redundancy elimination. Finally, 117 protein samples that constituted the positive training dataset and 7,458 proteins that constituted the negative training dataset were included. Eight feature groups containing 25 feature descriptors were used to formulate the protein sequences, generating a feature vector with 35,394 dimensions for each sample. The high dimensional feature pool was first filtered by an RF classifier with a less stringent standard (feature importance value >0.001). Finally, 15378 recorded features were retained. Note that this feature selection step was performed to filter out the extremely sparse features, instead of using as final feature optimization.

B. GRANULAR SUPPORT VECTOR-BASED UNDER-SAMPLING

NGs were identified from the SVM model, and 10 NSVs were extracted in each iteration. Finally, a balanced negative granular support vector dataset is generated, $BNGSV = [NG_1, NG_2, \ldots, NG_{700}]$. Intrinsically, the negative training samples were ranked in ascending order according to their distance from the hyperplane, in which NG_1 is the closest sample set and NG_{700} is the furthest sample set. To measure the performance of the GSV-US method, each fused training dataset FTD_i ($i = 1, 2, \ldots, 688$) was trained on the SVM model and evaluated on independent test datasets (refer to section 2.3.2 for details). Fig. 2 illustrates the value of corresponding metrics versus the FTD_i. The ACC value varied from 0.87 to 0.95 in the first 400 FTD_i subsets ($i < 400$), and it dramatically decreased when the value of i was larger than 400 (Fig. 2(a)). The findings were similar for the MCC value (Fig. 2(d)). The PRE value gradually decreased as i increased (Fig. 2(b)). The AUC value first increased ($i < 180$) then varied among 0.97∼0.98 ($180 < i < 580$) and then finally decreased (Fig. 2(e)).

Fig. 2 shows that the SEN value is roughly increased with i (Fig. 2(c)), while the PRE value is decreased with i. Equations (2) and (3) indicate that the SEN and the SPE metrics measure the predictive ability of a model in positive cases, but they have different focuses. An increase in the SEN value indicates that fewer true positive samples are misclassified. An SVM model trained on FTD that with a larger i is capable of finding more the positive samples. This can be explained as follows. Initially, (i.e., $i = 1$) the NGS is composed of the NGs that are closest to the positive samples. Therefore, the distance between the hyperplane and training datasets is relatively small (Fig. 2(f)), and the positive samples are more likely to be misclassified as negative. When i increases, the hyperplane moves relatively farther from positives samples, and the distance between the hyperplane and training datasets is gradually increased, which results

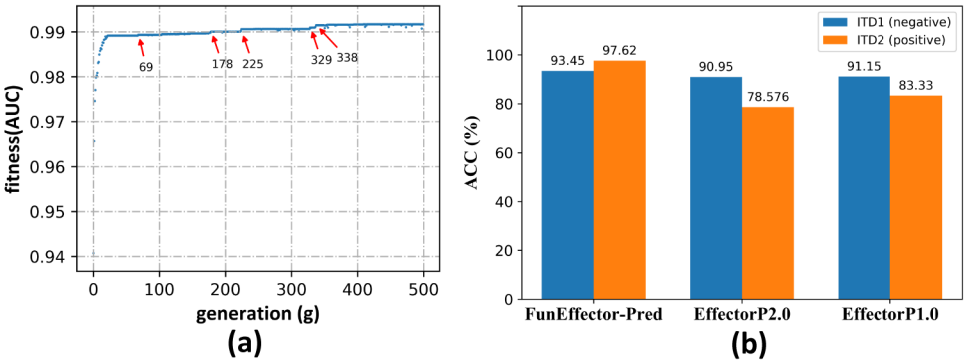

FIGURE 3. Fitness of genetic algorithm with respect to generations and model comparison with independent test data.

in positive samples likely being correctly classified. As a result, the SEN value is roughly increases with i. The PRE value measures the probability that actual positive samples are predicted as positive. A decrease of PRE value for an SVM model trained on an FTD that with a larger i, especially when i is larger than 400, will result in more negative samples being misclassified as positive.

The ACC, AUC and MCC evaluate overall performance of the model, and their values are significantly decrease when i becomes larger, as described above, suggesting that these NSVs are noisy samples and, hence, not beneficial for the improvement of the model. Furthermore, random searching of all negative samples requires increased computational resources and is time consuming. Therefore, the first 4,00 NGS, denoted as NGS-GA, were used for GA-based negative protein under-sampling.

C. UNDER SAMPLING USING A GENETIC ALGORITHM

To eliminate redundant NSVs and achieve an optimal combination among individuals of NGS-GA, the GA-US method was performed. The number of populations was set to 35, the chromosome length and the number of generations (g) were set to 117 and 500, respectively (refer to section 2.3.3 for details). Fig. 3(a) shows the change in fitness (AUC) verses g. Each point represents the average AUC of 35 chromosomes. The AUC values obviously increased from 0.9407 to 0.9892 in the first 23 generations, however, once g becomes greater than 23, the AUC value remained unchanged for most of the generations.Several ''jumping points'' (indicated by read arrows in Fig. 3(a)) were observed where the AUC value obviously increased, i.e., at generations of 69, 178, 225, 329 and 338. Finally, the maximum AUC value reached 0.9917 at the 500th generation. Therefore, the genes namely NSVs, retained in the final offspring were kept as the desired under-sampled negative training dataset, denoted as US-NTD, for the final model build. Thus, the balanced training dataset, denoted as BTD, was composed of the US-NTD and the positive training dataset.

To examine the efficiency of the GA-US method, we compared it with random under-sampling and un-resampling methods. To examine random under-sampling, we randomly selected the same number of negatives samples as positive samples and repeated the procedure 20 times. The performance was evaluated with 10-fold CV. The results are presents in Table 1. The highest PRE, MCC SEN values were achieved by the GA-US method. Note that although the model trained on original imbalanced dataset achieved the highest ACC value, the other 4 metrics of the model were very low because of the imbalance issue. Hence, the GA-US method resulted in the prediction performance.

D. IMPROVING PREDICTIVE PERFORMANCE USING FEATURE SELECTION

Next, to reduce the number of features, we employed the dimension reduction method MRMD 2.0 [94] for feature selection. The 15378 features retained in the above feature pre-selection step were further reduced to 3221 dimensions. The performance was evaluated with 10-fold CV. The results are shown in Table1, where the dimension reduced features resulted in the best ACC, PRE, SEN and MCC values. The results indicate that more than 78% of the 15378 features are redundant, and only 21% of the features are discriminative and important for classification.

In order to detect the most informative features for fungal effector classification, the feature importance was evaluated using mRMR [95]. The first 100 important features (according to the feature importance scores of mRMR) are presented in Supplementary table S2.

Fifty-four of the top 100 features were generated by the Moran autocorrelation descriptor and Geary autocorrelation descriptor of the autocorrelation feature group. The two descriptors represent the spatial autocorrelation [96] of the amino acid index (AAI), namely the physicochemical and biochemical properties [97], which implies that the AAI spatial correlation among amino acids is the most informative parameter for fungal effector classification. More specially, we found that among the eight physicochemical properties examined in this study, the ''CHAM820102'' property, which represents the free energy of the solution in water, plays a key role, as 15 of the 54 important autocorrelation

TABLE 1. Comparison of three different resampling methods and feature selection on 10-fold CV.

Sampling method	Feature selection	ACC (%)	PRE (%)	SEN (%)	MCC (%)
Without resampling	___	98.10	45.00	6.89	17.20
Random under-sampling	___	83.33	84.11	83.51	67.50
GA-US	___	85.67	94.64	84.00	73.23
GA-US	MRMD 2.0	**91.60**	**95.98**	**86.51**	**83.93**

features were calculated by this index. Additionally, eight features were calculated by ''DAYM780201'' property (relative mutability), seven features were calculated by each of ''BHAR880101'' property (average flexibility indices), ''CHAM820101'' property (polarizability parameter) and ''CHOC760101'' property (residue accessible surface area in tripeptide). Spatial correlation usually occurred among the amino acids that with a distance (nlag value) larger than 15.

Eleven of the 100 features belong to the grouped amino acid composition group. In this group, the 20 amino acids were categorized into several classes according to physicochemical properties, e.g., hydrophobicity, charge and molecular size [98]. Informative patters of this group include postivecharger.aromatic.gap3 (a positively charged amino acid followed by three gaps, namely three amino acid without classes restriction, and an aromatic amino acid on the end), aromatic.uncharger.gap0, aromatic.uncharger.alphaticr and so on (Supplementary tale S2). We found that most of the patterns included an aromatic amino acid, suggesting that phenylalanine, tryptophan and tyrosine are informative for classification. 11 of the 100 features were generated from the amino acid composition group, which calculates the frequency of each amino acid type in a protein sequence. Informative patterns of this group include GV.gap5 (a Gly followed by five gaps and a Val at the end), IA.gap1, TL.gap2, GV, LT.gap2, TL.gap1, TG.gap2,VG.gap5, LL.gap4, VA and I. This finding indicates that the Ile, Gly, Val, Leu and Thr, as well as their combinations, are important for effector prediction. Other informative features included protein hydrophobicity and hydrophilicity and chemical properties of cysteine, which coincides with the features important for EffectorP, where hydrophilic amino acids (serine and cysteine) are reported to be informative [3].

E. COMPARISON WITH EXISTING TOOLS USING INDEPENDENT TEST DATA

To examine the predictive power of the proposed fungal effector classifier, we compared it with other predictors including EffectorP1.0 [3] and EffectorP2.0 [6]. As the EffectorP model cannot be re-trained using a customized dataset, ITDs were used for the model comparison. We trained our predictive model on the BTD dataset, and the current best predictors were trained on the training sets presented in their respective studies. These models were tested on ITD1 and ITD2, separately. The results are presented in Fig. 3(b). The proposed FunEffector-Pred achieved the highest accuracy of 93.45% on ITD1, which was 2.5% and 2.3% higher than accuracy of EffectorP2.0 and EffectorP1.0, respectively. FunEffector-Pred also exhibited an improvement of 19.05% and 14.29% with respect to EffectorP2.0 and EffectorP1.0, respectively, on dataset ITD2. These results indicate that our method represents a significant improvement in accuracy over the existing prediction algorithm.

IV. CONCLUSION

In this study, a new method was designed to effectively learn from imbalanced datasets. An active learning strategy (GSV-US) combined with a GA was used for majority class dnder-sampling. Using the proposed GSV-US and GA, a fungal effector predictor was developed. When an independent test dataset was evaluated, the FunEffector-Pred significantly outperformed the existing predictors that for fungal effector identification. FunEffector-Pred is effective due to the extraction of informative negative samples and elimination of the redundant, noisy samples as well as the screening of discriminative features. Several feature patterns that are informative for fungal effector identification are reported for the first time. Although the proposed method achieves some improvements, there is still room to further enhance its performance using an ensemble algorithm and by incorporating more informative heterogeneous features. It should be noted that the FunEffector-Pred is specifically designed to predict fungal effectors, investigation of the applicability of GSV-US and GAs for other types of imbalanced datasets in future work is worthwhile.

V. DATA AVAILABILITY

All data and source code were available at http://lab.malab.cn/∼wangchao/softwares/software.html

VI. CONFLICT OF INTERESTS

The authors have declared no competing interests.

ACKNOWLEDGMENT

(Chao Wang and Pingping Wang contributed equally to this work.)

REFERENCES

[1] M. C. Fisher, D. A. Henk, C. J. Briggs, J. S. Brownstein, L. C. Madoff, S. L. McCraw, and S. J. Gurr, "Emerging fungal threats to animal, plant and ecosystem health," *Nature*, vol. 484, no. 7393, pp. 186–194, Apr. 2012.

[2] D. A. Jones, S. Bertazzoni, C. J. Turo, R. A. Syme, and J. K. Hane, "Bioinformatic prediction of plant–pathogenicity effector proteins of fungi," *Current Opinion Microbiol.*, vol. 46, pp. 43–49, Dec. 2018.

[3] J. Sperschneider, D. M. Gardiner, P. N. Dodds, F. Tini, L. Covarelli, K. B. Singh, J. M. Manners, and J. M. Taylor, "EffectorP: Predicting fungal effector proteins from secretomes using machine learning," *New Phytologist*, vol. 210, no. 2, pp. 743–761, Apr. 2016.

[4] L. Cheng, C. Qi, H. Zhuang, T. Fu, and X. Zhang, "GutMDisorder: A comprehensive database for dysbiosis of the gut microbiota in disorders and interventions," *Nucleic Acids Res.*, vol. 48, no. D1, pp. D554–D560, Jan. 2020.

[5] V. G. A. A. Vleeshouwers and R. P. Oliver, "Effectors as tools in disease resistance breeding against biotrophic, hemibiotrophic, and necrotrophic plant pathogens," *Mol. Plant-Microbe Interact.*, vol. 27, no. 3, pp. 196–206, Mar. 2014.

[6] J. Sperschneider, P. N. Dodds, D. M. Gardiner, K. B. Singh, and J. M. Taylor, "Improved prediction of fungal effector proteins from secretomes with EffectorP 2.0," *Mol. Plant Pathol.*, vol. 19, no. 9, pp. 2094–2110, Sep. 2018.

[7] H. Sonah, R. K. Deshmukh, and R. R. Bélanger, "Computational prediction of effector proteins in fungi: Opportunities and challenges," *Frontiers Plant Sci.*, vol. 7, p. 14, Feb. 2016.

[8] J. Sperschneider, P. N. Dodds, D. M. Gardiner, J. M. Manners, K. B. Singh, and J. M. Taylor, "Advances and challenges in computational prediction of effectors from plant pathogenic fungi," *PLoS Pathogens*, vol. 11, no. 5, 2015, Art. no. e1004806.

[9] M. Urban, A. Cuzick, K. Rutherford, A. Irvine, H. Pedro, R. Pant, V. Sadanadan, L. Khamari, S. Billal, S. Mohanty, and K. E. Hammond-Kosack, "PHI-base: A new interface and further additions for the multi-species pathogen–host interactions database," *Nucleic Acids Res.*, vol. 45, no. D1, pp. D604–D610, Jan. 2017.

[10] S. Ertekin, J. Huang, L. Bottou, and L. Giles, "Learning on the border: Active learning in imbalanced data classification," in *Proc. 16th ACM Conf. Inf. Knowl. Manage.*, Lisbon, Portugal, Nov. 2007, pp. 127–136, doi: 10.1145/1321440.1321461.

[11] Y.-H. Zhu, J. Hu, X.-N. Song, and D.-J. Yu, "DNAPred: Accurate identification of DNA-binding sites from protein sequence by ensembled hyperplane-distance-based support vector machines," *J. Chem. Inf. Model.*, vol. 59, no. 6, pp. 3057–3071, Jun. 2019.

[12] M. Zięba and J. M. Tomczak, "Boosted SVM with active learning strategy for imbalanced data," *Soft Comput.*, vol. 19, no. 12, pp. 3357–3368, Dec. 2015.

[13] Y. Ding, J. Tang, and F. Guo, "Identification of drug-target interactions via multiple information integration," *Inf. Sci.*, vols. 418–419, pp. 546–560, Dec. 2017.

[14] Y. Ding, J. Tang, and F. Guo, "Identification of drug-side effect association via multiple information integration with centered kernel alignment," *Neurocomputing*, vol. 325, pp. 211–224, Jan. 2019.

[15] M. Zhang, F. Li, T. T. Marquez-Lago, A. Leier, C. Fan, C. K. Kwoh, K.-C. Chou, J. Song, and C. Jia, "MULTiPly: A novel multi-layer predictor for discovering general and specific types of promoters," *Bioinformatics*, vol. 35, no. 17, pp. 2957–2965, Sep. 2019.

[16] L. Cheng, H. Zhao, P. Wang, W. Zhou, M. Luo, T. Li, J. Han, S. Liu, and Q. Jiang, "Computational methods for identifying similar diseases," *Mol. Therapy-Nucleic Acids*, vol. 18, pp. 590–604, Dec. 2019.

[17] L. Cheng, "Computational and biological methods for gene therapy," *Current Gene Therapy*, vol. 19, no. 4, p. 210, Nov. 2019.

[18] G. Wang, Y. Wang, W. Feng, X. Wang, J. Y. Yang, Y. Zhao, Y. Wang, and Y. Liu, "Transcription factor and microRNA regulation in androgen-dependent and-independent prostate cancer cells," *BMC Genomics*, vol. 9, Sep. 2008, Art. no. S22.

[19] G. Wang, Y. Wang, M. Teng, D. Zhang, L. Li, and Y. Liu, "Signal transducers and activators of Transcription-1 (STAT1) regulates microRNA transcription in interferon γ-stimulated HeLa cells," *PLoS ONE*, vol. 5, no. 7, Jul. 2010, Art. no. e11794.

[20] X. Zhao, Q. Jiao, H. Li, Y. Wu, H. Wang, S. Huang, and G. Wang, "ECFS-DEA: An ensemble classifier-based feature selection for differential expression analysis on expression profiles," *BMC Bioinf.*, vol. 21, no. 1, Dec. 2020, Art. no. 43.

[21] Q. Jiang, G. Wang, S. Jin, Y. Li, and Y. Wang, "Predicting human microRNA-disease associations based on support vector machine," *Int. J. Data Mining Bioinf.*, vol. 8, no. 3, pp. 282–293, 2013.

[22] B. Liu, S. Chen, K. Yan, and F. Weng, "IRO-PsekGCC: Identify DNA replication origins based on pseudo k-Tuple GC composition," *Frontiers Genet.*, vol. 10, p. 842, Sep. 2019.

[23] B. Liu, Z. Luo, and J. He, "SgRNA-PSM: Predict sgRNAs on-target activity based on position-specific mismatch," *Mol. Therapy-Nucleic Acids*, vol. 20, no. 5, pp. 323–330 Jun. 2020.

[24] Q. Zou and Q. Ma, "The application of machine learning to disease diagnosis and treatment," *Math. Biosciences*, vol. 320, Feb. 2020, Art. no. 108305.

[25] Q. Zou, K. Qu, Y. Luo, D. Yin, Y. Ju, and H. Tang, "Predicting diabetes mellitus with machine learning techniques," *Frontiers Genet.*, vol. 9, Nov. 2018.

[26] X.-J. Zhu, C.-Q. Feng, H.-Y. Lai, W. Chen, and L. Hao, "Predicting protein structural classes for low-similarity sequences by evaluating different features," *Knowl.-Based Syst.*, vol. 163, pp. 787–793, Jan. 2019.

[27] J.-X. Tan, S.-H. Li, Z.-M. Zhang, C.-X. Chen, W. Chen, H. Tang, and H. Lin, "Identification of hormone binding proteins based on machine learning methods," *Math. Biosci. Eng.*, vol. 16, no. 4, pp. 2466–2480, 2019.

[28] Y. Zhao, F. Wang, and L. Juan, "MicroRNA promoter identification in *Arabidopsis* using multiple histone markers," *BioMed Res. Int.*, vol. 2015, 2015, Art. no. 861402.

[29] B. Liu, C. Li, and K. Yan, "DeepSVM-fold: Protein fold recognition by combining support vector machines and pairwise sequence similarity scores generated by deep learning networks," *Briefings Bioinf.*, Oct. 2019, Art. no. bbz098, doi: 10.1093/bib/bbz098.

[30] L. Wei, Q. Zou, M. Liao, H. Lu, and Y. Zhao, "A novel machine learning method for cytokine-receptor interaction prediction," *Combinat. Chem. High Throughput Screening*, vol. 19, no. 2, pp. 144–152, Jan. 2016.

[31] Y. Wang, F. Shi, L. Cao, N. Dey, Q. Wu, A. S. Ashour, R. S. Sherratt, V. Rajinikanth, and L. Wu, "Morphological segmentation analysis and texture-based support vector machines classification on mice liver fibrosis microscopic images," *Current Bioinf.*, vol. 14, no. 4, pp. 282–294, Apr. 2019.

[32] S. P. Wang, Q. Zhang, J. Lu, and Y.-D. Cai, "Analysis and prediction of nitrated tyrosine sites with the mRMR method and support vector machine algorithm," *Current Bioinf.*, vol. 13, no. 1, pp. 3–13, Feb. 2018.

[33] L. Wei, C. Zhou, R. Su, and Q. Zou, "PEPred-suite: Improved and robust prediction of therapeutic peptides using adaptive feature representation learning," *Bioinformatics*, vol. 35, no. 21, pp. 4272–4280, Nov. 2019.

[34] H. Wang, J. Wang, L. Zhang, P. Sun, N. Du, and Y. Li, "A sequential segment based alpha-helical transmembrane protein alignment method," *Int. J. Biol. Sci.*, vol. 14, no. 8, pp. 901–906, 2018.

[35] L. Zhang, H. Wang, L. Yan, L. Su, and D. Xu, "OMPcontact: An outer membrane protein inter-barrel residue contact prediction method," *J. Comput. Biol.*, vol. 24, no. 3, pp. 217–228, Mar. 2017.

[36] W. Chen, P. Feng, T. Liu, and D. Jin, "Recent advances in machine learning methods for predicting heat shock proteins," *Current Drug Metabolism*, vol. 20, no. 3, pp. 224–228, Oct. 2018.

[37] G. Wang, X. Luo, J. Wang, J. Wan, S. Xia, H. Zhu, J. Qian, and Y. Wang, "MeDReaders: A database for transcription factors that bind to methylated DNA," *Nucleic Acids Res.*, vol. 46, no. D1, pp. D146–D151, Jan. 2018.

[38] B. Liu, X. Gao, and H. Zhang, "BioSeq-analysis2.0: An updated platform for analyzing DNA, RNA and protein sequences at sequence level and residue level based on machine learning approaches," *Nucleic Acids Res.*, vol. 47, no. 20, p. e127, Nov. 2019.

[39] X. Chen, M. J. Pérez-Jiménez, L. Valencia-Cabrera, B. Wang, and X. Zeng, "Computing with viruses," *Theor. Comput. Sci.*, vol. 623, pp. 146–159, Apr. 2016.

[40] X. Chen, J.-D. Qiu, S.-P. Shi, S.-B. Suo, S.-Y. Huang, and R.-P. Liang, "Incorporating key position and amino acid residue features to identify general and species-specific ubiquitin conjugation sites," *Bioinformatics*, vol. 29, no. 13, pp. 1614–1622, Jul. 2013.

[41] H.-B. Shen and K.-C. Chou, "PseAAC: A flexible Web server for generating various kinds of protein pseudo amino acid composition," *Anal. Biochem.*, vol. 373, no. 2, pp. 386–388, Feb. 2008.

[42] Z. Chen, P. Zhao, F. Li, A. Leier, T. T. Marquez-Lago, Y. Wang, G. I. Webb, A. I. Smith, R. J. Daly, K.-C. Chou, and J. Song, "IFeature: A Python package and Web server for features extraction and selection from protein and peptide sequences," *Bioinformatics*, vol. 34, no. 14, pp. 2499–2502, Jul. 2018.

[43] R. Muhammod, S. Ahmed, D. M. Farid, S. Shatabda, A. Sharma, and A. Dehzangi, "PyFeat: A Python-based effective feature generation tool for DNA, RNA and protein sequences," *Bioinformatics*, vol. 35, no. 19, pp. 3831–3833, Oct. 2019.

[44] B. Liu, "BioSeq-analysis: A platform for DNA, RNA and protein sequence analysis based on machine learning approaches," *Briefings Bioinf.*, vol. 20, no. 4, pp. 1280–1294, Jul. 2019.

[45] C. Z. Cai, "SVM-prot: Web-based support vector machine software for functional classification of a protein from its primary sequence," *Nucleic Acids Res.*, vol. 31, no. 13, pp. 3692–3697, Jul. 2003.

[46] Y. Ding, J. Tang, and F. Guo, "Predicting protein-protein interactions via multivariate mutual information of protein sequences," *BMC Bioinf.*, vol. 17, no. 1, 2016, Art. no. 398.

[47] C. Jia, Y. Zuo, and Q. Zou, "O-GlcNAcPRED-II: An integrated classification algorithm for identifying O-GlcNAcylation sites based on fuzzy undersampling and a K-means PCA oversampling technique," *Bioinformatics*, vol. 34, no. 12, pp. 2029–2036, Jun. 2018.

[48] L. Wei, C. Zhou, H. Chen, J. Song, and R. Su, "ACPred-FL: A sequence-based predictor using effective feature representation to improve the prediction of anti-cancer peptides," *Bioinformatics*, vol. 34, no. 23, pp. 4007–4016 2018.

[49] X. Qiang, C. Zhou, X. Ye, P.-F. Du, R. Su, and L. Wei, "CPPred-FL: A sequence-based predictor for large-scale identification of cell-penetrating peptides by feature representation learning," *Briefings Bioinf.*, vol. 21, no. 1, pp. 11–23 2018.

[50] L. Wei, P. Xing, R. Su, G. Shi, Z. S. Ma, and Q. Zou, "CPPred-RF: A sequence-based predictor for identifying cell-penetrating peptides and their uptake efficiency," *J. Proteome Res.*, vol. 16, no. 5, pp. 2044–2053, May 2017.

[51] J. Han, X. Han, Q. Kong, and L. Cheng, "PsSubpathway: A software package for flexible identification of phenotype-specific subpathways in cancer progression," *Bioinformatics*, vol. 2019, Art. no. btz894.

[52] L. Cheng, Y. Hu, J. Sun, M. Zhou, and Q. Jiang, "DincRNA: A comprehensive Web-based bioinformatics toolkit for exploring disease associations and ncRNA function," *Bioinformatics*, vol. 34, no. 11, pp. 1953–1956, Jun. 2018.

[53] L. Cheng, P. Wang, R. Tian, S. Wang, Q. Guo, M. Luo, W. Zhou, G. Liu, H. Jiang, and Q. Jiang, "LncRNA2Target v2.0: A comprehensive database for target genes of lncRNAs in human and mouse," *Nucleic Acids Res.*, vol. 47, no. D1, pp. D140–D144, Jan. 2019.

[54] B. Liu, Y. Zhu, and K. Yan, "Fold-LTR-TCP: Protein fold recognition based on triadic closure principle," *Briefings Bioinf.*, Dec. 2019, doi: 10.1093/bib/bbz139.

[55] A. R. Bhattacharya, J. Liu, and S. Chakraborty. *A Generic Active Learning Framework for Class Imbalance Applications*. Accessed: Jan. 2009. [Online]. Available: https://pdfs.semanticscholar.org/7fd1/53c0f6b8965f445bd3e7b2a06c774278b0a8.pdf

[56] Y. Tang, Y.-Q. Zhang, N. V. Chawla, and S. Krasser, "SVMs modeling for highly imbalanced classification," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 1, pp. 281–288, Feb. 2009.

[57] A. J. Jones, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. London, U.K.: MIT Press, 1993.

[58] S. Zou, Y. Huang, Y. Wang, J. Wang, and C. Zhou, "SVM learning from imbalanced data by GA sampling for protein domain prediction," in *Proc. 9th Int. Conf. Young Comput. Sci.*, Hunan, China, Nov. 2008, pp. 982–987.

[59] J. Ha and J.-S. Lee, "A new under-sampling method using genetic algorithm for imbalanced data classification," in *Proc. 10th Int. Conf. Ubiquitous Inf. Manage. Commun. (IMCOM)*, Jan. 2016, Art. no. 95.

[60] I. Benchaji, S. Douzi, and B. El Ouahidi, "Using genetic algorithm to improve classification of imbalanced datasets for credit card fraud detection," in *Smart Data and Computational Intelligence*. Cham, Switzerland: Springer, 2018.

[61] D. Li, L. Luo, W. Zhang, F. Liu, and F. Luo, "A genetic algorithm-based weighted ensemble method for predicting transposon-derived piRNAs," *BMC Bioinf.*, vol. 17, no. 1, p. 11, Aug. 2016.

[62] S. Akbar, M. Hayat, M. Iqbal, and M. A. Jan, "IACP-GAEnsC: Evolutionary genetic algorithm based ensemble classification of anticancer peptides by utilizing hybrid feature space," *Artif. Intell. Med.*, vol. 79, pp. 62–70, Jun. 2017.

[63] L. Fu, B. Niu, Z. Zhu, S. Wu, and W. Li, "CD-HIT: Accelerated for clustering the next-generation sequencing data," *Bioinformatics*, vol. 28, no. 23, pp. 3150–3152, Dec. 2012.

[64] Q. Zou, G. Lin, X. Jiang, X. Liu, and X. Zeng, "Sequence clustering in bioinformatics: An empirical study," *Briefings Bioinf.*, vol. 21, no. 1, pp. 1–10, 2020.

[65] A. Bateman, M. J. Martin, C. O'Donovan, M. Magrane, R. Apweiler, E. Alpi, R. Antunes, J. Ar-Ganiska, B. Bely, and M. Bingley, "Uniprot: A hub for protein information," *Nucleic Acids Res.*, vol. 43, no. D1, pp. D204–D212, Jan. 2015.

[66] J. J. A. Armenteros, K. D. Tsirigos, C. K. Sønderby, T. N. Petersen, O. Winther, S. Brunak, G. von Heijne, and H. Nielsen, "SignalP 5.0 improves signal peptide predictions using deep neural networks," *Nature Biotechnol.*, vol. 37, no. 4, pp. 420–423, Apr. 2019.

[67] J. J. Almagro Armenteros, M. Salvatore, O. Emanuelsson, O. Winther, G. von Heijne, A. Elofsson, and H. Nielsen, "Detecting sequence signals in targeting peptides using deep learning," *Life Sci. Alliance*, vol. 2, no. 5, Oct. 2019, Art. no. e201900429.

[68] A. Krogh, B. Larsson, G. von Heijne, and E. L. L. Sonnhammer, "Predicting transmembrane protein topology with a hidden Markov model: Application to complete genomes," *J. Mol. Biol.*, vol. 305, no. 3, pp. 567–580, Jan. 2001.

[69] W. Yang, X.-J. Zhu, J. Huang, H. Ding, and H. Lin, "A brief survey of machine learning methods in protein sub-golgi localization," *Current Bioinf.*, vol. 14, no. 3, pp. 234–240, Mar. 2019.

[70] H. Ding, W. Yang, H. Tang, P.-M. Feng, J. Huang, W. Chen, and H. Lin, "PHYPred: A tool for identifying bacteriophage enzymes and hydrolases," *Virologica Sinica*, vol. 31, no. 4, pp. 350–352, Aug. 2016.

[71] B. Liu and Y. Zhu, "ProtDec-LTR3.0: Protein remote homology detection by incorporating profile-based features into learning to rank," *IEEE Access*, vol. 7, pp. 102499–102507, 2019.

[72] J. Zhang and B. Liu, "A review on the recent developments of sequence-based protein feature extraction methods," *Current Bioinf.*, vol. 14, no. 3, pp. 190–199, Mar. 2019.

[73] B. Liu, F. Liu, X. Wang, J. Chen, L. Fang, and K.-C. Chou, "Pse-in-one: A Web server for generating various modes of pseudo components of DNA, RNA, and protein sequences," *Nucleic Acids Res.*, vol. 43, no. W1, pp. W65–W71, Jul. 2015.

[74] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[75] H. Lv, Z. M. Zhang, S. H. Li, J. X. Tan, W. Chen, and H. Lin, "Evaluation of different computational methods on 5-methylcytosine sites identification," *Briefings Bioinf.*, 2019, doi: 10.1093/bib/bbz048.

[76] X. Ru, L. Li, and Q. Zou, "Incorporating distance-based top-n-gram and random forest to identify electron transport proteins," *J. Proteome Res.*, vol. 18, no. 7, pp. 2931–2939, Jul. 2019.

[77] Z. Lv, S. Jin, H. Ding, and Q. Zou, "A random forest sub-Golgi protein classifier optimized via dipeptide and amino acid composition features," *Frontiers Bioengineering Biotechnol.*, vol. 7, p. 215, Sep. 2019.

[78] P. M. Granitto, C. Furlanello, F. Biasioli, and F. Gasperi, "Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products," *Chemometric Intell. Lab. Syst.*, vol. 83, no. 2, pp. 83–90, Sep. 2006.

[79] X. Zeng, W. Lin, M. Guo, and Q. Zou, "A comprehensive overview and evaluation of circular RNA detection tools," *PLOS Comput. Biol.*, vol. 13, no. 6, 2017, Art. no. e1005420.

[80] X. Zeng, X. Zhang, and Q. Zou, "Integrative approaches for predicting microRNA function and prioritizing disease-related microRNA using biological interaction networks," *Briefings Bioinf.*, vol. 17, no. 2, pp. 193–203, Mar. 2016.

[81] X. Zeng, W. Wang, C. Chen, and G. G. Yen, "A consensus community-based particle swarm optimization for dynamic community detection," *IEEE Trans. Cybern.*, early access, Sep. 23, 2019, doi: 10.1109/TCYB.2019.2938895.

[82] H. Xu, W. Zeng, X. Zeng, and G. G. Yen, "An evolutionary algorithm based on Minkowski distance for many-objective optimization," *IEEE Trans. Cybern.*, vol. 49, no. 11, pp. 3968–3979, Nov. 2019.

[83] H. Xu, W. Zeng, D. Zhang, and X. Zeng, "MOEA/HD: A multiobjective evolutionary algorithm based on hierarchical decomposition," *IEEE Trans. Cybern.*, vol. 49, no. 2, pp. 517–526, Feb. 2019.

[84] X. Zeng, S. Yuan, X. Huang, and Q. Zou, "Identification of cytokine via an improved genetic algorithm," *Frontiers Comput. Sci.*, vol. 9, no. 4, pp. 643–651, Aug. 2015.

[85] N. M. Razali and J. Geraghty, "Genetic algorithm performance with different selection strategies in solving TSP," in *Proc. World Congr. Eng.*, London, U.K, Jul. 2011, pp. 1134–1139.

[86] L. Wei, P. Xing, J. Zeng, J. Chen, R. Su, and F. Guo, "Improved prediction of protein–protein interactions using novel negative samples, features, and an ensemble classifier," *Artif. Intell. Med.*, vol. 83, pp. 67–74, Nov. 2017.
[87] L. Wei, S. Wan, J. Guo, and K. K. Wong, "A novel hierarchical selective ensemble classifier with bioinformatics application," *Artif. Intell. Med.*, vol. 83, pp. 82–90, Nov. 2017.
[88] R. Su, H. Wu, B. Xu, X. Liu, and L. Wei, "Developing a multi-dose computational model for drug-induced hepatotoxicity prediction based on toxicogenomics data," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 16, no. 4, pp. 1231–1239, Jul. 2019.
[89] Y. Hu, T. Zhao, T. Zang, Y. Zhang, and L. Cheng, "Identification of Alzheimer's disease-related genes based on data integration method," *Frontiers Genet.*, vol. 9, Jan. 2019, Art. no. 703.
[90] W. Chen, P. Feng, X. Song, H. Lv, and H. Lin, "IRNA-m7G: Identifying N7-methylguanosine sites by fusing multiple features," *Mol. Therapy-Nucleic Acids*, vol. 18, pp. 269–274, Aug. 2019.
[91] B. Liu and K. Li, "IPromoter-2L2.0: Identifying promoters and their types by combining smoothing cutting window algorithm and sequence-based features," *Mol. Therapy-Nucleic Acids*, vol. 18, pp. 80–87, Dec. 2019.
[92] X. Zeng, S. Zhu, X. Liu, Y. Zhou, R. Nussinov, and F. Cheng, "DeepDR: A network-based deep learning approach to in *silico* drug repositioning," *Bioinformatics*, vol. 35, no. 24, pp. 5191–5198, Dec. 2019.
[93] X. Zeng, Y. Zhong, W. Lin, and Q. Zou, "Predicting disease-associated circular RNAs using deep forests combined with positive-unlabeled learning methods," *Briefings Bioinf.*, 2019, doi: 10.1093/bib/bbz080.
[94] Q. Zou, J. Zeng, L. Cao, and R. Ji, "A novel features ranking metric with application to scalable visual and bioinformatics data classification," *Neurocomputing*, vol. 173, pp. 346–354, Jan. 2016.
[95] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.
[96] R. R. Sokal and N. L. Oden, "Spatial autocorrelation in biology: 1. Methodology," *Biol. J. Linnean Soc.*, vol. 10, no. 2, pp. 199–228, Jun. 1978.
[97] S. Kawashima, H. Ogata, and M. Kanehisa, "Aaindex: Amino acid index database," *Nucleic Acids Res.*, vol. 27, no. 1, pp. 368–369, Jan. 1999.
[98] T.-Y. Lee, Z.-Q. Lin, S.-J. Hsieh, N. A. Bretaña, and C.-T. Lu, "Exploiting maximal dependence decomposition to identify conserved motifs from a group of aligned signal sequences," *Bioinformatics*, vol. 27, no. 13, pp. 1780–1787, Jul. 2011.