Multi-View Budgeted Learning under Label and Feature Constraints Using Label-Guided Graph-Based Regularization

Christopher T. Symons SYMONSCT@ORNL.GOV
Oak Ridge National Laboratory, 1 Bethel Valley Rd., Oak Ridge, TN 37831 USA

Itamar Arel ITAMAR@IEEE.ORG
Department of Electrical and Computer Engineering, The University of Tennessee, 1508 Middle Dr., Knoxville, TN 37996 USA

Abstract

Budgeted learning under constraints on both the amount of labeled information and the availability of features at test time pertains to a large number of real world problems. Ideas from multi-view learning, semi-supervised learning, and even active learning have applicability, but a common framework whose assumptions fit these problem spaces is non-trivial to construct. We leverage ideas from these fields based on graph regularizers to construct a robust framework for learning from labeled and unlabeled samples in multiple views that are non-independent and include features that are inaccessible at the time the model would need to be applied. We describe examples of applications that fit this scenario, and we provide experimental results to demonstrate the effectiveness of knowledge carryover from training-only views.

1. Introduction

As learning algorithms are applied to more complex applications, relevant information can be found in a wider variety of forms, and the relationships between these information sources are often quite complex. The assumptions that underlie most learning algorithms do not readily or realistically permit the incorporation of many of the data sources that are available, despite an implicit understanding that useful information exists in these sources. When multiple information sources are available, they are often partially redundant, highly interdependent, and contain noise as well as other information that is irrelevant to the problem under study. In this paper, we are focused on a framework whose assumptions match this reality, as well as the reality that labeled information is usually sparse. Most significantly, we are interested in a framework that can also leverage information in scenarios where many features that would be useful for learning a model are not available when the resulting model will be applied.

As with constraints on labels, there are many practical limitations on the acquisition of potentially useful features. A key difference in the case of feature acquisition is that the same constraints often don't pertain to the training samples. This difference provides an opportunity to allow features that are impractical in an applied setting to nevertheless add value during the model-building process. Unfortunately, there are few machine learning frameworks built on assumptions that allow effective utilization of features that are only available at training time. In this paper we formulate a knowledge carryover framework for the budgeted learning scenario with constraints on features and labels. The approach is based on multi-view and semi-supervised learning methods that use graph-encoded regularization. Our main contributions are the following: (1) we propose and provide justification for a methodology for ensuring that changes in the graph regularizer using alternate views are performed in a manner that is target-concept specific, allowing value to be obtained from noisy views; and (2) we demonstrate how this general set-up can be used to effectively improve models by leveraging features unavailable at test time.

The rest of the paper is structured as follows. In Section 2, we outline real-world problems to motivate the approach and describe relevant prior work. Section

Appearing in *the 28th International Conference on Machine Learning, Workshop on Combining Learning Strategies to Reduce Label Cost*, Bellevue, WA, USA, 2011. Copyright 2011 by the author(s)/owner(s).

3 describes the graph construction process and the learning methodologies that are employed. Section 4 provides preliminary discussion regarding theoretical motivation for the method. In Section 5, effectiveness of the approach is demonstrated in a series of experiments employing modified versions of two well-known semi-supervised learning algorithms. Section 6 concludes the paper.

2. Background and Motivation

Constraints imposed on feature acquisition come from a variety of sources and can be found in many real-world applications. Often these problems fit naturally into a multi-view setting, in which feature sets can be reasonably partitioned into disjoint cohesive sets that are somewhat redundant. Consider satellite image analysis where ground sensors are available in areas that training examples are pulled from, but not available where the model would be applied. Applying traditional approaches results in a standard satellite image analysis problem that ignores the availability of ground-level data. Another example is a medical study where a battery of tests is performed, and results from all of these procedures are available for subjects of the study (the training set). An applied diagnostic that is dependent upon all of these tests would be prohibitively expensive. Therefore, the standard approach is to independently consider the separate tests to find one procedure that seems to be the best diagnostic, while inter-related, useful information from the rest of the study typically remains unused. We formulate a framework that allows such training-only observations to effectively improve a model that operates on a feature set that does not include these observations.

We view this as a budgeted, multi-view learning problem. Since most applications that fit this scenario will also have few labeled and many unlabeled examples, we also treat the problem as semi-supervised. We assume that we have l labeled examples, $\{(x_i, y_i)\}_{i=1}^{l}$ and u unlabeled examples, $\{x_i\}_{i=l+1}^{l+u}$. In addition, we have j distinct views of each example, $x_i = (x_i^1, x_i^2, ..., x_i^j)$. For simplicity, we will assume $y \in \{+1, -1\}$, with multi-class classifiers being constructed of multiple one-vs-all binary classifiers. To facilitate discussion, we will refer to a feature set available at both training and test time as a *primary view*, and we will refer to a feature set available in training only as a *secondary view*. For example, satellite image features could be a primary view used for classification, and corresponding ground-sensor features could be a secondary view not available when the model would be applied but that encapsulates useful information about some or all of the training examples.

Related work on semi-supervised, multi-view regularization (Sindhwani et al., 2005; Sindhwani & Rosenberg, 2008) employs two regularizers, but the regularization using the unlabeled examples is purely unsupervised. The result is that even though a tradeoff can be made and the unlabeled regularization can be de-emphasized, in cases where a feature set is particularly noisy with regard to the target concept, it is difficult to obtain benefit from such a view, particularly if the labeled set is small. In contrast, we demonstrate how changes in the construction of the regularizer that are informed by the labeled information provide benefit consistently. In particular, it becomes safer to include such information, since it is unlikely to make the regularization less effective than single-view, semi-supervised regularization.

3. Methodology

The framework we employ is one based on semi-supervised learning using graph-based regularization across separate views.

3.1. The Graph Laplacian

The graph Laplacian from spectral graph theory (Chung, 1997) is an important device in many semi-supervised algorithms. In certain manifold-based learning methods (Lin & Zha, 2008), the assumption is that the data lie on a low-dimensional manifold that cuts through the ambient space, and a graph is used to represent that manifold. The expectation, which is not always correct, is that the graph varies smoothly with respect to the target problem (i.e. examples from different classes are rarely linked, similar examples from the perspective of the target problem are linked, etc.). Then, the Laplacian of the graph is used to find a space that roughly represents that manifold. Since it works well in practice, in our experiments, we use the unnormalized form of the graph Laplacian, which is defined as follows:

$$L(u,v) = \begin{cases} d_v, & \text{if } u = v \\ -1, & \text{if } u \text{ and } v \text{ are adjacent} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where d is the degree (number of incident edges) of a vertex, and adjacency refers to a neighboring connection in the graph.

Typically, semi-supervised learning methods based on the graph Laplacian separate the manifold discovery process from the learning process so that manifold discovery is performed using unlabeled data only. We use both label information and multiple views to help ensure that the graph is indeed smooth with respect to the target concept being learned.

3.2. Random Subspace Multi-View Smoothing

When attempting to use secondary views in the construction of a graph regularizer, one must be cognizant of the potential for the secondary information to be much noisier than the primary information. Therefore, we need a method to alter the graph that attempts to use only the concept-relevant information from the secondary views. In the general setup we are addressing, the only information we have about the target concept is from the labels.

In semi-supervised learning, the number of labeled points is often very small in relation to the size of the graph. Applying label information directly through the use of a classifier can be problematic, since many classes contain subclasses or clusters, such that enforcing links between vertices based on labels alone is unwise. For example, creating an edge between two very dissimilar points can create a graph that violates the Riemannian assumptions that underpin the intuition of many of these approaches. We would like to have some sense that the end result will preserve some semblance of geodesic distance along the true manifold when these distances are obscured by noise in the ambient space. Changes based on a single classifier would be akin to label propagation, where every error could strongly affect the graph in a negative way.

We utilize a method based on random subspace selection (Bryll et al., 2003; Skurichina & Duin, 2001) in order to allow the labels to influence the graph construction in a robust way. The approach is simple in that we randomly select many feature splits and use them to train simple classifiers, each of which represents one subspace. We care very little about accuracy in the normal sense, because we are not looking to make a real classification. We simply want to place examples together in such a way as to provide some sense of smoothness according to our target, and therefore, if all of the classifiers were completely accurate, we would expect to end up ignoring any in-class clusters, and thus we would still have difficulty finding a useful graph.

We use the classifications of many classifiers from different subspaces to find a refined similarity that reflects information about our target concept in the various views. If two examples have similar feature subsets that are useful according to our labeled data, then we want them to be classified together. If they both lack good predictive features in particular subspaces, then they will often be misclassified together, which is what we want. Another advantage to using random subspaces is that we can expect the classifiers to more effectively utilize the labeled data based on the fact that random subspace selection reduces the dimensionality of the feature space on which each classifier learns.

Although one must specify the number of neighbors to retain and the number of splits to use in the graph construction (see Algorithm 1), the method seems to be rather robust with regard to these choices. In our experiments we perform 100 random feature splits for each view, resulting in two hundred subspace classifiers per view trained on the labeled data. We then construct a nearest neighbor graph based on cosine similarity in the primary view while weighting the similarity scores based on the percentage of shared classifications using theses random subspace models. We then build the unnormalized graph Laplacian. Based on common practice from related literature, we set the number of nearest neighbors, k, equal to 8. We test the effect of the graph changes using two different base algorithms that use the Laplacian matrix of the graph. The first learning algorithm is the Laplacian Eigenmap (LEM) approach described in (Belkin & Niyogi, 2004), and the second is the Laplacian Regularized Least Squares (LapRLS) approach described in (Belkin et al., 2006).

Algorithm 1 Graph Construction

```
  Input: data $\{(x_i, y_i)\}_{i=1}^{l}$, $\{x_i\}_{i=l+1}^{l+u}$ in each view,
  numNeighbors $k$, numSplits $s$
  for each view $v_j$ do
    for $index = 1$ to $s$ do
      Randomly split feature set into two equal parts
      Train linear classifier (SVM) on each split
      Classify each sample point using the classifiers
    end for
  end for
  Assign edge weights = (cosine similarity) × (percent
  of time classified together)
  Retain $k$ nearest neighbors
```

3.3. Transductive LEM Classifier

As described in (Belkin & Niyogi, 2004), we construct a linear classifier in a new space that allows transductive classification. The coefficients for the new dimensions are set by minimizing the sum of squared error on the labeled data. In other words, the coefficient vector **a** is obtained using the following equation:

$$a = (EE^T)^{-1}Ec \tag{2}$$

where c is a vector representing the class labels and the entries of E are the eigenfunctions of the Laplacian matrix, $\lambda_k v_{i,k}$; i is the index of the labeled point in the matrix, and k is the index in the new low-dimensional space. The mapping starts with the eigenvector associated with the first non-zero eigenvalue, and includes as many eigenvectors as the number of dimensions desired.

3.4. LEM Out-of-Sample Extension

Because LEM is inherently transductive, it only creates a mapping for an unlabeled example if it is part of the set used in the graph construction. For an out-of-sample extension, we use the Nystrom Formula as described in (Ouimet & Bengio, 2005). It employs the Laplacian matrix as a data-dependent Kernel function K_D in the following formula in order to map a new point into each dimension k of the new decision space:

$$f_k(x) = \frac{\sqrt{n}}{\lambda_k}\sum_{i=1}^{n} v_{ik}K_D(x, x_i) \tag{3}$$

where n is the size of the original dataset, and (λ_k, v_k) are the k-th eigenvalue and eigenvector.

3.5. Laplacian Regularized Least Squares

The second algorithm that we use to test the approach is Laplacian Regularized Least Squares (LapRLS) (Belkin et al., 2006), which uses two regularizers, including the Laplacian matrix. Once again, we use the same graph construction method to produce the multi-view-derived Laplacian matrix that is used in this algorithm. In this case the output function that is learned is the following:

$$f(x) = \sum_{i=1}^{l+u} \alpha_i K(x_i, x), \tag{4}$$

where K is the $(l+u) \times (l+u)$ Gram matrix over labeled and unlabeled points, and α is the following learned coefficient vector:

$$\alpha = (JK + \gamma_A lI + \frac{\gamma_I l}{(l+u)^2}LK)^{-1}Y, \tag{5}$$

with L being the Laplacian matrix described above, I being the $(l+u) \times (l+u)$ identity matrix, J being the $(l+u) \times (l+u)$ diagonal matrix with the first l diagonal entries equal to 1 and the rest of the entries equal to 0, and Y being the $(l+u)$ label vector, $Y = [y_1, ..., y_l, 0, ..., 0]$. See (Belkin et al., 2006) for details.

The modifications we employ are all during the graph construction phase. This means that we can train a LapRLS learner using a primary view in a straightforward manner since the secondary view information is encoded into the regularization term, $\frac{\gamma_I l}{(l+u)^2}LK$, via the matrix, L. While LapRLS avoids the need to select the number of dimensions as in the Laplacian Eigenmap approach, it does have its own parameters that control the effect of the unlabeled data. For all of our LapRLS experiments, we use the following parameters, as suggested for manifold regularization in (Belkin et al., 2006): $\gamma_A l = 0.005$, $\frac{\gamma_I l}{(l+u)^2} = 0.045$.

4. Theoretical Discussion

Although a thorough theoretical analysis is beyond the scope of this paper, we suggest how the general approach fits into existing theoretical work. The framework described in this paper can be considered to rely on a notion of compatibility, χ, as described in (Balcan & Blum, 2006; Balcan, 2008). The notion of compatibility is based on finding a model that has a low *unlabeled error rate*. In the case of a graph regularizer, this can indicate that the function being learned *agrees with the graph* and would not label two connected nodes with different class labels. Of course, if the graph incorrectly connects examples from different classes, then the target function itself does not have an unlabeled error rate of zero, even if some hypotheses do. Since compatibility is defined as an expectation over samples, in the multi-view setting the same theoretical arguments hold if the graph encoded notion of compatibility is derived from alternate views in addition to unlabeled data.

In (Balcan, 2008), various sample complexity bounds are provided. In some cases an assumption is made that the target function's unlabeled error rate is low (essentially zero), and in other cases the bounds depend on the unlabeled error of c^*, the true target function. For example, Theorem 2.3.2 provides a sample complexity bound in the realizable case ($c^* \in C$) that depends upon the unlabeled error of the target, c^*. A graph constructed over noisy samples is likely to have many "errors." Therefore, the first assumption is too simplistic for many real-world situations. Using unlabeled data alone, the target function's unlabeled error cannot be bounded at all, since it is entirely possible that similarity in the ambient feature space does not reflect similarity in terms of the target concept at all. In other words, the number of mistakes in the notion

Table 1. High-level classifier comparison: A linear SVM is trained on a single view consisting of just pixel features or all features combined into a single set. The same two sets are used for a LEM classifier, including a version that uses random subspace smoothing based on all features as a single set. This is compared to the method in this paper (MV-LEM), using all features for training, but only pixel features at test time. Transductive results are provided in addition to the inductive results on the set-aside test set.

	Features Used		Avg. Error Rate	
Classifier	Train	Test	Trans	Inductive
SVM	Pix	Pix	.099	.098
SVM	All	All	.081	.083
LEM	Pix	Pix	.083	.068
LEM	All	All	.085	.068
RS-LEM	All	All	.111	.081
MV-LEM	All	Pix	.073	.057

of compatibility itself (the graph) cannot be bound while ignoring all information concerning the target concept. Although labeled and unlabeled error are of different types, it should still be possible to use supervised PAC-learning bounds on generalization error to provide a bound on the unlabeled error rate of c^*, meaning that use of label information in the construction of the graph can bound this error with respect to the target, allowing bounds to be applied to the overall sample complexity.

5. Experimental Results

In order to demonstrate the effectiveness of the approach, we utilize the Multiple Features Dataset available through the UCI Machine Learning Repository (Frank & Asuncion, 2010). The dataset is an image recognition task over 2000 handwritten digits. It is a 10-class problem containing 200 examples of each digit. Each example is composed of 6 distinct features sets, which we use as 6 separate views; one primary and five secondary. The views are the following: 240 pixel averages in 2 × 3 windows (Pix); 216 profile correlations (Fac); 76 Fourier coefficients of the digit shapes (Fou); 64 Karhunen-Loéve features (Kar); 47 Zernike moments (Zer); and 6 morphological features (Mor). Additional information on the data can be found in (Van Breukelen et al., 1998). Each of our experimental results is the averaged error over 10 random splits of the data into a semi-supervised training set of 100 labeled and 900 unlabeled examples and a separate test set of 1000 examples for inductive testing. Table 1 shows a comparison of single view learning methods. When the training and test features are the same, it indicates that a single view was used; i.e. either the pixel features (Pix) or all features combined into a single feature set. In the case of the Multi-View Laplacian Eigenmap (MV-LEM), the views are treated separately, with pixel features being the only ones used for inductive testing and the other views providing information through the graph construction process as described above. It is interesting to note that the best performance is obtained with a model that only has access to the pixel values at test time and that the standard Laplacian Eigenmap approach does not improve with straightforward addition of the other features. The best performance is obtained by a careful approach that recognizes the potentially redundant information across multiple views and the hard realities one must face when attempting to include more features without the ability to increase the size of the labeled data.

Table 2. Knowledge carryover comparisons using LEM. Classification using primary view only. Each classifier uses the graph Laplacian for dimensionality reduction, retaining 40 eigenfunctions. Graph construction uses no smoothing (None) , random subspace smoothing based on the primary view only (Prime RS), or the cumulative effect of random subspace smoothing using all of the views (Both RS).

	Features Used		Avg. Error Rate	
Smoothing	Training	Test	Trans	Inductive
None	Pix	Pix	.083	.068
Prime RS	Pix	Pix	.095	.059
Both RS	All	Pix	.064	.054
None	Fac	Fac	.136	.124
Prime RS	Fac	Fac	.122	.101
Both RS	All	Fac	.072	.087
None	Fou	Fou	.307	.301
Prime RS	Fou	Fou	.316	.308
Both RS	All	Fou	.063	.227
None	Kar	Kar	.128	.114
Prime RS	Kar	Kar	.122	.092
Both RS	All	Kar	.065	.076
None	Zer	Zer	.276	.256
Prime RS	Zer	Zer	.276	.256
Both RS	All	Zer	.064	.203

Table 2 and Table 3 compare the effect of different graph construction methods using the LEM learner and the LapRLS learner. The smoothing of the graph takes one of the following three forms: no smoothing; random subspace smoothing via the primary view, or the cumulative effect of random subspace smoothing via all views. In this case, regardless of the method and regardless of the primary view, the addition of information from the other views during training always provides a significant level of improvement to the classifier that operates on the primary view only. We do not include the morphological features as a primary view, since the 6 features are not sufficient to generate

Table 3. Knowledge carryover comparisons using LapRLS. Classification using primary view only. Graph construction uses random subspace smoothing based on the primary view only (Prime RS) or the cumulative effect of random subspace smoothing using all of the views (Both RS).

	Features Used		Avg. Error Rate	
Smoothing	Training	Test	Trans	Inductive
Prime RS	Pix	Pix	.185	.213
Both RS	All	Pix	.162	.195
Prime RS	Fac	Fac	.113	.107
Both RS	All	Fac	.070	.078
Prime RS	Fou	Fou	.337	.349
Both RS	All	Fou	.318	.332
Prime RS	Kar	Kar	.180	.181
Both RS	All	Kar	.168	.172
Prime RS	Zer	Zer	.331	.329
Both RS	All	Zer	.304	.307

useful models for this 10 class problem.

6. Conclusion

We have demonstrated the use of principles from multi-view and semi-supervised learning for budgeted learning in the face of realistic constraints on the availability of both features and labels. Our experiments show consistent improvement when the only difference in training is the use of secondary views (not available at test time) to modify the Laplacian matrix used for regularization based on the approach outlined in this paper. This general learning framework can also admit the insertion of expert knowledge in other forms; e.g. via feature-based active learning.

Acknowledgments

Research sponsored by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the U.S. Department of Energy under contract no. DE-AC05-00OR22725. This manuscript has been authored by a contractor of the U.S. Government under Contract No. DE-AC05-00OR22725. Accordingly, the U.S. Government retains a non-exclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.

References

Balcan, Maria-Florina. *New Theoretical Frameworks for Machine Learning.* Phd thesis, Carnegie Mellon University, 2008.

Balcan, Maria-Florina and Blum, Avrim. An augmented PAC model for semi-supervised learning. In Chapelle, Olivier, Scholkopf, Bernhard, and Zien, Alexander (eds.), *Semi-Supervised Learning.* MIT Press, Cambridge, MA, 2006.

Belkin, Mikhail and Niyogi, Partha. Semi-supervised learning on Riemannian manifolds. *Machine Learning*, 56:209–239, 2004.

Belkin, Mikhail, Niyogi, Partha, and Sindhwani, Vikas. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7: 2399–2434, 2006.

Bryll, Robert, Gutierrez-Osuna, Ricardo, and Quek, Francis. Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets. *Pattern Recognition*, 36:1291–1302, 2003.

Chung, Fan R. K. *Spectral Graph Theory.* American Mathematical Society, Providence, RI, 1997.

Frank, A. and Asuncion, A. UCI machine learning repository, 2010. URL http://archive.ics.uci.edu/ml.

Lin, Tong and Zha, Hongbin. Riemannian manifold learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):796–809, 2008.

Ouimet, Marie and Bengio, Yoshua. Greedy spectral embedding. In *the 10th Workshop on Artificial Intelligence and Statistics (AISTATS)*, 2005.

Sindhwani, Vikas and Rosenberg, David S. An RKHS for multi-view learning and manifold co-regularization. In *Proceedings of the 25th International Conference on Machine Learning*, Helsinki, Finland, 2008.

Sindhwani, Vikas, Niyogi, Partha, and Belkin, Mikhail. A co-regularization approach to semi-supervised learning with multiple views. In *Workshop on Learning with Multiple Views, 22nd International Conference on Machine Learning*, Bonn, Germany, 2005.

Skurichina, Marina and Duin, R. P. W. Bagging and the random subspace method for redundant feature spaces. In Kittler, J. and Roli, F. (eds.), *Multiple Classifier Systems (MCS) 2001*, volume 2096, pp. 1–10. LNCS, 2001.

Van Breukelen, M., Duin, R. P. W., Tax, D. M. J., and Den Hartog, J.E. Handwritten digit recognition by combined classifiers. *Kybernetika*, 34(4):381–386, 1998.