Graph and Manifold Co-Regularization

Claudio Saccà, Michelangelo Diligenti, Marco Gori
Dipartimento di Ingegneria dell'Informazione
University of Siena, Via Roma 54, Siena, Italy
Email: {claudiosacca, diligmic, marco}@dii.unisi.it

***Abstract*—Classical foundations of Statistical Learning Theory rely on the assumption that the input patterns are independently and identically distributed. However, in many applications, the inputs, represented as feature vectors, are also embedded into a network of pairwise relations. Transductive approaches like graph regularization rely on the network topology without considering the feature vectors. Semi-supervised approaches like Manifold Regularization learn a function taking the feature vectors as input, while being smooth over the network connections. In this latter case, the connectivity information is processed at training time, but is still neglected during generalization, as the final classification decision takes only the feature vector representations as input. This paper presents and evaluates a model merging the advantages of graph regularization and kernel machines for transductive classification problems.**

I. Introduction

Learning in a transductive environment assumes that the test data is available at training time [1]. Semi-supervised approaches [2] are able to employ both unsupervised and supervised data in training, but they do not assume that the unsupervised portion is the entire data population. Therefore, semi-supervised methods typically learn a classification function that can be later used to generalize over any new input pattern.

Manifold Regularization (MR) [3] is a semi-supervised classification approach, which assumes that the data points are connected by links indicating that they share some similarity according to some metric. The MR assumption states that solutions should fit the examples, while being smooth over the connections. MR works independently on how the connections are built and, in this paper, we are interested in the cases where the manifold structure is not extracted from the input feature space, but the manifold expresses some relational knowledge about the learning problem like HTML links in a Web page classification task. *Graph regularization* (GR) [4] refers to a class of purely transductive approaches, processing the network formed by the pattern relationships. GR is a special case of MR, where the ambient norm is removed and a quadratic loss is used for the fitting of the supervised data.

When working in a transductive environment, it can be erroneously assumed that MR always improves GR, as it can consider both the topology and the feature representations at the same time. However, this is not always true as MR learns a function over the input space, meaning that even if it considers the topology to train the function weights, only the pattern representations in the input space will be used to classify them. Classification accuracy can significantly drop when the input space is not rich enough to allow MR to encode any information gathered from the topology in its weights via the representations of the patterns. On the other hand, when the feature representation is rich, MR seems to be in a better position than GR to learn the classification function, as it can use information that GR can not directly process. Similarly to what done in the context of manifold co-regularization [5] we assume to make predictions dependent on each other by enforcing consistency among them. Running graph and MR in parallel and adding a consistency constraint over each pair of values computed by the two methods, will be proved to be a simple but powerful way to improve the generalization capabilities of the overall model. In particular, the GR level provides the final output of the classification, with the consistency constraints making GR depend on the underlying feature vectors through the MR layer. Unlike pure manifold co-regularization, this allows to generalize well also in the cases where the feature representations are weak. The experimental results show that the proposed method improves over state-of-the-art transductive and semi-supervised learning approaches in several binary and multi-label classification tasks.

II. Manifold Regularization

Given a sample of patterns from an unknown distribution in some metric space, MR creates the manifold by connecting each pattern with its neighbors according to some selected metric. The manifold can be represented by a graph $\mathcal{G}$, where each node is associated to a pattern, and connected patterns in the manifold are represented by links between the corresponding nodes. Let $|\mathcal{G}|$ indicate the number of nodes in $\mathcal{G}$. Each link is also assigned a weight, which is inversely proportional to the distance between the patterns associated to the linked nodes. We assume that the solution can be expressed as a finite kernel expansion: $f(\boldsymbol{x}) = \sum_{i=0}^{|\mathcal{G}|} w_i K(\boldsymbol{x}, \boldsymbol{x}_i)$ where w_i is a weight associated to the i-th element of $\mathcal{G}$ and K is the kernel associated to the considered RKHS and $\boldsymbol{x}_i$ is the representation of the pattern associated to the i-th node of $\mathcal{G}$

Let us indicate as $\boldsymbol{f}$ the vector of values of the function over the nodes in the manifold $\boldsymbol{f} = [f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_{|\mathcal{G}|})]^T$. Since the function is a linear combination of kernel evaluated over the data points: $\boldsymbol{f} = \boldsymbol{K}\boldsymbol{w}$, where $\boldsymbol{K}, \boldsymbol{w}$ are the gram matrix with its (i,j)-th element equal to $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and the vector of weights. The cost function can be expressed as:

$$\underset{\boldsymbol{w}}{\operatorname{argmin}}\ \boldsymbol{w}^T \boldsymbol{K}\boldsymbol{w} + \lambda_l L(\boldsymbol{S}_l \boldsymbol{K}\boldsymbol{w}, \boldsymbol{y}) + \frac{\lambda_c}{2|\mathcal{G}|^2} \boldsymbol{w}^T \boldsymbol{K}\boldsymbol{L}\boldsymbol{K}\boldsymbol{w}$$

where λ_c and λ_l are parameters weighting the contribution of the GR and the labeled parts, $\boldsymbol{S}_l$ is a diagonal matrix having its (i,i)-th element equal to 1 if $\boldsymbol{x}_i$ is labeled (this matrix selects only the elements corresponding to supervised patterns from a vector of values), L is a loss function, $\boldsymbol{y}$ is the vector of size $|\mathcal{G}|$ having in the u-th position the target classification

score of node u if it is labeled and 0 otherwise, L is a loss function, penalizing the distance of the $f(\boldsymbol{x}_k)$ from the desired output y_k and $\boldsymbol{L}$ is the graph Laplacian. In particular, it holds that $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$ with $\boldsymbol{W}$ the adjacency matrix of graph $\mathcal{G}$ ((i,j)-th element equal to w^g_{ij}) and the diagonal matrix $\boldsymbol{D}$ given by $D_{ii} = \sum_{i \in \mathcal{G}} w^g_{ij}$. MR was originally proposed to establish the links between pairs of patterns based on their proximity in the input space. However, the methodology is by no means limited to this, as the formulation does not make any assumption on how the graph is built. In particular, we are interested to employ the MR schema where the graph encodes relational information that is external to the feature space.

III. Graph Regularization

Let $\mathcal{G}$ be a graph, where each node corresponds to a pattern and whose edges are assigned a weight $w^g_{uv} \geq 0$ representing the strength of the connection between the u-th and v-th patterns. Like for MR, $\mathcal{G}$ can be represented via its adjacency matrix $\boldsymbol{W}$, whose $(u,v) - th$ element is equal to w^g_{uv}. GR computes a vector of values $\boldsymbol{f'}$, where its v-th element $f'(v)$ assigns a score to the v-th node of $\mathcal{G}$. $\boldsymbol{f'}$ attempts at providing a good fitting of the target vector on the labeled nodes, while generalizing to unlabeled nodes by enforcing "smooth" variations over the graph connections. In particular, the learning problem determines the vector $\boldsymbol{f}^\star$ minimizing the cost functional, $\frac{1}{2}\left\|\boldsymbol{f'} - \boldsymbol{y}\right\|^2 + \frac{\lambda}{2}\boldsymbol{f'}^T \boldsymbol{R}_\mathcal{G} \boldsymbol{f'}$ where $\boldsymbol{R}_\mathcal{G}$ is a regularization matrix defined to penalize non-smooth solutions, $\boldsymbol{y}$ is the vector of target scores, whose non-supervised entries are set to 0, and $0 \leq \lambda \leq 1$ is a constant determining the trade off between regularization and error over the training nodes. The (unique) optimal solution $\boldsymbol{f}^\star$ can be computed by finding the stationary points of the cost function.

Now, let's define the cumulative weighted distance between the values computed for pairs of connected nodes as

$$C_\mathcal{G}[\boldsymbol{f'}] = \frac{1}{2}\sum_{u=1}^{|\mathcal{G}|}\sum_{v=1}^{|\mathcal{G}|} w^g_{uv}(f'(u) - f'(v))^2$$

Rearranging the terms,

$$C_\mathcal{G}[\boldsymbol{f'}] = \frac{1}{2}\sum_{u=1}^{|\mathcal{G}|}\sum_{v=1}^{|\mathcal{G}|} \frac{(w^g_{uv} + w^g_{vu})}{2}(f'(u) - f'(v))^2 \ .$$

Let us we define the symmetric weights $\bar{w}^g_{uv} = \frac{(w^g_{uv} + w^g_{vu})}{2}$, and let $\bar{\boldsymbol{W}}$ be a symmetric square matrix having $\bar{w}^g_{uv}$ as (u,v)-th element and let $\boldsymbol{D}$ be a diagonal matrix with its u-th element d_u equal to $\sum_{v=1}^{|\mathcal{G}|} \bar{w}^g_{uv}$. Therefore, $\mathrm{C}_\mathcal{G}[\boldsymbol{f'}] = \boldsymbol{f'}^T(\boldsymbol{D} - \bar{\boldsymbol{W}})\boldsymbol{f'}$ Setting $\boldsymbol{R}_\mathcal{G} = \boldsymbol{D} - \bar{\boldsymbol{W}}$ into the cost function and computing the stationary point, computes the optimal score vector as $\boldsymbol{f}^\star = \left(\boldsymbol{I} + \lambda\boldsymbol{D} - \lambda\bar{\boldsymbol{W}}\right)^{-1}\boldsymbol{y}$. Since $\boldsymbol{I} + \lambda\boldsymbol{D} - \lambda\bar{\boldsymbol{W}}$ is diagonally dominant and, therefore, invertible for $\lambda > 0$. Thus, the optimal solution $\boldsymbol{f}^\star$ exists and is uniquely defined by the graph and the supervised vector of target scores.

IV. Learning from Relations and Features

Let us assume to have input patterns in the form of feature vectors and some relational external knowledge among them, represented via a graph $\mathcal{G}$, whose nodes are the input patterns. GR and MR (used in a transductive context) both assign predictions over the nodes of the graph. Let $f'(i)$ be the score assigned by GR to the i-th node of $\mathcal{G}$ and $f(\boldsymbol{x}_i)$ be the score estimated via MR on the vectorial representation $\boldsymbol{x}_i$ of the pattern associated to the i-th node of the graph. The relationships between connected nodes on the graph are represented by causal relationships among the corresponding predictions at the MR and GR levels, respectively. The predictions of the two levels can be made dependent on each other by enforcing consistency among the corresponding output values. The GR level will provide the output of this merged model. Figure 1 shows the resulting causal relations among the variables, where causal relations coming from the input graph and from consistency enforcement levels are depicted using solid and dashed lines, respectively.

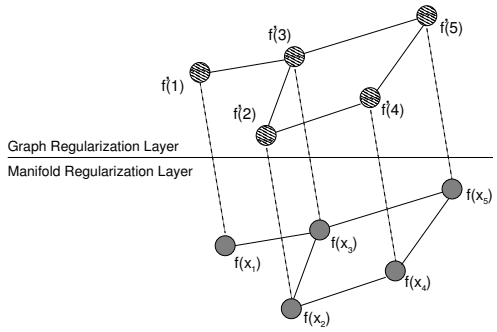

Fig. 1. Graph representing the co-regularization merged model. Inter-level connections coming from the input graph are depicted using solid lines, whereas dashed lines represent the causal relations introduced by the consistency enforcement between layers.

This schema is implemented by the following cost function implementing a graph-manifold co-regularization framework:

$$\begin{aligned} C(\boldsymbol{w}, \boldsymbol{f'}) &= \boldsymbol{w}^T\boldsymbol{K}\boldsymbol{w} + \lambda_l L_l(\boldsymbol{S}_l\boldsymbol{K}\boldsymbol{w}, \boldsymbol{y}) + \\ &+ \tfrac{\lambda_c}{2|\mathcal{G}|^2}\boldsymbol{w}^T\boldsymbol{K}\boldsymbol{L}\boldsymbol{K}\boldsymbol{w} + \\ &+ \tfrac{\lambda_m}{2|\mathcal{G}|^2}\boldsymbol{f'}^T\boldsymbol{L}\boldsymbol{f'} + \lambda_d L_d\big(\boldsymbol{f'}, (\boldsymbol{S}_l\boldsymbol{K}\boldsymbol{w} + \boldsymbol{y})\big) \end{aligned}$$

where the overall vector of function values is expressed by the kernel expansion $\boldsymbol{f} = \boldsymbol{K}\boldsymbol{w}$ and $\boldsymbol{f'}$ is a vector arranging the values for the variables on the GR layer. L_l, L_d are two loss functions measuring the fitting of the supervised data and the consistency of the MR and GR values over all elements of the vectors, respectively. λ_l is a meta-parameter weighting the contribution coming from fitting the supervised data, λ_c weights the smoothness of the output data over the GR graph, λ_d penalizes the non-consistency across the layers and, finally, λ_m enforces smoothness over the manifold.

When no feature representations are available for the patterns at the lower layer, the kernel machine can not learn any prediction. In this case all elements of $\boldsymbol{w}$ will be equal to zero because this minimizes the function regularizer. Therefore, it holds $\boldsymbol{w}^T\boldsymbol{K}\boldsymbol{w} = 0$, $L_l(\boldsymbol{S}_l\boldsymbol{K}\boldsymbol{w}, \boldsymbol{y}) = \sum_{x_k \in \mathcal{L}} L_l(0, y_i)$ is a constant, $L_d\big(\boldsymbol{f'}, \boldsymbol{S}_l\boldsymbol{K}\boldsymbol{w} + \boldsymbol{y}\big) = L_d\big(\boldsymbol{f'}, \boldsymbol{y}\big)$ and $\boldsymbol{w}^T\boldsymbol{K}\boldsymbol{L}\boldsymbol{K}\boldsymbol{w} = 0$. It is straightforward to see that this model collapses to GR if L_d is the quadratic loss $L_d(x,y) = |x-y|^2$.

On the other hand, when the kernel machine is able to perfectly encode the topology in its weights, we have reached

a solution such that $\boldsymbol{w}^T\boldsymbol{K}\boldsymbol{L}\boldsymbol{K}\boldsymbol{w} = 0$. It is straightforward to see that in this case the cost function will be minimized by selecting $\boldsymbol{f}'$ such that $\boldsymbol{f}' = \boldsymbol{K}\boldsymbol{w}$, as this will yield a null contribution from both terms depending on $\boldsymbol{f}'$ in the cost function: $L_d\big(\boldsymbol{f}', \boldsymbol{K}\boldsymbol{w}\big) = \boldsymbol{f}'^T\boldsymbol{L}\boldsymbol{f}' = 0$. In a more general (and interesting) case, some feature representation of the patterns is available but it is not necessarily rich enough to allow a perfect encoding of the topology. In this case, the proposed model will make the two underlying models play together to find a better overall solution.

In the rest of the paper we will assume that the manifold has been built from external relational knowledge because the proposed methodology is particularly effective for the integration of external knowledge, which is often difficult to encode by MR in its learned weights.

Assuming L_d and L_l to be the quadratic loss, the gradient of the cost function with respect to the model parameters $\boldsymbol{w}, \boldsymbol{f}'$ can be expressed as,

$$\begin{aligned}\frac{\partial C(\boldsymbol{w}, \boldsymbol{f}')}{\partial \boldsymbol{w}} &= \boldsymbol{K}\boldsymbol{w} + \lambda_l \boldsymbol{K}\boldsymbol{S}_l(\boldsymbol{S}_l\boldsymbol{K}\boldsymbol{w} - \boldsymbol{y}) + \qquad (1)\\ &+ \frac{\lambda_c}{|\mathcal{G}|^2}\boldsymbol{K}\boldsymbol{L}\boldsymbol{K}\boldsymbol{w} - \lambda_d \boldsymbol{K}\boldsymbol{S}_l(\boldsymbol{f}' - \boldsymbol{S}_l\boldsymbol{K}\boldsymbol{w} - \boldsymbol{y})\\ \frac{\partial C(\boldsymbol{w}, \boldsymbol{f}')}{\partial \boldsymbol{f}'} &= \lambda_d\big(\boldsymbol{f}' - \boldsymbol{S}_l\boldsymbol{K}\boldsymbol{w} - \boldsymbol{y}\big) + \frac{\lambda_m}{|\mathcal{G}|^2}\boldsymbol{L}\boldsymbol{f}' \qquad (2)\end{aligned}$$

The cost function is a quadratic form, whose unique minimum is found where the gradient vanishes.

The solution could be found with any linear system optimization algorithm but this would be impractical when $\boldsymbol{L}$ and $\boldsymbol{K}$ are large. Therefore, we instead directly optimize $C(\boldsymbol{w}, \boldsymbol{f}')$ by alternatively performing a gradient descent step for $\boldsymbol{w}$ and $\boldsymbol{f}'$ using the gradients expressed in eq. 1 and 2. In particular, resilient gradient descent was empirically found to converge very quickly to the solution and was therefore used in all the presented experiments.

The merged model has four hyper-parameters and optimizing these hyper-parameters via exhaustive search is not feasible for large datasets. However, it is possible to rely on the parameter semantic to guide the optimization. In particular, the best λ_l, λ_c can be selected via crossvalidation while keeping $\lambda_d = \lambda_m = 0$. This is equivalent to searching the best hyper-parameters for MR alone. Then, λ_d and λ_m are selected via crossvalidation while keeping λ_l, λ_c fixed. While not optimal, this methodology has been very effective in our experiments, while never searching over more than two parameters at the same time (like MR).

V. Experimental Results

AAN dataset. This dataset[1] contains 380 scientific publications manually classified into three research areas (*"Machine Translation"*, *"Dependency Parsing"* and *"Summarization"*). The AAN dataset is a relational dataset as the papers are linked via citations. Each paper is associated to a title represented as bag-of-words. We have compared four classification methods on detecting whether each paper belongs to one class of the dataset. In particular, a standard SVM implemented with linear kernel trained in the primal as described in [6], an Laplacian SVM implementing MR using the citations to build the manifold (implemented via svmlight [7]), GR run over the citation network and the proposed model coupling manifold and graph regularization. Since MR performances depend on being able to encode the relationships in their weights via the input features, the title terms have been randomly dropped in each pattern representation with a variable probability p. When $p = 0$, all the initial information is preserved, whereas $p = 1$ means that the feature representations become empty. In this latter case, the merged model performs like GR, whereas SVM and MR can not compute any output value. In this case, the classifier simply returns the most probable class according to prior probabilities computed over the training set. All the experiments have been averaged over 10 runs, where at each run 50%, 20%, 30% of patterns have been randomly selected for inclusion in the train, validation and test sets, respectively. Being a trasductive context, patterns selected for inclusion in the validation or test sets have not been dropped from the training set but they have been just provided as unlabeled data. Figure 2 presents the macro accuracy results: MR outperforms SVMs, GR results do not depend on the number of dropped features and they are outperformed by MR when no features are dropped. However, when features are removed, MR performances show a quick degradation, being soon outperformed by GR. The merged model is performing like GR when all features are dropped but it outperforms all other methods in most of the other cases with statistically significant (95% t-test) gains for most configurations.

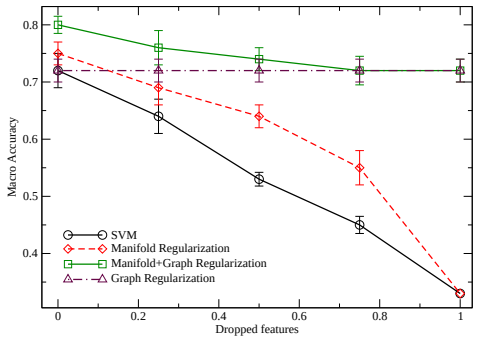

Fig. 2. Results on the AAN dataset for a varying number of dropped features.

CORA dataset. The CORA research paper dataset [8] collects papers classified into a topic hierarchy[2]. CORA is a relational data set, because the citations provide relations among papers. In our experimental settings, we have considered only the first level of the hierarchy, which contains 10 classes. The feature vector associated to each paper is formed by the paper title represented as bag-of-words . The CORA dataset contains 37000 unique papers. However, some of these papers have no title, they have been therefore discarded to allow a fair comparison against methods based on processing the feature vectors. The total number of considered papers has therefore been reduced to 19397, from which we randomly

[1] http://clair.si.umich.edu/homepage/downloads/aanrelational/

[2] http://people.cs.umass.edu/~mccallum/data.html

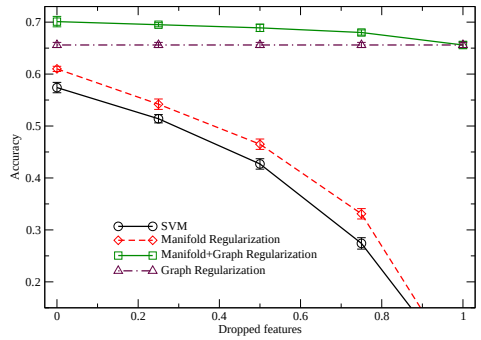

Fig. 3. Results on CORA for a varying number of dropped features.

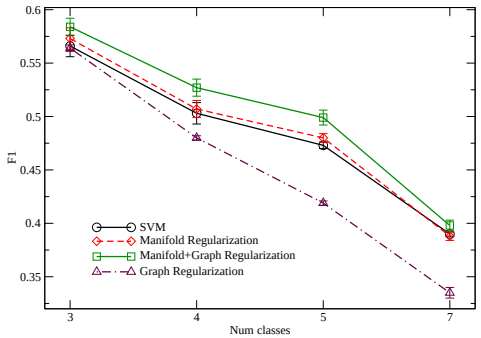

Fig. 4. Results on the Artminer dataset varying number of considered classes.

sampled 10000 documents. Since the dataset is used as a transductive relational dataset, 10 folds have been created by randomly sampling at each round the 20% of the papers for which supervisions are kept. The remaining 30% and 50% labelled documents have been included in the validation and test set, respectively. All reported results have been obtained as an average over ten different samples of the data. Figure 3 shows that MR consistently outperforms SVMs classifiers, but both methods' performances strongly decline when many features are removed. GR outperforms MR in all conditions in this task, as citations carry more relevant information than title words and manifold Regularization can not effectively encode the citations in the weights. The merged model is performing like GR when all features are dropped but it outperforms all other methods in most of the other cases with statistically significant (95% t-test) gains.

Arnetminer dataset. The goal of this experiment is to predict the movie genre by looking at the movie title, director and producers' names. Any pair of movies with the same director, writer and/or producer are also linked in the network. The Arnetminer Movie dataset[3], which contains information about 18000 movies and associated directors, writers and actors [9]. A set of tags is assigned to each movie and we selected the movies with at least one tag containing at least one of the following keywords: *horror*, *drama*, *comedy*, *television*, *teen*, *musical*, *adventure*. Each of this tag corresponds to an underlying genre that we wish to predict. Any movie that is not associated to any tag containing these keywords has been discarded. The resulting dataset contains 10894 movies. Please note that this is a multi-label dataset, since movies can be associated to multiple associated genres. The classification experiments have been repeated over 10 different random splits of 50%, 25%, 25% of the patterns for the training, validation and test sets, respectively. At each round, a binary classifier f_c for each class c is trained. For the multi-label dataset the results are measured in terms of micro averaged F1. The reported results have been obtained as the average over the 10 samples, where the meta-parameters have been selected via cross-validation on the validation set at each iteration. Figure 4 shows the results when the classification is performed over all available classes or when further reducing the dataset to the top most common N genres. In particular, the results show statistically significant (95% t-test) gains for the merged proposed model for most configurations.

[3]http://arnetminer.org/lab-datasets/soinf

VI. Conclusions

This paper presented a co-regularization framework to integrate relational knowledge into kernel machines. This model collapses into graph regularization when working in a pure discrete domain and to manifold regularization when dealing with rich feature pattern representations. In all the cases in between, the model has been experimentally proved to improve over the single underlying models.

Acknowledgments. This research was partially supported by the research grant PRIN2009, "Learning Techniques in Relational Domains and Their Applications" (2009LNP494) from the Italian MURST.

References

[1] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proceedings of International Conference on Machine Learning (ICML)*. Morgan Kaufmann, 1999, pp. 200–209.

[2] O. Chapelle, B. Schölkopf, A. Zien *et al.*, *Semi-supervised learning*. MIT press Cambridge, MA:, 2006, vol. 2.

[3] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *The Journal of Machine Learning Research*, vol. 7, p. 2434, 2006.

[4] D. Zhou, J. Huang, and B. Schölkopf, "Learning from labeled and unlabeled data on a directed graph," in *Proceedings of the 22nd international conference on Machine learning*. ACM, 2005, pp. 1036–1043.

[5] V. Sindhwani and D. S. Rosenberg, "An rkhs for multi-view learning and manifold co-regularization," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 976–983.

[6] O. Chapelle, "Training a support vector machine in the primal," *Neural Computation*, vol. 19, no. 5, pp. 1155–1178, 2007.

[7] T. Joachims, *Making large scale SVM learning practical. Advances in Kernel Methods - Support Vector Learning*. MIT press, 1999.

[8] A. McCallum, K. Nigam, J. Rennie, and K. Seymore, "Automating the construction of internet portals with machine learning," *Information Retrieval Journal*, vol. 3, pp. 127–163, 2000, www.research.whizbang.com/data.

[9] J. Tang, J. Sun, C. Wang, and Z. Yang, "Social influence analysis in large-scale networks," in *Proceedings of the Fifteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'2009)*, 2009, pp. 807–816.