The Conjunctive Disjunctive Node Kernel

Dinh Tran Van[1], Alessandro Sperduti[1] and Fabrizio Costa[2]

1- Department of Mathematics, Padova University
Trieste, 63, 35121 Padova, Italy

2- Bioinformatics Group, Department of Computer Science, Freiburg University
Georges-Kohler-Allee 106, 79110 Freiburg, Germany

Abstract. Gene-disease associations are inferred on the basis of similarities between the proteins encoded by genes. Biological relationships used to define similarities range from interacting proteins, proteins that participate in pathways and protein expression profiles. Though graph kernel methods have become a prominent approach for association prediction, most solutions are based on a notion of information diffusion that does not capture the specificity of different network parts. Here we propose a graph kernel method that explicitly models the configuration of each gene's context. An empirical evaluation on several biological databases shows that our proposal is competitive w.r.t. state-of-the-art kernel approaches.

1 Introduction and Related Work

Predictive systems for gene-disease associations are often based on a notion of similarity between genes. A common strategy is to encode relations between genes as a network and use graph based techniques to make useful inferences. In the last decades, a number of graph kernel methods have been proposed that directly exploit transitive properties in biological networks. The prototypical method is the Diffusion kernel (DK) [2] inspired by the heat diffusion phenomenon. The key idea is to allow a given amount of *heat* placed on nodes to *diffuse* through the edges. The similarity between two nodes v_i, v_j is then defined as the amount of heat starting from v_i and reaching v_j within a given time interval. In DK the heat flow is proportional to the number of paths connecting two nodes, which introduces an undesired bias that penalize peripheral nodes w.r.t. central ones. This problem is tackled by a modified version of DK called Markov exponential diffusion kernel (MED) [3] where a Markov matrix replaces the Laplacian matrix. Another kernel called Markov diffusion kernel (MD) [4], exploits instead the notion of *diffusion distances*, a measure of similarity between patterns of heat diffusion. The Regularized Laplacian kernel (RL) [5] represents instead a normalized version of the random walk with restart model and defines the node similarity as the number of paths connecting two nodes with different lengths. All these approaches can be applied to dense networks with high degree nodes. A drawback of these approaches is however their relatively low discriminative capacity. This is in part due to the fact that information is processed in an additive and independent fashion which prevents them from accurately modeling the configuration of each gene's context. To address this issue here we

This work was supported by the University of Padova, Strategic Project BIOINFOGEN.

propose to employ a *decompositional* graph kernel (DGK) [1] technique in which the similarity function between graphs can be formed by decomposing each graph into subgraphs and by devising a valid local kernel between the subgraphs. To exploit its higher discriminative capacity we first decompose the network in a collection of connected sparse graphs and then we develop a suitable kernel, that we call Conjunctive Disjunctive Node Kernel (CDNK).

2 Method

We start from the type of similarity notion computed by decomposition kernels between graph instances and adapt it to express the similarity between nodes in a single network. In this work we use three key ideas: 1) genes are labeled using their functional profile, 2) we transform the network to distinguish highly connected components from sparsely connected ones, and 3) we transform the neighborhood of each gene in a sparse high dimensional vector that can be easily processed by standard machine learning techniques such as SVMs.

2.1 Gene Labeling

Gene-disease associations networks typically represent genes as nodes labeled with a gene identifier. Here we take a different approach: since we want to build a predictive system based on the configuration of each gene's context, we use as labels a discretized functional annotation based on the *gene ontology* [10]. More precisely, we use the gene ontology to construct binary vectors representing the bag-of-words encoding for each gene (i.e. if a GO-term is associated with the gene). The resulting representations are then clustered so that genes with similar description profiles receive the same class identifier as label. Finally, node labels are used to define the graph isomorphism problem.

2.2 Network Decomposition

In gene-disease associations networks it is not uncommon to find nodes with high degrees. Unfortunately these cases cannot be effectively processed by decomposition kernels based on exact neighborhood matches because of the high number of neighborhood subgraphs. As an alternative, we propose to decompose the network in a linked collection of sparse sub-networks where each node has a reduced connectivity. More precisely we distinguish two types of edges: *conjunctive* and *disjunctive* edges. Nodes linked by conjunctive edges are going to be used jointly to define the notion of context. Nodes linked by disjunctive edges are instead used to define features based only on the pairwise co-occurrence of the genes at the endpoints. The aim of the following decompositions is to link sparse sub-networks (which comprise only conjunctive edges) via disjunctive edges.

Definitions. A graph $G = (V, E)$ is a structure that consists of a node set $V(G)$ and an edge set $E(G)$. The *distance* between two nodes u and v, notated as $\mathcal{D}(u, v)$, is the length of the shortest path between them. The *neighborhood* with radius r of a node v is the set of nodes at a distance no greater than r from

v and denoted by $N_r(v)$. The *neighborhood subgraph* with radius r of node v, denoted by $\mathcal{N}_r^v$, is the subgraph formed by the nodes in the neighborhood with radius r of v and the relative edges with endpoints in $N_r(v)$.

K-core decomposition [8]: The node set is partitioned in two groups on the basis of the degree of each node w.r.t. a threshold degree D. The node partition is used to induce the conjunctive vs disjunctive edge partition: edges that have endpoints in the same part are marked as conjunctive, while edges with endpoints in different parts are marked as disjunctive. We apply the k-core decomposition iteratively considering only the graph induced by the conjunctive edges until no node has a degree greater than D.

Clique decomposition [9]: All the cliques (completely connected subgraphs) with a number of nodes greater than a threshold size C are identified. Nodes in a clique seem to share same properties. Therefore, a new 'representative' node is added to the network for each clique. The endpoints of all edges incident on the clique's nodes are transferred to the representative node. Disjunctive edges are introduced to connect each node in the clique to the representative. Finally all edges with both endpoints in the clique are removed.

In our work a network is transformed by applying first the k-core decomposition and then the clique decomposition.

2.3 Node Graph Kernels

We start from the Neighborhood Subgraph Pairwise Distance Kernel (NSPDK) [6] and adapt it to express the similarity between nodes in a single network. The key idea in NSPDK is to decompose graphs in small fragments and count how many pairs of fragments are shared between two instances. We introduce two improvements: 1) we partition the features according to the individual node's neighborhood, and 2) feature construction distinguishes between disjunctive and conjunctive edges.

2.3.1 The Neighborhood Subgraph Pairwise Distance Kernel

The NSPDK is an instance of convolution kernel [1] where given a graph $G \in \mathcal{G}$ and two rooted graphs A_u, B_v, the relation $R_{r,d}(A_u, B_v, G)$ is true iff $A_u \cong \mathcal{N}_r^u$ is (up to isomorphism $\cong$) a neighborhood subgraph of radius r of G and so is $B_v \cong \mathcal{N}_r^v$, with roots at distance $\mathcal{D}(u,v) = d$. We denote R^{-1} as the inverse relation that returns all pairs of neighborhoods of radius r at distance d in G, $R_{r,d}^{-1}(G) = \{A_u, B_v | R_{r,d}(A_u, B_v, G) = true\}$. The kernel $\kappa_{r,d}$ over $\mathcal{G} \times \mathcal{G}$, counts the number of such fragments in common in two input graphs:

$$\kappa_{r,d}(G, G') = \sum_{\substack{A_u, B_v \in R_{r,d}^{-1}(G) \\ A'_{u'}, B'_{v'} \in R_{r,d}^{-1}(G')}} \mathbf{1}_{A_u \cong A'_{u'}} \cdot \mathbf{1}_{B_v \cong B'_{v'}},$$

where $\mathbf{1}_{A \cong B}$ is the *exact matching function* that returns 1 if A is isomorphic to B and 0 otherwise. Finally, the NSPDK is defined as $K(G, G') = \sum_r \sum_d \kappa_{r,d}(G, G')$,

The degree is defined by only considering incident conjunctive edges.

where for efficiency reasons, the values of r and d are upper bounded to a given r^* and d^*, respectively.

2.3.2 The Conjunctive Disjunctive Node Kernel

We extend NSPDK and define a node kernel $K(G_u, G_{u'})$ between two copies of the same network G where we distinguish the nodes u and u' respectively. The idea is to define the features of a node u as the subset of NSPDK features that always have the node u as one of the roots. In addition we distinguish between two types of edges, termed *conjunctive* and *disjunctive* edges. We consider only conjunctive edges when computing distances and hence when we induce neighborhood subgraphs. When choosing the pair of neighborhoods to form a single feature, we additionally consider roots u and v that are not at distance d but such that u is connected to w via a disjunctive edge and such that w is at distance d from v. In this way disjunctive edges can still allow an 'information flow' even if their endpoints are only considered in a pairwise fashion and not jointly.

Formally, we define two relations: the *conjunctive relation* $R^{\wedge}_{r,d}(A_u, B_v, G_u)$ is true iff (i) $A_u \cong \mathcal{N}^u_r$ is a neighborhood subgraph of radius r of G_u and so is $B_v \cong \mathcal{N}^v_r$, and (ii) $\mathcal{D}(u, v) = d$; the *disjunctive relation* $R^{\vee}_{r,d}(A_u, B_v, G_u)$ is true iff (i) $A_u \cong \mathcal{N}^u_r$ and $B_v \cong \mathcal{N}^u_r$ are true, (ii) $\exists w$ s.t. $\mathcal{D}(w, v) = d$, and (iii) (u, w) is a disjunctive edge. We define $\kappa_{r,d}$ on the inverse relations ${R^{\wedge}_{r,d}}^{-1}$ and ${R^{\vee}_{r,d}}^{-1}$

$$\kappa_{r,d}(G_u, G_{u'}) = \sum_{\substack{A_u, B_v \in {R^{\wedge}_{r,d}}^{-1}(G_u) \\ A'_{u'}, {B'}_{v'} \in {R^{\wedge}_{r,d}}^{-1}(G_{u'})}} \mathbf{1}_{A_u \cong A'_{u'}} \cdot \mathbf{1}_{B_v \cong {B'}_{v'}} + \sum_{\substack{A_u, B_v \in {R^{\vee}_{r,d}}^{-1}(G_u) \\ A'_{u'}, {B'}_{v'} \in {R^{\vee}_{r,d}}^{-1}(G_{u'})}} \mathbf{1}_{A_u \cong A'_{u'}} \cdot \mathbf{1}_{B_v \cong {B'}_{v'}}.$$

The CDNK is finally defined as $K(G_u, G_v) = \sum_r \sum_d \kappa_{r,d}(G_u, G_v)$, where once again for efficiency reasons, the values of r and d are upper bounded to a given r^* and d^*.

3 Evaluation

We perform an empirical evaluation of the predictive performance of several kernel based methods on two of the databases used in [3].

BioGPS: A gene co-expression network is constructed from BioGPS dataset, which contains 79 tissues, measured with the Affymetrix U133A array. Edges are inserted when the pairwise Pearson correlation coefficient (PCC) between genes is larger than 0.5.

Pathways: Pathway information is retrieved from KEGG, Reactome, PharmGKB and the Pathway Interaction Database. If a couple of proteins co-participate in any pathway, the two corresponding genes are linked.

To evaluate the performance of graph node kernels we analyze the *gene prioritization*, i.e. given a set of genes known to be associated to a given disease, gene prioritization is the task to rank the candidate genes based on their probabilities to be related to that disease. Similar to the evaluation process used in

This is identical to the NSPDK relation $R_{r,d}(A_u, B_v, G)$.

[3], we choose 12 diseases with at least 30 confirmed genes. For each disease, we construct a positive set $\mathcal{P}$ with all confirmed disease genes, and a negative set $\mathcal{N}$ which contains random genes associated at least to one disease class which is not related to the class that is defining the positive set. In [3] the ratio between the dataset sizes is chosen as $|\mathcal{N}| = \frac{1}{2}|\mathcal{P}|$. The predictive performance of each method is evaluated via a leave-one-out cross validation: one gene is kept out in turn and the rest are used to train an SVM model. We compute a decision score q_i for the test gene g_i as the top percentage value of score s_i among all candidate gene scores. We collect all decision scores for every gene in the training set to form a global decision score list on which we compute the AUC ROC.

Model Selection: The hyper parameters of the various methods are set using a k-fold on a dataset set that is then never used in the predictive performance estimate. We try the values for diffusion parameter in DK and MED in $\{10^{-3}, 10^{-3}, 10^{-2}, 10^{-1}\}$, time steps in MD in $\{1, 10, 100\}$ and RL parameter in $\{1, 4, 7\}$. For CDNK, we try for the degree threshold values in $\{10,\ 15,\ 20\}$, clique size threshold in $\{4,\ 5\}$, maximum radius in $\{1,\ 2\}$, maximum distance in $\{2,\ 3,\ 4\}$. Finally, the regularization trade off parameter C for the SVM is searched in $\{10^{-5},\ 10^{-4},\ 10^{-3},\ 10^{-2},\ 10^{-1}, 1,\ 10,\ 10^2\}$.

4 Results and Discussion

Table 1 shows the AUC performance of the models trained by using different graph node kernels on 11 gene-disease association problems using the BioGPS and Pathways datasets to materialize the gene relation network. In the table, the best result on each disease is marked in bold. We note that CDNK ranks first in 7 out of 11 cases using both networks. CDNK is the top performant kernel also when considering the average AUC ROC and the average rank with 73.3/2.0, 76.5/1.8, with a difference w.r.t. the second best of 5.5% and 1% on BioGPS and Pathways, respectively. MED and RL show similar and moderate results with small gap between them. DK and MD exhibit modest performance on average and are ranked last in many cases: 7 times out of 11 for MD in BioGPS and 10 out of 11 for DK in Pathways. Finally note that although CDNK has state of the art performance, there are cases that would lead to a significant decrease in the quality of the results, namely when networks have high *average* node degree as this would lead to very sparse and fragmented decompositions.

5 Conclusions

We have shown how decomposing a network in a set of connected sparse graphs allows us to take advantage of the discriminative power of CDNK, a novel decomposition kernel, to achieve state-of-the-art results. In future work we will investigate how to 1) decompose networks in a data driven way and 2) extend the CDNK approach to gene-disease association problems exploiting multiple heterogeneous information sources.

	BioGPS					**Pathways**				
Disease	K1	K2	K3	K4	K5	K1	K2	K3	K4	K5
1	52/5	57/4	59/3	59/2	**65/1**	75/5	76/4	79/3	79/2	**80/1**
2	82/2	79/3	75/4	75/5	**88/1**	55/5	65/4	77/3	77/2	**81/1**
3	64/4	60/5	72/2	**72/1**	66/3	55/5	63/4	64/3	66/2	**67/1**
4	65/4	58/5	68/3	68/2	**72/1**	54/5	65/4	**74/1**	74/2	66/3
5	64/5	64/4	67/2	66/3	**76/1**	53/5	56/4	63/2	63/3	**68/1**
6	75/2	70/5	71/4	71/3	**79/1**	83/5	93/4	97/2	**97/1**	93/3
7	73/3	67/5	75/2	**76/1**	69/4	85/5	88/4	89/2	**90/1**	89/3
8	74/5	**77/1**	76/3	76/2	75/4	54/5	66/4	72/3	72/2	**73/1**
9	**72/1**	66/5	68/3	70/2	67/4	53/5	65/2	64/4	64/3	**81/1**
10	54/3	50/5	56/2	51/4	**78/1**	69/3	65/5	74/2	**74/1**	67/4
11	58/4	51/5	59/3	59/2	**72/1**	54/5	69/4	75/2	74/3	**77/1**
$\overline{AUC}$	66.6	63.5	67.8	67.5	**73.3**	62.7	70.1	75.3	75.5	**76.5**
$\overline{Rank}$	3.5	4.3	2.8	2.5	**2.0**	4.8	3.9	2.5	2.0	**1.8**

Table 1: *Predictive performance on 11 gene-disease associations using networks induced by the BioGPS and the Pathway database. We report the AUC ROC and the rank for each kernel method: K1 = DK, K2 = MD, K3 = MED, K4 = RL, K5 = CDNK.*

References

[1] Haussler, David. Convolution kernels on discrete structures. Vol. 646. Technical report, Department of Computer Science, University of California at Santa Cruz, 1999.

[2] Kondor, Risi Imre, and John Lafferty, Diffusion kernels on graphs and other discrete input spaces. *ICML*. Vol. 2. 2002.

[3] Chen. B, et al. Disease gene identification by using graph kernels and Markov random fields. Science China Life Sciences 57.11 (2014): 1054-1063.

[4] Fouss F, et al. An experimental investigation of graph kernels on a collaborative recommendation task. Proceedings of the 6th international conference on data mining 2006. ICDM 2006, 863-868.

[5] Chebotarev P and Shamis E. The matrix forest therem and measuring relations in small social groups. Automation and Remote Control 1997, 58(9):1505-1514.

[6] C. Fabrizio, and K. De Grave, Fast neighborhood subgraph pairwise distance kernel. Proceedings of the 26th, International Conference on Machine Learning. Omnipress, 2010.

[7] Goh KI et al, The human disease network. Proceedings of the National Academy of Sciences 104.21 (2007): 8685-8690.

[8] A. Hamelin, et al, K-core decomposition: A tool for the visualization of large scale networks. arXiv preprint cs/0504107 (2005).

[9] R. E. Tarjan, Decomposition by clique separators, Discrete Math, vol. 55, no. 2, pp. 221-232, July. 1985.

[10] Gene Ontology Consortium. The Gene Ontology (GO) database and informatics resource. Nucleic acids research,(2004) 32(suppl 1), D258-D261.