NRC Publications Archive Archives des publications du CNRC

Utilization of Gene Ontology in Semi-supervised Clustering

Doan, Duong, D.; Wang, Yunli; Pan, Youlian

This publication could be one of several versions: author's original, accepted manuscript or the publisher's version. / La version de cette publication peut être l'une des suivantes : la version prépublication de l'auteur, la version acceptée du manuscrit ou la version de l'éditeur.

For the publisher's version, please access the DOI link below. / Pour consulter la version de l'éditeur, utilisez le lien DOI ci-dessous.

Publisher's version / Version de l'éditeur:

<https://doi.org/10.1109/CIBCB.2011.5948467>

2011 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB 2011) Proceedings, 2011-05-01

NRC Publications Record / Notice d'Archives des publications de CNRC:

<https://nrc-publications.canada.ca/eng/view/object/?id=64227bbc-8d1f-44dc-9447-0bfcde158fd8>

<https://publications-cnrc.canada.ca/fra/voir/objet/?id=64227bbc-8d1f-44dc-9447-0bfcde158fd8>

Access and use of this website and the material on it are subject to the Terms and Conditions set forth at

<https://nrc-publications.canada.ca/eng/copyright>

READ THESE TERMS AND CONDITIONS CAREFULLY BEFORE USING THIS WEBSITE.

L'accès à ce site Web et l'utilisation de son contenu sont assujettis aux conditions présentées dans le site

<https://publications-cnrc.canada.ca/fra/droits>

LISEZ CES CONDITIONS ATTENTIVEMENT AVANT D'UTILISER CE SITE WEB.

Questions? Contact the NRC Publications Archive team at

PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca. If you wish to email the authors directly, please see the first page of the publication for their contact information.

Vous avez des questions? Nous pouvons vous aider. Pour communiquer directement avec un auteur, consultez la première page de la revue dans laquelle son article a été publié afin de trouver ses coordonnées. Si vous n'arrivez pas à les repérer, communiquez avec nous à PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca.

Utilization of Gene Ontology in Semi-supervised Clustering

Duong D. Doan*

Faculty of Computer Science, University
of New Brunswick, 540 Windsor Street,
Fredericton, NB, Canada
b89ct@unb.ca

Yunli Wang

Institute for Information Technology,
National Research Council Canada, 46
Dineen Drive, Fredericton, NB, Canada
Yunli.Wang@nrc-cnrc.gc.ca

Youlian Pan

Institute for Information Technology,
National Research Council Canada, 1200
Montreal Road, Ottawa, ON, Canada
youlian.pan@nrc-cnrc.gc.ca

Abstract— Semi-supervised clustering incorporating biological relevance as a prior knowledge has been favoured over the past decade. However, selection of prior knowledge has been a challenge. We generate prior knowledge from GO terms at different levels of GO hierarchy and use them to study their impact on the performance of subsequent clustering of microarray data by using MPCKMeans and GOFuzzy. We evaluate the performance by F-measure and the number of specific GO terms and transcription factors. The clustering result with prior knowledge generated from lower levels of GO hierarchy have higher F-measure and more number of specific GO terms and transcription factors. MPCKMeans with prior knowledge generated from multiple levels in the GO hierarchy outperforms GOFuzzy with prior knowledge from the first level in the GO hierarchy. A small amount (1-2%) of prior knowledge can improve semi-supervised clustering result substantially and the more specific prior knowledge is generally more efficient in guiding the semi-supervised clustering process.

Keywords—Gene Ontology, semi-supervised clustering.

I. INTRODUCTION

Cells in a living organism carry identical set of genes. However, different subsets of these genes are expressed under different conditions, in different tissues and in different periods of a time series. It is commonly believed that co-expressed genes are probably functionally related or co-regulated by certain transcription factors. Therefore we can predict functions of an unknown gene based on known functions of other genes in a group sharing similar attributes. Grouping data items into clusters based on the similarity of their attributes is the ultimate goal of clustering. Clustering methods can be classified into two main groups, unsupervised and semi-supervised. In unsupervised clustering, there is no information about the membership of data items to predefined classes or relationships between data items. In semi-supervised clustering, there is limited amount of prior knowledge to guide the clustering process. Even with very little amount of prior knowledge, in normal circumstances, semi-supervised clustering produces a better result over unsupervised clustering [1], [2], [11].

Research on semi-supervised clustering has been very active over the past decade. Bilenko et al. [2] present a comprehensive semi-supervised clustering algorithm,

MPCKMeans, that integrates both constraint learning and metric learning in clustering processes. MPCKMeans assumes that some limited knowledge about a dataset is known. For example, some specific data points of the dataset must be or must not be in the same cluster. If two data points must be in the same cluster, a *must-link* pairwise constraint is created between them. Otherwise, if two data points must not be in the same cluster, a *cannot-link* pairwise constraint is created between the two data points. Besides using pairwise constraints for initializing clusters, MPCKMeans minimizes instances of violating such constraints in each resulted cluster. Because generating prior knowledge manually is time consuming, it is therefore important to automatically generate prior knowledge with high accuracy for semi-supervised clustering methods. Costa et al. [4] generate pairwise constraints from image data while Schonhuth et al. [5] generate pairwise constraints from transcription factor data; they both use a mixture model of multivariate Gaussians with diagonal covariance matrix to cluster data. Gao et al. [6] use partial background information from different sources and in various forms to aid the clustering process. For example, background knowledge in the biomedical literature can be used as a prior knowledge. Chopra et al. [7] present an interesting clustering model that can generate multiple versions of clusters from a single dataset by projecting the dataset onto different sets of landmark genes associated with different biological processes selected from Gene Ontology. Dotan-Cohen et al. [8] and Kang et al. [9] use the structure of Gene Ontology to calculate similarities between genes. Macintyre et al. [10] do not use the structure of Gene Ontology but instead the gene ontology profiles and gene expression contexts to cluster data. Tari et al. [11] develop a semi-supervised clustering algorithm GOFuzzy c-means based on fuzzy c-means clustering algorithm and use prior knowledge generated from the cutoff of the first level of the Gene Ontology hierarchical structure to train their clustering model. Prior knowledge is used to initialize fuzzy memberships of a gene; the gene has different scores of memberships in different clusters. In such a way, GOFuzzy integrate membership distribution of the genes having GO annotation together with their other features and group them others without GO annotation based on the same other feature. GOFuzzy c-means shows some potential good clustering results with high z-scores. Even though all of these approaches

*The work was done while DDD was at NRC-IIT working with YW

show the effectiveness of prior knowledge in training clustering systems, none of them studies the influences of different ways of generating prior knowledge on clustering results.

In this paper, we propose a semi-supervised clustering framework by generating pairwise constraints from Gene Ontology at different levels as prior knowledge and use MPCKMeans platform [[2]] for subsequent clustering. Unlike GOFuzzy c-means where pairwise constraints are generated from the first level of GO hierarchy, we generate crisp knowledge from a lower level of the GO hierarchy or fuzzy knowledge from multiple levels of the GO hierarchy. We use F-measure to evaluate performances of different methods of generating prior knowledge for the Spellman, Gasch, and Tari's YeastA and YeastB datasets [[11], [12], [13]], and the Brassica time-series dataset [[14]]. We compare the performance of MPCKMeans using fuzzy knowledge generated from multiple levels of GO with the performance of GOFuzzy. Evaluation based on gene and transcription factor enrichment analysis shows that MPCKMeans using fuzzy knowledge produces much better results than those of GOFuzzy with prior knowledge generated from the first level of GO hierarchy.

II. METHODS

A. Datasets

We use 5 datasets to evaluate our approach. The first two datasets are yeast microarray datasets: Gasch [[13]] with 6152 genes and 173 attributes, and Spellman [[12]] with 6178 genes and 82 attributes. We also use Tari's YeastA and YeastB [[11]], two variations of the Spellman and Gasch datasets, respectively, where YeastA has 6221 genes and 80 attributes, and YeastB has 6152 genes and 93 attributes. The last dataset is a time-series canola (*Brassica napus*) microarray dataset [[14]] with 1229 genes, three different time points during seed development. For the evaluation purposes, we only use 3957 genes in Gasch, 3960 genes in Spellman, 3962 genes in YeastA, and 2957 genes in YeastB that are annotated with GO biological process. Similarly, we only keep 507 genes in the canola dataset that have GO biological process. Genes that are not annotated with GO biological process are excluded from our experiments.

B. Generating prior knowledge

We use Gene Ontology [15] to generate prior knowledge for Gasch, Spellman, YeastA, YeastB, and canola datasets. Gene Ontology (GO) is a controlled vocabulary of terms for describing gene product characteristics and organized in an acyclic hierarchical structure, where general GO terms are placed near the root and specific GO terms near the leaf nodes. Any two adjacent GO terms on the same path are linked by arcs describing an offspring node either *is-a* or *part-of* its parental node. Gene Ontology is organized in three domains: biological process, molecular function, and cellular component, and generic across different species. Because GO is quite broad, GO Slim, a subset of GO terms, is sometime preferably used. Besides generic GO Slim, which is species-independent for common use, there are many GO Slims targeting at specific species, such as TAIR for plant and SGD for yeast. In order to

compare the performance of our clustering framework with that of GOFuzzy c-means that only accepts old versions of GO annotations, we use generic GO Slim compiled in July 2006, GO Slim SGD compiled in September 2005, and Yeast GO annotation compiled on September 2005 for Gasch, Spellman, YeastA, and YeastB datasets.

Since GO annotation for *Brassica* is not available, we use that of its close relatives, *Arabidopsis*, in this study. We use generic GO Slim compiled in September 2009, GO Slim TAIR compiled in September 2009, and *Arabidopsis* GO annotation compiled in November 2009 for the *Brassica* dataset. We build a GO hierarchy for all GO terms associated with biological process for all genes in each dataset. These GO terms are organized in a tree with the root node GO:0008150 for biological process.

C. Knowledge from the first level of GO hierarchy

We make a cut at the first level from the root node of the GO hierarchy tree. Specifically, we select general GO terms at the first level of the GO hierarchy. For any two genes sharing the same general GO term or their GO terms are descendants of the same general GO term, we create a *must-link* pairwise constraint between these two genes. We use the pairwise constraints as prior knowledge for MPCKMeans and for the evaluation. Table I displays a list of 43 general GO terms obtained from the first level of the GO hierarchy for yeast datasets. We obtain 47 general GO terms from the first level of GO hierarchy for the Brassica dataset.

D. Knowledge from lower levels of GO hierarchy

Instead of generating pairwise constraints from general GO terms at the first level of the GO hierarchy, we use GO terms at lower levels. We use general GO terms at levels 2, 3, 4, and 5, respectively, to generate prior knowledge. Specifically, a *must-link* pairwise constraint is created between a pair of genes sharing a common general GO term at that level. Genes that share an upper levels GO terms, but not at this level are excluded from the experiment. For example, we generated 63 general GO terms at the second level from the YeastA dataset. These 63 GO terms do not include those at the first level unless some of them are at the second level in one of the alternative GO hierarchy path. To evaluate the results using these pairwise constraints as prior knowledge, we eliminate any gene from the input data that does not have a GO in the selected general GO term set. We tested this approach using only YeastA dataset. At each level of GO hierarchy, 50 clusters ($k=50$) are generated.

E. Fuzzy knowledge from multiple levels of GO hierarchy

We generate prior knowledge for MPCKMeans from all GO terms at various levels of GO hierarchy. Specifically, we calculate the similarity between any two GO terms in Gene Ontology using the G-SESAME tool [[16]]. The similarity between two GO terms is in the range from 0 to 1, where 0 means two GO terms are completely unrelated and 1 means they are very closely related. Let $S_{GO}(go; go_l)$ be the similarity between two GO terms go and go_l computed by G-SESAME based on the GO hierarchy. We compute the maximum semantic similarity $Sim(go, GO)$ between a GO term go and a

set of GO terms $GO = go_1, \dots, go_k$ using the following formula [[16]]:

$$Sim(go, GO) = \max_{1 \leq l \leq k} [S_{GO}(go, go_l)]$$

Assuming that gene g_i is annotated by a set of GO terms $GO_i = go_{i1}, go_{ip}$ and gene g_j is annotated by a set of GO terms $GO_j = go_{j1}, \dots, go_{jq}$, the semantic similarity between g_i and g_j is computed as follows [[16]]:

$$Sim(g_i, g_j) = [\sum_{1 \leq l \leq p} Sim(go_{il}, GO_j) + \sum_{1 \leq h \leq q} Sim(go_{jh}, GO_i)] / (p + q)$$

TABLE I. LIST OF 43 GENERAL GO TERMS FROM THE FIRST LEVEL OF GO HIERARCHY

GO term	Description
GO:0000746	conjugation
GO:0000910	cytokinesis
GO:0006091	generation of precursor metabolites and energy
GO:0006259	DNA metabolism
GO:0006350	cellular transcription
GO:0006412	protein biosynthesis
GO:0006457	protein folding
GO:0006464	protein modification process
GO:0006519	cellular amino acid and derivative metabolic process
GO:0006725	cellular aromatic compound metabolic process
GO:0006766	vitamin metabolic process
GO:0006810	transport
GO:0006950	response to stress
GO:0006997	nucleus organization
GO:0007005	mitochondrion organization
GO:0007010	cytoskeleton organization
GO:0007031	peroxisome organization
GO:0007033	vacuole organization
GO:0007047	cellular cell wall organization
GO:0007049	cell cycle
GO:0007059	chromosome segregation
GO:0007114	cell budding
GO:0007124	pseudohyphal growth
GO:0007126	meiosis
GO:0007165	signal transduction
GO:0016044	cellular membrane organization
GO:0016050	vesicle organization
GO:0016070	RNA metabolic process
GO:0016192	vesicle-mediated transport
GO:0019725	cellular homeostasis
GO:0030435	sporulation resulting in formation of a cellular spore
GO:0032196	transposition
GO:0032989	cellular component morphogenesis
GO:0042221	response to chemical stimulus
GO:0042254	ribosome biogenesis
GO:0044255	cellular lipid metabolic process
GO:0044257	cellular protein catabolic process
GO:0044262	cellular carbohydrate metabolic process
GO:0045333	cellular respiration
GO:0046483	heterocycle metabolic process
GO:0051186	cofactor metabolic process
GO:0051276	chromosome organization
GO:0070271	protein complex biogenesis

Once the similarity between any two genes in the dataset is computed, we select those pairs of genes whose similarities between two genes are greater than a threshold 0.85 and use a subset of these pairs as labeled data to train MPCKMeans. We modify the objective function of MPCKMeans to handle fuzzy knowledge. Specifically, instead of using the same penalty cost (i.e., 1) for all violated pairwise constraints, we use the similarity associated with each pairwise constraint as the cost for violation. Even though fuzzy knowledge is used to guide the clustering process, a gene can only belong to a single cluster in the clustering results of MPCKMeans.

F. Clustering Algorithms

Three algorithms are used to cluster gene expression data: KMeans, MPCKMeans and GOFuzzy. KMeans is an unsupervised clustering algorithm that aims to partition a dataset into k clusters, where the total distance between data points in each cluster and cluster centers is minimized. MPCKMeans [[2]] is a comprehensive semi-supervised clustering algorithm by incorporating prior knowledge into KMeans. It integrates both constraint learning and metric learning in its clustering process. Prior knowledge is in the form of *must-link* and *cannot-link* pairwise constraints, which is part of input data to MPCKMean. If a pairwise constraint is violated, a penalty costs is added into the cost function. Penalty costs are different from each other depending on the squared distances between two data points of violated constraints. Metric learning is to ensure that distances between data points in the same cluster are minimized while distances between data points of different clusters are maximized. In our experiment, only *must-link* pairwise constraints are used.

GO Fuzzy c-means is developed from the fuzzy c-means clustering algorithm [[11]]. It uses general GO terms from the first level of the GO hierarchy to determine the number of clusters, to initialize and to update the membership of genes in the clustering process. GO Fuzzy c-means algorithm computes membership of each gene to different clusters.

G. Evaluation Methods

When knowledge from a single level of the GO hierarchy is used, clustering results are evaluated based on pairwise F-measure [[2]]. When the knowledge from multiple levels of the GO structure is used, clustering results are evaluated by unique GO terms and unique transcription factors. Specifically, how many unique GO terms and unique transcription factors associated with the clustering result? Pairwise F-measure is calculated as follows.

$$Precision = \text{Pairs correctly predicted} / \text{Pairs predicted},$$

$$Recall = \text{Pair correctly predicted} / \text{Total correct pairs},$$

$$F\text{-Measure} = 2 \times Precision \times Recall / (Precision + Recall).$$

The number of *Pairs correctly predicted* and the number of *Pairs predicted* are subjective to the algorithm used. Whereas, the number of *Total correct pairs* is independent of the algorithm. Values of *Precision*, *Recall* and *F-Measure* are in the range [0..1].

We also evaluate the performance of MPCKMeans, GOFuzzy, and KMeans by unique GO terms and unique transcription factors. A good cluster ideally would have many genes associated with similar biological processes or some common transcription factors. These biological processes, represented by a GO terms, or the common transcription factors are called *significant GO terms* or *significant transcription factors*. We use Gene Ontology Analyzer package (GOAL) [[17]] to extract GO terms and transcription factors (TFs) associated with each cluster of genes. We examine if there is any GO term or transcription factor over-represented with each cluster than it would be by chance. Buferroni correction is used for testing statistical significance and p-value is set to ≤ 0.005 . A maximum of seven steps on a GO tree search beginning from leaf is performed and transcription factors are included for analysis. We use transcription factors associated with genes from [[18]]. Only GO terms and transcription factors associated with more than two genes are returned. We follow the method proposed in [[7]] to evaluate clustering results based on unique GO terms analysis and extend this evaluation method for unique transcription factor analysis. We compare the number of *significant GO terms* and *transcription factors* associated with our MPCKMeans clustering results to those obtained from KMeans and GOFuzzy. GO terms and transcription factor associated with the clustering results of KMeans are *original GO terms* and *original transcription factors* [[7]], which are generated by a standard unsupervised clustering method. Any *significant GO term* or *transcription factor* appearing in both MPCKMeans and KMeans, or in both GOFuzzy and KMeans is an *overlapping GO term* or *overlapping transcription factor*. Any *significant GO term* or *transcription factor* appearing in MPCKMeans but not in KMeans, or appearing in GOFuzzy but not in KMeans is a *unique GO term* or *unique transcription factor*. A *unique GO term* or *unique transcription factor* indicates new found biological evidence associated with a cluster that cannot be found by other conventional clustering methods. Because the ultimate target of gene clustering is to group co-expressed genes so that to predict functions of unknown genes based on functions of other known genes, new found biological evidence is an indicator of a good clustering method. Therefore, a clustering method is considered better when it generates higher number of *unique GO terms* and *unique transcription factors* [[7]].

III. RESULTS

We use five-fold cross validation; four among five folds are used to train MPCKMeans and one fold is used to evaluate its performance. The algorithm runs 10 times to increase the accuracy of results.

A. Performance with prior knowledge from the first level of GO hierarchy

The numbers of clusters for the yeast and Brassica datasets are determined from literature [[7], 14] rather than from the number of general GO terms obtained from the first level of the GO hierarchy. There are 60 clusters generated for the Gasch dataset, 50 clusters for the YeastA, YeastB, and Spellman

datasets. Figure 1 shows the performances of MPCKMeans trained by pairwise constraints from the first level of the GO hierarchy. When there is no prior knowledge provided, F-Measure is very small (below 0.08) for all four yeast datasets. F-Measure is relatively unchanged when the number of pairwise constraints increases to 1000. When the number of pairwise constraints increases from 1000 to 1500, F-Measure steeply increases and reaches as high as 0.19. It is worth mentioning that the total number of constraints for each microarray dataset is very large. For example the total number of constraints is 1227032 in YeastA and 1222832 in YeastB. When we use only 1.2% (1500 randomly selected pairwise constraints for YeastA) of the total number of constraints, the F-Measure rises from 0.062 to 0.17.

The performance of our approach for the *Brassica* dataset with 24 clusters is shown in Fig. 2. When there is no prior knowledge provided, F-Measure is 0.16 and it does not change much when the number of pairwise constraints increases to 200. When we continue increase the number of constraints from 200 to 1000, F-Measure markedly increases to about 0.64 and remains relatively constant when the number of pairwise constraints increases from 1000 to 2000.

Figure 1. Performance of MPCKMeans with the YeastA, YeastB, Gasch and Spellman datasets.

B. Performance with prior knowledge from low levels of GO hierarchy

We compare influences of prior knowledge generated from different levels of the GO hierarchy on the performance of MPCKMeans. Because the performance is relatively independent from the dataset used when the prior knowledge is generated from the first level of the GO hierarchy, we conducted this experiments based only on the YeastA dataset. The performance of MPCKMeans is shown in Fig. 3. At the lower levels of GO constraint, the F-measure markedly increases when the number of pairwise constraints increases from 100 to 1000. The slope of increase between 100 and 500 constraints appears to be correlated with GO levels. The overall performance is the best when the constraints are from 5th GO level.

We repeat the above experiment by generating different number of clusters at different GO level from the YeastA datasets. There are 63, 114, 77, and 48 clusters at GO levels 2, 3, 4, and 5, respectively. Fig. 4 shows the result. The change in the number of clusters does not seem to significantly affect the performance of MPCKMeans (Figs. 3, 4).

C. Performance with fuzzy prior knowledge from multiple levels of GO hierarchy

Tables II and III display the performance of GOFuzzy and MPCKMeans ran in 10 times on the YeastA dataset, using GO and transcription factor analysis. To compare the performance of GOFuzzy with MPCKMeans and KMeans, we transform fuzzy clustering by using GOFuzzy into crisp clusters. Recall that in GoFuzzy, a gene is assigned a membership score to each individual cluster. By transforming GoFuzzy into crisp clusters, we keep only genes in a cluster whose membership score is greater than a chosen threshold. Two thresholds 0.037 and 0.035 are used to generate two different crisp clustering results, *GOFuzzy*_{0.037} and *GOFuzzy*_{0.035}, from the fuzzy clustering results of GOFuzzy.

Figure 2. Performance of MPCKMeans with the *Brassica* dataset.

Figure 3. Performance of MPCKMeans with YeastA dataset, generating 50 clusters for all low cutoff levels.

Figure 4. Performance of MPCKMeans with YeastA dataset, generating different number of clusters for each cutoff level.

TABLE II. INTERSECTION BETWEEN SIGNIFICANT GO TERMS GENERATED BY DIFFERENT CLUSTERING METHODS.

	# of original GO terms in KMeans	# of overlapping GO terms in KMeans and other methods	# of unique GO terms in KMeans and other methods
<i>GOFuzzy</i> _{0.037}	4022	240	348
<i>GOFuzzy</i> _{0.035}	4022	803	1802
MPCKMeans	4022	1897	2004

TABLE III. INTERSECTION BETWEEN SIGNIFICANT TRANSCRIPTION FACTORS GENERATED BY DIFFERENT CLUSTERING METHODS.

	# of original TFs in KMeans	# of overlapping TFs in KMeans and other methods	# of unique TFs in KMeans and other methods
<i>GOFuzzy</i> _{0.037}	69	0	2
<i>GOFuzzy</i> _{0.035}	69	2	27
MPCKMeans	69	18	51

TABLE IV. NON-OVERLAPPED TRANSCRIPTION FACTORS DISCOVERED BY DIFFERENT CLUSTERING METHODS.

Unique transcription factors associating with clustering results of MPCKMeans	ARG80, ARG81, ARO80, BAS1, CBF1, CHA4, DAL82, FHL1, GAT1, GCN4, GCR2, GLN3, HAP5, LEU3, MET31, MET4, PHO2, PUT3, RAP1, RCS1, SFP1, STP1
Unique transcription factors associating with clustering results of GOFuzzy _{0.035}	ARG81, FHL1, GCR2, LEU3, MET4, RAP1, RTG3, SFP1, STP1
Unique transcription factors associating with clustering results of GOFuzzy _{0.037}	MET4, STP1

MPCKMeans using fuzzy knowledge from all levels of GO hierarchy can generate a larger number of unique GO terms and transcription factors compared to those generated by GO Fuzzy using knowledge from GO terms at the first level of GO hierarchy. Table IV displays non-overlapped unique transcription factors from clustering results of GOFuzzy and MPCKMeans. All the unique transcription factors discovered by GOFuzzy_{0.035} and GOFuzzy_{0.037} are also discovered by MPCKMeans but many unique transcription factors discovered by MPCKMeans are not discovered by GOFuzzy_{0.035} or GOFuzzy_{0.037}. MPCKMeans with fuzzy knowledge from all levels of GO hierarchy reveals new biological aspects associating with clusters than GOFuzzy with knowledge from the first level of GO hierarchy. This is strong evidence that MPCKMeans produces more biologically meaningful clustering results than those of GOFuzzy.

IV. DISCUSSIONS AND CONCLUSIONS

We present a semi-supervised clustering framework to cluster gene expression data by using the MPCKMeans as the clustering algorithm and present different approaches of generating pairwise constraints from Gene Ontology as prior knowledge for MPCKMeans. We demonstrated a better clustering result by using knowledge from lower levels of GO hierarchy than those from the first level because the lower level of a GO term the more specific, indicating any pair of genes that are commonly annotated to such GO term are closely related. Thus pairwise constraints provided for MPCKMeans during the training process would be more accurate and meaningful.

In this study we generate prior knowledge at first five levels of GO hierarchy and demonstrated that the more specific GO terms are generally more effective constraints in the semi-clustering process. It is possible that GO terms at a lower level, such as 6 or 7 could be more effective constraints. Nevertheless, it is evident from Figs 3 and 4 that level 5 GO terms are sufficiently good constraints.

Even though GO Fuzzy c-means shows some potential good clustering results, there a limitation with the clustering

method. GO terms from the first level of the GO hierarchy are too general. Thus two genes sharing the same general GO term may not necessarily be closely related. As the result, the prior knowledge generated solely from the first level may not be sufficiently good in training a clustering model.

Our framework can accept pairwise constraints generated from other biological sources such as transcription factor or protein-protein interaction networks. Moreover, knowledge generated is tested on the MPCKMeans algorithm but can be used for any semi-supervised clustering algorithm that accepts pairwise constraints. Our evaluation shows that pairwise constraints generated from low levels of the GO hierarchy better guide clustering process of MPCKMeans than those from the first level of GO. When pairwise constraints generated from multiple levels of the GO hierarchy are used, MPCKMeans outperforms GO Fuzzy with prior knowledge generated from the first level of GO hierarchy.

REFERENCES

- [1] Y. Pan, J. D. Pylatuk, J. Ouyang, A. F. Famili, P. R. Fobert, "Discovery of functional genes for systemic acquired resistance in *Arabidopsis thaliana* through integrated data mining". *Journal of Bioinformatics and Computational Biology* 2004, 2(4): 639-655.
- [2] M. Bilenko, S. Basu, R. J. Mooney. "Integrating constraints and metric learning in semi-supervised clustering". In: *Proceedings of the Twenty-First International Conference on Machine Learning* 2004, pp. 81-88.
- [3] Z. Lu, T. K. Leen. "Semi-supervised clustering with pairwise constraints: a discriminative approach". In: *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics*, 2007.
- [4] I. G. Costa, R. Krause, L. Opitz, A. Schliep. "Semi-supervised learning for the identification of syn-expressed genes from fused microarray and in situ image data". *BMC Bioinformatics* 2007, 8 (Suppl 10):S3.
- [5] A. Schonhuth, I. G. Costa, A. Schliep. "Semi-supervised clustering of yeast gene expression data". In: *Okada A. et al. (eds.) Cooperation in Classification and Data Analysis* 2009, pp. 151-159.
- [6] J. Gao, P. N. Tan, H. Cheng. "Semi-supervised clustering with partial background information". In: *Proceedings of the Sixth SIAM International Conference on Data Mining* 2006, pp. 487-491.
- [7] P. Chopra, J. Kang, J. Yang, H. J. Cho, H. S. Kim, M-G. Lee. "Microarray data mining using landmark gene-guided clustering". *BMC Bioinformatics* 2008, 9:92.
- [8] D. Dotan-Cohen, S. Kasif, A. A. Melkman. "Seeing the forest for the trees: using the Gene Ontology to restructure hierarchical clustering". *Bioinformatics* 2009, 25(14): pp. 1789-1795.
- [9] B. Y. Kang, S. Ko, D. W. Kim. "SICAGO: Semi-supervised cluster analysis using semantic distance between gene pairs in Gene Ontology". *Bioinformatics* 2010, 26(10): pp. 1384-1385.
- [10] G. Macintyre, J. Bailey, D. Gustafsson, I. Haviv, A. Kowalczyk. "Using Gene Ontology annotations in exploratory microarray clustering to understand cancer etiology". *Pattern Recognition Letters* 2010, Article in Press.
- [11] L. Tari, C. Baral, S. Kim. "Fuzzy c-means clustering with prior biological knowledge". *Journal of Biomedical Informatics* 2009, 42(1): 74-81
- [12] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, B. Futcher. "Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization". *Mol Biol Cell* 1998, 9(12): pp. 3273-3297.
- [13] A. P. Gasch, P. T. Spellman, C. M. Kao, O. Carmel-Harel, M. B. Eisen, G. Storz, D. Botstein, P. O. Brown. "Genomic expression programs in the response of yeast cells to Environmental Changes". *Mol Biol Cell* 2000, 11(12): pp. 4241-4257.

- [14] Y. Huang, L. Chen, L. Wang, K. Vijayan, S. Phan, Z. Liu, L. Wan, A. Ross, D. Xiang, R. Datla, Y. Pan, J. Zou. "Probing the endosperm gene expression landscape in *Brassica napus*, *BMC Genomics* 2009, 10:256.
- [15] Gene Ontology. <http://www.geneontology.org>. Last accessed May 21, 2010
- [16] Z. Du, L. Li, C. F. Chen, P. S. Yu, J. Z. Wang. "G-SESAME: web tools for go term based gene similarity analysis and knowledge discovery". *Nucleic Acids Research* 2009, 37: pp. W345-W349.
- [17] A. B. Tchagang, A. Gawronski, H. Berube, S. Phan, F. Famili, Y. Pan. "GOAL: A software tool for assessing biological significance of gene groups". *BMC Bioinformatics* 2010, 11: 229.
- [18] C. T. Harbison, D. B. Gordon, T. I. Lee, N. J. Rinaldi, K. D. Macisaac, T. W. Danford, N. M. Hannett, J. B. Tagne, D. B. Reynolds, J. Yoo, E. G. Jennings, J. Zeitlinger, D. K. Pokholok, M. Kellis, P. A. Rolfe, K. T. Takusagawa, E. S. Lander, D. K. Gifford, E. Fraenkel, R. A. Young. "Transcriptional regulatory code of a eukaryotic genome". *Nature* 2004, 431: pp. 99-104.