Applied Network Science

RESEARCH Open Access

Feature extraction with spectral clustering for gene function prediction using hierarchical multi-label classification

Miguel Romero*, Oscar Ramírez, Jorge Finke and Camilo Rocha

*Correspondence: miguelangel.romero@javerianacali.edu.co

Department of Electronics and Computer Science, Pontificia Universidad Javeriana, Calle 18 N 118-250, Cali 760031, Colombia

Abstract

Gene annotation addresses the problem of predicting unknown associations between gene and functions (e.g., biological processes) of a specific organism. Despite recent advances, the cost and time demanded by annotation procedures that rely largely on in vivo biological experiments remain prohibitively high. This paper presents a novel in silico approach for to the annotation problem that combines cluster analysis and hierarchical multi-label classification (HMC). The approach uses spectral clustering to extract new features from the gene co-expression network (GCN) and enrich the prediction task. HMC is used to build multiple estimators that consider the hierarchical structure of gene functions. The proposed approach is applied to a case study on *Zea mays*, one of the most dominant and productive crops in the world. The results illustrate how in silico approaches are key to reduce the time and costs of gene annotation. More specifically, they highlight the importance of: (1) building new features that represent the structure of gene relationships in GCNs to annotate genes; and (2) taking into account the structure of biological processes to obtain consistent predictions.

Keywords: Hierarchical classification, Supervised learning, Spectral clustering, Shap values, Gene function prediction, *Zea mays*

Introduction

Identifying the association of genes to functions is key to gain insight into how genomes serve as blueprints for life, e.g., to develop treatments for specific conditions or enhance tolerance to environmental stresses (Rust et al. 2002; Vandepoele et al. 2009; Yandell and Ence 2012). Numerous studies have used co-expression data to predict specific biological functions and processes (Oti et al. 2008; Romero et al. 2020; Stuart 2003; van Dam et al. 2017). Intuitively, genes are reported to co-express whenever they are simultaneously active, which suggests that they are associated to one or more common biological processes.

Under this hypothesis, characterizing gene interactions as a gene co-expression network (GCN) may assist to identify unknown functional annotations in a genome. Co-expression networks are generally represented as undirected weighted graphs, where vertices denote genes and weighted edges indicate the strength of the co-expression

between two genes. A detailed analysis of the structure and distribution of gene relationships in GCNs provides additional clues that facilitate the prediction of gene functions (Valentini 2009).

However, the cost and time requirements to annotate genes using in vivo biological experimentation remains prohibitively high (Cho et al. 2015; Zhou et al. 2005). To overcome this limitation, hybrid approaches that integrate existing knowledge of gene-function associations and in silico methods have been proposed (Cho et al. 2016; Deng et al. 2003; Luo et al. 2007; Romero et al. 2022). While they have shown great promise, given the extreme combinatorial nature of the problem, annotating genes in an efficient manner remains an open challenge.

Functional annotations are defined by the Gene Ontology (GO), which contains three main types of annotations: biological processes, molecular functions, and cellular component (Gene Ontology Consortium 2019). These annotations, commonly known as GO terms, are structured in a hierarchy and defined as a directed acyclic graph (DAG). Gene annotation approaches generally ignore the relationships among biological processes, even though these relationships are key to improve the accuracy and avoid inconsistency in predictions. A prediction is said to be inconsistent w.r.t. the GO hierarchy when a gene is inferred to have a particular function a, but it is not inferred to have all ancestor of a. In other words, an inconsistent prediction states that the prediction does not satisfy the ancestral relations between GO terms. Satisfying ancestral constraints is often referred to as the *true-path rule* in GO (Valentini 2009; Ashburner et al. 2000) and as the *hierarchical constraint* in HMC (Vens et al. 2008).

This paper presents a feature extraction approach for in silico annotation of genes. It follows a network-based approximation that uses cluster analysis and hierarchical multi-label classification (HMC) for building a predictor that assigns functions to genes satisfying the true-path rule. Cluster analysis plays the role of enriching the information available for predicting gene-function associations by extracting new features that represent structural properties of the GCN. That is, co-expression relations are used to identify gene clusters that ultimately help in associating functions to genes (i.e., guilt by association, see Petsko (2009)). It has been shown in Romero et al. (2022) that new features built from the GCN and associations between genes and functions with the spectral clustering algorithm are key to improve the prediction performance in the gene annotation problem. The results in Romero et al. (2022) show that using other features associated to structural properties of the GCN and gene functional information lead to lower performance.

Furthermore, the extracted features are filtered (using SHAP) based on their impact in the prediction task and HMC is used to predict gene-function associations that take into account the relations between biological functions. The proposed approach illustrates how the performance of gene annotation is improved by combining: (1) new information extracted from the GCN; and (2) classification methods that consider the relation between gene functions.

This approach is applied to a case study on *Zea mays*, one of the most dominant and productive crops. *Zea mays* serves a variety of purposes, including animal feed and derivatives for human consumption and ethanol (Zhou et al. 2020). The co-expression information used in the study is imported from the ATTED-II database (Obayashi et al.

2018). The resulting GCN, modeled as a weighted graph, comprises 26,131 vertices (i.e., genes) and 44,621,533 edges. The functional information (i.e., known gene-function associations) is taken from DAVID Bioinformatics Resources (Huang et al. 2009). It contains a total of 255,865 annotations of biological processes for maize, i.e., pathways to which a gene contributes. The results highlight the importance of extracted features that represent structural properties of the GCN and the hierarchical structure of biological processes with HMC to improve prediction performance. Ultimately, the results provide experimental (in silico) evidence that the proposed approach is a viable and promising approximation to gene function prediction.

This paper is a significant extended version of Romero et al. (2022) that:

- Addresses the gene function prediction as a hierarchical multi-label classification problem by considering the structure of gene functions. That is ancestral relationships are represented as a DAG (Gene Ontology Consortium 2019).
- Analyzes a larger functional database for the case study of maize. The number of genes associated to at least one function increased from 5361 to 10,049. The new dataset consists of 255 865 associations between genes and functions, and 7021 relations between functions.
- Concludes that the ancestral relations between functions and the features extracted from the GCN improve the prediction performance in the gene function prediction task when addressed as a hierarchical multi-label classification problem.

The remainder of the paper is organized as follows. "Preliminaries" section reviews some preliminaries. "Clustering-based feature extraction" section introduces the approach to extract features from the gene co-expression network using cluster analysis. The proposed approach to predict gene functions, based on hierarchical multi-label classification is presented in "Hierarchical multi-label classification for gene function prediction" section. "Case study: Zea mays" section presents the case study for the *Zea mays* species. Finally, "Related work and concluding remarks" section draws some concluding remarks and future research directions.

Preliminaries

This section presents preliminaries on spectral clustering, gene co-expression networks, gene function prediction, hierarchical multi-label classification, and SHAP feature contribution.

Spectral clustering

The aim of applying cluster analysis on a network is to identify groups of vertices sharing a (parametric) notion of similarity (Yu 2003; Rodriguez et al. 2019). Usually, distance or centrality metrics are used for clustering. Spectral clustering is a clustering method with foundations in algebraic graph theory (Jia et al. 2014). It has been shown that spectral clustering has better overall performance across different areas of applications (Murugesan et al. 2021). Given a graph G, the spectral clustering decomposition of G can be represented by the equation $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{L}$ is the Laplacian, $\mathbf{D}$ is the degree (i.e., a diagonal matrix with the number of edges incident to each node), and $\mathbf{A}$ the adjacency

matrices of G. Spectral clustering uses, say, the n eigenvectors associated to the n smallest nonzero eigenvalues of $\mathbf{L}$. In this way, each node of the graph gets a coordinate in $\mathbb{R}^n$. The resulting collection of eigenvectors serve as input to a clustering algorithm (e.g., k-means) that groups the nodes in n clusters.

Gene co-expression network

A gene co-expression network (GCN) is represented as an undirected graph where each vertex represents a gene and each edge the level of co-expression between two genes.

Definition 1 Let V be a set of genes, E a set of edges that connect pairs of genes, and $w : E \to \mathbb{R}_{\geq 0}$ a weight function. A *(weighted) gene co-expression network* is a weighted graph $G = (V, E, w)$.

The set of genes V in a co-expression network is particular to the genome under study. The correlation of expression profiles between each pair of genes is measured, commonly, using the Pearson correlation coefficient. Every pair of genes is assigned and ranked according to a relationship measure, and a threshold is used as a cut-off value to determine E. The weight function w denotes the strength of the co-expression between each pair of genes in V. For example, in the ATTED-II database, the co-expression relation between any pair of genes is measured as a z-score expressed as a function of the co-expression index LS (Logit Score) (Obayashi et al. 2018; Obayashi and Kinoshita 2011).

Gene function prediction

In an annotated gene co-expression network, each gene is associated with the collection of biological functions to which it is related (e.g., through in vivo experiments).

Definition 2 Let A be a set of biological functions. An *annotated gene co-expression network* is a gene co-expression network $G = (V, E, w)$ complemented with an annotation function $\phi : V \to 2^A$.

The problem of predicting gene functions can be explained as follows. Given an annotated co-expression network $G = (V, E, w)$ with annotation function ϕ, the goal is to use the information represented by ϕ, together with additional information (e.g., features of G), to obtain a function $\psi : V \to 2^A$ that extends ϕ. Associations between genes and functions not present in ϕ have either not been found through in vivo experiments, or do not exist in a biological sense. The new associations identified by ψ are a suggestion of functions that need to be verified through in vivo experiments. The function ψ can be built from a predictor of gene functions, e.g., based on a supervised machine learning model.

Hierarchical multi-label classification

Node classification refers to the task of predicting a node class for an input data based on the information of other nodes in the network (Bhagat et al. 2011). In general, node classification problems can be categorize into three different types: *binary classification* refers to predict one attribute (target) with two classes (for example,

positive and negative) (Khan and Madden 2010); *multi-class classification* refers to the case where the attribute to be predicted has more than two classes and are mutually exclusive (for example, the brand of a car) (Mills 2021); and *multi-label classification* refers to predicting an attribute with at least two classes, but where an instance could be associated to more than one class (for example, the gene function prediction problem) (Xu et al. 2020).

Although the aforementioned prediction methods are frequently used, they do not consider hierarchical relations between classes. For such scenarios, hierarchical multi-label classification (HMC) addresses the task of structured output prediction where the classes are organized into a hierarchy and an instance may belong to multiple classes. In many problems, such as gene function prediction, classes inherently satisfy these conditions (Levatić et al. 2015). Authors in Silla and Freitas (2011) expose that there are two types of methods to explore the hierarchical structure. First, *top-down or local classifiers* refer to partially predict the classes in the hierarchy from the top to the bottom. Second, *big-bang or global classifiers* refer to use a single classifier that considers the entire hierarchy at once.

Classifiers that ignore the class relationships, by predicting only the leaf classes in the hierarchy or predicting each class independently, often lead to *inconsistent predictions*. This refers to the fact that a node is inferred to have a particular class a, but the outcome of the classifier fails to infer the node's association to all ancestor classes of a in the hierarchy. In other words, an inconsistent prediction states that the prediction does not satisfy the hierarchy for some class a. Satisfying ancestral constraints is often referred to as the *true-path rule* in GO (Valentini 2009; Ashburner et al. 2000) and as the *hierarchical constraint* in HMC (Vens et al. 2008).

Figure 1 illustrates the four HMC methods used in this work: *Local classifier per node* (*lcn*) consists of training one binary classifier for each class in the hierarchy except the root. *Local classifier per parent node* (*lcpn*) consists of training a multi-label classifier for each parent node in the hierarchy to distinguish between its child classes. *Local classifier per level* (*lcl*) consists of training one multi-label classifier for each level of the class hierarchy except for the root. *Global classifier* consists of building a single multi-label classifier taking into account the hierarchy as a whole during a single run. The global classifier can assign classes at potentially every level of the hierarchy to an instance.

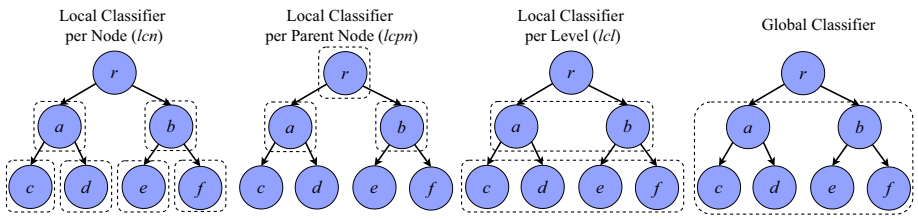

Fig. 1 Example of global and local methods for hierarchical multi-label classification. Given a hierarchy of classes (r, a, b, c, d, e, and f), the dashed boxes show the number of classifiers required for each method. Note that the *lcn*, *lcpn*, *lcl*, and global classifiers require 6, 4, 3, and 1 predictors, respectively

SHAP feature contribution

The performance of classification algorithms is partly determined by the features used to train a particular predictor. SHAP (SHapley Additive exPlanation) is a framework that computes the importance values for each feature in a dataset using concepts from game theory (Lundberg and Lee 2017; Lundberg et al. 2020). SHAP assigns Shapely values to explain which features in the model are the most important for prediction by calculating the changes in the prediction when features are conditioned. Given a predictor and a training set, SHAP computes a matrix with the same dimensions of the predictor's output containing the Shapely values for each instance and class. For example, in a binary classification problem and a training set of n instances, the output of SHAP is a matrix of dimension $n \times 2$ (there are two classes, positive and negative). In multi-label classification problems, the output is a matrix of dimension $n \times 2$ for each class, since classes are not mutually exclusive and the outcome is either positive or negative for each class.

Clustering-based feature extraction

The approach for extracting features from the GCN using a clustering algorithm and Gene Ontology term enrichment is presented. It combines information from the GCN, and the associations between genes and functions to create features capturing topological properties of the GCN.

The inputs of the approach are a GCN, denoted by $G = (V, E, w)$, a set of (biological) functions A, an annotation function $\phi : V \to 2^A$, and a set $K = \{k_0, \ldots, k_{m-1}\}$ for sampling the number of clusters. The annotation function ϕ must satisfy true-path rule for the GO hierarchy (Ashburner et al. 2000; Valentini 2009). That is, if a gene is associated to a function, then it must also be associated to every ancestor of the function in the hierarchy, and if a gene is not associated to a function, then it must not be associated to any of its descendants.

The outputs are two feature matrices J_G and J_F, of dimension $V \times A \cdot K \to [0, 1]$, specifying the likelihood of the genes V to be associated to the functions in A when the graph is decomposed in m clusters. Matrices J_G and J_F correspond to the GCN (that is the graph G) and an affinity graph defined the next subsection.

The feature extraction approach consists of three stages, which are depicted in Fig. 2. First, an affinity graph F with information in ϕ is created from G. Second, the spectral clustering algorithm is applied to both G and its enriched version F for the m different number of clusters specified in K. Third, the Gene Ontology term enrichment technique is used to create m features for each function $a \in A$, corresponding to the number of clusters in K.

Affinity graph creation

An affinity graph $F = (V, E, w_F)$ between G and ϕ is built. Its weight function is defined as the mean between the co-expression weight specified by w and the proportion of shared functions between genes specified by ϕ.

Definition 3 The *weight function* $w_F : V \times V \to [0, 1]$ is defined for any $u, v \in V$ as

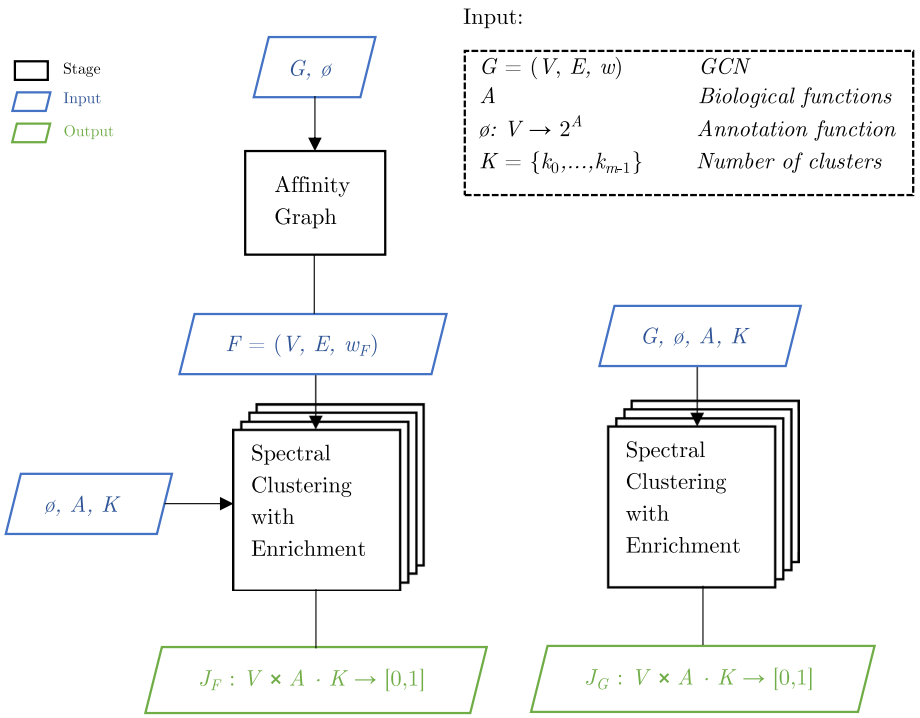

Fig. 2 The clustering-based feature extraction approach consists of three stages. Namely, creation of affinity graph, clustering computation, and Gene Ontology term enrichment. Its inputs are a GCN, denoted by $G = (V, E, w)$, a set of functions A, an annotation function $\phi : V \to 2^A$, and a set $K = \{k_0, \ldots, k_{m-1}\}$. Its output are two feature matrices (for both G and its enriched version F) of dimension $V \times A \cdot K \to [0, 1]$ that specify how likely it is for the genes to be associated to the functions in A when the graph is decomposed m clusters, each of size k_i, for $0 \leq i \leq m$

$$w_F(u, v) = \frac{1}{2}\left(\frac{w(u, v) - 1}{\max(w) - 1} + \frac{|\phi(u) \cup \phi(v)|}{|\phi(u) \cap \phi(v)|}\right),$$

where $\max(w)$ denotes the maximum value in the range of w (which exists because w is finite).

Under the assumption that at least one element in the range of w is greater than 1, it is guaranteed that the range of w_F is $[0, 1]$ (because $w : V \times V \to [1, \infty)$). This is indeed the case, in practice, because the co-expression between two genes in the GCN is quantified in terms of the z-score, which is highly unlikely to be 1 for all pairs of genes.

Gene clustering

The spectral clustering algorithm is applied independently to each graph $X \in \{G, F\}$ to decompose X (i.e., group the genes V) using the number of clusters specified by $K = \{k_0, \ldots, k_{m-1}\}$. The decomposition of X is performed m times, once per k in K. The adjacency matrices of the weighted and undirected graphs G and F are used as the precomputed affinity matrices required for the spectral clustering algorithm. The outcome of the clustering algorithm is an assignment from nodes to clusters of size k, for each $k \in K$. More precisely, the outputs of this stage are the matrices

$I_X : V \times K \rightarrow [0,1]$, where each column $0 \leq i < m$ represents the decomposition of X in k_i clusters.

Gene enrichment

The goal of this stage is to produce a matrix $J_X : V \times A \cdot K \rightarrow [0,1]$ for each $X \in \{G, F\}$, specifying how likely it is for the genes to be associated to every function $a \in A$ when X is decomposed in the given number of clusters.

For each decomposition from the previous stage (i.e., each column of the matrices I_X) and function $a \in A$, the resulting clusters are used to compute whether a significant number of members associated to function a is (locally) present. Intuitively, if genes that are grouped together have a strong co-expression relation and most of the group are associated to gene function a, then the remaining genes are also likely to be associated to a (i.e., guilt by association, see Petsko (2009)). In this way, for each $v \in V$, $a \in A$, and $k \in K$, the entry $J_X(v, a \cdot k)$ is a p-value indicating if the function a is over-represented in the decomposition of k clusters of X. This process is commonly known as Gene Ontology term enrichment and may use different statistical tests, such as, Fisher's exact test (Yon Rhee et al. 2008).

Hierarchical multi-label classification for gene function prediction

This section presents the approach for gene function prediction using HMC to create a predictor, enriched with the information of the features created in "Clustering-based feature extraction" section.

The GO hierarchy is defined as a directed acyclic graph (DAG) containing three main types of annotations: biological processes, molecular functions, and cellular component (Gene Ontology Consortium 2019). This work focuses on biological processes, i.e., a subgraph of the GO hierarchy that contains 28 roots (i.e., functions in the GO hierarchy with null indegree). This subgraph is denoted as $H = (A, R)$, where A is the set of biological processes and R the binary relation representing ancestral relations between pairs of biological processes (i.e., $(a, b) \in R$ means that function b is ancestor of function a in the GO hierarchy). The topological-sorting traversal algorithm presented in Romero et al. (2022) is used to transform the GO hierarchy of biological processes into a tree. As a result, the hierarchy is split into several components, i.e., subtrees of H called sub-hierarchies. Each sub-hierarchy, $H' = (A', R')$ with $A' \subseteq A$, $R' \subseteq R$, and $r \in A'$ the root, is associated to a subgraph $G' = (V', E', w)$ containing all genes $v \in V$ associated to r, i.e., $V' = \phi^{-1}(r)$. Note that, the proposed approach is independently applied to each sub-hierarchy.

The inputs of the approach are a sub-hierarchy $H' = (A', R')$, a subgraph of the GCN, denoted by $G' = (V', E', w)$, where $V' \subseteq V$ and $E' \subseteq E$, an annotation function $\phi : V \rightarrow 2^{A'}$, the matrices J_G and J_F resulting from "Clustering-based feature extraction" section, and a constant value $c \in [0,1]$ for feature selection. The output is a function $\psi : V' \times A' \rightarrow [0,1]$, specifying, for each gene $v \in V'$, the probability $\psi(v, a)$ of v being associated to function $a \in A'$.

First, sub-matrices J'_G and J'_F are created from J_G and J_F, by respectively considering only the genes $V' \subseteq V$ and functions $A' \subseteq A$. These sub-matrices represent structural properties of the GCN subgraph G', and associations between genes and functions based

on multiple partitions of each graph. Figure 3 illustrates the prediction approach. The reminder of this section is devoted to detailing the prediction approach.

SHAP filters the extracted features with more impact in the prediction task, and HMC is used to predict associations between genes and functions without inconsistencies (i.e., complying the true-path rule). Since local HMC methods use more than one predictor per hierarchy, the feature selection is executed for each predictor independently, considering only the features related to the functions being predicted, denoted by $A'' \subseteq A'$. For example, consider the function hierarchy and a local classifier per level method depicted in Fig. 4. The predictor for level 1 predicts functions *a* and *b*, so only the features associated to functions *a* and *b* are considered for the feature selection.

Feature selection

The aim of feature selection is to produce a matrix $J : V' \times \Theta(c) \rightarrow [0, 1]$ by selecting a reduced number of significant features from J'_G and J'_F. The number of selected features is denoted by $0 \leq \Theta(c) \leq 2m \cdot |A''|$, where $m \cdot |A''|$ is the number of features in each matrix J'_G and J'_F, denoted as q (that is $q = m \cdot |A''|$).

Feature selection is conveyed from J'_G and J'_F to J using SHAP. Let J'_{G+F} denote the matrix resulting from extending J'_G with the q features of J'_F. That is, for each $v \in V'$, the

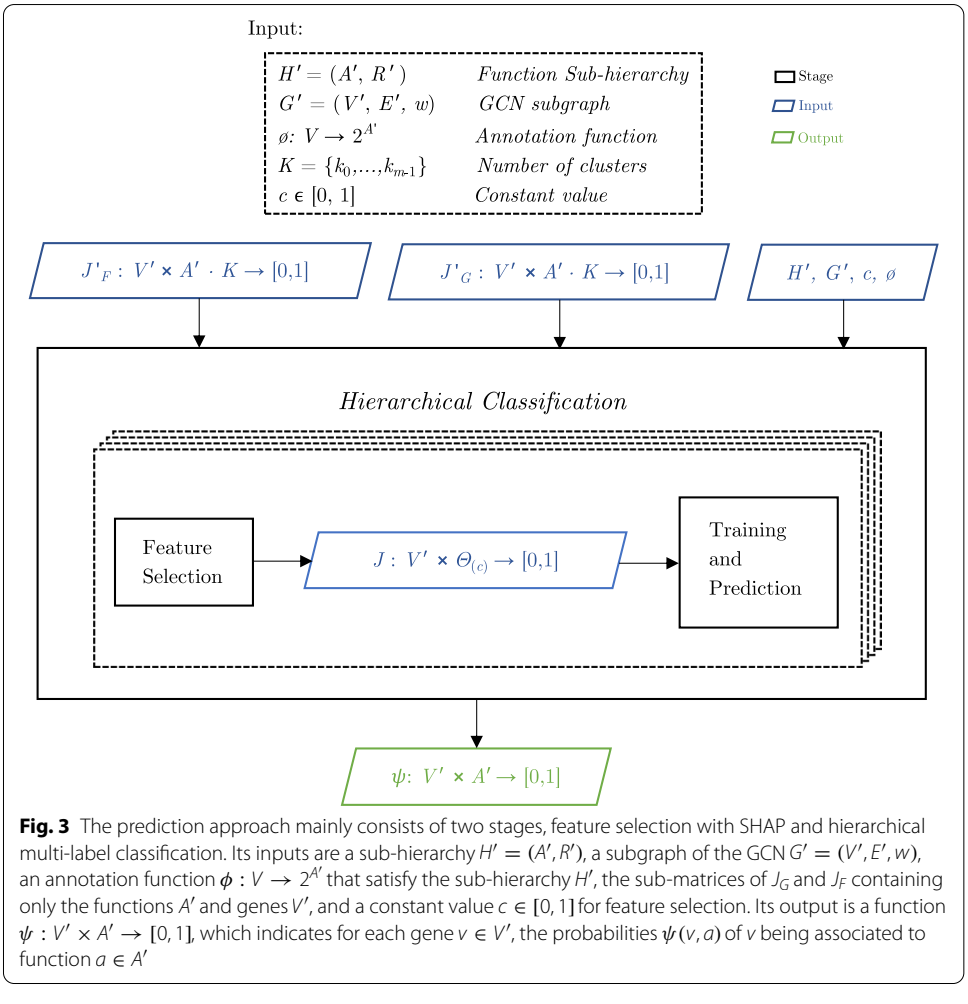

Fig. 3 The prediction approach mainly consists of two stages, feature selection with SHAP and hierarchical multi-label classification. Its inputs are a sub-hierarchy $H' = (A', R')$, a subgraph of the GCN $G' = (V', E', w)$, an annotation function $\phi : V \rightarrow 2^{A'}$ that satisfy the sub-hierarchy H', the sub-matrices of J_G and J_F containing only the functions A' and genes V', and a constant value $c \in [0, 1]$ for feature selection. Its output is a function $\psi : V' \times A' \rightarrow [0, 1]$, which indicates for each gene $v \in V'$, the probabilities $\psi(v, a)$ of v being associated to function $a \in A'$

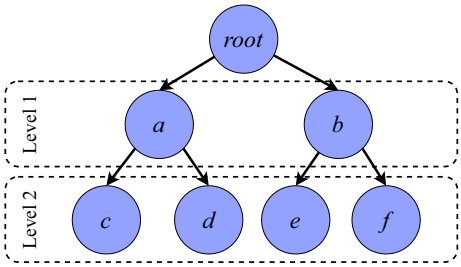

Fig. 4 Gene function prediction considering the function hierarchy and using a local classifier per level method. The predictor for the level 1 predicts functions *a* and *b*, so only the features from J'_G and J'_F associated to functions *a* and *b* are considered for the feature selection

expression $J'_{G+F}(v, \_)$ denotes a function with domain $[0, 2q)$ and range $[0, 1]$, where the values in $[0, q)$ denote the p-values associated to v in G and the values in $[q, 2q)$ the ones associated to v in the enriched version of G. For each entry $J'_{G+F}(v, j)$, with $v \in V'$ and $0 \leq j < 2q$, the mean absolute SHAP value $s_{(v,j)}$ is computed after a large enough number of Shapely values are computed (executions of SHAP). Features are selected based on the cutoff

$$c \cdot \sum_{j=0}^{2q-1} s_{(v,j)},$$

i.e., on the sum of mean absolute values by a factor of the input constant c. The first $\Theta(c)$ features, sorted from greater to lower mean absolute SHAP value, are selected as to reach the given cutoff.

Note that the input constant c is key for selecting the number of significant features. The idea is to set c so as to find a balance between prediction efficiency and the computational cost of building the predictor.

Training and prediction

This stage comprises a process that combines two supervised machine learning techniques/tools to build the predictor ψ. In particular, stratified k-fold cross-validation and hierarchical multi-label classification are used sequentially in a pipeline.

The pipeline takes as input the matrix J, which specifies the significant features of J'_G and J'_F, the sub-hierarchy H' and the annotation function ϕ. First, k-fold is applied to split the dataset into k different folds for cross validation (note that k is not related to the input K). That is, each fold is used as a test set, while the remaining $k - 1$ folds are used for training. Recall that k-fold cross validation aims to overcome overfitting in training. Furthermore, one or multiple random forest classifiers are build and used for prediction, the number of classifiers depends on the HMC method. Randoms forest is selected for this approach since it is a tree-based and multi-label classification algorithm, which is interpretable (SHAP can be applied). The parameter values used for random forest classifiers, differently from the default scikit-learn values, are: 200 estimators (*n_estimators*) and minimum number of samples of 5 (*min_samples_split*).

Additionally, some HMC methods require an extra step to keep prediction consistent w.r.t. the sub-hierarchy H' (i.e., comply the true-path rule). The probability of association

between a function $v \in V'$ and a function $a \in A'$ must be lower than the probability of association between the same gene and the ancestor of a in H'. To satisfy this constraint cumulative probabilities are computed throughout the paths in H'. That is, for each gene $v \in V$ and functions $(a, b) \in R$, the predicted probability of the association between v and a is multiplied by the predicted probability of association between v and b (its ancestor). This process is repeated for every path in the hierarchy from the root to the leaves.

The output of this stage is the predictor ψ, i.e., the probabilities of associations between the genes in V' and functions A'. Note that the predictor ψ satisfies the true-path rule.

Performance evaluation

It is often the case in HMC datasets that individual classes have few positive instances. In genome annotation, typically only a few genes are associated to specific functions. This implies that for most classes (deeper in the hierarchy), the number of negative instances by far exceeds the number of positive instances. Hence, the real focus is recognizing the positive instances (predict associations between genes and functions), rather than correctly predicting the negative ones (predict that a function is not associated to a given gene). Although ROC curves are better known, their area under the curve is higher if a model correctly predicts negative instances, which is not suitable for HMC problems.

For this reasons, the measures (based on the precision-recall (PR) curve) introduced by Vens et al. (2008) are used for evaluation.

Area under the average PR curve

The first metric transforms the multi-label problem into a binary one by computing the precision and recall for all functions A' together. This corresponds to micro-averaging the precision and recall.

The output of the prediction stage are the probabilities of associations between genes V' and functions A'. Thereby, instead of selecting a single threshold to compute precision and recall, multiple thresholds are used to create a PR curve. In the PR curve each point represent the precision and recall for a give threshold that can be computed as:

$$\overline{\text{Prec}} = \frac{\sum_i TP_i}{\sum_i TP_i + \sum_i FP_i}, \quad \text{and} \quad \overline{\text{Rec}} = \frac{\sum_i TP_i}{\sum_i TP_i + \sum_i FN_i}.$$

Note that i ranges over all functions A', i.e., precision and recall are computed for all functions together. The area under this curve is denoted as AU($\overline{\text{PRC}}$).

Average area under the PR curves

The second metric corresponds to the (weighted) average of the areas under the PR curves for all functions A'. This metric, referred as macro-average of precision and recall, can be computed as follows:

$$\overline{\text{AUPRC}}_{w_1,w_2,\ldots,w_{|A'|}} = \sum_i w_i \cdot \text{AUPRC}_i.$$

If the weights of all functions are the same (i.e., $1/|A'|$) the metric is denoted as $\overline{\text{AUPRC}}$. In addition, weights can also be defined based on the number of genes associated to functions in ϕ, i.e., $w_a = |\phi^{-1}(a)| / \sum_i |\phi^{-1}(i)|$ for $a \in A$. In the later case, denoted

as $\overline{\text{AUPRC}_w}$, more frequent functions get higher weight. Note that one point in the weighted PR curve corresponds to the (weighted) average of the AUPRC of all functions A' given a threshold.

Case study: *Zea mays*

Next section describes a case study on applying the feature extraction and prediction approach presented in "Clustering-based feature extraction" and "Hierarchical multi-label classification for gene function prediction" sections to maize (*Zea mays*). First, the maize data used for the case study is described. Second, the proposed approach is applied to the maize data. Lastly, the performance of the proposed approach is compared to two models trained using each set of features J_G and J_F, independently.

Data description and feature extraction

The co-expression information used in the study is imported from the ATTED-II database (Obayashi et al. 2018). The gene co-expression network $G = (V, E, w)$ comprises 26 131 vertices (genes) and 44 621 533 edges. In this case, a z-score threshold of 1 is used as the cut-off measure for G, i.e., E contains edges e that satisfy $w(e) \geq 1$ (most of them satisfying $w(e) > 1$). Note that the highest value is assigned to the strongest connections. The functional information for this network is taken from DAVID Bioinformatics Resources (Huang et al. 2009) (2021 update); it contains annotations of biological processes, i.e., pathways to which a gene contributes. It is important to note that genes may be associated to several biological processes, and biological processes may be associated to multiple genes. The database comprises 3 924 biological processes A and 7 021 ancestral relations R between these functions, that represent the hierarchy $H = (A, R)$ of the GO (Gene Ontology Consortium 2019). A total of 255 865 association between genes and functions are considered, these associations represent the annotation function $\phi : V \to 2^A$.

The feature extraction approach is applied with the inputs G, A, ϕ and $K = \{10, 20, \ldots, 100\}$ (values are incremented in steps of 10 up to 100). The outputs are the feature matrices J_G and J_F that specify how likely it is for the maize genes V to be associated to the biological processes A when the graph is decomposed in the number of clusters in K.

Moreover, only functions associated to more than 200 genes have been considered, so the number of functions in the resulting sub-hierarchies is tractable regarding the dimension of the output of SHAP (see "Preliminaries" section). Recall that the Gene Ontology hierarchy splits into 28 sub-hierarchies when considering only biological processes. Additionally, all sub-hierarchies with less than 10 functions are discarded and the topological-sorting algorithm introduced in Romero et al. (2022) is used to transform the sub-hierarchies, represented as DAGs, into trees. For each ancestral relation $(a, b) \in R$ (b is ancestor of a), the algorithm assigns a weight as the ratio of the number of genes associated to the a to the number of genes associated to b. Then, for each function $a \in A'$ with more than one parent, only the one with the higher weight remains (ties are broken arbitrarily).

As result, there are 5 sub-hierarchies of biological processes. Table 1 describes each sub-hierarchy H', starting by the root term r and its description, following the

Table 1 Resulting sub-hierarchies H' of biological processes for maize

Root	Description	Functions	Genes	Functions per level
GO:0050896	Response to stimulus	13	1733	5/5/2
GO:0051179	Localization	25	1497	3/5/9/6/1
GO:0065007	Biological regulation	37	2647	2/5/11/10/4/2/2
GO:0008152	Metabolic process	92	6596	8/18/38/12/7/6/2
GO:0009987	Cellular process	92	8005	13/19/19/17/13/8/2

The identifier and description of each root function r is presented in the first and second columns, respectively. The third column shows the number of functions A' within each sub-hierarchy and the fourth column shows the number maize genes in the GCN subgraph G' associated to H'. The last column shows the number of functions per level, e.g., the first sub-hierarchy has 3 levels and there are 5, 5, and 2 functions on each level

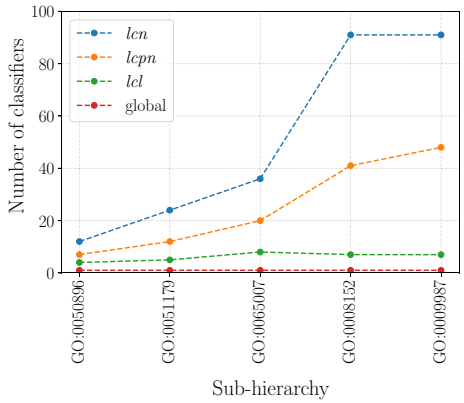

Fig. 5 Number of classifiers trained per HMC method and sub-hierarchy. The *lcn* requires $|A'| - 1$ classifiers. The *lcpn* requires as many classifiers as functions with children in H'. The *lcl* requires as many classifiers as the number of levels in H'. At last, the global method requires one classifier per hierarchy

number of functions A' and the number of genes V' in the associated GCN subgraph G'. The prediction approach is applied to each sub-hierarchy H' independently. The remaining input parameter for the prediction approach is $c = 0.9$ (recall that this parameter is used to filter the most relevant features according to their mean SHAP value). Figure 5 depicts the number of classifiers trained per HMC method and sub-hierarchy. Note that the global method requires one classifier per hierarchy, while the *lcn* requires $|A'| - 1$ classifiers.

Summary of results

Figure 6 presents the prediction performance of the proposed approach measured with the AU($\overline{\text{PRC}}$) (denoted as *micro*) for four HMC methods, namely, local classifier per node (*lcn*), local classifier per parent node (*lcpn*), local classifier per level (*lcl*), and global classifier. In general, it can be seen that all methods get a high area under the average PR curve, but the global classifier outperforms the local methods for all sub-hierarchies. The proposed approach identifies the associations between genes and functions by using the features extracted from the GCN G and the affinity graph F, and considering the ancestral relations of the biological processes. The global method

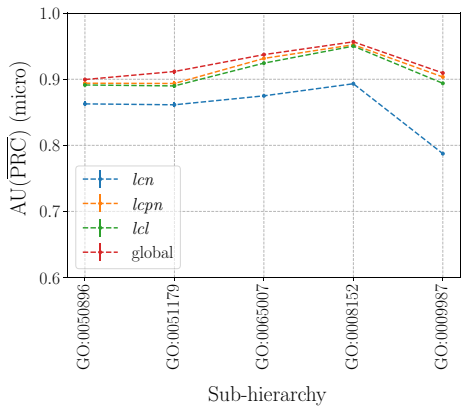

Fig. 6 Prediction performance of the proposed approach measured with the area under the average PR curve, i.e., AU($\overline{\text{PRC}}$). The performance is measured independently per sub-hierarchy

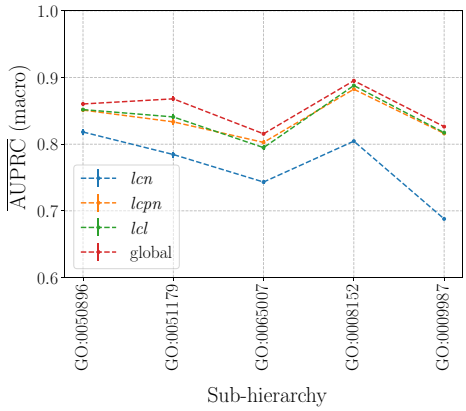

Fig. 7 Prediction performance of the proposed approach measured with the average area under the PR curve, i.e., $\overline{\text{AUPRC}}$. The performance is measured independently per sub-hierarchy

obtains the best performance, followed by the *lcpn* and the *lcl*. Using multi-label classifiers is better than using a binary classifier for each function, i.e., *lcn* method.

The micro score measures the overall performance of all functions within a sub-hierarchy without distinguishing between them. Figure 7 presents the prediction performance measured with the $\overline{\text{AUPRC}}$, denoted as *macro*. The macro score measure the prediction performance for each function individually and then takes the average. The conclusion is similar, the global method outperforms the local ones.

Finally, Fig. 8 illustrates the prediction performance measured with the $\overline{\text{AUPRC}_w}$, denoted as *macro weighted*. This score weights the individual performance of each function according to the number of genes associated to it. Thereby, the leaves and deeper functions in a sub-hierarchy always get lower weight than the others. Note that the deeper a functions is in a sub-hierarchy, the lower the predicted probabilities becomes. The global method outperforms the locals again. The conclusion is consistent with the three metrics, using clustering techniques to extract features from the

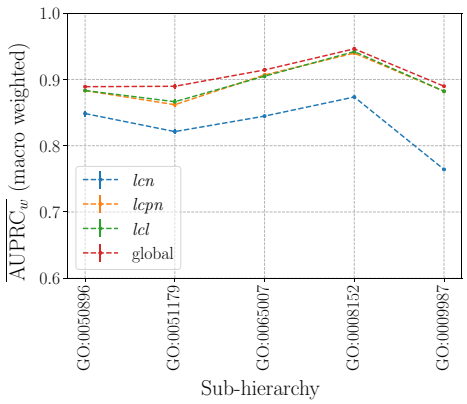

Fig. 8 Prediction performance of the proposed approach measured with the average area under the PR curve, i.e., $\overline{\text{AUPRC}_w}$. The performance is measured independently per sub-hierarchy

Table 2 Number of extracted and filtered features used for the global method per sub-hierarchy

Root	Total	Filtered
GO:0050896	239	124
GO:0051179	479	263
GO:0065007	713	402
GO:0008152	1812	796
GO:0009987	1813	853

Recall that the extracted features are filtered using the mean SHAP values to select the more important with a cutoff defined by the input constant *c*

GCN and considering the hierarchical structure of the biological processes seems to be key for the gene function production task.

It has been shown in Romero et al. (2022) that the new features built from the GCN, and the associations between genes and functions with the spectral clustering algorithm are key to improve the prediction performance in the gene annotation problem (w.r.t. other features of the GCN and gene functional information). However, the feature extraction approach presented in "Clustering-based feature extraction" section produces two different sets of features, namely, J_G and J_F, that are combined and used for prediction. The individual relevance of each set of features for the gene annotation problem is analyzed by (i) looking at the distribution of the filtered features for the global method and (ii) comparing the performance of the prediction task using each set of features independently. Table 2 presents the number of extracted and filtered features used for the global method per sub-hierarchy. Recall that the features are filtered using the mean SHAP values to select the more important ones with a cutoff defined by the input constant c.

Figure 9 illustrates the distribution of the filtered features for the global method per sub-hierarchy. Note that, even though the features from the affinity graph F (i.e., J_F) are more important, features from the GCN G (i.e., J_G) are also selected for all sub-hierarchies. Figure 10 shows the prediction performance of the global HMC method trained using the features J_G and J_F independently, and the proposed approach (i.e.,

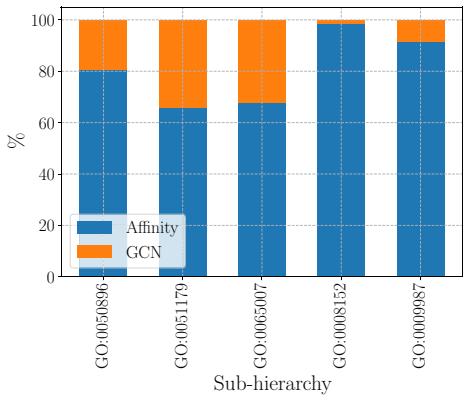

Fig. 9 Distribution of the filtered features from J_G and J_F for the global method per sub-hierarchy

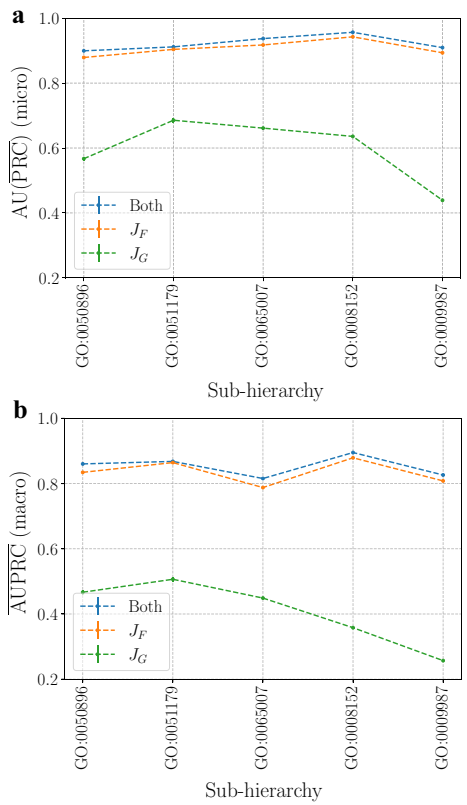

Fig. 10 Prediction performance of the global method trained using the features J_G and J_F independently, and the proposed approach (i.e., their combination) measured with AU($\overline{\text{PRC}}$) and $\overline{\text{AUPRC}}$. The performance is measured independently per sub-hierarchy

their combination) measured with AU($\overline{\text{PRC}}$) and $\overline{\text{AUPRC}}$. The combination of both sets of features, extracted from the GCN and the affinity graph is key to improve the performance of the proposed approach for all sub-hierarchies.

Related work and concluding remarks

Related work

Zhou et al. (2020) presented an approach to predict functions of maize proteins using graph convolutional networks. In particular, an amino acid sequence of proteins and the GO hierarchy were used to predict functions of proteins with a deep graph convolutional network model (DeepGOA). Their results showed that DeepGOA is a powerful tool to integrate amino acid data and the GO structure to accurately annotate proteins. Similarly, the work presented in Cruz et al. (2020) aims to predict the phenotypes and functions associated to maize genes using: (i) hierarchical clustering based on datasets of transcriptome (set of molecules produced in transcription) and metabolome (set of metabolites found within an organism); and (ii) GO enrichment analyses. Their results showed that profiling individual plants is a promising experimental design for narrowing down the lab-field gap. Gligorijević et al. (2018) proposed a network fusion method based on multimodal deep autoencoders to extract high-level features of proteins from multiple interaction networks. This method, called *deepNF*, relied on a deep learning technique that captures relevant protein features from different complex, non-linear interaction networks. Their results showed that extracting new features from biological networks is key to annotate gene with functions. The work in Zhao et al. (2019) is also closely related. They presented Gene Ontology hierarchy preserving hashing (HPHash), a gene function prediction method that retains the hierarchical order between GO terms. It used a hierarchy preserving hashing technique based on the taxonomic similarity between terms to capture the GO hierarchy. Hashing functions were used to compress the gene-term association matrix, where the semantic similarity between genes was used to predict the functions of the genes. Their results showed that HPHash preserves the GO hierarchy and improves prediction performance.

In addition, the authors in Chen et al. (2018) presented *iFeature*, a Python-based toolkit for generating numerical feature representation schemes from protein sequences. It integrated algorithms for feature clustering, selection, and dimensionality reduction to facilitate training, analysis, and benchmarking of machine-learning models. In a related way, Mu et al. (2021) showed that feature extraction of protein sequences is helpful for prediction of protein functions or interactions. They introduced FEGS (Feature Extraction based on Graphical and Statistical features), a novel feature extraction model for protein sequences that combines graphical and statistical features. Their results showed that similarity analysis of protein sequences has applications in the study of gene annotation, gene function prediction, identification and construction of gene families, and gene discovery.

Concluding remarks and future work

By combining network-based modeling, cluster analysis, interpretable machine learning, and hierarchical multi-label classification, the approach presented in this paper introduces a novel method to address the gene function prediction problem. It aims to predict the association probability between each gene and function by taking advantage of the GCN spectral decomposition, the information available of associations between genes and functions, and the ancestral relations between the functions (i.e., the GO hierarchy).

A case study on *Zea mayz* (maize) is presented. Using the structural information of the gene co-expression network (extracted by a spectral clustering algorithm) and considering the hierarchical structure of the biological processes (using HMC) seems to be the key for the improved performance of the proposed approach. More precisely, the global HMC method, which considers all features available for a sub-hierarchy to build a single classifier, outperforms the other methods in relation to the three metrics that were used (namely, AU($\overline{\text{PRC}}$) , $\overline{\text{AUPRC}}$, and $\overline{\text{AUPRC}_w}$).

The results presented in Romero et al. (2022) show that the features extracted from the GCN using spectral clustering lead to better prediction performance in the gene function prediction task (addressed as an independent binary classification problem per function). In this work, it has been shown that considering the ancestral relations between functions to produce an outcome that satisfies its hierarchical structure (i.e., complies the true-path rule or hierarchical constraint), based on the features extracted from the GCN, improves the performance in the gene function prediction task (addressed as a hierarchical multi-label classification problem).

Two main lines of work can be considered for future work. First, applying the proposed approach to identify genes associated to specific stresses (e.g., low temperature, salinity) can help to reduce the set of candidate genes that respond to treatments for in vivo validation. Second, exploring transfer learning techniques (especially, domain adaptation) to enrich the building of the classifiers using information from other organisms (datasets), not only can lead to higher prediction performance, but also can enable the proposed approach on organisms without a wealth of significant functional information.

Abbreviations

AUPRC:: Area under precision-recall curve; DAG:: Directed acyclic graph; GO:: Gene ontology; GCN:: Gene co-expression network; HMC:: Hierarchical multi-label classification; *lcl*:: Local classifier per level; *lcn*:: Local classifier per node; *lcpn*:: Local classifier per parent node.

Acknowledgements

Not applicable.

Author contributions

MR and OR proposed the original idea. JF and CR provide advice on algorithms concepts and implementation. MR and OR structured the methodology and performed the analysis. MR, OR, JF, and CR wrote the manuscript. All authors read and approved the final manuscript.

Authors information

Miguel Romero Ph.D. Student in Engineering and Applied Sciences at the Pontificia Universidad Javeriana, in Cali (Colombia). He earned a B.S. degree in Economics and Systems Engineering from the Escuela Colombiana de Ingeniería *Julio Garavito*, Bogotá (Colombia). He has experience in rewrite logic, network analysis, machine learning, algorithms, and competitive programming. He works on the development of mathematical models and algorithms that allow identifying, from in silico omic characterization, the expression of phenotypic traits in different varieties of crops.

Oscar Ramírez M.Sc. Student in Engineering at the Pontificia Universidad Javeriana, in Cali (Colombia). He earned a B.S. degree in Electrical Engineering from the Escuela Colombiana de Ingeniería ***Julio Garavito***, Bogotá (Colombia). He has experience in network analysis and algorithms. He works on the development of mathematical models and algorithms for the identification of uncharacterized associations between genes and biological functions in ***Zea mays***. Specially, identifying those associations related to key stress types, such as, low temperature tolerance and common rust resistant.

Jorge Finke Professor in the Department of Electronics and Computer Science at the Pontificia Universidad Javeriana, in Cali (Colombia). He earned a B.S., a M.Sc. and a Ph.D. degree in Systems theory from The Ohio State University, Columbus, OH. More than 10 years of experience in developing representations of complex, large-scale data, and innovative approaches to predictive modeling and analysis, including cutting-edge research in academia as well as real-world consulting for companies across multiple industries. My motivation is to discover new and disruptive opportunities that enable organizations to solve problems and achieve strategic goals.

Camilo Rocha Associate Professor in the Department of Electronics and Computer Science at the Pontificia Universidad Javeriana, in Cali (Colombia). Starting February 2020, he has been appointed Dean of Engineering and Sciences. He earned a B.S. and a M.Sc. degree in Informatics from the Universidad de los Andes (Bogotá), and a M.Sc. degree in Mathematics and a Ph.D. degree in Computer Science from the University of Illinois at Urbana-Champaign. His main research

interests are in formal methods, algorithms, and software engineering, more specifically on techniques for building reliable software systems.

Funding
This work was partially funded by the OMICAS program: Optimización Multiescala In-silico de Cultivos Agrícolas Sostenibles (Infraestructura y Validación en Arroz y Caña de Azúcar), anchored at the Pontificia Universidad Javeriana in Cali and funded within the Colombian Scientific Ecosystem by The World Bank, the Colombian Ministry of Science, Technology and Innovation, the Colombian Ministry of Education and the Colombian Ministry of Industry and Turism, and ICETEX, under GRANT ID: FP44842-217-2018. The second author was partially supported by Fundación CeiBA.

Availability of data and materials
The datasets analyzed for the current study are publicly available from different sources. They can be found in the following locations: Gene co-expression data of ***Oryza sativa Japonica*** is available on ATTED-II Obayashi et al. (2018). Functional data of rice genes is available on the DAVID Bioinformatics Resources Huang et al. (2009). Hierarchical data of Gene Ontology terms are available on the GOATOOLS Python library Klopfenstein et al. (2018). The data collected, cleaned, and processed from the above sources as used in the case study can be requested to the authors. A workflow implementation is publicly available: Project name: clustering_hmc. Project home page: https://github.com/migueleci/clustering_hmc. Operating system(s): platform independent. Programming language: Python 3. Other requirements: None. License: GNU GPL v3.

Declarations

Ethics approval and consent to participate
All data were anonymized and collected in accordance to paragraph 23 of the German federal law, German Protection against Infection Act ("Infektionsschutzgesetz"), which regulates the prevention and control of infectious diseases in humans. Therefore, ethical approval and informed consent were not required.

Consent for publication
Not applicable, because all data displayed in this publication are surveillance-based data, obtained in accordance with the German Protection against Infection Act ("***Infektionsschutzgesetz***").

Competing interests
The authors declare that they have no competing interests.

Received: 1 March 2022 Accepted: 9 April 2022
Published online: 10 May 2022

References

Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G (2000) Gene ontology: tool for the unification of biology. Nat Genet 25(1):25–29. https://doi.org/10.1038/75556

Bhagat S, Cormode G, Muthukrishnan S (2011) Node classification in social networks. In: Aggarwal CC (ed) Social network data analytics, Springer US, Boston, pp 115–148. https://doi.org/10.1007/978-1-4419-8462-3_5

Chen Z, Zhao P, Li F, Leier A, Marquez-Lago TT, Wang Y, Webb GI, Smith AI, Daly RJ, Chou K-C, Song J (2018) iFeature: a python package and web server for features extraction and selection from protein and peptide sequences. Bioinformatics 34(14):2499–2502. https://doi.org/10.1093/bioinformatics/bty140

Cho H, Berger B, Peng J (2016) Compact integration of multi-network topology for functional analysis of genes. Cell Syst 3(6):540–5485. https://doi.org/10.1016/j.cels.2016.10.017

Cho H, Berger B, Peng J (2015) Diffusion component analysis: unraveling functional topology in biological networks. In: RECOMB 2015, Springer, Cham, pp 62–64

Cruz DF, DeMeyer S, Ampe J, Sprenger H, Herman D, VanHautegem T, DeBlock J, Inzé D, Nelissen H, Maere S (2020) Using single-plant-omics in the field to link maize genes to functions and phenotypes. Mol Syst Biol. https://doi.org/10.15252/msb.20209667

Deng M, Zhang K, Mehta S, Chen T, Sun F (2003) Prediction of protein function using protein-protein interaction data. J Comput Biol 10(6):947–960. https://doi.org/10.1089/106652703322756168

Gene Ontology Consortium (2019) The Gene Ontology Resource: 20 years and still GOing strong. Nucleic Acids Res 47(D1):330–338. https://doi.org/10.1093/nar/gky1055

Gligorijević V, Barot M, Bonneau R (2018) deepNF: deep network fusion for protein function prediction. Bioinformatics 34(22):3873–3881. https://doi.org/10.1093/bioinformatics/bty440

Huang DW, Sherman BT, Lempicki RA (2009) Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. Nat Protoc 4(1):44–57. https://doi.org/10.1038/nprot.2008.211

Jia H, Ding S, Xu X, Nie R (2014) The latest research progress on spectral clustering. Neural Comput Appl 24(7–8):1477–1486. https://doi.org/10.1007/s00521-013-1439-2

Khan SS, Madden MG (2010) A survey of recent trends in one class classification. In: Coyle L, Freyne J (eds) Artificial intelligence and cognitive science, vol 6206, Springer, Berlin, pp 188–197. https://doi.org/10.1007/978-3-642-17080-5_21

Klopfenstein DV, Zhang L, Pedersen BS, Ramírez F, Warwick Vesztrocy A, Naldi A, Mungall CJ, Yunes JM, Botvinnik O, Weigel M, Dampier W, Dessimoz C, Flick P, Tang H (2018) GOATOOLS: a python library for gene ontology analyses. Sci Rep 8(1):10872. https://doi.org/10.1038/s41598-018-28948-z

Levatić J, Kocev D, Džeroski S (2015) The importance of the label hierarchy in hierarchical multi-label classification. J Intell Inf Syst 45(2):247–271. https://doi.org/10.1007/s10844-014-0347-y
Lundberg SM, Erion G, Chen H, DeGrave A, Prutkin JM, Nair B, Katz R, Himmelfarb J, Bansal N, Lee S-I (2020) From local explanations to global understanding with explainable ai for trees. Nat Mach Intell 2(1):2522–5839
Lundberg S, Lee SI (2017) A unified approach to interpreting model predictions. arXiv:1705.07874 [cs, stat] . arXiv:1705.07874
Luo F, Yang Y, Zhong J, Gao H, Khan L, Thompson DK, Zhou J (2007) Constructing gene co-expression networks and predicting functions of unknown genes by random matrix theory. BMC Bioinform 8(1):299. https://doi.org/10.1186/1471-2105-8-299
Mills P (2021) Solving for multi-class: a survey and synthesis. arXiv:1809.05929 [cs, stat] . arXiv:1809.05929
Mu Z, Yu T, Liu X, Zheng H, Wei L, Liu J (2021) FEGS: a novel feature extraction model for protein sequences and its applications. BMC Bioinform 22(1):297. https://doi.org/10.1186/s12859-021-04223-3
Murugesan N, Cho I, Tortora C (2021) Benchmarking in cluster analysis: a study on spectral clustering, DBSCAN, and K-Means. In: Data analysis and rationality in a complex world, Springer, Cham, pp 175–185. http://link.springer.com/10.1007/978-3-030-60104-1_20. Accessed 30 Sep 2021
Obayashi T, Kinoshita K (2011) COXPRESdb: a database to compare gene coexpression in seven model animals. Nucleic Acids Res 39(Database):1016–1022. https://doi.org/10.1093/nar/gkq1147
Obayashi T, Aoki Y, Tadaka S, Kagaya Y, Kinoshita K (2018) ATTED-II in 2018: a plant coexpression database based on investigation of the statistical property of the mutual rank index. Plant Cell Physiol 59(1):3–3. https://doi.org/10.1093/pcp/pcx191
Oti M, van Reeuwijk J, Huynen MA, Brunner HG (2008) Conserved co-expression for candidate disease gene prioritization. BMC Bioinform 9(1):208. https://doi.org/10.1186/1471-2105-9-208
Petsko GA (2009) Guilt by association. Genome Biol 10(4):104. https://doi.org/10.1186/gb-2009-10-4-104
Rodriguez MZ, Comin CH, Casanova D, Bruno OM, Amancio DR, Costa LF, Rodrigues FA (2019) Clustering algorithms: a comparative approach. PLoS ONE 14(1):0210236. https://doi.org/10.1371/journal.pone.0210236
Romero M, Finke J, Rocha C (2022) A top-down supervised learning approach to hierarchical multi-label classification in networks. App Netw Sci 7(1):8. https://doi.org/10.1007/s41109-022-00445-3
Romero M, Finke J, Quimbaya M, Rocha C (2020) In-silico gene annotation prediction using the co-expression network structure. In: Complex networks and their applications, vol VIII, Springer, Cham, pp 802–812
Romero M, Ramírez Ó, Finke J, Rocha C (2022) Supervised gene function prediction using spectral clustering on gene co-expression networks. In: Benito RM, Cherifi C, Cherifi H, Moro E, Rocha LM, Sales-Pardo M (eds) Complex networks and their applications X, vol 1016, Springer, Cham, pp 652–663. https://doi.org/10.1007/978-3-030-93413-2_54
Rust AG, Mongin E, Birney E (2002) Genome annotation techniques: new approaches and challenges. Drug Discov Today 7(11):70–76. https://doi.org/10.1016/S1359-6446(02)02289-4
Silla CN, Freitas AA (2011) A survey of hierarchical classification across different application domains. Data Min Knowl Disc 22(1–2):31–72. https://doi.org/10.1007/s10618-010-0175-9
Stuart JM (2003) A gene-coexpression network for global discovery of conserved genetic modules. Science 302(5643):249–255. https://doi.org/10.1126/science.1087447
Valentini G (2009) True path rule hierarchical ensembles. In: Multiple classifier systems, Springer, Berlin, pp 232–241
van Dam S, Võsa U, van der Graaf A, Franke L, de Magalhães JP (2017) Gene co-expression analysis for functional classification and gene-disease predictions. Briefings Bioinform. https://doi.org/10.1093/bib/bbw139
Vandepoele K, Quimbaya M, Casneuf T, De Veylder L, Van de Peer Y (2009) Unraveling transcriptional control in arabidopsis using cis-regulatory elements and coexpression networks. Plant Physiol 150(2):535–546. https://doi.org/10.1104/pp.109.136028
Vens C, Struyf J, Schietgat L, Džeroski S, Blockeel H (2008) Decision trees for hierarchical multi-label classification. Mach Learn 73(2):185–214. https://doi.org/10.1007/s10994-008-5077-3
Xu D, Shi Y, Tsang IW, Ong Y-S, Gong C, Shen X (2020) Survey on multi-output learning. IEEE Trans Neural Netw Learn Syst 31(7):2409–2429. https://doi.org/10.1109/TNNLS.2019.2945133
Yandell M, Ence D (2012) A beginner's guide to eukaryotic genome annotation. Nat Rev Genet 13(5):329–342. https://doi.org/10.1038/nrg3174
Yon Rhee S, Wood V, Dolinski K, Draghici S (2008) Use and misuse of the gene ontology annotations. Nat Rev Genet 9(7):509–515. https://doi.org/10.1038/nrg2363
Yu S (2003) Multiclass spectral clustering. In: Proceedings ninth IEEE international conference on computer vision, IEEE, Nice, France, pp 313–3191. https://doi.org/10.1109/ICCV.2003.1238361
Zhao Y, Fu G, Wang J, Guo M, Yu G (2019) Gene function prediction based on gene ontology hierarchy preserving hashing. Genomics 111(3):334–342. https://doi.org/10.1016/j.ygeno.2018.02.008
Zhou Y, Young JA, Santrosyan A, Chen K, Yan SF, Winzeler EA (2005) In silico gene function prediction using ontology-based pattern identification. Bioinformatics 21(7):1237–1245. https://doi.org/10.1093/bioinformatics/bti111
Zhou G, Wang J, Zhang X, Guo M, Yu G (2020) Predicting functions of maize proteins using graph convolutional network. BMC Bioinform 21(S16):420. https://doi.org/10.1186/s12859-020-03745-6

Publisher's Note