OXFORD

Briefings in Functional Genomics, 19(5–6), 2020, 350–363

doi: 10.1093/bfgp/elaa013
Advance Access Publication Date: 22 June 2020
Review Paper

Machine learning-based approaches for disease gene prediction

Duc-Hau Le

Corresponding author: Duc-Hau Le, Department of Computational Biomedicine, Vingroup Big Data Institute, Hanoi, Vietnam. Tel.: +84 912324564; E-mail: hauldhut@gmail.com

Abstract

Disease gene prediction is an essential issue in biomedical research. In the early days, annotation-based approaches were proposed for this problem. With the development of high-throughput technologies, interaction data between genes/proteins have grown quickly and covered almost genome and proteome; thus, network-based methods for the problem become prominent. In parallel, machine learning techniques, which formulate the problem as a classification, have also been proposed. Here, we firstly show a roadmap of the machine learning-based methods for the disease gene prediction. In the beginning, the problem was usually approached using a binary classification, where positive and negative training sample sets are comprised of disease genes and non-disease genes, respectively. The disease genes are ones known to be associated with diseases; meanwhile, non-disease genes were randomly selected from those not yet known to be associated with diseases. However, the later may contain unknown disease genes. To overcome this uncertainty of defining the non-disease genes, more realistic approaches have been proposed for the problem, such as unary and semi-supervised classification. Recently, more advanced methods, including ensemble learning, matrix factorization and deep learning, have been proposed for the problem. Secondly, 12 representative machine learning-based methods for the disease gene prediction were examined and compared in terms of prediction performance and running time. Finally, their advantages, disadvantages, interpretability and trust were also analyzed and discussed.

Key words: disease gene prediction; machine learning; binary classification; unary classification; semi-supervised learning; advanced learning models

Introduction

Disease gene prediction, the task of predicting the most plausible candidate disease genes, is an essential issue in biomedical research, and a variety of approaches have been proposed [1–3]. Most of the early methods, including POCUS [4], SUSPECTS [5], ENDEAVOUR [6] and ToppGene [7], have prioritized candidate genes by annotating them with respect to biological structures or functions and comparing their annotations with those of already known disease genes. These annotation-based approaches are limited in that they fail to capture indirect relationships between genes whose common features or functions are not yet annotated. To overcome this challenge, gene prioritization methods guided by biological networks have recently been proposed [3, 8–11]. The emergence of such network-based methods compared to annotation-based ones is due to the coverage of interactome data, where recent high-throughput technologies have yielded a vast amount of interaction data between cellular molecules and the interaction data covered most of genome and proteome.

Machine learning is generally techniques to learn a system from data, and they are recently applied successfully to various significant biomedical problems [12–14] such as genome annotation [15], classification of microarray data [16, 17], inference of gene regulatory networks [18], prediction of drug-target

Duc-Hau Le received his Ph.D. degree in 2012 with research interests in bioinformatics and computational biology. So far, he has published many papers on prestigious journals. He is now leading the Department of Computational Biomedicine, Vingroup Big Data Institute in Vietnam, which aims to develop/apply computational methods to build and analyze high-throughput biomedical data in order to improve the screen, diagnosis, and treatment of complex diseases.

[19, 20], the discovery of gene–gene interaction in disease data [21, 22], drug discovery [23–25] and personalized medicine [26–28]. In particular, it has been applied to the prediction of disease-associated genes [29, 30]. The problem is often formulated as a classification task, where the known disease gene and relevant biomedical data are used to train a classifier, which is then used to predict novel disease genes.

Briefly, at the early, for machine learning-based studies, the task of disease gene prediction was usually approached as a binary classification problem, where the training data consist of positive and negative training samples [29], such as Decision Trees (DT) [31, 32], k-Nearest Neighbor (k-NN) [33], Naive Bayesian (NB) classifier [34, 35], binary Support Vector Machines (SVM) classifier [36–38] and Artificial Neural Networks (ANN) techniques [39]. In the binary classification-based methods, positive training samples were known disease genes; meanwhile, negative training samples were the remaining ones. This is a limitation of the binary classification-based methods because the negative training samples should be actual non-disease genes. However, it is nearly impossible in biomedical researches to construct this set. Thus, more practical approaches have been proposed. Indeed, unary/one-class classifiers learned from only positive samples have been introduced [40–42]. Those studies used one-class SVM [43] and kernel-based data fusion methods [44] to integrate data from different resources. The remaining set may contain unknown disease genes; thus, semi-supervised learning (SSL) methods such as a binary semi-supervised [45] and positive and unlabeled (PU) learning techniques [46, 47] were proposed. Those SSL-based classifiers are learned from both labeled (i.e. known disease genes) and unlabeled (i.e. the remaining genes) sets. Experiment results showed that the SSL-based methods [46, 47] outperform binary SVM [36] and k-NN [33] as well as the one-class SVM-based methods proposed in [40, 42]. Recently, more advanced methods, including ensemble learning [48, 49], matrix factorization (MF) [50–52] and deep learning [53–58] have been proposed for the disease gene prediction. Ensemble learning-based methods combine the outcome of single classifiers trained on either different learning models or datasets into final prediction. Meanwhile, inspired by the recommendation problem, the MF-based methods have also been used for the disease gene prediction. Finally, with the ability to learning representation of input data, deep learning-based methods have shown their improvement in prediction performance.

In this study, we reviewed almost machine learning-based approaches for the disease gene prediction. To this end, we first drew a roadmap of the machine learning-based methods for the task. Although classification techniques have extensively been used for the disease gene prediction problem using many kinds of annotated genes/proteins data, to the best of our knowledge, there has been no comparative study of those techniques that are based on the same data representation except our previous studies between the binary classifiers [29] and between the more realistic classifiers [30]. Those studies usually represented data in feature vectors and kernel matrices for binary and unary, PU learning classifiers, respectively. Therefore, we additionally examined and compared the prediction performance and running time of 12 typical classification techniques, including binary, unary, semi-supervised, ensemble and deep learning-based methods for the task based on a vectorial representation of samples. More specifically, five traditional binary classification techniques have been used for the task such as DT, k-NN, NB, ANN and SVM. In addition, two one-class classification techniques, including one-class SVM [43] and one-class Hempstalk [59] were also compared. The former was also proposed for the task [40, 42]; meanwhile, the later has not been yet. Moreover, three SSL-based methods, including one binary SSL- and two PU learning-based methods, which have also been proposed for the task, were additionally examined. Finally, two modern methods, including an ensemble learning- and a deep learning-based ones, were also compared. Besides comparing the 12 methods in terms of prediction performance and running time, their advantages and disadvantages were also analyzed. Based on our experiment, tree- and neural network-based methods have shown to be good methods in terms of both accuracy (ACC) and the area under the receiver operating characteristic curve (AUROC) measures. For the two one-class methods, the one-class Hempstalk was shown to be the best; meanwhile, the one-class SVM was shown to be the worst overall competitive methods. Interestingly, PU learning-based methods, which were proven to be better than traditional binary and one-class SVM methods in previous studies [46, 47], achieved low prediction performance in our experiment. Finally, modern methods such as an ensemble method, RandomForest [60], and a deep learning-based method, DNN [61], actually showed their power for the task in terms of both the performance measures. For practical use, the running time of the 12 methods were also compared. In addition, their interpretability and trust were also discussed.

Preliminary

Figure 1 shows a common scheme of machine learning-based approaches for the disease gene prediction. An approach (e.g. binary, unary or SSL-based classification) to the problem should be selected at the beginning. For the data preparation, training data, which are usually known (labeled) and unknown (unlabeled) disease genes annotated with biomedical data (e.g. -omics data) are collected. Then the data are preprocessed by normalization, transformation and featurization techniques. Features of genes can be either carefully selected based on evidence of the significant difference between disease and non-disease genes or automatically learned from the input data. After that, the preprocessed data are used to train a learning model. In the second stage, the learning model (e.g. two-class SVM, one-class SVM or PU learning) is first selected to fit the chosen approach; then, it is trained using the preprocessed data. The trained model with optimized parameters is finally selected after testing and validation processes. At the last stage, the trained model is used to predict novel disease genes from the remaining ones.

A road of machine learning-based approaches for the disease gene prediction

At the early, the disease gene prediction was usually approached as a binary classification problem, where the training data consist of positive and negative training samples. Positive training samples were known disease genes; meanwhile, negative training samples were often randomly selected from the remaining ones. The remaining set may contain unknown disease genes; meanwhile, the negative training samples should be actual non-disease genes. However, there is no database stored such the genes (no proven negatives), because it is often the case in biology that not observing an association does not imply the association does not exist. Therefore, to reduce this uncertainty, another approach to the disease gene prediction is unary/one-class classification, in which the classifier is learned only from known disease genes. Due to the fact that the remaining set

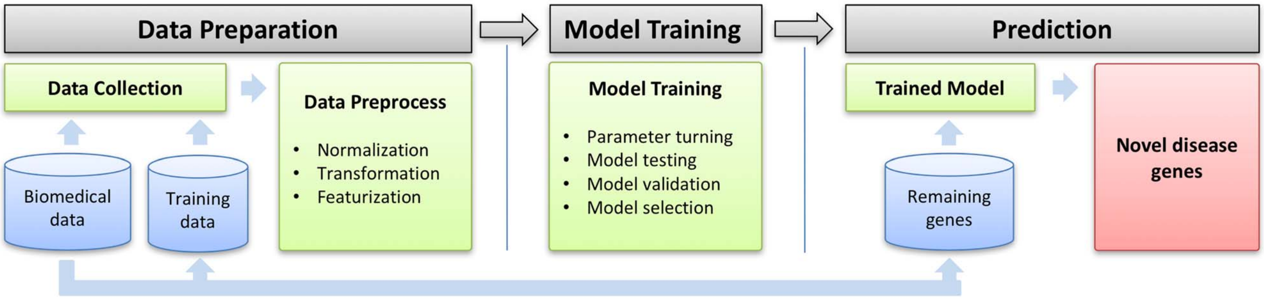

Figure 1. A common scheme of machine learning-based approaches for the disease gene prediction.

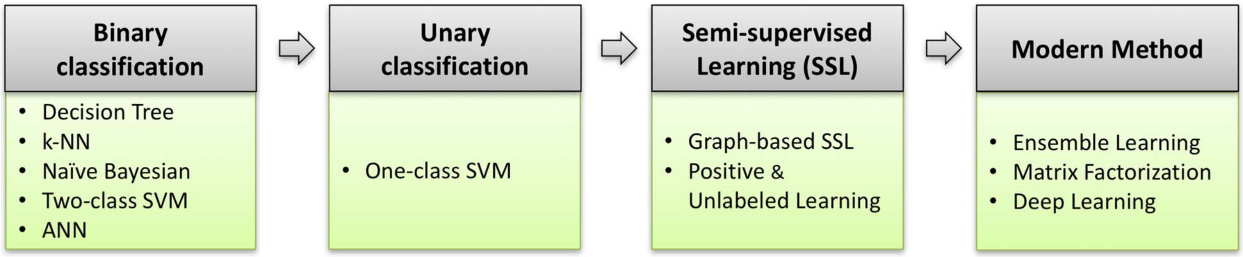

Figure 2. A roadmap of machine learning-based methods for the disease gene prediction.

may contain unknown disease genes, SSL-based methods were proposed to the problem, where the classifier is learned from both labeled and unlabeled set. In particular, PU learning-based methods were proposed. In the method, the classifier is learned from both positive training samples (i.e. known disease genes) and unlabeled samples (the remaining genes). Recently, more advanced methods, which are able to combine single classifiers (i.e. ensemble learning models), inspired from recommendation problem (i.e. MF methods) or able to learn the representation of data (i.e. deep learning models), have also been proposed for the disease gene prediction problem. The performance of the machine learning-based method is often dependent on the size of training data (i.e. known disease genes). Therefore, to gain the prediction performance, proposed machine learning-based methods usually used all disease genes (i.e. general disease) for training. Only a few of them have been introduced for disease groups (e.g. cancer, metabolism and infectious diseases) or specific diseases with a large number of known genes (e.g. Parkinson, primary immunodeficiency diseases – PID). We here reviewed all the machine learning-based approaches for the problem. Figure 2 shows a roadmap, and Table 1 summarizes machine learning-based methods proposed for the problem.

Binary classification

Decision tree

DT classifier is a simple and widely used classification technique that is capable of breaking down a complex decision-making process into a collection of simpler decisions [62, 63]. This is one of the early binary classification techniques proposed for the disease gene prediction [31, 32]. They were introduced based on distinctive sequence features of known disease proteins compared to all human proteins. In those studies, disease proteins obtained from Online Mendelian Inheritance in Man (OMIM) [64] were set as the positive training set; meanwhile, the negative training set with the same size was randomly selected from the human genome and presumably was not known to be involved in diseases.

K-nearest neighbors

K-nearest neighbors (k-NN) algorithm is a non-parametric method used for classification and regression [65]. With the growth of interaction data between proteins, Xu and Li [33] used a k-NN algorithm for the identification of novel disease genes. In the study, topological properties of proteins were constructed from a human protein–protein interaction (PPI) network. To build a positive training set, they also collected known disease genes from OMIM. Based on a result of [66] that the human genome may contain thousands of essential genes having features that differ significantly from both disease genes and the other genes, they constructed the negative training set by randomly selected proteins from a control set. This control set contains all human genome, excluding disease and essential proteins. The use of topological properties of proteins was based on their investigation of the significant distinction of the properties between the disease gene set and the control set.

Naive Bayesian classifier

NB classifier is a simple probabilistic classifier based on applying Bayes' theorem with independence assumptions between predictors [67]. Calvo *et al.* [34] and Lage *et al.* [35] used this algorithm to identify human disease genes by integrating multiple types of genomic, phenotypic and interactomics data. In particular, the study [34] built a NB classifier based on eight different genomic datasets to identify human mitochondrial disease genes. Finally, they reported that their method outperformed a DT-based method, CART [62], and a boosting AdaBoost algorithm [68]. The training samples included mitochondrial proteins curated by the MitoP2 database [69] and non-mitochondrial proteins annotated to localize to other cellular compartments. Meanwhile, the study [35] relied on a simple assumption that

Table 1. A summary of machine learning-based methods for the disease gene prediction

Method	Input data	Data representation/integration	General/specific diseases
Binary classification			
Decision tree [31, 32]	Protein sequence, known genes from OMIM	Feature vector	General disease
k-nearest neighbors [33]	PPI network, known genes from OMIM	Feature vector	General disease
Naive Bayesian [34]	Multi-omicss data, mitochondrial proteins	Feature vector	General disease
Naive Bayesian [35]	PPI network, phenotype similarity, known genes from GeneCard and OMIM	Feature vector	General disease
Two-class support vector machine [36]	PPI network, protein sequence	Feature vector	General disease
Two-class support vector machine [37]	PPI network, protein sequence, protein functional information, disease ontology	Feature vector	Specific diseases
Two-class support vector machine [38]	PPI network, gene expression, molecular alterations, mouse studies	Feature vector	Specific diseases
Artificial neural network [73]	PPI network	Feature vector	Specific diseases
Artificial neural network [74]	Gene expression of case-control samples	Feature vector	General disease
Unary classification			
One-class SVM [41, 42]	Domain vocabularies	Kernel matrix	General disease
One-class SVM [40]	Multiple biomedical data	Kernel matrix	General disease
Semi-supervised classification			
Binary semi-supervised [45]	Multiple–omics data, PPI network	Feature vector	General disease
PU learning (biased SVM) [47]	Multiple biomedical data	Kernel matrix	
PU learning (multi-level SVM) [46]	Protein domain, PPI network, gene ontology	Feature vector	
Modern methods			
Random forest [29]	Multiple biomedical data	Feature vector	General disease
Ensemble PU learning [48]	Gene expression, PPI network, gene ontology	Feature vector	Disease group
Matrix factorization [50, 52]	Gene expression, gene network, disease similarity, known gene-phenotype associations	Similarity and association matrices	General disease
Deep neural network [55]	Protein sequence, PPI network	Features learned from the data	Disease group
Node2vec and autoencoder [57]	PPI networks	Features learned from the data	Specific disease
Multimodal deep belief network [54]	PPI networks, gene ontology, disease similarity	Features learned from the data	General disease
Graph convolutional network [53]	Gene network, gene expression, gene orthology, disease similarity	Features learned from the data	General disease

mutations in different members of a protein complex (which was predicted from a PPI network) lead to comparable phenotypes, where the similarity between phenotypes can be calculated by text mining. Therefore, they built a Bayesian predictor based on phenotypic similarities and confidence scores of interactions between proteins. The positive training samples for that Bayesian predictor are known disease genes collected from GeneCard [70] and OMIM, whereas the negative training samples are genes in the genome excluded known disease genes.

Two-class Support Vector Machines

The SVM method attempts to map the input feature space into a new high dimensional feature space and then finds an optimal separating hyperplane, which can be used to discriminate between positive and negative samples [71]. Based on both interaction and sequence data of protein, Smalter *et al.* [36] used a two-class SVM to the problem and constructed the training set in the same way as in [33]. They showed that their method is better than the k-NN method [33], which only based on interaction data. The SVM technique then was also used in some studies for the disease gene prediction [37, 38]. Specifically, in addition to the PPI network and sequence data, a study [37] used protein functional information at the molecular level. Moreover, unlike the above-mentioned methods where the classifier was trained based on all disease genes associated with all diseases, the study [37] trained the learning model for each disease ontology term; therefore, only disease terms with at least 10 known associated genes were considered. The positive set is all of such the known disease genes, whereas the negative training set includes genes associated with other diseases and 10% the remaining genes in the genome. Likewise, the study [38] used a two-class SVM classifier to identify genes related to PID. This classifier was trained using 69 binary features of known PID genes and non-PID genes. The trained classifier was then applied to predict 1442 candidate PID genes.

Artificial neural network

ANNs are computational models inspired by animals' central nervous systems in the brain that are capable of machine learning and pattern recognition as a classifier [72]. A multi-layer perceptron, a functional link ANN and a two-class SVM classifiers were used to identify novel disease genes for four complex diseases (i.e. Cancer, Type 1 Diabetes, Type 2 Diabetes and Ageing) using eight topological features calculated from a PPI network [73]. For each of those diseases, the positive training set was the known associated genes, whereas the negative training set was constructed by the same procedure as in [33]. The result showed that the two-class SVM classifier archived higher accuracy than the other two methods. However, the functional link ANN has a lower computation cost compared to the multi-layer perception and the SVM classifiers. Moreover, Xiao *et al.* [74] used differential expression data of different diseases to build ANN classifiers, including three layers (i.e. input layer, hidden layer and output layer), where the number of input neurons was set to a number of case-control expression datasets. The positive training set also contained known disease genes, whereas the negative training set consists of the same number of genes randomly selected from all non-known disease genes. To avoid overfitting of trained ANN classifier, 1000 classifiers were trained based on 1000 randomly selected negative training sets. A comparison was performed and showed that their method was better than the DT-based method [32], but worse than an annotation-based method, ENDEAVOUR [6].

Unary classification

One-class classification, also known as unary classification, tries to identify objects of a specific class amongst all objects, by learning from a training set containing only the objects of that class. That is different from and more complicated than the traditional classification problem, which tries to distinguish between two or more classes with the training set containing objects from all the classes. An example is the classification of the operational status of a nuclear plant as 'normal' [75]. In particular, one-class SVM proposed by [43] was widely used for the disease gene prediction. The primary strategy is to find a hyperplane that separates points representing the disease genes from the origin with the largest possible margin. Finally, it considers a gene more likely to be a disease gene if the representing point of the gene lies farther in the direction of this hyperplane. Indeed, Yu *et al.* [41] used one-class SVM as one of the linear ranking algorithms for the disease gene prediction by text mining on five different domain vocabularies and two text representation schemes. In which, each classifier was trained on an individual domain vocabulary. This means that only one view of genes was considered in the kernel learning process (also known as single kernel learning). They only used linear kernels because the dimensionality of the data is very high. Also using the one-class SVM, De Bie *et al.* [40] formulated the problem as a novelty detection where one tries to model the training genes only with nine kinds of data sources from Ensembl database [76] (i.e. microarray, DNA sequence, EST data, gene ontology (GO) annotations, InterPro domains, KEGG pathway, motifs, binding data and literature). Therefore, this one-class SVM-based classifier was learned on a number of different views on the genes. This is also known as a multiple kernel learning (MKL) method. Three kinds of the kernel, including a linear, two variants of Gaussian kernel, were tested. Finally, they showed that their method outperformed the annotation-based method ENDEAVOUR [6]. More recently, this research group applied the MKL techniques to the disease gene prediction based on multi-view text mining [42].

Semi-supervised classification

The one-class classification-based methods such as one-class SVM seems to be the best-fit way since only a positive training set consisting of known disease genes is needed for the training task. However, it is particularly true in high dimension with a few examples that density estimation from a few positive samples is known to be very challenging. Besides, the remaining set may contain unknown disease genes. Thus, SSL-based methods were proposed.

Binary semi-supervised classification

SSL is halfway between supervised and unsupervised learning since both labeled data and unlabeled data are used in the training process [77]. Since labeling often requires much human labor, whereas unlabeled data is far easier to obtain, SSL is beneficial in many problems in bioinformatics such as prediction of cancer recurrence [78], predicting interactions between HIV-1 and human proteins [79], inference of gene regulatory networks [18] and so on. Nguyen and Ho [45] used the SSL model for the disease gene prediction. In particular, they formed the positive training set and negative training set in the same way as in [33]. However, the initial positive training set was extended by adding neighbors of disease proteins (i.e. they are unlabeled) in a human PPI network. In addition to the PPI network data, they used other genomic and proteomic data to construct an eight-feature vector for each protein. Finally, a graph-based SSL [80, 81] was applied to train the classifier. As a result, they showed that their method is better than the k-NN and two-class SVM in the same benchmark and settings.

Positive and unlabeled learning

A common task of all the binary classification-based methods is the construction of non-disease genes as the negative training set to train binary classifiers, which is nearly impossible in biomedical researches. To overcome this limitation, recent studies proposed to learn the scoring function only from known and unknown gene set, by formulating the problem as positive (P) and unlabeled (U) learning (PU learning) [82, 83]. This is known to be a dominant paradigm when a set of candidates has to be ranked in terms of similarity to a set of positive data [69]. In machine learning, PU learning is a collection of semi-supervised techniques for training binary classifiers on positive and unlabeled samples only. This machine learning technique was applied to the disease gene prediction problem [46, 47]. Indeed, they were motivated by the principle of PU learning that information provided by the distribution of unlabeled examples can improve the scoring function (Figure 3). Therefore, a simple and efficient strategy to solve a PU learning problem is to: (1) Assign negative labels to elements in U; (2) Train a binary classifier to discriminate P from U, allowing errors in the training labels. More specifically, the study [47] proposed a method, namely ProDiGe, which used a biased SVM classifier [84]. This binary classifier assigned large weights for positive samples during training to account for the fact that they represent high-confidence samples (i.e. known disease genes);

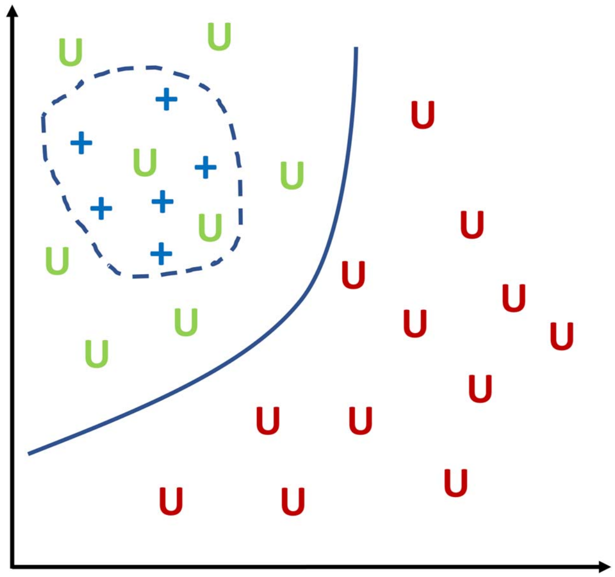

Figure 3. An illustration of PU learning. When only positive examples (+) are used, a classification method may create a dot line boundary. By using additional unlabeled samples (U), the border could be the solid line.

meanwhile, the 'negative' samples (collected from the remaining set) may contain false negatives (i.e. unknown disease genes), which they hope to discover. In addition, to reduce variance, a bootstrap procedure was added to that biased SVM to form a bagging SVM classifier [85]. Nine datasets collected from the Ensembl database [76] were used to describe genes. After extending their method to learn simultaneously from such multiple heterogeneous data sources, they showed that their method outperformed the one-class SVM MKL introduced in [40, 42] in both mean and optimized (i.e. based on MKL) variants of ProDiGe. In addition, ProDiGe was also reported outperforming a network-based method, PRINCE [86]. However, ProDiGe still contains some limitations caused by the random choice of random subsets from U to train multiple classifiers to discriminate P from U since the random subsets could still include unknown disease genes. And thus, individual classifiers are not accurate, and this will affect the overall performance of the final classifier. Besides, ProDiGe treated all samples in the random subset equally. Therefore, Yang *et al*. [46] proposed a different strategy, namely PUDI, to overcome those limitations for PU learning that is applied to the disease gene prediction. Indeed, they partitioned U into four labeled sets (i.e. reliable negative set, likely positive set, likely negative set and weak negative set) relying on their likelihoods being positive/negative class. Finally, they built multi-level weighted SVMs [87] to identify novel disease genes. As a result, they reported that PUDI outperformed ProDiGe, the method based on two-class SVM [36] and the method based on k-NN [33].

Modern methods

The performance of a single classifier for predicting disease-gene associations is highly dependent on the benchmark datasets; thus, recent studies have combined single classifiers by ensemble learning methods for the task. Indeed, Random Forest (RF) [60], a decision-tree-based ensemble learning method, was shown to be the best among binary classification methods [29]. The ensemble strategy has also been used to combine PU learning methods [48, 49] for the disease gene prediction problem.

The task of disease-gene association has been recently formulated as a recommendation problem, where a known association between a disease and a gene is considered as an occurred event between a user and an item, respectively. Therefore, matrix factorization (MF) methods, which have been popularly used for the recommendation task, could be directly applied in the disease gene prediction problem. In addition to the known disease–gene associations, features of diseases and genes embedded in disease similarities and gene similarities can be used to aid the prediction of disease–gene associations [50, 52].

However, the similarity information and domain knowledge-based extracted features of diseases and genes might not fully reflect complicated relationships between the features and between instances and labels. Thus, deep learning with the ability to exploiting the high order and nonlinear relationship could achieve high performance in many applications, including the disease gene prediction problem [53–55, 57]. Indeed, deep learning methods have been exploited to predict genes associated with general disease [53, 54] and specific diseases such as Parkinson's [57] and infectious diseases [55] using different types of biomedical data. For specific diseases, Barman *et al*. [55] firstly extracted features from protein sequences and topological features from a PPI network, then employed some binary classifiers, including a Deep Neural Network (DNN) [61], SVM, NB and RF for predicting genes associated with infectious diseases. Meanwhile, Peng *et al*. [57] used Node2vec [88] to extract the vector representation of each gene in a PPI network, then an autoencoder [89] was used to reduce the dimension of the obtained vector. Finally, an SVM classifier was used to predict novel genes associated with Parkinson's disease.

Previous studies have shown that combining multiple types of biomedical data could improve the prediction performance of disease–gene association by multi-view learning techniques such as the MKL techniques [40, 42]. To leverage the advantage of deep learning in data fusion, a recent study [54] has used a multimodal deep belief network (DBN) [61] to combine sub-models trained on PPI networks and GO terms to learn cross-modality representations for the disease gene prediction problem. To exploit more potential relationships between entities in a graph, a graph convolutional network (GCN) [90], which combines first-order approximation of spectral graph convolutions and ANN, was proposed. This technique has been recently used in combination with the MF method to exploit similarity networks of diseases and genes for the prediction of disease–gene associations [53].

Performance examination

Although a diversity of classification-based methods has been proposed for the disease gene prediction there is no comprehensive comparison among them due to the difference in data representation, performance assessment of each method. In previous studies, we compared the prediction performance among binary classification [29] and among advanced classification [30] methods for the disease gene prediction using the vectorial representation of data. Here, we compared 12 typical classification-based methods including five traditional binary (i.e. DT, k-NN, NB, SVM and ANN), two unary (i.e. one-class SVM and one-class Hempstalk [59]), three semi-supervised (i.e. binary SSL, biased SVM and multi-level SVM) and two modern methods (i.e. an ensemble model, RF and a deep learning model, DNN) methods for the disease gene prediction based on the vectorial representation of data.

Building the feature set

We collected interactomic, genomic and proteomic data for genes/proteins. Detail procedure to create the feature set can be referred to our previous studies [29, 30]. Briefly, the feature set is comprised of topological features (i.e. degree, 1-N index, 2-N index, distance to disease genes and positive topology coefficient) calculated from a human PPI network [91]. These features are collected based on an observation about their significant difference between two groups of disease and non-disease genes [33]. Besides, the two groups are also different in length of the protein sequence [31, 32] and a number of GO [92] terms annotating proteins [4, 93]. Also, a larger number of domains and binding sites may allow mutation to more easily corrupt protein functions [94]; thus, a protein domain data from InterPro [95, 96] was collected via BioMart tool [97] to calculate the number of domains and binding sites for each protein. Finally, the evolution rate of a gene may contribute to the likelihood of hereditary disease [32, 66]; thus, homolog gene data [98] were used to calculate the evolution rate for each protein. In summary, a total of 10 features representing topological, sequence, structural, annotation and evolutionary properties of gene/protein were collected and normalized in the range of [0, 1] for all experiments.

Creation of the training set

Figure 4 shows a workflow of building the training set for different classification methods for the disease gene prediction. Briefly, known disease genes were collected from OMIM [64], then they were mapped to the human PPI network to obtain known disease proteins (i.e. the positive training set, P). P was then used as the training set for the unary classification methods. The remaining proteins (R) in the human PPI network consists of unknown disease proteins, non-disease proteins and 2056 essential proteins. Then, based on a result of the study [66] that human genome may contain thousands of essential genes having features which differ significantly from both disease genes and the other genes, we excluded essential proteins collected from DEG [99], BioMart [97] and DAVID [100] to construct an unlabeled set (U), which contains unknown disease proteins and non-disease proteins. For binary classification methods, we additionally built the negative training samples (N) by randomly selecting a set (RS) having the same size with the positive training set (P) from the unlabeled set (U). Following the same procedure proposed by [45], we further collected neighbors (U_P) of known disease proteins (P) in the human PPI network from U to build the training set for the binary SSL-based method. Finally, both P and RS were used for training PU-based methods. In particular, for the biased SVM-based method, RS was also used as N in the binary classification method, but samples in RS were given smaller weights compared to that in P during the training process [47]. The set RS was further partitioned into four labeled sets [i.e. reliable negative set (RN), likely positive set (LP), likely negative set (LN), and weak negative set (WN)] based on the extent to which an unknown gene is relevant to diseases of interest as required for the multi-level weighted SVM-based method used in PUDI [46].

Performance comparison

We tested typical machine learning-based methods for the disease gene prediction, including five traditional binary classifiers (i.e. DT, k-NN, NB, SVM, and ANN). In addition, two unary classification methods, including one-class SVM [43] and one-class Hempstalk [59], were also tested. We also examined three SSL-based classification methods, including a binary SSL-based classification method, transductive SVM (TSVM) [101] and two PU learning, i.e. biased SVM-based and multilevel SVM-based methods. Finally, two modern methods, including an ensemble learning method, RF [60] and a DNN, were also examined. For DT, k-NN, NB, ANN and RF, we used their implementation in WEKA [102], a Java-based machine learning tool. For SVM-based methods such as the two-class SVM, the one-class SVM, the biased SVM and multilevel SVM, we used LibSVM [103] package in Weka. For the one-class Hempstalk method, a Weka package, namely 'oneClassClassifier [104],' was also used. For the DNN method, we used another Weka package, i.e. WekaDeeplearning4j [105]. Finally, for the TSVM method, we used its implementation in the SVMlin package [106].

To determine the parameters for the model selection, we faithfully used the settings that have been reported in the literature. For instance, we selected a non-linear kernel function (i.e. radial basic function) and other parameters such as error penalty for miss-classed samples and complexity of non-linear optimal separating hyper-plane for the binary SVM as in [36]. For k-NN, the optimal number of neighbors (k) of 3 was selected as in [33]. For DT, two default parameters, including confidence factor for pruning and a minimum number of instances per leaf was set to 0.25 and 2, respectively. For ANN, a number of input neurons were set to the number of features (i.e. 10), and a number of hidden neurons was set to an empirical value [i.e. an average of the number of features and number of classes (i.e. 2)]. In addition, the sigmoid threshold function was used for every neuron. Finally, trees were built with unlimited depth for RF. Weights for the PU learning methods were optimized to find the best classifiers. More specifically, we found a ratio between P and RS is 1.1 for the best biased SVM-based PU learning method. Meanwhile, the weights for the multi-level SVM-based PU method were 1.5, 1, 1, 1.1, 1.2 for P, LP, WN, LN, RN, respectively. For the DNN-based method, we tested with several architectures of the model (i.e. number of hidden layers and number of neurons per each layer), and found that DNN worked best for one hidden layer having 32 neurons. For the other methods, their default parameters were used.

The prediction performance of each method was assessed by accuracy (ACC) (i.e. defined as the number of true predictions overall predictions) for all methods and AUROC (i.e. area under the receiver operating characteristic curve) for methods which deal with more than one class (i.e. the two one-class classifiers were excluded) using a 10-fold cross-validation scheme on the training set. Also, the selection of RS was repeated 100 times to avoid sampling bias and performance variance. Then, the final performance was an average value of the two measures (i.e. ACC and AUROC). The running times of the 12 methods were also compared. They are average values for one trial of the cross-validation scheme performed on a workstation with OS Ubuntu 18.04.4 LTS (64-bit), Intel Core i7-4510 U CPU @ 2GHz × 4 cores and 16GB RAM.

Figure 5A shows the performance comparison of the 12 methods for the disease gene prediction. For the five traditional classifiers (i.e. DT, k-NN, NB, SVM and ANN), ANN was the best classifier in terms of both measures (i.e. ACC = 0.71, AUROC = 0.78), meanwhile NB (ACC = 0.63) and SVM (AUROC = 0.68) were the worst classifiers in terms of ACC and AUROC, respectively. For the two one-class classifiers, only ACC values were calculated for both of them. Interestingly, the one-class Hempstalk was

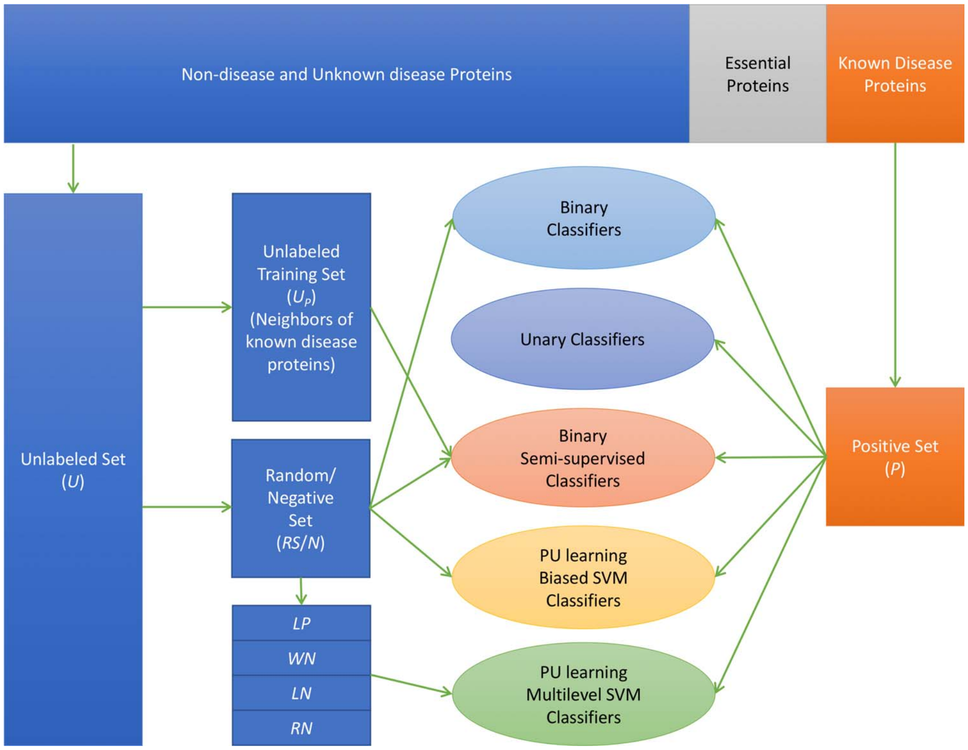
Non-disease and Unknown disease Proteins
Essential Proteins
Known Disease Proteins
Unlabeled Set (U)
Unlabeled Training Set (U_P) (Neighbors of known disease proteins)
Random/ Negative Set (RS/N)
LP
WN
LN
RN
Binary Classifiers
Unary Classifiers
Binary Semi-supervised Classifiers
PU learning Biased SVM Classifiers
PU learning Multilevel SVM Classifiers
Positive Set (P)

Figure 4. A workflow of building the training set for different classification methods for the disease gene prediction.

the best (ACC = 0.83); meanwhile, the one-class SVM achieves worst performance (ACC = 0.50) overall competitive methods. This is because the one-class SVM [43] is only based on the class probability estimation strategy; meanwhile, the one-class Hempstalk is additionally based on density/distribution strategies [59]. Among the three SSL-based methods, the binary SSL-based method, TSVM, was the best in terms of both measures (i.e. ACC = 0.69 and AUROC = 0.76). This is because the training set of TSVM was enriched with neighbors of the known disease proteins. Thus, the method can exploit advanced features of both network- and machine learning-based approaches for the disease gene prediction. Indeed, the 'disease module' principle (i.e. genes/proteins associated with the same/similar disease tend to locate closely in gene/protein interaction network) has been widely used in network-based methods for the disease gene prediction [3]. In contrast to the result shown in [46], Figure 5A shows that the multi-level SVM-based PU learning method performed worse than the biased SVM-based one in terms of both measures. Finally, the two modern methods (i.e. RF and DNN) are comparable and among the best in terms of both measures. They are only worse than the one-class Hempstalk in terms of ACC.

Regarding running time, Figure 5B shows that simple methods such as DT, k-NN, and NB require much little running time compared to the others. Especially, neural network-based methods such as ANN and DNN were much computationally expensive (i.e. DNN was 186, 93, and 1686 times slower than DT, k-NN and NB, respectively).

Although the one-class SVM achieved the poorest performance for general disease in our experiment, a recent study on the disease gene prediction in acute myeloid leukemia cancer has shown the dominance of the one-class SVM method compared to other binary classification and PU learning approaches [107]. This indicates that the performance of a single classification-based method for predicting disease–gene associations is highly dependent on the benchmark datasets. However, it holds for ensemble methods that they usually achieve good performance compared to others. Indeed, RF, an ensemble method, is the best among binary classification methods [29]. Similarly, recent ensemble strategies have combined the outcome of single PU classifiers trained on either different learning models or datasets into final prediction [48, 49], and shown to be better than the other binary and PU learning-based methods. In the next sections, we are going to discuss more the advantages and disadvantages of the machine learning-based methods for the disease gene prediction task.

Advantages and disadvantages of the prediction models

The 'No Free Lunch' theorem in machine learning states that no one algorithm works best for every problem [108]. Many factors need to be considered such as properties of the data (e.g. number of training samples, dimensionality, datatype, relationship, sparsity, distribution, etc.), system requirement (e.g. speed, performance, memory usage, etc.) the problem itself (e.g. linearly/nonlinearly separable), etc. when assessing the use of machine learning models for a problem. For the disease gene prediction, those factors have also been considered. For instance, the relationships between diseases and genes are known to be non-linear. In addition, to increase the number of training samples, general disease, disease group or ones having many known associated genes are often considered. Features of a gene or a disease–gene association are carefully selected or automatically learned from the data.

In the previous section, we have drawn a roadmap of machine learning-based methods for the disease gene prediction problem and examined their performance for general disease with

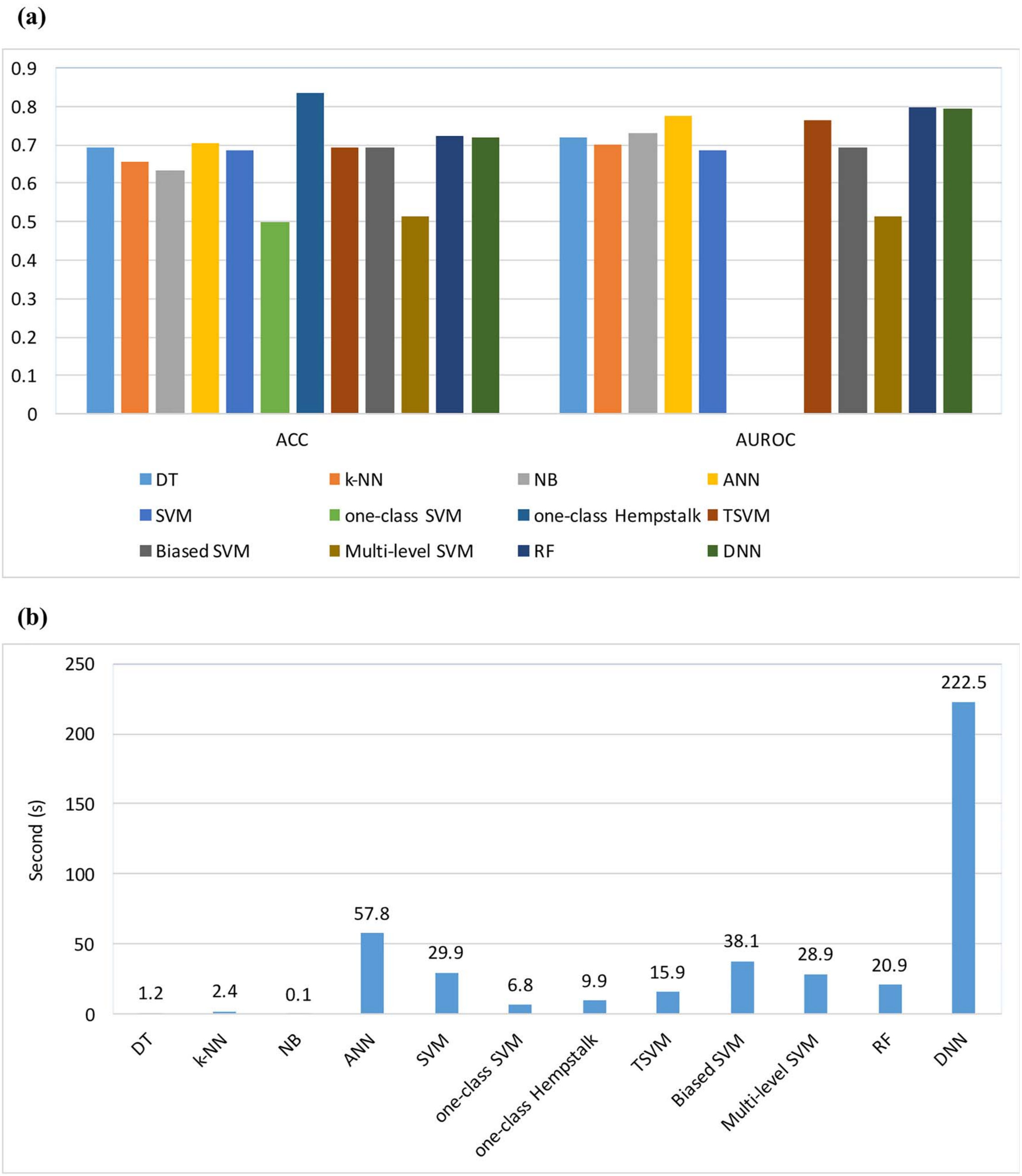

Figure 5. Comparison of 12 classification methods for the disease gene prediction in terms of (**A**) ACC and AUROC and (**B**) running time.

10 carefully designed features. The roadmap showed that the evolution in terms of both approach and the prediction performance of machine learning methods for the problem started with traditional binary, then unary, semi-supervised classifiers, and recently with modern machine learning models such as ensemble and deep learning. However, after examining some of those methods, we observed that the unary classifiers were better/worse in terms of ACC than traditional binary classifiers depending on the one-class classification algorithm. In addition, it was stated that the multi-level SVM classifier better fits the problem than the biased SVM classifier [46]; however, its performance in terms of both ACC and AUROC is worse than that of the biased one in our experiment.

For the traditional binary classifiers, DT [62, 63] and ANN [72] appeared to be among the best methods in terms of both ACC and AUROC. They only go after the best method (i.e. one-class Hempstalk) in terms of ACC and the two modern methods RF and DNN in terms of both performance measures. Interestingly, RF [60] is an ensemble of DT, and DNN [61] is also derived from ANN. Thus, we could conclude that tree- and neural network-based methods are among the most suitable techniques for the disease gene prediction.

Considering the running time (Figure 5B) and both performance measures, the best methods would be tree-based methods (i.e. DT and RF). In addition, besides a disadvantage as DT is prone to overfitting, DT has many advantages such as it is robust to outliers, scalable and able to naturally model non-linear decision boundaries thanks to their hierarchical structure. Moreover, it requires little effort in data preparation since data normalization and scaling are not required for DT [62, 63]. More importantly, it is easy to interpret visually when the trees only contain several levels. As a tree-based ensemble method, RF is not as easy to visually interpret and requires more running time; however, it has many advantages such as not prone to overfitting, reducing the variance, and it can handle data with high dimension, feature correlation, class imbalance and missing values. In addition, RF can extract important features; this also helps interpret the model's prediction. In contrast, neural network-based methods such as ANN and DNN are good to model the non-linear data with a large number of input features. However, they are black-box models and computationally expensive. Especially for deep learning models, they often require a very large amount of data for training and much more expertise to tune the model (i.e. setting the architecture and hyperparameters).

Besides the abovementioned tree- and neural network-based methods, the binary SSL-based method, TSVM [101], was also shown good performance in terms of both performance measures. It also performed better than the binary SVM in terms of AUROC. This indicates the important contribution of unlabeled data (in this case, they are neighbors of known disease genes in the human PPI network) to the prediction performance. However, as an SVM-based method, TSVM requires a large amount of time to process for a large dataset and require more expertise to tune the model (i.e. kernel functions and hyper-parameters).

Finally, NB and k-NN achieve moderate performance in terms of both measures. Common advanced features of the two methods are that they are both simple to implement (e.g. little parameter tuning is required) and computationally fast. However, NB requires an assumption of the independence between features [67]; meanwhile, k-NN does not work very well on datasets with a large number of features [65]. In the next section, the interpretability and trust of the prediction models for the disease gene prediction will also be discussed.

Interpretability and trust of the prediction models

To build trust with practitioners once a prediction is made, we must make the prediction models more interpretable. Unfortunately, most of the advanced machine learning models such as ensemble [29, 48, 49] and deep neural [53–55, 57] models are complex black boxes that are not able to explain why they reached the prediction. There is often a trade-off between the interpretability and the accuracy of a prediction model [109]. If a model is simple, it is often more interpretable but also yields lower accuracy. However, the definition of simplicity is also ambiguous since what is simple for one person may not be so for another. Unfortunately, the disease-gene association is a complex relationship; thus, advanced machine learning methods are more useful to model such the relationship. Thus, we should not expect the advanced methods are simple to be interpretable.

There are some ways to trust the result of a machine learning model for the disease gene prediction problem. Firstly, a model is often derived from well-established mathematical and statistical theories [110]; thus, its prediction can be trusted. Secondly, if a model provides valid reasons for its prediction, it builds our trust for that model. For example, if the model can select which features are important in its prediction and to which features it gave more weight, it is reasonable for the practitioners to trust that model. For our experiments, all features of genes/proteins were carefully selected based on evidence of the significant difference between disease and non-disease genes/proteins (see Building the feature set section). However, those features may affect differently on the prediction specified with different weights in the trained model. Thirdly, the model should show its ability to make reliable predictions. Therefore, if we could provide other sources of trust, such as its ability to correctly predicting for unseen data (i.e. not in the training set), then its prediction can be trusted. This can be done by separating the known data from a data source into training and testing sets, where the training set is used to build the model, while the testing set is used to test how well the trained model can predict the unseen data. This procedure can be extended in a way that the model is trained on a data source and then tested in an independent one. More interestingly, for the disease gene prediction, the trust can be strengthened in that novel predictions (i.e. novel disease-associated genes) can be either experimentally validated or supported by evidence from the literature.

A reason that affects the trust of a practitioner once a prediction is made is its uncertainty. The uncertainty of a prediction is especially important in biomedical applications, in which the disease gene prediction is one of them. Therefore, quantifying the uncertainty of a prediction is a crucial issue in predicting disease–gene associations. Basically, uncertainties can be derived from aleatoric and epistemic sources [111]. The former (i.e. aleatoric uncertainty) is also known as statistical uncertainty, where the returned outcome may differ each time we run the same experiment. For the disease gene prediction problem, this kind of uncertainty can be derived from the complex relationships between diseases and genes. The later one (i.e. epistemic uncertainty) is also known as systematic uncertainty and is due to a lack of knowledge about the problem. For instance, given a disease of interest, the number of known associated genes may not be enough for training the prediction model. This is the reason why most of the machine learning models have been proposed for general disease, disease group, or ones with many known associated genes. These two sources of uncertainty can be solved by proposing advanced machine learning methods and providing enough data to better learn the complex relationships between diseases and genes. Indeed, more modern machine learning techniques such as ensemble [29, 48, 49] and deep learning [53–55, 57] models have improved the prediction performance.

The uncertainty of model prediction is generally measured by the mean and variance of a set of predictions from the model. To generate the set of predictions, methods are often proposed to make multiple sets of training data from the original one or to make multiple models by changing the model's parameters, consequently making multiple predictions. To build new datasets, bootstrap sampling is a representative technique that builds new datasets by sampling with replacement from the original dataset [112]. However, this technique can be applied only to simple models with normality assumptions are made about the sampling distributions. Ensemble methods with bootstrap sampling techniques have been used for the disease gene prediction [29, 47]. For complex models in terms of the size and architecture, such as deep learning models, which require a significant amount of time to fit, other techniques have been proposed. For instance, dropout, an empirical technique to prevent overfitting in deep learning models, was proposed to estimate the model uncertainty at prediction time [113]. Indeed, the technique ignores a random subset of neurons in a network layer at every batch evaluation; thus, it was used to make multiple models (i.e. different sub-networks with a different subset of hyper-parameters).

Conclusions and future perspectives

In general, machine learning-based approaches usually consider the disease gene prediction as a classification problem. Binary classification-based methods have limitations in defining the negative training set, including non-disease genes, because no proven non-disease genes exist. Meanwhile, unary classification-based methods relying only on the positive training set (i.e. known disease genes) can avoid the definition of the negative training set but ignore the unknown (remaining) set, which may contain unknown disease genes. Therefore, SSL methods, which can exploit both known and unknown sets, have shown to be more reasonable. Recently, advanced methods combining different strategies have taken advances of them, thus improved the prediction performance. In this study, the roadmap of machine learning-based methods for the disease gene prediction was drawn. Twelve typical methods were further reviewed and compared in terms of both prediction performance and running time. The advantages and disadvantages of each model were also analyzed. Finally, their interpretability and the trust of their prediction were also discussed.

Except for the unary classification-based methods, other classification-based methods have to define the negative training set clearly or implicitly. Thus, to avoid this definition, graph-based SSL methods, which connect hidden and observed labels represented as nodes on a graph, then information is propagated from observed labels over the graph, should be considered for the disease gene prediction problem. Indeed, a conditional random field, a discriminative model specified over an undirected graph, was proposed for candidate gene prioritization by simultaneously exploiting both network and annotation information directly without attempting to convert the network information into features or vice versa [114]. This is the reason why network-based approaches, which rely on natural connections (e.g. in protein-interaction networks or gene networks) between known and unknown disease genes in the form of 'disease module,' are usually used for this problem (also known as network medicine) [3, 115–117]. Recent MF-based methods [50–52] are partially based on this idea in a way that known disease–gene associations are represented as a bipartite network; meanwhile, the connections between genes and between diseases are represented by similarity networks of genes and diseases, respectively. The network-based approaches can also ignore the design of features for genes, where their properties are often embedded in the networks.

Finally, most machine learning-based techniques for the disease gene prediction problem are supervised or semi-supervised learning. This is also a limitation since some diseases do not have a known molecular basis, which also means that the construction of labeled data is impossible since there are no known genes associated with them prior to the learning task. Therefore, unsupervised learning models should be used for this kind of disease. Accompany with advances of MKL for the data integration, unsupervised MKL [118] can be considered for this case.

Key Points

- Disease gene prediction, an important issue in biomedical research, is usually approached as a classification problem in machine learning.
- A roadmap of machine learning-based methods for the disease gene prediction (from traditional binary to unary, to semi-supervised learning and to modern methods) was drawn and reviewed.
- Twelve representative machine learning-based methods for the disease gene prediction were examined and compared in terms of prediction performance and running time.
- The advantages and disadvantages of the machine learning-based methods for the disease gene prediction were analyzed.
- The interpretability and the trust of the machine learning-based models for the disease gene prediction were discussed.

Funding

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.01-2017.14.

References

1. Kann MG. Advances in translational bioinformatics: computational approaches for the hunting of disease genes. *Brief Bioinform* 2009;**11**(1):96–110.
2. Tranchevent L-C, *et al.* A guide to web tools to prioritize candidate genes. *Brief Bioinform* 2010;**12**(1):22–32.
3. Wang X, Gulbahce N, Yu H. Network-based methods for human disease gene prediction. *Brief Funct Genomics* 2011;**10**(5):280–293.
4. Turner F, Clutterbuck D, Semple C. POCUS: mining genomic sequence annotation to predict disease genes. *Genome Biol* 2003;**4**(11):R75.
5. Adie EA, *et al.* SUSPECTS: enabling fast and effective prioritization of positional candidates. *Bioinformatics* 2006;**22**(6):773–774.
6. Aerts S, *et al.* Gene prioritization through genomic data fusion. *Nat Biotechnol* 2006;**24**(5):537–544.
7. Chen J, *et al.* Improved human disease candidate gene prioritization using mouse phenotype. *BMC Bioinformatics* 2007;**8**(1):392.
8. Le D-H, Kwon Y-K. GPEC: a Cytoscape plug-in for random walk-based gene prioritization and biomedical evidence collection. *Comput Biol Chem* 2012;**37**:17–23.
9. Le D-H, Kwon Y-K. Neighbor-favoring weight reinforcement to improve random walk-based disease gene prioritization. *Comput Biol Chem* 2013;**44**:1–8.
10. Le D-H, Dang V-T. Ontology-based disease similarity network for disease gene prediction. *Vietnam J Comput Sci* 2016;**3**(3):197–205.
11. Le D-H, Pham V-H. HGPEC: a Cytoscape app for prediction of novel disease-gene and disease-disease associations and evidence collection based on a random walk on heterogeneous network. *BMC Syst Biol* 2017;**11**(1):61.

12. Tarca AL, *et al.* Machine learning and its applications to biology. *PLoS Comput Biol* 2007;**3**(6):e116.
13. Yousef A, Moghadam Charkari N. A novel method based on new adaptive LVQ neural network for predicting protein-protein interactions from protein sequences. *J Theor Biol* 2013;**336**:231–239.
14. Li Y, Wu F-X, Ngom A. A review on machine learning principles for multi-view biological data integration. *Brief Bioinform* 2018;**19**(2):325–340.
15. Yip KY, Cheng C, Gerstein M. Machine learning and genome annotation: a match meant to be? *Genome Biol* 2013;**14**(5):205.
16. Basford KE, McLachlan GJ, Rathnayake SI. On the classification of microarray gene-expression data. *Brief Bioinform* 2013;**14**(4):402–410.
17. Le D-H, Van NT. Meta-analysis of whole-transcriptome data for prediction of novel genes associated with autism spectrum disorder. In: *Proceedings of the 8th International Conference on Computational Systems-Biology and Bioinformatics*. Nha Trang City, Viet Nam: ACM, 2017, 56–61.
18. Maetschke SR, *et al.* Supervised, semi-supervised and unsupervised inference of gene regulatory networks. *Brief Bioinform* 2013.
19. Ding H, *et al.* Similarity-based machine learning methods for predicting drug-target interactions: a brief review. *Brief Bioinform* 2013.
20. Le D-H, Nguyen D-P, Dao A-M. Significant path selection improves the prediction of novel drug-target interactions. In: *SoICT 2016*. Ho chi Minh City, Vietnam: ACM, 2016, 30–35.
21. Upstill-Goddard R, *et al.* Machine learning approaches for the discovery of gene-gene interactions in disease data. *Brief Bioinform* 2012;**14**(2):251–260.
22. Okser S, Pahikkala T, Aittokallio T. Genetic variants and their interactions in disease risk prediction—machine learning and network perspectives. *BioData Min* 2013; **6**(1):5.
23. Chen H, *et al.* The rise of deep learning in drug discovery. *Drug Discov Today* 2018.
24. Nguyen PH, Le D-H. Drug repositioning by bipartite local models. In: *2018 5th NAFOSTED Conference on Information and Computer Science (NICS)*, 2018.
25. Le D-H, Nguyen-Ngoc D. Drug repositioning by integrating known disease-gene and drug-target associations in a semi-supervised learning model. *Acta Biotheor* 2018;**66**(4):315–331.
26. Nguyen GTT, Le D-H. A matrix completion method for drug response prediction in personalized medicine. In: *Proceedings of the Ninth International Symposium on Information and Communication Technology*. Danang City, Viet Nam: ACM, 2018, 410–415.
27. Le D-H, Pham V-H. Drug response prediction by globally capturing drug and cell line information in a heterogeneous network. *J Mol Biol* 2018;**430**(18, Part A): 2993–3004.
28. Le D-H, Nguyen-Ngoc D. Multi-task regression learning for prediction of response against a panel of anti-cancer drugs in personalized medicine. In: *2018 1st International Conference on Multimedia Analysis and Pattern Recognition (MAPR)*, 2018.
29. Le D-H, Xuan Hoai N, Kwon Y-K. A comparative study of classification-based machine learning methods for novel disease gene prediction. In: Nguyen V-H, Le A-C, Huynh V-N (eds). *Knowledge and Systems Engineering*, Vol. **577–588**. Springer International Publishing, 2015.
30. Le D-H, Nguyen M-H. Towards more realistic machine learning techniques for prediction of disease-associated genes. In: *Proceedings of the Sixth International Symposium on Information and Communication Technology*. Hue City, Vietnam: ACM, 2015, 116–120.
31. Lospez-Bigas N, Ouzounis CA. Genome-wide identification of genes likely to be involved in human genetic disease. *Nucleic Acids Res* 2004;**32**(10):3108–3114.
32. Adie E, *et al.* Speeding disease gene discovery by sequence based candidate prioritization. *BMC Bioinformatics* 2005;**6**(1):55.
33. Xu J, Li Y. Discovering disease-genes by topological features in human protein-protein interaction network. *Bioinformatics* 2006;**22**(22):2800–2805.
34. Calvo S, *et al.* Systematic identification of human mitochondrial disease genes through integrative genomics. *Nat Genet* 2006;**38**(5):576–582.
35. Lage K, *et al.* A human phenome-interactome network of protein complexes implicated in genetic disorders. *Nat Biotechnol* 2007;**25**(3):309–316.
36. Smalter A, Lei SF, Chen X-W. Human disease-gene classification with integrative sequence-based and topological features of protein-protein interaction networks. In: *IEEE International Conference on Bioinformatics and Biomedicine, 2007. BIBM 2007*. IEEE, 2007.
37. Radivojac P, *et al.* An integrated approach to inferring gene–disease associations in humans. *Proteins* 2008;**72**(3):1030–1037.
38. Keerthikumar S, *et al.* Prediction of candidate primary immunodeficiency disease genes using a support vector machine learning approach. *DNA Res* 2009;**16**(6):345–351.
39. Jiabao S, Patra JC, Yongjin L. Functional link artificial neural network-based disease gene prediction. In: *International Joint Conference on Neural Networks, 2009. IJCNN 2009*, 2009.
40. De Bie T, *et al.* Kernel-based data fusion for gene prioritization. *Bioinformatics* 2007;**23**(13):i125–i132.
41. Yu S, *et al.* Comparison of vocabularies, representations and ranking algorithms for gene prioritization by text mining. *Bioinformatics* 2008;**24**(16):i119–i125.
42. Yu S, *et al.* Gene prioritization and clustering by multi-view text mining. *BMC Bioinformatics* 2010;**11**(1):28.
43. Schölkopf B, *et al.* Estimating the support of a high-dimensional distribution. *Neural Comput* 2001;**13**(7):1443–1471.
44. Lanckriet GRG, *et al.* Learning the kernel matrix with semidefinite programming. *J Mach Learn Res* 2004;**5**:27–72.
45. Nguyen T-P, Ho T-B. Detecting disease genes based on semi-supervised learning and protein-protein interaction networks. *Artif Intell Med* 2012;**54**(1):63–71.
46. Yang P, *et al.* Positive-unlabeled learning for disease gene identification. *Bioinformatics* 2012;**28**(20):2640–2647.
47. Mordelet F, Vert J-P. ProDiGe: prioritization of disease genes with multitask machine learning from positive and unlabeled examples. *BMC Bioinformatics* 2011;**12**(1):389.
48. Yang P, *et al.* Ensemble positive unlabeled learning for disease gene identification. *PLoS One* 2014;**9**(5):e97079.
49. Jowkar G-H, Mansoori EG. Perceptron ensemble of graph-based positive-unlabeled learning for disease gene identification. *Comput Biol Chem* 2016;**64**:263–270.
50. Natarajan N, Dhillon IS. Inductive matrix completion for predicting gene–disease associations. *Bioinformatics* 2014;**30**(12):i60–i68.
51. Luo P, *et al.* *Predicting Gene-Disease Associations with Manifold Learning*. Cham: Springer International Publishing, 2018.

52. Zeng X, *et al*. Probability-based collaborative filtering model for predicting gene–disease associations. *BMC Med Genet* 2017;**10**(5):76.
53. Han P, *et al*. GCN-MF: disease-gene association identification by graph convolutional networks and matrix factorization. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. Anchorage, AK, USA: Association for Computing Machinery, 2019, 705–713.
54. Luo P, *et al*. Enhancing the prediction of disease–gene associations with multimodal deep learning. *Bioinformatics* 2019;**35**(19):3735–3742.
55. Barman RK, *et al*. Identification of infectious disease-associated host genes using machine learning techniques. *BMC Bioinformatics* 2019;**20**(1):736.
56. Koohi-Moghadam M, *et al*. Predicting disease-associated mutation of metal-binding sites in proteins using a deep learning approach. *Nat Mach Intell* 2019;**1**(12):561–567.
57. Peng J, Guan J, Shang X. Predicting Parkinson's disease genes based on Node2vec and autoencoder. *Front Genet* 2019;**10**(226).
58. Chen X, *et al*. A deep learning approach to identify association of disease-gene using information of disease symptoms and protein sequences. *Anal Methods* 2020.
59. Hempstalk K, Frank E, Witten I. One-class classification by combining density and class probability estimation. In: Daelemans W, Goethals B, Morik K (eds). *Machine Learning and Knowledge Discovery in Databases*. Berlin Heidelberg: Springer, 2008, 505–519.
60. Breiman L. Random forests. *Mach Learn* 2001;**45**(1):5–32.
61. LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature* 2015;**521**(7553):436–444.
62. Breiman L, *et al*. *Classification and Regression Trees*. Monterey, CA: Wadsworth & Brooks, 1984.
63. Quinlan JR. Induction of decision trees. *Mach Learn* 1986;**1**(1):81–106.
64. Amberger J, *et al*. McKusick's online Mendelian inheritance in man (OMIM®). *Nucleic Acids Res* 2009;**37**(suppl 1):D793–D796.
65. Altman NS. An introduction to kernel and nearest-neighbor nonparametric regression. *Am Stat* 1992;**46**(3):175–185.
66. Tu Z, *et al*. Further understanding human disease genes by comparing with housekeeping genes and other genes. *BMC Genomics* 2006;**7**(1):31.
67. Rish I. An empirical study of the naive Bayes classifier. In: *IJCAI 2001 workshop on empirical methods in artificial intelligence*, 2001.
68. Schapire RE. *A brief introduction to boosting*. In: *IJCAI*, 1999.
69. Prokisch H, *et al*. MitoP2: the mitochondrial proteome database-now including mouse data. *Nucleic Acids Res* 2006;**34**(suppl 1):D705–D711.
70. Safran M, *et al*. GeneCards TM 2002: towards a complete, object-oriented, human gene compendium. *Bioinformatics* 2002;**18**(11):1542–1543.
71. Cortes C, Vapnik V. Support-vector networks. *Mach Learn* 1995;**20**(3):273–297.
72. Lek S, Park YS. Artificial Neural Networks. In: Jørgensen SE, Fath BD (eds). *Encyclopedia of Ecology*. Oxford: Academic Press, 2008, 237–245.
73. Sun J, Patra JC, Li Y. Functional link artificial neural network-based disease gene prediction. In: *Proceedings of the 2009 International Joint Conference on Neural Networks*. Atlanta, Georgia, USA: IEEE Press, 2009, 425–432.
74. Xiao Y, *et al*. Differential expression pattern-based prioritization of candidate genes through integrating disease-specific expression data. *Genomics* 2011;**98**(1):64–71.
75. Martinus D, Tax J. One-class classification: concept-learning in the absence of counterexamples. PhD thesis, Delft University of Technology, 2001.
76. Cunningham F, *et al*. Ensembl 2019. *Nucleic Acids Res* 2018;**47**(D1):D745–D751.
77. Chapelle O, Schölkopf B, Zien A. *Semi-supervised Learning*, Vol. **2**. Cambridge: MIT Press, 2006.
78. Shi M, Zhang B. Semi-supervised learning improves gene expression-based prediction of cancer recurrence. *Bioinformatics* 2011;**27**(21):3017–3023.
79. Qi Y, *et al*. Semi-supervised multi-task learning for predicting interactions between HIV-1 and human proteins. *Bioinformatics* 2010;**26**(18):i645–i652.
80. Sabes PN, Jordan MI. Advances in neural information processing systems. In: Tesauro G, Touretzky D, Leed T (eds). *Advances in Neural Information Processing Systems*. CiteSeer, 1995.
81. Zhu X, Ghahramani Z, Lafferty J. *Semi-supervised Learning Using Gaussian Fields and Harmonic Functions*. ICML, 2003.
82. Denis F, Gilleron R, Letouzey F. Learning from positive and unlabeled examples. *Theor Comput Sci* 2005;**348**(1): 70–83.
83. Letouzey F, Denis F, Gilleron R. Learning from positive and unlabeled examples. In: Arimura H, Jain S, Sharma A (eds). *Algorithmic Learning Theory*. Berlin Heidelberg: Springer, 2000, 71–85.
84. Liu B, *et al*. Partially supervised classification of text documents. In: *Machine Learning-International Workshop Then Conference*, 2002, 387–394.
85. Mordelet F, Vert JP. A bagging SVM to learn from positive and unlabeled examples. *Pattern Recognition Letters*, 2014;**37**:201–209.
86. Vanunu O, *et al*. Associating genes and protein complexes with disease via network propagation. *PLoS Computional Biology* 2010;**6**(1):e1000641.
87. Liu T, *et al*. *Partially Supervised Text Classification with Multi-Level Examples*. AAAI, 2011.
88. Grover A, Leskovec J. node2vec: scalable feature learning for networks. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.
89. Hinton GE, Salakhutdinov RR. Reducing the dimensionality of data with neural networks. *Science* 2006;**313**(5786):504–507.
90. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks arXiv preprint arXiv: 1609.02907. 2016.
91. Brown KR, Jurisica I. Online predicted human interaction database. *Bioinformatics* 2005;**21**(9):2076–2082.
92. The UniProt, C. The universal protein resource (UniProt) in 2010. *Nucleic Acids Res* 2010;**38**(suppl_1):D142–D148.
93. Freudenberg J, Propping P. A similarity-based method for genome-wide prediction of disease-relevant human genes. *Bioinformatics* 2002;**18**(suppl 2):S110–S115.
94. Jonsson PF, Bates PA. Global topological features of cancer proteins in the human interactome. *Bioinformatics* 2006;**22**(18):2291–2297.
95. Apweiler R, *et al*. The InterPro database, an integrated documentation resource for protein families, domains and functional sites. *Nucleic Acids Res* 2001;**29**(1):37–40.

96. Hunter S, *et al.* InterPro in 2011: new developments in the family and domain prediction database. *Nucleic Acids Res* 2011;**40**(D1):D306–D312.
97. Smedley D, *et al.* BioMart—biological queries made easy. *BMC Genomics* 2009;**10**(1):22.
98. Sayers EW, *et al.* Database resources of the National Center for biotechnology information. *Nucleic Acids Res* 2011;**39**(suppl 1):D38–D51.
99. Luo H, *et al.* DEG 10, an update of the database of essential genes that includes both protein-coding genes and noncoding genomic elements. *Nucleic Acids Res* 2014;**42**(D1):D574–D580.
100. Dennis G, *et al.* DAVID: database for annotation, visualization, and integrated discovery. *Genome Biol* 2003;**4**(9): R60.
101. Sindhwani V, Keerthi SS. *Large scale semi-supervised linear SVMs*. In: *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Seattle, Washington, USA: ACM, 2006, 477–484.
102. Hall M, *et al.* The WEKA data mining software: an update. *ACM SIGKDD Explor* 2009;**11**(1):10–18.
103. Chang C-C, Lin C-J. LIBSVM: a library for support vector machines. *ACM Trans Intell Syst Technol* 2011;**2**(3):27.
104. oneClassClassifier. *oneClassClassifier: Performs One-class Classification on a Dataset*. https://weka.sourceforge.io/packageMetaData/oneClassClassifier/index.html.
105. Lang S, *et al.* WekaDeeplearning4j: a deep learning package for Weka based on Deeplearning4j. *Knowl-Based Syst* 2019;**178**:48–50.
106. Sindhwani V, Keerthi SS. Newton methods for fast solution of semi-supervised linear SVMs. In: *Large Scale Kernel Machines*, 2007, 155–174.
107. Vasighizaker A, Sharma A, Dehzangi A. A novel one-class classification approach to accurately predict disease-gene association in acute myeloid leukemia cancer. *PLoS One* 2019;**14**(12):e0226115.
108. Fernández-Delgado M, *et al.* Do we need hundreds of classifiers to solve real world classification problems? *J Mach Learn Res* 2014;**15**(1):3133–3181.
109. Johansson U, *et al.* Trade-off between accuracy and interpretability for predictive in silico modeling. *Future Med Chem* 2011;**3**(6):647–663.
110. Sugiyama M. *Introduction to Statistical Machine Learning*. Morgan Kaufmann, 2015.
111. Kiureghian AD, Ditlevsen O. Aleatory or epistemic? Does it matter? *Struct Saf* 2009;**31**(2):105–112.
112. Hesterberg T. Bootstrap. *WIREs Comp Stats* 2011;**3**(6): 497–526.
113. Gal Y, Ghahramani Z. Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In: *Proceedings of the 33rd International Conference on International Conference on Machine Learning*, Vol. **48**. New York, NY, USA: JMLR.org, 2016, 1050–1059.
114. Xie B, *et al. Conditional random field for candidate gene prioritization*. In: *Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics*. ACM, 2013.
115. Barabasi A-L, Gulbahce N, Loscalzo J. Network medicine: a network-based approach to human disease. *Nat Rev Genet* 2011;**12**(1):56–68.
116. Zhang F, *et al.* A network medicine approach to build a comprehensive atlas for the prognosis of human cancer. *Brief Bioinform* 2016;**17**(6):1044–1059.
117. Piro RM. Network medicine: linking disorders. *Hum Genet* 2012;**131**(12):1811–1820.
118. Zhuang J, *et al.* Unsupervised multiple kernel learning. In: Chun-Nan H, Wee Sun L (eds). *Proceedings of the Asian Conference on Machine Learning*. PMLR: Proceedings of Machine Learning Research, 2011, 129–144.