Supervised Gene Function Prediction Using Spectral Clustering on Gene Co-expression Networks

Miguel Romero(✉), Óscar Ramírez, Jorge Finke, and Camilo Rocha

Department of Electronics and Computer Science, Pontificia Universidad Javeriana, Cali, Colombia
miguel.romero@javerianacali.edu.co

Abstract. Gene annotation addresses the problem of predicting unknown functions that are associated to the genes of a specific organism (e.g., biological processes). 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 an in silico approach to the annotation of genes that follows a network-based representation, and combines techniques from multivariate statistics (spectral clustering) and machine learning (gradient boosting). Spectral clustering is used to enrich the gene co-expression network (GCN) with currently known gene annotations. Gradient boosting is trained on features of the GCN to build an estimator of the probability that a gene is involved in a given biological process. The proposed approach is applied to a case study on *Zea mays*, one of the world's most dominant and productive crop. Broadly speaking, the main results illustrate how computational experimentation narrows down the time and costs in efforts to annotate the functions of genes. More specifically, the results highlight the importance of network science, multivariate statistics, and machine learning techniques in reducing types I and II prediction errors.

1 Introduction

An important pillar for gaining insight into how genomes serve as blueprints for life is understanding the association of genes with as yet unknown functions [25,29]. Developing treatments that use genomic information about organisms to treat specific conditions, including—for example—approaches that enhance tolerance levels to environmental stresses, has motivated a significant body of research [28]. Nonetheless, the cost and time demanded by in vivo biological experimentation to annotate large sets of genes remains prohibitively high [5,32]. Hybrid approaches that integrate existing knowledge of gene-function associations and in silico methods have been introduced to overcome this limitation [6,8,15,24]. The ability to cope with the extreme combinatorial nature of gene annotation enables computational experimentation to narrow down the effort, time, and costs.

A number of studies have shown that the representation and analysis of gene co-expression networks (GCNs) are a useful framework for guiding in silico annotation of genes [20,27]. The crux of GCN-based analysis is that it exploits the topology of the network and offers a rich source of new information for predicting gene-function

R. M. Benito et al. (Eds.): COMPLEX NETWORKS 2021, SCI 1016, pp. 652–663, 2022.
https://doi.org/10.1007/978-3-030-93413-2_54

associations [26]. In practice, such approaches remain valid as long as the data for gene-function associations is relatively complete and the underlying co-expression network is tractable. Therefore, unlocking the full potential of network-based gene annotation demands efficient approaches that are scalable to long datasets.

Main Contribution. This paper presents a novel approach for in silico annotation of genes. It follows a network-based approximation that uses clustering and machine learning for building a predictor that assigns functions to genes. The role of clustering is to enrich the available information for gene-function associations by creating new features that are later used for supervised learning. More precisely, new features are built by taking into account clusters that seem relevant to the specific biological function under scrutiny. These new features are filtered based on the impact that is made on prediction, before a machine learning algorithm is used to build the predictor. The proposed approach illustrates how the performance of gene annotation is improved based on the new information obtained from the clustering of the GCN.

The approach is applied to a case study on *Zea mays*, the world's most dominant and productive crop. *Zea mays* is used for a variety of purposes, including animal feed and derivatives for human consumption, and ethanol [31]. The co-expression information used in the case study is borrowed from the ATTED-II database [18]. The resulting GCN, modeled as a weighted graph, comprises 26,131 vertices (genes) and 44,621,533 edges (binary co-expression gene relations). The functional information (known gene-function associations) is taken from AmiGO [2]; it contains annotations of biological processes, i.e., pathways to which a gene contributes. A total of 5,361 genes are associated to 3,285 functions. Two benchmarks are introduced and used to contrast the results of the approach. The comparison highlights the importance of multivariate statistics and machine learning techniques in reducing type I (i.e., false positives) and type II (i.e., false negatives) prediction errors. Ultimately, this case study provides experimental (in silico) evidence that the proposed approach is a viable and promising approximation to gene function prediction.

Related Work. The authors in [31] predict functions of maize proteins using graph convolutional networks. In particular, amino acid sequence of proteins and the Gene Ontology (GO) hierarchy are used to predict functions of proteins with a deep graph convolutional network model (DeepGOA). The results show that DeepGOA integrates amino acid data and the GO structure to accurately annotate proteins. The work in [7] 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. The results show that profiling individual plants is a promising experimental design for narrowing down the lab-field gap. Finally, a prediction of protein functions for *Zea mays* is presented in [17]. The approach, called PiZeam, is built using a method of interacting orthologs (genes are said to be *orthologs* if they evolved from a common ancestor). PiZeam demonstrates that the protein functions of maize can be predicted based on protein sequence data of other organisms because orthologs tend to retain the same function [10]).

Outline. The remainder of the paper is organized as follows. Section 2 gathers some preliminaries. The proposed approach is presented in Sect. 3. Section 4 presents a case

study for the *Zea mays* species. Finally, Sect. 5 draws some concluding remarks and future research directions.

2 Preliminaries

This section presents preliminaries on gene co-expression networks, gene function prediction, spectral clustering, and the SHAP technique.

2.1 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, with the help of 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 how strongly co-expressed are 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) [18,19].

2.2 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.

2.3 Spectral Clustering

The goal of clustering classification on a network is to identify groups of vertices sharing a (parametric) notion of similarity [23]. Usually, distance or centrality metrics are used for clustering. Spectral clustering is an important clustering method due to its precise foundation from algebraic graph theory [11]. It has been shown that spectral clustering has better overall performance, but with somewhat more instability compared to other algorithms [16]. 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. This technique 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.

2.4 SHAP

In general, the performance of classification algorithms is determined by the features used for training a particular prediction. SHAP (SHapley Additive exPlanation) is a framework that allows us to compute importance values for each feature using concepts from game theory [13]. Given a predictor and a training set, 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. A key advantage of SHAP is that its plots depict the contributions of different weights of features in a predictor [14].

3 Clustering-Based Function Prediction

This section presents the approach for gene function prediction based on spectral clustering. The approach combines multivariate statistics and supervised learning techniques to create a predictor enriched with the information of clusters. The predictor takes into account features that capture topological properties of the GCN.

The approach can be independently applied to each function in the set of gene functions A to be predicted. Formally, the inputs of the approach are a GCN, denoted by $G = (V, E, w)$, an annotation function $\phi : V \to 2^A$, a (biological) function $a \in A$, a set $K = \{k_0, \ldots, k_{m-1}\}$ for sampling the number of clusters, and a constant value $c \in [0, 1]$ for feature selection. The output is a function $\psi_a : V \to [0, 1]$, which indicates for each gene $v \in V$, the probability $\psi_a(v)$ of v having the function a.

The proposed approach consists of four stages. First, an enriched graph with information in ϕ is created from G. Second, m features are created for both G and its enriched version obtained in the previous stage, corresponding to the m number of clusters in K. Third, these new features are filtered by selecting those with more impact in the prediction task. Fourth, supervised learning is used to build the predictor ψ_a. These stages are depicted and detailed in Fig. 1. They have been implemented in Python and are available at https://github.com/migueleci/geneclust. The rest of this section is devoted to detailing each of the four stages in the approach.

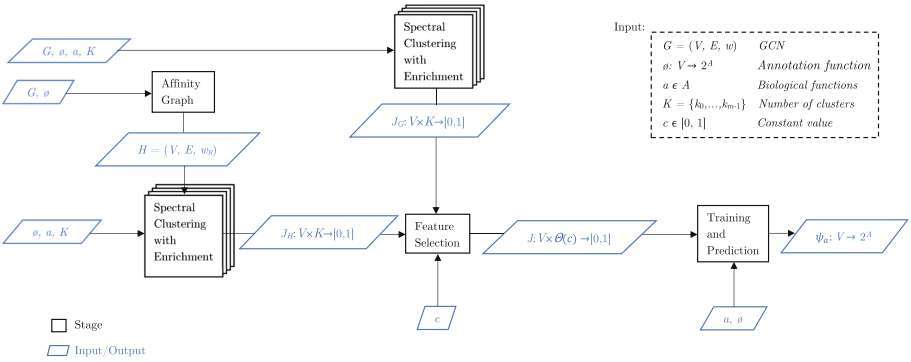

Fig. 1. The clustering-based approach is split into four stages. Namely, creation of affinity graph, clustering computation, feature selection, and training and prediction. Its inputs are a GCN, denoted by $G = (V, E, w)$, an annotation function $\phi : V \to 2^A$, a function $a \in A$, a set $K = \{k_0, \ldots, k_{m-1}\}$, and a real number $c \in [0, 1]$. Its output is a predictor ψ_a, which indicates, for each gene $v \in V$, the probability $\psi_a(v)$ of v having function a.

3.1 Affinity Graph Creation

An affinity graph $H = (V, E, w_H)$ 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 as specified by ϕ.

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

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

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

It is guaranteed that the range of w_H is $[0, 1]$ under the assumption that at least one element in the range of w is greater than 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.

3.2 Gene Enrichment with Clustering

For each graph $X \in \{G, H\}$, the goal of this stage is to produce a matrix $J_X : V \times K \to [0, 1]$ that specifies how likely it is for the genes to be associated to function a when X is decomposed in a given number of clusters. This is achieved in two steps, namely, by using clustering and computing a measure for each node in each cluster.

The decomposition of X is performed m times, once for each value k in K. The approach uses spectral clustering for finding the k clusters. The output of the clustering

algorithm is an assignment from nodes to clusters. Each cluster is used to gather and compute information with the goal of deciding whether a significant number of members associated to function a is (locally) present. Intuitively, if genes grouped together have a strong co-expression relation and most of them in the group are associated to gene function a, the remaining genes are also likely to be associated to a (guilt by association, see [21]). In this way, for each $v \in V$ and $k \in K$, the entry $J_X(v,k)$ specifies 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 [30].

3.3 Feature Selection

Matrices J_G and J_H represent structural properties of the GCN. They also represent associations between genes and functions based on partitions of each graph. The goal of this stage is to produce a matrix $J : V \times \Theta(c) \to [0,1]$ by selecting a reduced number of significant features $0 \leq \Theta(c) \leq 2m$ from J_G and J_H.

Feature selection is conveyed from J_G and J_H to J using SHAP. Let J_{G+H} denote the matrix resulting from extending J_G with the m columns of J_H. That is, for each $v \in V$, the expression $J_{G+H}(v, \_)$ denotes a function with domain $[0, 2m)$ and range $[0, 1]$, where the values in $[0, m)$ denote the p-values associated to v in G and the values in $[m, 2m)$ the ones associated to v in H. For each entry $J_{G+H}(v, j)$, with $v \in V$ and $0 \leq j < 2m$, the mean absolute SHAP value $s_{(v,j)}$ is computed after a large enough number of Shapely values are computed for feature j (executions of SHAP). Features are selected based on the cutoff

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

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.

3.4 Training and Prediction

This stage comprises a process that combines different supervised machine learning techniques/tools to finally build the predictor ψ_a. In particular, stratified k-fold cross-validation, the Synthetic Minority Over-sampling Technique (SMOTE) [3], hyper-parameter tuning [1], and XGBoost [4] are used sequentially in a pipeline. Stratified k-fold and over-sampling aim to overcome overfitting and learning bias. SMOTE is used to handle imbalanced datasets for underrepresented classes; it synthesizes new examples of the minority class from the existing ones. Hyper-parameter tuning aims to improve the performance of the prediction by optimizing parameters of the classifier such as, e.g., learning rate and maximum depth of trees. Gradient boosting decision tree algorithm, namely, XGBoost [4], is used as classifier.

The pipeline takes as input the matrix J, which specifies the significant features of J_G and J_H, the annotation function ϕ, and the set $\phi^{-1}(a)$ of genes associated to a. First, k-fold is applied to split the dataset into k different folds (this k has nothing to do with the input K). Each fold is used as test set, while the remaining $k-1$ folds are used for training. Furthermore, the training set is balanced using SMOTE to over-sample the minority class. The balanced training set is used to tune the following hyper-parameters of the XGBoost classifier: maximum tree depth for base learners, minimum sum of instance weight needed in a child, boosting learning rate, and subsample ratio of the training instance.

The prediction is carried out using the best estimator, i.e., the estimator with the combinations of parameters' values that achieve the best performance. The output are the probabilities of associations between the genes in V and function a, namely, the predictor ψ_a.

4 Case Study: *Zea Mays*

Next section describes a case study on applying the approach presented in Sect. 3 to maize (*Zea mays*). First, the maize data used for the study is described. Second, two benchmarks are introduced to compare the performance of the approach. Finally, the outcome of the proposed approach is contrasted with the benchmarks.

4.1 Data Description and Feature Selection

The co-expression information used in the study is borrowed from the ATTED-II database [18]. 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 AmiGO [2]; 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. A total of 5,361 genes are associated to 3,285 functions, comprising 20.5% of the genes in V.

Only 121 (3.7%) functions are associated to more than 10% of the genes in the GCN; that is, the data is highly imbalanced in most cases. For this reason, only biological processes corresponding to functions of level 1 in the function hierarchy defined by [9] are used for the prediction. For example, if a gene is associated to the function *response to light stimulus* and *response to stimulus* is its ancestor of level 1, then the gene will be associated to *response to stimulus*. Note that the functions of level 1 in the hierarchy are the more general ones.

As result, there are 19 biological processes of level 1, twelve of which are associated to more than 40 genes. Thus, the final dataset of associations between genes and functions is more balanced for applying supervised learning. Table 1 lists the 12 biological processes used for the prediction. In the end, the function $\phi : V \rightarrow 2^A$ for G associates $|A| = 12$ functions (biological processes of level 1 in function hierarchy) to 5,361 genes. The remaining input parameters are $K = \{10, 20, \ldots, 100\}$ and $c = 0.9$.

Table 1. Biological processes A of level 1 in the Gene Ontology hierarchy [9] used for prediction. The identifier and name of each function is presented in the first and second columns, respectively. The third column shows the number of maize genes associated to each function.

Term	Description	Genes	% of GCN
GO:0009987	Cellular process	4,269	16.34
GO:0008152	Metabolic process	3,047	11.66
GO:0065007	Biological regulation	1,492	5.71
GO:0051179	Localization	872	3.34
GO:0050896	Response to stimulus	851	3.26
GO:0023052	Signaling	307	1.17
GO:0032502	Developmental process	124	0.47
GO:0000003	Reproduction	78	0.30
GO:0032501	Multicellular organismal process	76	0.29
GO:0022414	Reproductive process	76	0.29
GO:0044419	Biological process involved in interspecies interaction between organisms	48	0.18
GO:0002376	Immune system process	47	0.18

Figure 2 depicts the distribution of filtered features using SHAP for each function in A. Note that, in most cases, the features coming from the affinity graph H are more important (or have more impact) for the prediction task.

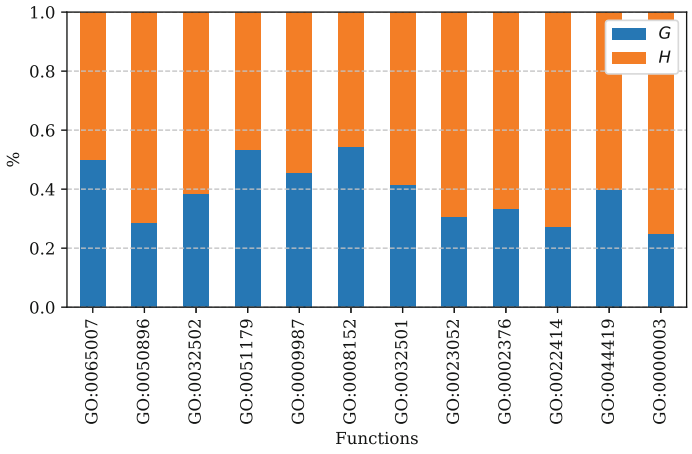

Fig. 2. Distribution of selected features using SHAP, from the 20 features corresponding to $K = \{10, 20, \ldots, 100\}$ for G and H. The features from the affinity graph H have more impact than those from G.

4.2 Benchmarks

Two models are built to benchmark the performance of the proposed approach. First, a baseline model using only the matrix of associations between genes and functions (i.e., matrix representation of function ϕ) as input features for the prediction. Second, a model including some topological properties of the GCN as additional features (together with the representation of ϕ) to train the predictor.

The topological properties included for each gene in V are the degree, average neighbor degree, eccentricity, clustering coefficient, closeness centrality, betweenness centrality, PageRank, Kleinberg's authority score, Kleinberg's hub scores, and coreness. These measures were computed with the help of igraph [12], an open source and free collection of network analysis tools available in several programming languages.

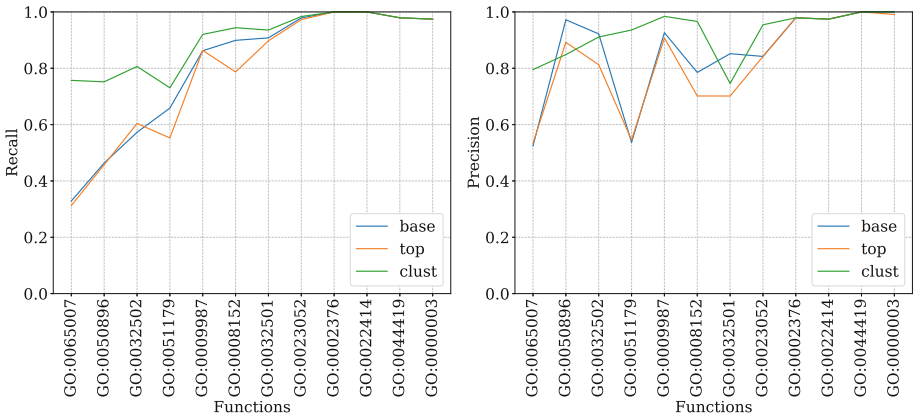

Fig. 3. Prediction performance measured with recall and precision score for the prediction of the 12 functions in A with the proposed approach and the benchmarks. The approach is labeled as *clust*, the baseline benchmark is labeled as *base*, and the one including the topological properties is labeled as *top*.

4.3 Summary of Results

Figure 3 presents the prediction performance of the proposed approach and its comparison to the benchmarks by using the recall and precision scores. It can be seen that the proposed approach outperforms both benchmarks in terms of the recall score. That is, the proposed approach is better at identifying the associations between genes and functions even though datasets are highly imbalanced in some cases. For example, recall scores for the function GO:0065007 are 0.76, 0.33, and 0.31 for *clust*, *base*, and *top*, respectively. That is, the proposed approach improves the performance in relation to the benchmarks.

The precision score measures how many of the predicted associations are relevant (i.e., true w.r.t. ϕ). Therefore, predictions that are not part of ϕ are considered as suggestions of possible gene-function associations. For this reason, the recall score is relevant

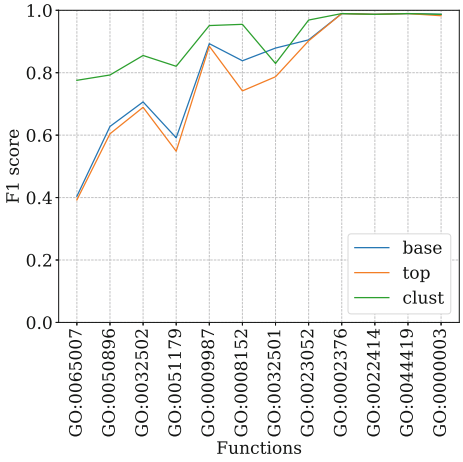

Fig. 4. Prediction performance measured by the F1 score with the proposed approach *clust*, and the benchmarks *base* and *top*.

to measure the performance of the predictions in relation to the known associations in ϕ. Note that both (recall and precision) scores of the proposed approach are higher than 0.75 for all functions A. Furthermore, the precision score of the proposed approach is lower than the benchmarks for 3 functions. In these cases, false positives can be considered to suggest candidate associations.

Figure 4 presents the F1 scores for the proposed approach and the benchmarks. Recall that the F1 score is the harmonic mean of precision p and recall r, and it is defined as $\text{F1} = \frac{2pr}{p+r}$. The F1 score of the proposed approach is at least as good as the base benchmark for 11 out 12 functions. The exception is in relation to the GO:0032502 function.

As final word on the choice of the clustering algorithm, it must be noted that other clustering algorithms, such as DBSCAN [22], were evaluated. However, spectral clustering showed the best and more consistent performance.

5 Concluding Remarks and Future Work

By combining network-based modeling, clustering, and supervised machine learning, 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, taking advantage of the GCN spectral decomposition, and the information available of associations between genes and functions. A comparison between the proposed approach and two benchmarks on a *Zea mayz* case study is presented. Using the structural information of the network, computed by a spectral clustering algorithm, is likely to be the key for the good performance of other GCN-based predictors. The proposed approach outperforms the two benchmarks, especially in terms of the recall score.

Two main lines of work can be considered for future work. First, gathering more information of associations between genes and functions for *Zea mays* is required. This way, it would be possible to use the functions beyond level 1 in the Gene Ontology hierarchy, therefore including more specific functions and their corresponding hierarchical constraints. Second, applying the proposed approach to identify genes associated to specific stresses, such as low temperature, can help to reduce the set of candidate genes that respond to treatments for in vivo validation.

Acknowledgments. 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.

References

1. Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. J. Mach. Learn. Res. **13**(10), 281–305 (2012)
2. Carbon, S., Mungall, C.: Gene Ontology Data Archive, July 2018. Type: dataset
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
4. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794 (2016)
5. Cho, H., Berger, B., Peng, J.: Diffusion component analysis: unraveling functional topology in biological networks. In: Przytycka, T.M. (ed.) RECOMB 2015. LNCS, vol. 9029, pp. 62–64. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16706-0_9
6. Cho, H., Berger, B., Peng, J.: Compact integration of multi-network topology for functional analysis of genes. Cell Syst. **3**(6), 540-548.e5 (2016)
7. Cruz, D.F., et al.: Using single-plant-omics in the field to link maize genes to functions and phenotypes. Mol. Syst. Biol. **16**(12), e9667 (2020)
8. Deng, M., Zhang, K., Mehta, S., Chen, T., Sun, F.: Prediction of protein function using protein-protein interaction data. J. Comput. Biol. **10**(6), 947–960 (2003)
9. Gene Ontology Consortium: The gene ontology resource: 20 years and still GOing strong. Nucl. Acids Res. **47**(D1), D330–D338 (2019)
10. Jensen, R.A.: Orthologs and paralogs - we need to get it right. Genom. Biol. **2**(8), 1–3 (2001)
11. Jia, H., Ding, S., Xu, X., Nie, R.: The latest research progress on spectral clustering. Neural Comput. Appl. **24**(7), 1477–1486 (2013). https://doi.org/10.1007/s00521-013-1439-2
12. Ju, W., Li, J., Yu, W., Zhang, R.: iGraph: an incremental data processing system for dynamic graph. Front. Comput. Sci. **10**(3), 462–476 (2016)
13. Lundberg, S., Lee, S.-I.: A Unified Approach to Interpreting Model Predictions. arXiv:1705.07874, November 2017
14. Lundberg, S.M., et al.: From local explanations to global understanding with explainable AI for trees. Nat. Mach. Intell. **2**(1), 2522–5839 (2020)
15. Luo, F., et al.: Constructing gene co-expression networks and predicting functions of unknown genes by random matrix theory. BMC Bioinform. **8**(1), 299 (2007)

16. Murugesan, N., Cho, I., Tortora, C.: Benchmarking in cluster analysis: a study on spectral clustering, DBSCAN, and K-Means. In: Chadjipadelis, T., Lausen, B., Markos, A., Lee, T.R., Montanari, A., Nugent, R. (eds.) IFCS 2019. SCDAKO, pp. 175–185. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-60104-1_20
17. Musungu, B., Bhatnagar, D., Brown, R.L., Fakhoury, A.M., Geisler, M.: A predicted protein interactome identifies conserved global networks and disease resistance subnetworks in maize. Front. Genet. **6** (2015)
18. Obayashi, T., Aoki, Y., Tadaka, S., Kagaya, Y., Kinoshita, K.: 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), e3–e3 (2018)
19. Obayashi, T., Kinoshita, K.: COXPRESdb: a database to compare gene coexpression in seven model animals. Nucl. Acids Res. **39**, D1016–D1022 (2011)
20. Oti, M., van Reeuwijk, J., Huynen, M.A., Brunner, H.G.: Conserved co-expression for candidate disease gene prioritization. BMC Bioinform. **9**(1), 208 (2008)
21. Petsko, G.A.: Guilt by association. Genom. Biol. **10**(4), 104 (2009)
22. Rehman, S.U., Asghar, S., Fong, S., Sarasvady, S.: DBSCAN: past, present and future. In: The Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014), pp. 232–238, Bangalore, India, February 2014
23. Rodriguez, M.Z., et al.: Clustering algorithms: a comparative approach. PLoS One **14**(1), e0210236 (2019)
24. Romero, M., Finke, J., Quimbaya, M., Rocha, C.: In-silico gene annotation prediction using the co-expression network structure. In: Cherifi, H., Gaito, S., Mendes, J.F., Moro, E., Rocha, L.M. (eds.) COMPLEX NETWORKS 2019. SCI, vol. 882, pp. 802–812. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-36683-4_64
25. Rust, A.G., Mongin, E., Birney, E.: Genome annotation techniques: new approaches and challenges. Drug Discov. Today **7**(11), S70–S76 (2002)
26. Valentini, G.: True path rule hierarchical ensembles. In: Benediktsson, J.A., Kittler, J., Roli, F. (eds.) MCS 2009. LNCS, vol. 5519, pp. 232–241. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02326-2_24
27. van Dam, S., Võsa, U., van der Graaf, A., Franke, L., de Magalhães, J.P.: Gene co-expression analysis for functional classification and gene-disease predictions. Briefings Bioinform. **19**(4), 139 (2017)
28. Vandepoele, K., Quimbaya, M., Casneuf, T., De Veylder, L., Van de Peer, Y.: Unraveling transcriptional control in Arabidopsis using CIS-regulatory elements and coexpression networks. Plant Physiology **150**(2), 535–546 (2009)
29. Yandell, M., Ence, D.: A beginner's guide to eukaryotic genome annotation. Nat. Rev. Genet. **13**(5), 329–342 (2012)
30. Yon Rhee, S., Wood, V., Dolinski, K., Draghici, S.: Use and misuse of the gene ontology annotations. Nat. Rev. Genet. **9**(7), 509–515 (2008)
31. Zhou, G., Wang, J., Zhang, X., Guo, M., Yu, G.: Predicting functions of maize proteins using graph convolutional network. BMC Bioinform. **21**(S16), 420 (2020)
32. Zhou, Y., Young, J.A., Santrosyan, A., Chen, K., Yan, S.F., Winzeler, E.A.: In silico gene function prediction using ontology-based pattern identification. Bioinformatics **21**(7), 1237–1245 (2005)