Semi-Supervised Learning Literature Survey

Xiaojin Zhu

Computer Sciences TR 1530
University of Wisconsin – Madison
Last modified on July 19, 2008

Contents

1 FAQ 3

2 Generative Models 7
2.1 Identifiability 7
2.2 Model Correctness 9
2.3 EM Local Maxima 10
2.4 Cluster-and-Label 10
2.5 Fisher kernel for discriminative learning 10

3 Self-Training 11

4 Co-Training and Multiview Learning 11
4.1 Co-Training 11
4.2 Multiview Learning 13

5 Avoiding Changes in Dense Regions 13
5.1 Transductive SVMs (S3VMs) 13
5.2 Gaussian Processes 16
5.3 Information Regularization 17
5.4 Entropy Minimization 17
5.5 A Connection to Graph-based Methods? 17

6 Graph-Based Methods 18
6.1 Regularization by Graph 18
6.1.1 Mincut 18
6.1.2 Discrete Markov Random Fields: Boltzmann Machines . . 19
6.1.3 Gaussian Random Fields and Harmonic Functions 19
6.1.4 Local and Global Consistency 20
6.1.5 Tikhonov Regularization 20
6.1.6 Manifold Regularization 20
6.1.7 Graph Kernels from the Spectrum of Laplacian 21
6.1.8 Spectral Graph Transducer 22
6.1.9 Local Learning Regularization 22
6.1.10 Tree-Based Bayes 23
6.1.11 Some Other Methods 23
6.2 Graph Construction 24
6.3 Fast Computation 25
6.4 Induction 26
6.5 Consistency 27

6.6 Dissimilarity Edges, Directed Graphs, and Hypergraphs 28
6.7 Connection to Standard Graphical Models 29

7 Using Class Proportion Knowledge 29

8 Learning Efficient Encoding of the Domain from Unlabeled Data 30

9 Computational Learning Theory 32

10 Semi-supervised Learning in Structured Output Spaces 33
10.1 Generative Models 33
10.2 Graph-based Kernels 33

11 Related Areas 34
11.1 Spectral Clustering 34
11.2 Learning with Positive and Unlabeled Data 34
11.3 Semi-supervised Clustering 35
11.4 Semi-supervised Regression 35
11.5 Active Learning and Semi-supervised Learning 36
11.6 Nonlinear Dimensionality Reduction 37
11.7 Learning a Distance Metric 37
11.8 Inferring Label Sampling Mechanisms 39
11.9 Metric-Based Model Selection 40
11.10Multi-Instance Learning 41

12 Scalability Issues of Semi-Supervised Learning Methods 41

13 Do Humans do Semi-Supervised Learning? 41
13.1 Visual Object Recognition with Temporal Association 43
13.2 Infant Word-Meaning Mapping 44
13.3 Human Categorization Experiments 44

1 FAQ

Q: What's in this Document?
A: We review the literature on semi-supervised learning, which is an area in machine learning and more generally, artificial intelligence. There has been a whole spectrum of interesting ideas on how to learn from both labeled and unlabeled data, i.e. semi-supervised learning. This document originates as a chapter in the

author's doctoral thesis (Zhu, 2005). However the author will update the online version regularly to incorporate the latest development in the field. Please obtain the latest version at http://pages.cs.wisc.edu/~jerryzhu/research/ssl/semireview.html. The date below the title indicates its version. Older versions of the survey can be found at the same URL.

I recommend citation using the following bibtex entry:

```
@techreport{zhu05survey,
  author = "Xiaojin Zhu",
  title = "Semi-Supervised Learning Literature Survey",
  institution = "Computer Sciences, University of Wisconsin-Madison",
  number = "1530",
  year = 2005
}
```

The review is by no means comprehensive as the field of semi-supervised learning is evolving rapidly. It is difficult for one person to summarize the field. The author apologizes in advance for any missed papers and inaccuracies in descriptions. Corrections and comments are highly welcome. Please send them to jerryzhu@cs.wisc.edu.

Q: What is semi-supervised learning?

A: In this survey we focus on semi-supervised classification. It is a special form of classification. Traditional classifiers use only labeled data (feature / label pairs) to train. Labeled instances however are often difficult, expensive, or time consuming to obtain, as they require the efforts of experienced human annotators. Meanwhile unlabeled data may be relatively easy to collect, but there has been few ways to use them. Semi-supervised learning addresses this problem by using large amount of unlabeled data, together with the labeled data, to build better classifiers. Because semi-supervised learning requires less human effort and gives higher accuracy, it is of great interest both in theory and in practice.

Semi-supervised classification's cousins, semi-supervised clustering and regression, are briefly discussed in section 11.3 and 11.4.

Q: Can we really learn anything from unlabeled data? It sounds like magic.

A: Yes we can – under certain assumptions. It's not magic, but good matching of problem structure with model assumption.

Many semi-supervised learning papers, including this one, start with an introduction like: "labels are hard to obtain while unlabeled data are abundant, therefore semi-supervised learning is a good idea to reduce human labor and improve accuracy". Do not take it for granted. Even though you (or your domain expert) do not spend as much time in labeling the training data, you need to spend reasonable amount of effort to design good models / features / kernels / similarity functions for semi-supervised learning. In my opinion such effort is more critical than for supervised learning to make up for the lack of labeled training data.

Q: Does unlabeled data always help?
A: No, there's no free lunch. Bad matching of problem structure with model assumption can lead to degradation in classifier performance. For example, quite a few semi-supervised learning methods assume that the decision boundary should avoid regions with high $p(x)$. These methods include transductive support vector machines (TSVMs), information regularization, Gaussian processes with null category noise model, graph-based methods if the graph weights is determined by pairwise distance. Nonetheless if the data is generated from two heavily overlapping Gaussian, the decision boundary would go right through the densest region, and these methods would perform badly. On the other hand EM with generative mixture models, another semi-supervised learning method, would have easily solved the problem. Detecting bad match in advance however is hard and remains an open question.

Anecdotally, the fact that unlabeled data do not always help semi-supervised learning has been observed by multiple researchers. For example people have long realized that training Hidden Markov Model with unlabeled data (the Baum-Welsh algorithm, which by the way qualifies as semi-supervised learning on sequences) can reduce accuracy under certain initial conditions (Elworthy, 1994). See (Cozman et al., 2003) for a more recent argument. Not much is in the literature though, presumably because of the publication bias.

Q: How many semi-supervised learning methods are there?
A: Many. Some often-used methods include: EM with generative mixture models, self-training, co-training, transductive support vector machines, and graph-based methods. See the following sections for more methods.

Q: Which method should I use / is the best?
A: There is no direct answer to this question. Because labeled data is scarce, semi-supervised learning methods make strong model assumptions. Ideally one should use a method whose assumptions fit the problem structure. This may be difficult in reality. Nonetheless we can try the following checklist: Do the classes produce well clustered data? If yes, EM with generative mixture models may be a good choice; Do the features naturally split into two sets? If yes, co-training may be appropriate; Is it true that two points with similar features tend to be in the same class? If yes, graph-based methods can be used; Already using SVM? Transductive SVM is a natural extension; Is the existing supervised classifier complicated and

hard to modify? Self-training is a practical wrapper method.

Q: How do semi-supervised learning methods use unlabeled data?
A: Semi-supervised learning methods use unlabeled data to either modify or reprioritize hypotheses obtained from labeled data alone. Although not all methods are probabilistic, it is easier to look at methods that represent hypotheses by $p(y|x)$, and unlabeled data by $p(x)$. Generative models have common parameters for the joint distribution $p(x, y)$. It is easy to see that $p(x)$ influences $p(y|x)$. Mixture models with EM is in this category, and to some extent self-training. Many other methods are discriminative, including transductive SVM, Gaussian processes, information regularization, and graph-based methods. Original discriminative training cannot be used for semi-supervised learning, since $p(y|x)$ is estimated ignoring $p(x)$. To solve the problem, $p(x)$ dependent terms are often brought into the objective function, which amounts to assuming $p(y|x)$ and $p(x)$ share parameters.

Q: What is the difference between 'transductive learning' and 'semi-supervised learning'?
A: Different authors use slightly different names. In this survey we will use the following convention:

- 'Semi-supervised learning' refers to the use of both labeled and unlabeled data for training. It contrasts supervised learning (data all labeled) or unsupervised learning (data all unlabeled). Other names are 'learning from labeled and unlabeled data' or 'learning from partially labeled/classified data'. Notice semi-supervised learning can be either transductive or inductive.

- 'Transductive learning' will be used to contrast inductive learning. A learner is transductive if it only works on the labeled and unlabeled training data, and cannot handle unseen data. The early graph-based methods are often transductive. Inductive learners can naturally handle unseen data. Notice under this convention *transductive support vector machines* (TSVMs) are in fact inductive learners, because the resulting classifiers are defined over the whole space. The name TSVM originates from the intention to work only on the observed data (though people use them for induction anyway), which according to (Vapnik, 1998) is solving a simpler problem. People sometimes use the analogy that transductive learning is take-home exam, while inductive learning is in-class exam.

- In this survey semi-supervised learning refers to 'semi-supervised classification', where one has additional unlabeled data and the goal is classification. Its cousin 'semi-supervised clustering', where one has unlabeled data with some pairwise constraints and the goal is clustering, is only briefly discussed later in the survey.

We will follow the above convention in the survey.

Q: Where can I learn more?
A: A book on semi-supervised learning is (Chapelle et al., 2006c). An older survey can be found in (Seeger, 2001). I gave a tutorial at ICML 2007, the slides can be found at `http://pages.cs.wisc.edu/~jerryzhu/icml07tutorial.html`.

2 Generative Models

Generative models are perhaps the oldest semi-supervised learning method. It assumes a model $p(x, y) = p(y)p(x|y)$ where $p(x|y)$ is an identifiable mixture distribution, for example Gaussian mixture models. With large amount of unlabeled data, the mixture components can be identified; then ideally we only need one labeled example per component to fully determine the mixture distribution, see Figure 1. One can think of the mixture components as 'soft clusters'.

Nigam et al. (2000) apply the EM algorithm on mixture of multinomial for the task of text classification. They showed the resulting classifiers perform better than those trained only from L. Baluja (1998) uses the same algorithm on a face orientation discrimination task. Fujino et al. (2005) extend generative mixture models by including a 'bias correction' term and discriminative training using the maximum entropy principle.

One has to pay attention to a few things:

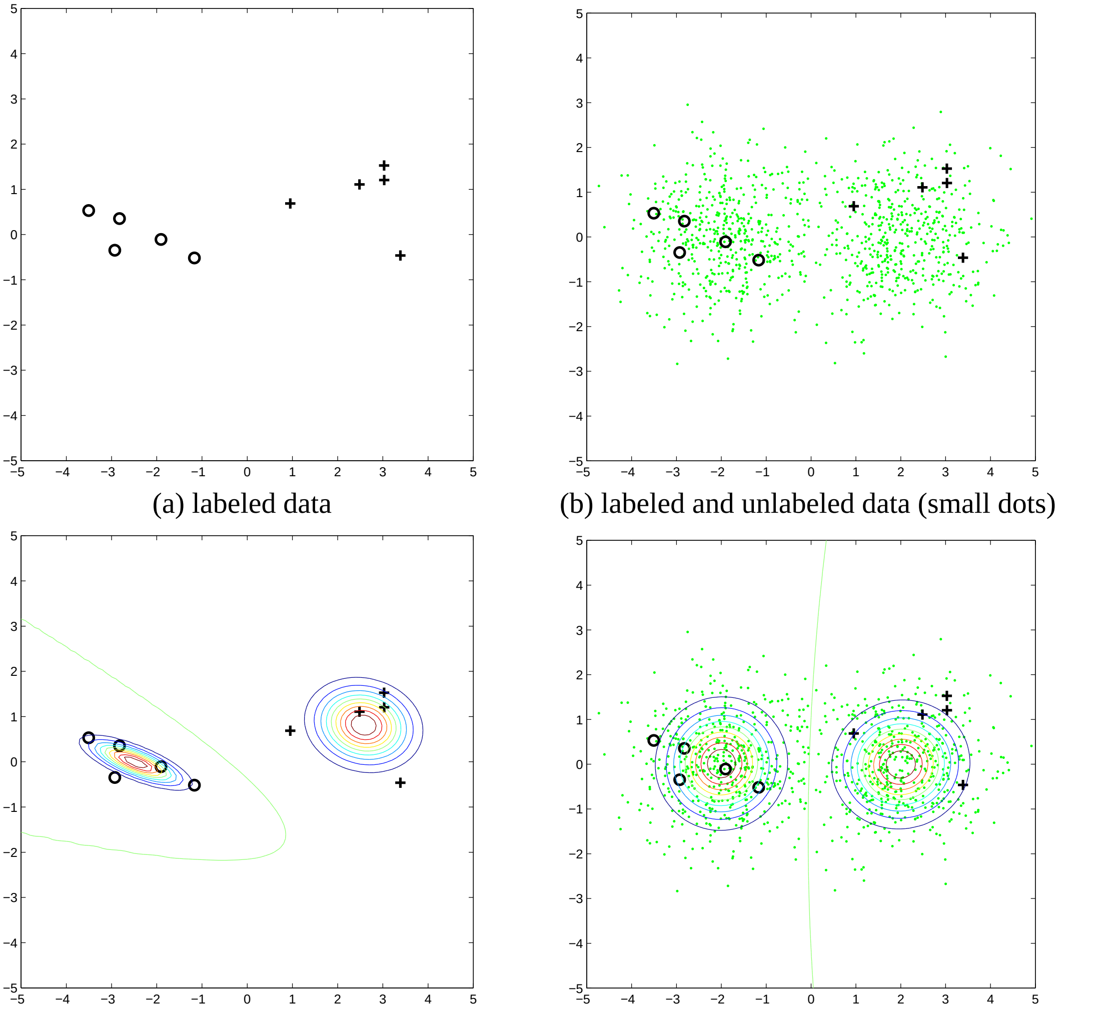
2.1 Identifiability

The mixture model ideally should be identifiable. In general let $\{p_\theta\}$ be a family of distributions indexed by a parameter vector θ. θ is identifiable if $\theta_1 \neq \theta_2 \Rightarrow p_{\theta_1} \neq p_{\theta_2}$, up to a permutation of mixture components. If the model family is identifiable, in theory with infinite U one can learn θ up to a permutation of component indices.

Here is an example showing the problem with unidentifiable models. The model $p(x|y)$ is uniform for $y \in \{+1, -1\}$. Assuming with large amount of unlabeled data U we know $p(x)$ is uniform in $[0, 1]$. We also have 2 labeled data points $(0.1, +1), (0.9, -1)$. Can we determine the label for $x = 0.5$? No. With our assumptions we cannot distinguish the following two models:

$$p(y = 1) = 0.2,\ p(x|y = 1) = \text{unif}(0, 0.2),\ p(x|y = -1) = \text{unif}(0.2, 1) \quad (1)$$

$$p(y = 1) = 0.6,\ p(x|y = 1) = \text{unif}(0, 0.6),\ p(x|y = -1) = \text{unif}(0.6, 1) \quad (2)$$



(a) labeled data

(b) labeled and unlabeled data (small dots)

(c) model learned from labeled data

(d) model learned from labeled and unlabeled data

Figure 1: In a binary classification problem, if we assume each class has a Gaussian distribution, then we can use unlabeled data to help parameter estimation.

which give opposite labels at $x = 0.5$, see Figure 2. It is known that a mixture of

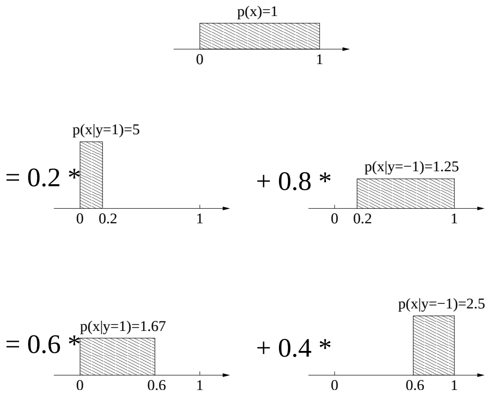


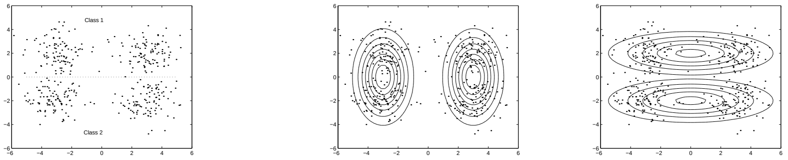
Figure 2: An example of unidentifiable models. Even if we known $p(x)$ (top) is a mixture of two uniform distributions, we cannot uniquely identify the two components. For instance, the mixtures on the second and third line give the same $p(x)$, but they classify $x = 0.5$ differently.

Gaussian is identifiable. Mixture of multivariate Bernoulli (McCallum & Nigam, 1998a) is not identifiable. More discussions on identifiability and semi-supervised learning can be found in e.g. (Ratsaby & Venkatesh, 1995) and (Corduneanu & Jaakkola, 2001).

2.2 Model Correctness

If the mixture model assumption is correct, unlabeled data is guaranteed to improve accuracy (Castelli & Cover, 1995) (Castelli & Cover, 1996) (Ratsaby & Venkatesh, 1995). However if the model is wrong, unlabeled data may actually hurt accuracy. Figure 3 shows an example. This has been observed by multiple researchers. Cozman et al. (2003) give a formal derivation on how this might happen.

It is thus important to carefully construct the mixture model to reflect reality. For example in text categorization a topic may contain several sub-topics, and will be better modeled by multiple multinomial instead of a single one (Nigam et al., 2000). Some other examples are (Shahshahani & Landgrebe, 1994) (Miller & Uyar, 1997). Another solution is to down-weighing unlabeled data (Corduneanu & Jaakkola, 2001), which is also used by Nigam et al. (2000), and by Callison-Burch et al. (2004) who estimate word alignment for machine translation.



(a) Horizontal class separation (b) High probability (c) Low probability

Figure 3: If the model is wrong, higher likelihood may lead to lower classification accuracy. For example, **(a)** is clearly not generated from two Gaussian. If we insist that each class is a single Gaussian, **(b)** will have higher probability than **(c)**. But **(b)** has around 50% accuracy, while **(c)**'s is much better.

2.3 EM Local Maxima

Even if the mixture model assumption is correct, in practice mixture components are identified by the Expectation-Maximization (EM) algorithm (Dempster et al., 1977). EM is prone to local maxima. If a local maximum is far from the global maximum, unlabeled data may again hurt learning. Remedies include smart choice of starting point by active learning (Nigam, 2001).

2.4 Cluster-and-Label

We shall also mention that instead of using an probabilistic generative mixture model, some approaches employ various clustering algorithms to cluster the whole dataset, then label each cluster with labeled data, e.g. (Demiriz et al., 1999) (Dara et al., 2002). Although they can perform well if the particular clustering algorithms match the true data distribution, these approaches are hard to analyze due to their algorithmic nature.

2.5 Fisher kernel for discriminative learning

Another approach for semi-supervised learning with generative models is to convert data into a feature representation determined by the generative model. The new feature representation is then fed into a standard discriminative classifier. Holub et al. (2005) used this approach for image categorization. First a generative mixture model is trained, one component per class. At this stage the unlabeled data can be incorporated via EM, which is the same as in previous subsections. However instead of directly using the generative model for classification, each labeled example is converted into a fixed-length Fisher score vector, i.e. the derivatives of log likelihood w.r.t. model parameters, for all component models (Jaakkola & Haussler, 1998). These Fisher score vectors are then used in a discriminative classifier like an SVM, which empirically has high accuracy.

3 Self-Training

Self-training is a commonly used technique for semi-supervised learning. In self-training a classifier is first trained with the small amount of labeled data. The classifier is then used to classify the unlabeled data. Typically the most confident unlabeled points, together with their predicted labels, are added to the training set. The classifier is re-trained and the procedure repeated. Note the classifier uses its own predictions to teach itself. The procedure is also called self-teaching or bootstrapping (not to be confused with the statistical procedure with the same name). The generative model and EM approach of section 2 can be viewed as a special case of 'soft' self-training. One can imagine that a classification mistake can reinforce itself. Some algorithms try to avoid this by 'unlearn' unlabeled points if the prediction confidence drops below a threshold.

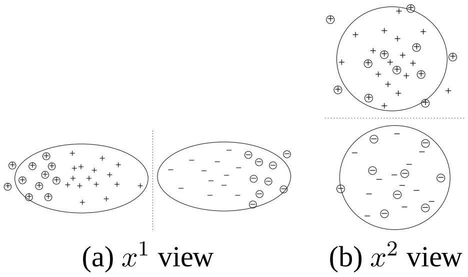
Self-training has been applied to several natural language processing tasks. Yarowsky (1995) uses self-training for word sense disambiguation, e.g. deciding whether the word 'plant' means a living organism or a factory in a give context. Riloff et al. (2003) uses it to identify subjective nouns. Maeireizo et al. (2004) classify dialogues as 'emotional' or 'non-emotional' with a procedure involving two classifiers.Self-training has also been applied to parsing and machine translation. Rosenberg et al. (2005) apply self-training to object detection systems from images, and show the semi-supervised technique compares favorably with a state-of-the-art detector.

Self-training is a wrapper algorithm, and is hard to analyze in general. However, for specific base learners, there has been some analyzer's on convergence. See e.g. (Haffari & Sarkar, 2007; Culp & Michailidis, 2007).

4 Co-Training and Multiview Learning

4.1 Co-Training

Co-training (Blum & Mitchell, 1998) (Mitchell, 1999) assumes that (i) features can be split into two sets; (ii) each sub-feature set is sufficient to train a good classifier; (iii) the two sets are conditionally independent given the class. Initially two separate classifiers are trained with the labeled data, on the two sub-feature sets respectively. Each classifier then classifies the unlabeled data, and 'teaches' the other classifier with the few unlabeled examples (and the predicted labels) they feel



(a) x^1 view (b) x^2 view

Figure 4: Co-Training: Conditional independent assumption on feature split. With this assumption the high confident data points in x^1 view, represented by circled labels, will be randomly scattered in x^2 view. This is advantageous if they are to be used to teach the classifier in x^2 view.

most confident. Each classifier is retrained with the additional training examples given by the other classifier, and the process repeats.

In co-training, unlabeled data helps by reducing the version space size. In other words, the two classifiers (or hypotheses) must agree on the much larger unlabeled data as well as the labeled data.

We need the assumption that sub-features are sufficiently good, so that we can trust the labels by each learner on U. We need the sub-features to be conditionally independent so that one classifier's high confident data points are *iid* samples for the other classifier. Figure 4 visualizes the assumption.

Nigam and Ghani (2000) perform extensive empirical experiments to compare co-training with generative mixture models and EM. Their result shows co-training performs well if the conditional independence assumption indeed holds. In addition, it is better to probabilistically label the entire U, instead of a few most confident data points. They name this paradigm co-EM. Finally, if there is no natural feature split, the authors create artificial split by randomly break the feature set into two subsets. They show co-training with artificial feature split still helps, though not as much as before. Collins and Singer (1999); Jones (2005) used co-training, co-EM and other related methods for information extraction from text. Balcan and Blum (2006) show that co-training can be quite effective, that in the extreme case only one labeled point is needed to learn the classifier. Zhou et al. (2007) give a co-training algorithm using Canonical Correlation Analysis which also need only one labeled point. Dasgupta et al. (Dasgupta et al., 2001) provide a PAC-style theoretical analysis.

Co-training makes strong assumptions on the splitting of features. One might wonder if these conditions can be relaxed. Goldman and Zhou (2000) use two learners of different type but both takes the whole feature set, and essentially use one learner's high confidence data points, identified with a set of statistical tests, in U to teach the other learning and vice versa. Chawla and Karakoulas (2005) perform empirical studies on this version of co-training and compared it against several other methods, in particular for the case where labeled and unlabeled data do not follow the same distribution. Later Zhou and Goldman (2004) propose a single-view multiple-learner Democratic Co-learning algorithm. An ensemble of learners with different inductive bias are trained separately on the complete feature of the labeled data. They then make predictions on the unlabeled data. If a majority of learners confidently agree on the class of an unlabeled point x_u, that classification is used as the label of x_u. x_u and its label is added to the training data. All learners are retrained on the updated training set. The final prediction is made with a variant of a weighted majority vote among all the learners. Similarly Zhou and Li (2005b) propose 'tri-training' which uses three learners. If two of them agree on the classification of an unlabeled point, the classification is used to teach the third classifier. This approach thus avoids the need of explicitly measuring label confidence of any learner. It can be applied to datasets without different views, or different types of classifiers. Balcan et al. (2005b) relax the conditional independence assumption with a much weaker expansion condition, and justify the iterative co-training procedure. Johnson and Zhang (2007) propose a two-view model that relaxes the conditional independence assumption.

4.2 Multiview Learning

More generally, we can define learning paradigms that utilize the agreement among different learners. The particular assumptions of Co-Training are in general not required by multiview learning models. Instead, multiple hypotheses (with different inductive biases, e.g., decision trees, SVMs, etc.) are trained from the same labeled data set, and are required to make similar predictions on any given unlabeled instance. Multiview learning has a long history (de Sa, 1993). It has been applied to semi-supervised regression (Sindhwani et al., 2005b; Brefeld et al., 2006), and the more challenging structured output spaces (Brefeld et al., 2005; Brefeld & Scheffer, 2006). Some theoretical analysis on the value of agreement among multiple learners can be found in (Leskes, 2005; Farquhar et al., 2006).

5 Avoiding Changes in Dense Regions

5.1 Transductive SVMs (S3VMs)

Discriminative methods work on $p(y|x)$ directly. This brings up the danger of leaving $p(x)$ outside of the parameter estimation loop, if $p(x)$ and $p(y|x)$ do not share parameters. Notice $p(x)$ is usually all we can get from unlabeled data. It is believed that if $p(x)$ and $p(y|x)$ do not share parameters, semi-supervised learning cannot help. This point is emphasized in (Seeger, 2001).

Transductive support vector machines (TSVMs)[1] builds the connection between $p(x)$ and the discriminative decision boundary by not putting the boundary in high density regions. TSVM is an extension of standard support vector machines with unlabeled data. In a standard SVM only the labeled data is used, and the goal is to find a maximum margin linear boundary in the Reproducing Kernel Hilbert Space. In a TSVM the unlabeled data is also used. The goal is to find a labeling of the unlabeled data, so that a linear boundary has the maximum margin on both the original labeled data and the (now labeled) unlabeled data. The decision boundary has the smallest generalization error bound on unlabeled data (Vapnik, 1998). Intuitively, unlabeled data guides the linear boundary away from dense regions.

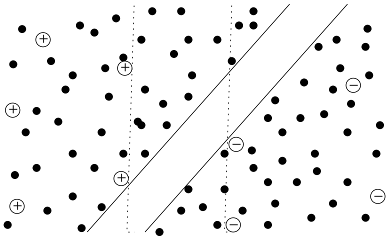


Figure 5: In TSVM, U helps to put the decision boundary in sparse regions. With labeled data only, the maximum margin boundary is plotted with dotted lines. With unlabeled data (black dots), the maximum margin boundary would be the one with solid lines.

However finding the exact transductive SVM solution is NP-hard. Major effort has focused on efficient approximation algorithms. Early algorithms (Bennett & Demiriz, 1999) (Demirez & Bennett, 2000) (Fung & Mangasarian, 1999) either cannot handle more than a few hundred unlabeled examples, or did not do so in experiments. The SVM-light TSVM implementation (Joachims, 1999) is the first widely used software.

De Bie and Cristianini (De Bie & Cristianini, 2004; De Bie & Cristianini, 2006b) relax the TSVM training problem, and transductive learning problems in general to semi-definite programming (SDP). The basic idea is to work with the binary label matrix of rank 1, and relax it by a positive semi-definite matrix without the rank constraint. The paper also includes a speech up trick to solve median-sized

[1] In recent papers, TSVMs are also called *Semi-Supervised Support Vector Machines* (S^{3}VM), because the learned classifiers can in fact be used inductively to predict on unseen data.

problems with around 1000 unlabeled points. Xu and Schuurmans (2005) present a similar multi-class version of SDP formulation, which results in multi-class SVM for semi-supervised learning. The computational cost of SDP is still expensive though.

TSVM can be viewed as SVM with an additional regularization term on unlabeled data. Let $f(x) = h(x) + b$ where $h \in \mathcal{H}_K$. The optimization problem is

$$\min_f \sum_{i=1}^{l} (1 - y_i f(x_i))_+ + \lambda_1 \|h\|^2_{\mathcal{H}_K} + \lambda_2 \sum_{i=l+1}^{n} (1 - |f(x_i)|)_+ \tag{3}$$

where $(z)_+ = \max(z, 0)$. The last term arises from assigning label sign$(f(x))$ to unlabeled point x. The margin on unlabeled point is thus sign$(f(x))f(x) = |f(x)|$. The loss function $(1 - |f(x_i)|)_+$ has a non-convex hat shape as shown in Figure 6, which is the root of the optimization difficulty.

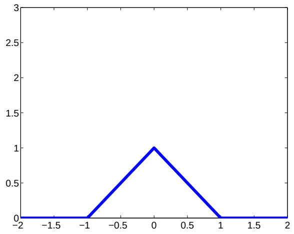


Figure 6: The TSVM loss function $(1 - |f(x_i)|)_+$

Chapelle and Zien (2005) propose ∇SVM, which approximates the hat loss $(1-|f(x_i)|)_+$ with a Gaussian function, and perform gradient search in the primal space. Sindhwani et al. (2006) use a deterministic annealing approach, which starts from an ‘easy’ problem, and gradually deforms it to the TSVM objective. In a similar spirit, Chapelle et al. (2006a) use a continuation approach, which also starts by minimizing an easy convex objective function, and gradually deforms it to the TSVM objective (with Gaussian instead of hat loss), using the solution of previous iterations to initialize the next ones. Collobert et al. (2006) optimize the hard TSVM directly, using an approximate optimization procedure known as concave-convex procedure (CCCP). The key is to notice that the hat loss is a sum of a convex function and a concave function. By replacing the concave function with a linear upper bound, one can perform convex minimization to produce an upper bound of the loss function. This is repeated until a local minimum is reached. The authors report significant speed up of TSVM training with CCCP. Sindhwani and Keerthi (2006) proposed a fast algorithm for *linear* S3VMs, suitable for large scale text applications. Their implementation can be found at `http://people.cs.uchicago.edu/~vikass/svmlin.html`.

With all the approximation solutions to TSVMs, it is interesting to understand just how good a global optimum TSVM can be. With the Branch and Bound search technique, Chapelle et al. (2006b) finds the global optimal solution for small datasets. The results indicate excellent accuracy. Although Branch and Bound will probably never be useful for large datasets, the results provide some ground truth, and points to the potentials of TSVMs with better approximation methods.

Weston et al. (2006) learn with a 'universum', which is a set of unlabeled data that is known to come from *neither* of the two classes. The decision boundary is encouraged to pass through the universum. One interpretation is similar to the maximum entropy principle: the classifier should be confident on labeled examples, yet maximally ignorant on unrelated examples.

Zhang and Oles (2000) point out that TSVMs may not behave well under some circumstances.

The maximum entropy discrimination approach (Jaakkola et al., 1999) also maximizes the margin, and is able to take into account unlabeled data, with SVM as a special case.

5.2 Gaussian Processes

Lawrence and Jordan (2005) proposed a Gaussian process approach, which can be viewed as the Gaussian process parallel of TSVM. The key difference to a standard Gaussian process is in the noise model. A 'null category noise model' maps the hidden continuous variable f to three instead of two labels, specifically to the never used label '0' when f is around zero. On top of that, it is restricted that unlabeled data points cannot take the label 0. This pushes the posterior of f away from zero for the unlabeled points. It achieves the similar effect of TSVM where the margin avoids dense unlabeled data region. However nothing special is done on the process model. Therefore all the benefit of unlabeled data comes from the noise model. A very similar noise model is proposed in (Chu & Ghahramani, 2004) for ordinal regression.

Chu et al. (2006) develop Gaussian process models that incorporate pairwise label relations (e.g. two points tends to have similar or different labels). Note such similar-label information is equivalent to those used in graph-based semi-supervised learning. Such models, using only similarity information, are applied to semi-supervised learning successfully. However dissimilarity is only briefly discussed, with many questions remain open.

There is a finite form of a Gaussian process in (Zhu et al., 2003c), in fact a joint Gaussian distribution on the labeled and unlabeled points with the covariance matrix derived from the graph Laplacian. Semi-supervised learning happens in the process model, not the noise model.

5.3 Information Regularization

Szummer and Jaakkola (2002) propose the information regularization framework to control the label conditionals $p(y|x)$ by $p(x)$, where $p(x)$ may be estimated from unlabeled data. The idea is that labels shouldn't change too much in regions where $p(x)$ is high. The authors use the mutual information $I(x;y)$ between x and y as a measure of label complexity. $I(x;y)$ is small when the labels are homogeneous, and large when labels vary. This motives the minimization of the product of $p(x)$ mass in a region with $I(x;y)$ (normalized by a variance term). The minimization is carried out on multiple overlapping regions covering the data space.

The theory is developed further in (Corduneanu & Jaakkola, 2003). Corduneanu and Jaakkola (2005) extend the work by formulating semi-supervised learning as a communication problem. Regularization is expressed as the rate of information, which again discourages complex conditionals $p(y|x)$ in regions with high $p(x)$. The problem becomes finding the unique $p(y|x)$ that minimizes a regularized loss on labeled data. The authors give a local propagation algorithm.

5.4 Entropy Minimization

The hyperparameter learning method in section 7.2 of (Zhu, 2005) uses entropy minimization. Grandvalet and Bengio (2005) used the label entropy on unlabeled data as a regularizer. By minimizing the entropy, the method assumes a prior which prefers minimal class overlap.

Lee et al. (2006) apply the principle of entropy minimization for semi-supervised learning on 2-D conditional random fields for image pixel classification. In particular, the training objective is to maximize the standard conditional log likelihood, and at the same time minimize the conditional entropy of label predictions on unlabeled image pixels.

5.5 A Connection to Graph-based Methods?

Let $p(x)$ be a probability distribution from which labeled and unlabeled data are drawn. Narayanan et al. (2006) prove that the 'weighted boundary volume', i.e. the surface integral $\int_S p(s)ds$ along a decision boundary S, is approximated by $\frac{\sqrt{\pi}}{N\sqrt{t}} f^\top L f$ when the number of iid data points N tends to infinity. Here L is the normalized graph Laplacian and f is an indicator function of the cut, and t is the bandwidth of the edge weight Gaussian function, which must tend to zero at a certain rate. This result suggests that S3VMs and related methods which seek a decision boundary that passes through low density regions, and graph-based semi-supervised learning methods which approximately compute the graph cut, might be more strongly connected that previously thought.

6 Graph-Based Methods

Graph-based semi-supervised methods define a graph where the nodes are labeled and unlabeled examples in the dataset, and edges (may be weighted) reflect the similarity of examples. These methods usually assume label smoothness over the graph. Graph methods are nonparametric, discriminative, and transductive in nature.

6.1 Regularization by Graph

Many graph-based methods can be viewed as estimating a function f on the graph. One wants f to satisfy two things at the same time: 1) it should be close to the given labels y_L on the labeled nodes, and 2) it should be smooth on the whole graph. This can be expressed in a regularization framework where the first term is a loss function, and the second term is a regularizer.

Several graph-based methods listed here are similar to each other. They differ in the particular choice of the loss function and the regularizer. We believe it is more important to construct a good graph than to choose among the methods. However graph construction, as we will see later, is not a well studied area.

6.1.1 Mincut

Blum and Chawla (2001) pose semi-supervised learning as a graph mincut (also known as st-cut) problem. In the binary case, positive labels act as sources and negative labels act as sinks. The objective is to find a minimum set of edges whose removal blocks all flow from the sources to the sinks. The nodes connecting to the sources are then labeled positive, and those to the sinks are labeled negative. Equivalently mincut is the *mode* of a Markov random field with binary labels (Boltzmann machine). The loss function can be viewed as a quadratic loss with infinity weight: $\infty \sum_{i \in L}(y_i - y_{i|L})^2$, so that the values on labeled data are in fact fixed at their given labels. The regularizer is

$$\frac{1}{2}\sum_{i,j} w_{ij}|y_i - y_j| = \frac{1}{2}\sum_{i,j} w_{ij}(y_i - y_j)^2 \tag{4}$$

The equality holds because the y's take binary (0 and 1) labels. Putting the two together, mincut can be viewed to minimize the function

$$\infty \sum_{i \in L} (y_i - y_{i|L})^2 + \frac{1}{2} \sum_{i,j} w_{ij} (y_i - y_j)^2 \tag{5}$$

subject to the constraint $y_i \in \{0, 1\}, \forall i$.

One problem with mincut is that it only gives hard classification without confidence (i.e. it computes the mode, not the marginal probabilities). Blum et al. (2004) perturb the graph by adding random noise to the edge weights. Mincut is applied to multiple perturbed graphs, and the labels are determined by a majority vote. The procedure is similar to bagging, and creates a 'soft' mincut.

Pang and Lee (2004) use mincut to improve the classification of a sentence into either 'objective' or 'subjective', with the assumption that sentences close to each other tend to have the same class.

6.1.2 Discrete Markov Random Fields: Boltzmann Machines

The proper but hard way is to compute the marginal probabilities of the discrete Markov random fields. This is inherently a difficult inference problem. Zhu and Ghahramani (2002) attempted exactly this, but were limited by the MCMC sampling techniques (they used global Metropolis and Swendsen-Wang sampling).

Getz et al. (2005) computes the marginal probabilities of the discrete Markov random field at any temperature with the Multi-canonical Monte-Carlo method, which seems to be able to overcome the energy trap faced by the standard Metropolis or Swendsen-Wang method. The authors discuss the relationship between temperatures and phases in such systems. They also propose a heuristic procedure to identify possible new classes.

6.1.3 Gaussian Random Fields and Harmonic Functions

The Gaussian random fields and harmonic function methods in (Zhu et al., 2003a) is a continuous relaxation to the difficulty discrete Markov random fields (or Boltzmann machines). It can be viewed as having a quadratic loss function with infinity weight, so that the labeled data are clamped (fixed at given label values), and a regularizer based on the graph combinatorial Laplacian Δ:

$$\begin{aligned} & \infty \sum_{i \in L} (f_i - y_i)^2 + 1/2 \sum_{i,j} w_{ij} (f_i - f_j)^2 && (6) \\ = \quad & \infty \sum_{i \in L} (f_i - y_i)^2 + f^\top \Delta f && (7) \end{aligned}$$

Notice $f_i \in \mathbb{R}$, which is the key relaxation to Mincut. This allows for a simple closed-form solution for the node marginal probabilities. The mean is known as a harmonic function, which has many interesting properties (Zhu, 2005).

Recently Grady and Funka-Lea (2004) applied the harmonic function method to medical image segmentation tasks, where a user labels classes (e.g. different organs) with a few strokes. Levin et al. (2004) use the equivalent of harmonic functions for colorization of gray-scale images. Again the user specifies the desired color with only a few strokes on the image. The rest of the image is used as unlabeled data, and the labels propagation through the image. Niu et al. (2005) applied the label propagation algorithm (which is equivalent to harmonic functions) to word sense disambiguation. Goldberg and Zhu (2006) applied the algorithm to sentiment analysis for movie rating prediction.

6.1.4 Local and Global Consistency

The local and global consistency method (Zhou et al., 2004a) uses the loss function $\sum_{i=1}^{n}(f_i - y_i)^2$, and the *normalized Laplacian* $D^{-1/2}\Delta D^{-1/2} = I - D^{-1/2}WD^{-1/2}$ in the regularizer,

$$1/2 \sum_{i,j} w_{ij}(f_i/\sqrt{D_{ii}} - f_j/\sqrt{D_{jj}})^2 \quad = \quad f^\top D^{-1/2}\Delta D^{-1/2} f \tag{8}$$

6.1.5 Tikhonov Regularization

The Tikhonov regularization algorithm in (Belkin et al., 2004a) uses the loss function and regularizer:

$$1/k \sum_i (f_i - y_i)^2 + \gamma f^\top S f \tag{9}$$

where $S = \Delta$ or Δ^p for some integer p.

6.1.6 Manifold Regularization

The manifold regularization framework (Belkin et al., 2004b) (Belkin et al., 2005) employs two regularization terms:

$$\frac{1}{l}\sum_{i=1}^{l} V(x_i, y_i, f) + \gamma_A ||f||_K^2 + \gamma_I ||f||_I^2 \tag{10}$$

where V is an arbitrary loss function, K is a 'base kernel', e.g. a linear or RBF kernel. I is a regularization term induced by the labeled and unlabeled data. For example, one can use

$$||f||_I^2 = \frac{1}{(l+u)^2} \hat{f}^\top \Delta \hat{f} \tag{11}$$

where $\hat{f}$ is the vector of f evaluations on $L \cup U$.

Sindhwani et al. (2005a) give a semi-supervised kernel that is not limited to the unlabeled points, but defined over all input space. The kernel thus supports induction. Essentially the kernel is a new interpretation of the manifold regularization framework above. Starting from a base kernel K defined over the whole input space (e.g. linear kernels, RBF kernels), the authors modify the RKHS by keeping the same function space but changing the norm. Specifically a 'point-cloud norm' defined by $L \cup U$ is added to the original norm. The point-cloud norm corresponds to $||f||_I^2$. Importantly this results in a new RKHS space, with a corresponding new kernel that deforms the original one along a finite-dimensional subspace given by the data. The new kernel is defined over the whole space, yet it 'follows the manifold'. Standard supervised kernel machines with the new kernel, trained on L only, are able to perform inductive semi-supervised learning. In fact they are equivalent to LapSVM and LapRLS (Belkin et al., 2005) with a certain parameter. Nonetheless finding the new kernel involves inverting a $n \times n$ matrix. Like many other methods it can be costly. Also notice the new kernel depends on the observed $L \cup U$ data, thus it is a random kernel.

6.1.7 Graph Kernels from the Spectrum of Laplacian

For kernel methods, the regularizer is a (typically monotonically increasing) function of the RKHS norm $||f||_K = f^\top K^{-1} f$ with kernel K. Such kernels are derived from the graph, e.g. the Laplacian.

Chapelle et al. (2002) and Smola and Kondor (2003) both show the spectral transformation of a Laplacian results in kernels suitable for semi-supervised learning. The diffusion kernel (Kondor & Lafferty, 2002) corresponds to a spectrum transform of the Laplacian with

$$r(\lambda) = \exp(-\frac{\sigma^2}{2}\lambda) \tag{12}$$

The regularized Gaussian process kernel $\Delta + I/\sigma^2$ in (Zhu et al., 2003c) corresponds to

$$r(\lambda) = \frac{1}{\lambda + \sigma} \tag{13}$$

Similarly the order constrained graph kernels in (Zhu et al., 2005) are constructed from the spectrum of the Laplacian, with non-parametric convex optimization. Learning the optimal eigenvalues for a graph kernel is in fact a way to (at least partially) improve an imperfect graph. In this sense it is related to graph construction.

Kapoor et al. (2005) learn both the graph weight hyperparameter, the hyperparameter for Laplacian spectrum transformation $r(\lambda) = \lambda + \delta$, and the noise model hyperparameter with evidence maximization. Expectation Propagation (EP) is used for approximation. The authors also propose a way to classify unseen points. This spectrum transformation is relatively simple.

6.1.8 Spectral Graph Transducer

The spectral graph transducer (Joachims, 2003) can be viewed with a loss function and regularizer

$$\min c(f-\gamma)^\top C(f-\gamma) + f^\top L f \tag{14}$$

$$\text{s.t.} f^\top 1 = 0 \text{and} f^\top f = n \tag{15}$$

where $\gamma_i = \sqrt{l_-/l_+}$ for positive labeled data, $-\sqrt{l_+/l_-}$ for negative data, l_- being the number of negative data and so on. L can be the combinatorial or normalized graph Laplacian, with a transformed spectrum. c is a weighting factor, and C is a diagonal matrix for misclassification costs.

Pham et al. (2005) perform empirical experiments on word sense disambiguation, comparing variants of co-training and spectral graph transducer. The authors notice spectral graph transducer with carefully constructed graphs ('SGT-Cotraining') produces good results.

6.1.9 Local Learning Regularization

The solution of graph-based methods can often be viewed as local averaging. For example, the harmonic function solution if we use an unnormalized Laplacian satisfies the averaging property:

$$f(x_i) = \frac{\sum_{j\sim i} w_{ij} f(x_j)}{\sum_{j\sim i} w_{ij}}. \tag{16}$$

In other words, the solution $f(x_i)$ at an unlabeled point x_i is the weighted average of its neighbors' solutions. Note the neighbors are usually unlabeled points too, so this is a self-consistent property. If we do not require $f(x_i)$ to equal the local average, but regularize f so they are close, we are using a regularizer of the special form $f^\top \Delta f$, where Δ is the unnormalized Laplacian.

A more general self-consistent property is obtained if one extends local averaging to a local linear fit. That is, one can build a local linear model from x_i's neighbors, and predict the value at x_i using this linear model. The solution $f(x_i)$ is regularized to be close to this predicted value. Note there will be n different linear models, one for each x_i. Wu and Schölkopf (2007) showed that such local linear model regularization can be written as $f^\top Rf$. The matrix R (which generalizes the Laplacian) has a special form which can be computed from X alone.

6.1.10 Tree-Based Bayes

Kemp et al. (2003) define a probabilistic distribution $P(Y|T)$ on discrete (e.g. 0 and 1) labellings Y over an evolutionary tree T. The tree T is constructed with the labeled and unlabeled data being the leaf nodes. The labeled data is clamped. The authors assume a mutation process, where a label at the root propagates down to the leaves. The label mutates with a constant rate as it moves down along the edges. As a result the tree T (its structure and edge lengths) uniquely defines the label prior $P(Y|T)$. Under the prior if two leaf nodes are closer in the tree, they have a higher probability of sharing the same label. One can also integrate over all tree structures.

The tree-based Bayes approach can be viewed as an interesting way to incorporate structure of the domain. Notice the leaf nodes of the tree are the labeled and unlabeled data, while the internal nodes do not correspond to physical data. This is in contrast with other graph-based methods where labeled and unlabeled data are all the nodes.

6.1.11 Some Other Methods

Szummer and Jaakkola (2001) perform a t-step Markov random walk on the graph. The influence of one example to another example is proportional to how easy the random walk goes from one to the other. It has certain resemblance to the diffusion kernel. The parameter t is important.

Chapelle and Zien (2005) use a density-sensitive connectivity distance between nodes i, j (a given path between i, j consists of several segments, one of them is the longest; now consider all paths between i, j and find the shortest 'longest segment'). Exponentiating the negative distance gives a graph kernel.

Bousquet et al. (2004) propose 'measure-based regularization', the continuous counterpart of graph-based regularization. The intuition is that two points are similar if they are connected by high density regions. They define regularization based on a known density $p(x)$ and provide interesting theoretical analysis. However it seems difficult in practice to apply the theoretical results to higher ($D > 2$) dimensional tasks.

6.2 Graph Construction

Although the graph is at the heart of graph-based semi-supervised learning methods, its construction has not been studied extensively. The issue has been discussed in (Zhu, 2005) Chapter 3 and Chapter 7. There are several distinct approaches:

1. Using domain knowledge. Balcan et al. (2005a) build graphs for video surveillance using strong domain knowledge, where the graph of webcam images consists of time edges, color edges and face edges. Such graphs reflect a deep understanding of the problem structure and how unlabeled data is expected to help.

2. Neighbor graphs. A few “standard” graphs are: kNN graph where each item is connected to its k-nearest-neighbor under some distance measure; ϵNN where connection happens within the radius ϵ; the edges can be weighted by a Gaussian function (a.k.a. heat kernel, RBF kernel) $w_{ij} = \exp(-\|x_i - x_j\}^2/\sigma^2)$, or unweighted (weight=1). Empirically, kNN weighted graph with small k tends to perform better.

   There are several tricks one can apply to such graphs. Carreira-Perpinan and Zemel (2005) build robust graphs from multiple minimum spanning trees by perturbation and edge removal. When using a Gaussian function as edge weights, the bandwidth of the Gaussian needs to be carefully chosen. Zhang and Lee (2006) derive a cross validation approach to tune the bandwidth for each feature dimension, by minimizing the leave-one-out mean squared error of predictions and given labels on labeled points. By invoking the matrix inversion lemma and careful pre-computation, the time complexity of LOO tuning is moderately reduced (but still at $O(u^3)$).

3. Local fit. Wang and Zhang (2006) perform an operation very similar to locally linear embedding (LLE) on the data points first, but constraining the LLE weights to be non-negative. These weights are then used as graph weights.

   Hein and Maier (2006) propose an algorithm to de-noise points sampled from a manifold. That is, data points are assumed to be noisy samples of some unknown underlying manifold. They used the de noising algorithm as a preprocessing step for graph-based semi-supervised learning, so that the graph can be constructed from better separated data points. Such preprocessing results in better semi-supervised classification accuracy.

6.3 Fast Computation

Many semi-supervised learning methods scale as badly as $O(n^3)$ as they were originally proposed. Because semi-supervised learning is interesting when the size of unlabeled data is large, this is clearly a problem. Many methods are also transductive (section 6.4). In 2005 several papers start to address these problems.

Fast computation of the harmonic function with conjugate gradient methods is discussed in (Argyriou, 2004). A comparison of three iterative methods: label propagation, conjugate gradient and loopy belief propagation is presented in (Zhu, 2005) Appendix F. Recently numerical methods for fast N-body problems have been applied to *dense* graphs in semi-supervised learning, reducing the computational cost from $O(n^3)$ to $O(n)$ (Mahdaviani et al., 2005). This is achieved with Krylov subspace methods and the fast Gauss transform.

The harmonic mixture models (Zhu & Lafferty, 2005) convert the original graph into a much smaller backbone graph, by using a mixture model to 'carve up' the original $L \cup U$ dataset. Learning on the smaller graph is much faster. Similar ideas have been used for e.g. dimensionality reduction (Teh & Roweis, 2002). The heuristics in (Delalleau et al., 2005) similarly create a small graph with a subset of the unlabeled data. They enables fast approximate computation by reducing the problem size.

Garcke and Griebel (2005) propose the use of sparse grids for semi-supervised learning. The main advantages are $O(n)$ computation complexity for sparse graphs, and the ability of induction. The authors start from the same regularization problem of (Belkin et al., 2005). The key idea is to approximate the function space with a finite basis, with sparse grids. The minimizer f in this finite dimensional subspace can be efficiently computed. As the authors point out, this method is different from the general kernel methods which rely on the representer theorem for finite representation. In practice the method is limited by data dimensionality (around 20). A potential drawback is that the method employs a regular grid, and cannot 'zoom in' to small interesting data regions with higher resolution.

Yu et al. (2005) solve the large scale semi-supervised learning problem by using a bipartite graph. The labeled and unlabeled points form one side of the bipartite split, while a much smaller number of 'block-level' nodes form the other side. The authors show that the harmonic function can be computed using the block-level nodes. The computation involves inverting a much smaller matrix on block-level nodes. It is thus cheaper and more scalable than working directly on the $L \cup U$ matrix. The authors propose two methods to construct the bipartite graph, so that it approximates the given weight matrix W on $L \cup U$. One uses Nonnegative Matrix Factorization, the other uses mixture models. The latter method has the additional benefit of induction, and is similar to the harmonic mixtures (Zhu &

Lafferty, 2005). However in the latter method the mixture model is derived based on the given weight matrix W. But in harmonic mixtures W and the mixture model are independent, and the mixture model serves as a 'second knowledge source' in addition to W.

The original manifold regularization framework (Belkin et al., 2004b) needs to invert a $(l+u) \times (l+u)$ matrix, and is not scalable. To speed up things, Sindhwani et al. (2005c) consider *linear manifold regularization*. Effectively this is a special case when the base kernel is taken to be the linear kernel. The authors show that it is advantageous to work with the primal variables. The resulting optimization problem can be much smaller if the data dimensionality is small, or sparse.

Tsang and Kwok (2006) scale manifold regularization up by adding in an ϵ-insensitive loss into the energy function, i.e. replacing $\sum w_{ij} \left(f(x_i) - f(x_j)\right)^2$ by $\sum w_{ij} \left(|f(x_i) - f(x_j)|_\epsilon\right)^2$, where $|z|_\epsilon = \max(|z| - \epsilon, 0)$. The intuition is that most pairwise differences $f(x_i) - f(x_j)$ are very small. By tolerating differences smaller than ϵ, the solution becomes sparse. They were able to handle one million unlabeled points in manifold regularization with this method.

6.4 Induction

Most graph-based semi-supervised learning algorithms are transductive, i.e. they cannot easily extend to new test points outside of $L \cup U$. Recently induction has received increasing attention. One common practice is to 'freeze' the graph on $L \cup U$. New points do not (although they should) alter the graph structure. This avoids expensive graph computation every time one encounters new points.

Zhu et al. (2003c) propose that new test point be classified by its nearest neighbor in $L \cup U$. This is sensible when U is sufficiently large. In (Chapelle et al., 2002) the authors approximate a new point by a linear combination of labeled and unlabeled points. Similarly in (Delalleau et al., 2005) the authors proposes an induction scheme to classify a new point x by

$$f(x) = \frac{\sum_{i \in L \cup U} w_{xi} f(x_i)}{\sum_{i \in L \cup U} w_{xi}} \tag{17}$$

This can be viewed as an application of the Nyström method (Fowlkes et al., 2004).

Yu et al. (2004) report an early attempt on semi-supervised induction using RBF basis functions in a regularization framework. In (Belkin et al., 2004b), the function f does not have to be restricted to the graph. The graph is merely used to regularize f which can have a much larger support. It is necessarily a combination of an inductive algorithm and graph regularization. The authors give the graph-regularized version of least squares and SVM. (Note such an SVM is different from the graph kernels in standard SVM in (Zhu et al., 2005). The former is inductive with both a graph regularizer and an inductive kernel. The latter is transductive with only the graph regularizer.) Following the work, Krishnapuram et al. (2005) use graph regularization on logistic regression. Sindhwani et al. (2005a) give a semi-supervised kernel that is defined over the whole space, not just on the training data points. These methods create inductive learners that naturally handle new test points.

The harmonic mixture model (Zhu & Lafferty, 2005) naturally handles new points as well. The idea is to model the labeled and unlabeled data with a mixture model, e.g. mixture of Gaussian. In standard mixture models, the class probability $p(y|i)$ for each mixture component i is optimized to maximize label likelihood. However in harmonic mixture models, $p(y|i)$ is optimized differently to minimize an underlying graph-based cost function. Under certain conditions, the harmonic mixture model converts the original graph on unlabeled data into a 'backbone graph', with the components being 'super nodes'. Harmonic mixture models naturally handle induction just like standard mixture models.

Several other inductive methods have been discussed in section 6.3 together with fast computation.

6.5 Consistency

The consistency of graph-based semi-supervised learning algorithms is an open research area. By consistency we mean whether classification converges to the right solution as the number of labeled and unlabeled data grows to infinity. Recently von Luxburg et al. (2005) (von Luxburg et al., 2004) study the consistency of *spectral clustering methods*. The authors find that the normalized Laplacian is better than the unnormalized Laplacian for spectral clustering. The convergence of the eigenvectors of the unnormalized Laplacian is not clear, while the normalized Laplacian always converges under general conditions. There are examples where the top eigenvectors of the unnormalized Laplacian do not yield a sensible clustering. The corresponding problem in semi-supervised classification needs further study. One reason is that in semi-supervised learning the whole Laplacian (normalized or not) is often used for regularization, not only the top eigenvectors.

Zhang and Ando (2006) prove that semi-supervised learning based on graph kernels is well-behaved in that the solution converges as the size of unlabeled data approaches infinity. They also derived a generalization bound, which leads to a way to optimizing kernel eigen-transformations.

6.6 Dissimilarity Edges, Directed Graphs, and Hypergraphs

So far a graph encodes label similarity. That is, two examples are connected if we prefer them to have the same label. Furthermore, if the edges are weighted, a larger weight means the two nodes are more likely to have the same label. The weights are always non-negative. However, sometimes we might also have *dissimilarity* information that two nodes should have different labels. In the general case, one can have both similarity and dissimilarity information on the same graph (e.g., "x_1 and x_2 should have the same label, while x_2 and x_3 should have different labels").

It is easy to see that simply encoding dissimilarity with negative edge weight is not appropriate: the energy function can become unbounded, and the objective becomes non-convex. Goldberg et al. (2007) defines a different graph energy function for dissimilarity edges. In particular, if x_i and x_j are dissimilar, one minimizes $w_{ij}(f(x_i) + f(x_j))^2$. Note the essential difference to similarity edges is the plus sign instead of minus sign, and w_{ij} stays non-negative. This forces $f(x_i)$ and $f(x_j)$ to have different signs and similar absolute values so they cancel each other out (the trivial solution of zeros is avoided by other similarity edges). The resulting energy function is still convex and can be easily solved using linear algebra. Tong and Jin (2007) adopt a different objective function as minimizing a ratio, which is solved by a semidefinite program.

Such similarity and dissimilarity edges are sometimes known as must-links and cannot-links in the context of semi-supervised clustering (or constrained clustering), which is discussed in Section 11.3.

For semi-supervised learning on directed graphs, Zhou et al. (2005b) take a hub - authority approach and essentially convert a directed graph into an undirected one. Two hub nodes are connected by an undirected edge with appropriate weight if they co-link to authority nodes, and vice versa. Semi-supervised learning then proceeds on the undirected graph.

Zhou et al. (2005a) generalize the work further. The algorithm takes a transition matrix (with a unique stationary distribution) as input, and gives a closed form solution on unlabeled data. The solution parallels and generalizes the normalized Laplacian solution for undirected graphs (Zhou et al., 2004a). The previous work (Zhou et al., 2005b) is a special case with the 2-step random walk transition matrix. In the absence of labels, the algorithm is the generalization of the normalized cut (Shi & Malik, 2000) on directed graphs.

Lu and Getoor (2003) convert the link structure in a directed graph into per-node features, and combines them with per-node object features in logistic regression. They also use an EM-like iterative algorithm.

Zhou et al. (2006a) propose to formulate relational objects using hypergraphs, where an edge can connect more than two vertices, and extend spectral clustering, classification and embedding to such hypergraphs.

6.7 Connection to Standard Graphical Models

The Gaussian random field formulation (Zhu et al., 2003a) is a standard undirected graphical model, with continuous random variables. Given labeled nodes (observed variables), the inference is used to obtain the mean (equivalently the mode) h_i of the remaining variables, which is the harmonic function. However the interpretation of the harmonic function as parameters for Bernoulli distributions at the nodes (i.e. each unlabeled node has label 1 with probability h_i, 0 otherwise) is non-standard.

Burges and Platt (2005) propose a *directed* graphical model, called Conditional Harmonic Mixing, that is somewhat between graph-based semi-supervised learning and standard Bayes nets. In standard Bayes nets there is one conditional probability table on each *node*, which looks at the values of all its parents and determines the distribution of the node. However in Conditional Harmonic Mixing there is one table on each *directed edge*. On one hand it is simpler because each table deals with only one parent node. On the other hand at the child node the estimated distributions from the parents may not be consistent, and the child takes the average distribution in KL divergence. Importantly the directed graph can contain loops, and there is always a unique global solution. It can be shown that the harmonic function can be interpreted as a special case of Conditional Harmonic Mixing.

7 Using Class Proportion Knowledge

It has long been noticed that constraining the class proportions on unlabeled data can be important for semi-supervised learning. By class proportion we refer to the proportion of instances classified into each class, e.g., 20% positive and 80% negative. Without any constrains on class proportion, various semi-supervised learning algorithms tend to produce unbalanced output. In the extreme case, all unlabeled data might be classified into one of the classes, which is undesirable.

For this reason, various semi-supervised learning methods have been using some form of class proportion constraints. The desired class proportions are either obtained as an input, which reflects domain knowledge, or estimated (by frequency or with smoothing) from the class proportions in the labeled dataset. For example, Zhu et al. (2003a) use a heuristic "class mean normalization" procedure to move towards the desired class proportions; S3VM methods explicitly fit the desired class proportions. In Joachims (1999); Chapelle et al. (2006b), it is a constraint on hard labels

$$\frac{1}{u}\sum_{i=l+1}^{l+u} y_i = \frac{1}{l}\sum_{i=1}^{l} y_i. \tag{18}$$

Note the y's on the left hand side are predicted labels, while on the right hand side are given constants. In Chapelle and Zien (2005), it is a constraint on continuous function predictions:

$$\frac{1}{u}\sum_{i=l+1}^{l+u} f(x_i) = \frac{1}{l}\sum_{i=1}^{l} y_i. \tag{19}$$

However, in these methods the class proportion constraint is combined with other model assumptions, e.g., label smoothness on a graph, or large separation in unlabeled data regions. Mann and McCallum (2007) show that class proportion by itself can be a useful regularizer for semi-supervised learning. Let $\tilde{p}$ be the multinomial distribution of desired class proportion, and $\tilde{p}_\theta$ be the class proportion produced by the current model θ. Note the latter is computed on unlabeled data. Mann and McCallum add the KL-divergence $KL(\tilde{p}\|\tilde{p}_\theta)$ as a regularizer to logistic regression,

$$\min_\theta -\sum_{i=1}^{l} \log p_\theta(y_i|x_i) + \lambda KL(\tilde{p}\|\tilde{p}_\theta). \tag{20}$$

The objective is reported to be non-convex. A gradient method is used for optimization. Results on several natural language processing tasks are good, and one advantage of this approach is its efficiency.

8 Learning Efficient Encoding of the Domain from Unlabeled Data

One can use unlabeled data to learn an efficient feature encoding of the problem domain. The labeled data is then represented using this new feature, and classification is done via standard supervised learning. The idea has been implicit in several works, e.g., kernel learning from graph Laplacian on labeled and unlabeled data (Zhu et al., 2005). One can also perform PCA on the unlabeled data, and use the resulting low dimensional representation

Ando and Zhang (2005); Johnson and Zhang (2007) build on a two-view feature generation framework, where the input features form two subsets with a feature split $x = (z_1, z_2)$. It is assumed that the two views are conditionally independent given class label y:

$$p(z_1, z_2|y) = p(z_1|y)p(z_2|y). \tag{21}$$

Unlike co-training, the views are not assumed to be individually sufficient for classification. The novelty lies in the definition of a large number of *auxiliary problems*. These are artificial classification tasks, using one view z_2 to predict some function of the other view $t_m(z_1)$, where m indices different auxiliary problems. Note the auxiliary problems can be defined and trained on unlabeled data. In particular, one can define a linear model ${w_m}^\top z_2$ to fit $t_m(z_1)$, and learn the weight w_m using all unlabeled data. The weight vector w_m has the same dimension as z_2. With auxiliary functions that reflect typical classification goals in the problem domain, one can imagine that some dimensions in the set of weights $\{w_1, \ldots, w_m, \ldots\}$ are more important, indicating the corresponding dimensions in z_2 are more useful. These dimensions (or a linear combination) can be compactly extracted by a Singular Value Decomposition on the matrix constructed from the weights, and act as a new and shorter representation of z_2. Similarly, z_1 has a new representation by exchanging the role of z_1 and z_2. Finally, the original representation (z_1, z_2) and the new representations of z_1 and z_2 are concatenated as the new representation of the instance x. This new representation contains the information of unlabeled data and auxiliary problems. One then perform standard supervised learning with labeled data using the new representation. The choice of auxiliary problems are of great importance to the success of semi-supervised learning in this setting.

Raina et al. (2007) consider the case when the unlabeled data *does not necessarily come from the classes to be classified*. For example, in an image categorization task the two classes can be elephants and rhinos, while the unlabeled data can be any natural scenes. The paper proposes a “self-taught learning” algorithm, in which the unlabeled data is used to learn a higher level representation that is tuned for the problem domain. For example, if the images were originally represented by pixels, the higher level representation might be small patches that correspond to certain semantics (e.g., edges). In particular, the algorithm finds a set of basis b, and each instance is a sparse weighted combination of bases, with weights a. The a, b are learned by the following optimization problem:

$$\min_{a,b} \quad \sum_{i=l+1}^{l+u} \|x_i - \sum_j a_{ij} b_j\|^2 + \beta \sum_j |a_{ij}| \tag{22}$$

$$s.t. \quad \|b_j\|_2 \leq 1, \forall j \tag{23}$$

It is reported that the sparsity is important for self-taught learning. Once the bases are learned, the labeled instances are represented by their weights on the bases. A supervised algorithm is then applied to the labeled data in this new representation.

9 Computational Learning Theory

In this survey we have primarily focused on various semi-supervised learning algorithms. The theory of semi-supervised learning has been touched upon occasionally in the literature. However it was not until recently that the computational learning theory community began to pay more attention to this interesting problem.

Leskes (2005) presents a generalization error bound for semi-supervised learning with multiple learners, an extension to co-training. The author shows that if multiple learning algorithms are forced to produce similar hypotheses (i.e. to agree) given the same training set, and such hypotheses still have low training error, then the generalization error bound is tighter. The unlabeled data is used to assess the agreement among hypotheses. The author proposes a new Agreement-Boost algorithm to implement the procedure.

Kaariainen (2005) presents another generalization error bound for semi-supervised learning. The idea is that the target function is in the version space. If a hypothesis is in the version space (revealed by labeled data), and is close to all other hypotheses in the version space (revealed by unlabeled data), then it has to be close to the target function. Closeness is defined as classification agreement, and can be approximated using unlabeled data. This idea builds on metric-based model selection (Section 11.9).

Balcan and Blum (2005) propose a PAC-style model for semi-supervised learning. This is the first PAC model that explains when unlabeled data might help (notice the classic PAC model cannot incorporate unlabeled data at all). There has been previous *particular* analysis for explaining when unlabeled data helps, but they were all based on specific settings and assumptions. In contrast this PAC model is a general, unifying model. The authors define an interesting quantity: the compatibility of a hypothesis w.r.t. the unlabeled data distribution. For example in SVM a hyperplane that cuts through high density regions would have low compatibility, while one that goes along gaps would have high compatibility. We note that the compatibility function can be defined much more generally. The intuition of the results is the following. Assuming a-priori that the target function has high compatibility with unlabeled data. Then if a hypothesis has zero training error (standard PAC style) *and* high compatibility, the theory gives the number of labeled and unlabeled data to guarantee the hypothesis is good. The number of labeled data needed can be quite small.

10 Semi-supervised Learning in Structured Output Spaces

In most of this paper we consider classification on individual instances. In this section we discuss semi-supervised learning in structured output spaces, e.g. for sequences and trees.

10.1 Generative Models

One example of generative models for semi-supervised sequence learning is the Hidden Markov Model (HMM), in particular the Baum-Welsh HMM training algorithm (Rabiner, 1989). It is essentially the sequence version of the EM algorithm on mixture models as mentioned in section 2. Baum-Welsh algorithm has a long history, well before the recent emergence of interest on semi-supervised learning. It has been successfully applied to many areas including speech recognition. It is usually not presented as a semi-supervised learning algorithm, but certainly qualifies as one. Some cautionary notes can be found in (Elworthy, 1994).

10.2 Graph-based Kernels

Many existing structured learning algorithms (e.g. conditional random fields, maximum margin Markov networks) can be endowed with a 'semi-supervised' kernel. Take the example of learning on sequences. One first creates a graph kernel on the union of all elements in the sequences (i.e. ignoring the sequence structure, treating the elements of a sequence as if they were individual instances). The graph kernel can be constructed with any of the above methods. Next one applies the graph kernel to a standard structured learning kernel machine. Such kernel machines include the kernelized conditional random fields (Lafferty et al., 2004) and maximum margin Markov networks (Taskar et al., 2003), which differ primarily by the loss function they use.

With a graph kernel the kernel machine thus perform semi-supervised learning on structured data. Lafferty et al. (2004) hinted this idea and tested it on a bioinformatics dataset. The graph kernel matrix they used is transductive in nature, which is defined only on elements in the training data. Altun et al. (2005) defines a graph kernel over the whole space by linearly combining the norms of a standard kernel and a graph regularization term, resulting in a nonlinear graph kernel similar to Sindhwani et al. (2005a). They use the kernel with a margin loss. Brefeld and Scheffer (2006) extend structured SVM with a multi-view regularizer, which penalizes disagreements between classifications on unlabeled data, where the classifiers operate on different feature subsets.

11 Related Areas

The focus of the survey is on classification with semi-supervised methods. There are some closely related areas with a rich literature.

11.1 Spectral Clustering

Spectral clustering is unsupervised. As such there is no labeled data to guide the process. Instead the clustering depends solely on the graph weights W. On the other hand semi-supervised learning for classification has to maintain a balance between how good the 'clustering' is, and how well the labeled data can be explained by it. Such balance is expressed explicitly in the regularization framework.

As we have seen in section 8.1 of (Zhu, 2005) and section 6.5 here, the top eigenvectors of the graph Laplacian can unfold the data manifold to form meaningful clusters. This is the intuition behind spectral clustering. There are several criteria on what constitutes a good clustering (Weiss, 1999).

The normalized cut (Shi & Malik, 2000) seeks to minimize

$$Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)} \tag{24}$$

The *continuous relaxation* of the cluster indicator vector can be derived from the normalized Laplacian. In fact it is derived from the second smallest eigenvector of the normalized Laplacian. The continuous vector is then discretized to obtain the clusters. De Bie and Cristianini (2006a) present an SDP relaxation of the normalized graph cut problem, including a series of relaxations between spectral relaxations above, and SDP relaxation. The SDP formulation can easily include partial label or constraint information, and therefore applicable for transductive classification.

The data points are mapped into a new space spanned by the first k eigenvectors of the normalized Laplacian in (Ng et al., 2001), with special normalization. Clustering is then performed with traditional methods (like k-means) in this new space. This is very similar to kernel PCA.

Fowlkes et al. (2004) use the Nyström method to reduce the computation cost for large spectral clustering problems. This is related to the method in (Zhu, 2005) Chapter 10.

Chung (1997) presents the mathematical details of spectral graph theory.

11.2 Learning with Positive and Unlabeled Data

In many real world applications, labeled data may be available from only one of the two classes. Then there is the unlabeled data, known to contain both classes.

There are two ways to formulate the problem: classification or ranking.

Classification Here one builds a classifier even though there is no negative example. It is important to note that with the positive training data one can estimate the positive class conditional probability $p(x|+)$, and with the unlabeled data one can estimate $p(x)$. If the prior $p(+)$ is known or estimated from other sources, one can derive the negative class conditional as

$$p(x|-) = \frac{p(x) - p(+)p(x|+)}{1 - p(+)} \tag{25}$$

With $p(x|-)$ one can then perform classification with Bayes rule. Denis et al. (2002) use this fact for text classification with Naive Bayes models.

Another set of methods heuristically identify some 'reliable' negative examples in the unlabeled set, and use EM on generative (Naive Bayes) models (Liu et al., 2002) or logistic regression (Lee & Liu, 2003).

Ranking Given a large collection of items, and a few 'query' items, ranking orders the items according to their similarity to the queries. Information retrieval is the standard technique under this setting, and we will not attempt to include the extensive literatures on this mature field. It is worth pointing out that graph-based semi-supervised learning can be modified for such settings. Zhou et al. (2004b) treat it as semi-supervised learning with positive data on a graph, where the graph induces a similarity measure, and the queries are positive examples. Data points are ranked according to their graph similarity to the positive training set.

11.3 Semi-supervised Clustering

Also known as constrained clustering or clustering with side information, this is the cousin of semi-supervised classification. The goal is clustering but there are some 'labeled data' in the form of *must-links* (two points must in the same cluster) and *cannot-links* (two points cannot in the same cluster). There is a tension between satisfying these constraints and optimizing the original clustering criterion (e.g. minimizing the sum of squared distances within clusters). Procedurally one can modify the distance metric to try to accommodate the constraints, or one can bias the search. We refer readers to a recent short survey (Grira et al., 2004) for the literatures.

11.4 Semi-supervised Regression

In principle all graph-based semi-supervised classification methods in section 6 are indeed function estimators. That is, they estimate 'soft labels' before making a classification. The function tries to be close to the targets y in the labeled set, and at the same time be smooth on the graph. Therefore these graph-based semi-supervised methods can also naturally perform regression. Some of the methods can be thought of as Gaussian processes with a special kernel that is constructed from unlabeled data.

Zhou and Li (2005a) proposed using co-training for semi-supervised regression. The paper used two kNN regressors, each with a different p-norm as distance measure. Like in co-training, each regressor makes prediction on unlabeled data, and the most confident predictions are used to train the other regressor. The confidence of a prediction on unlabeled point is measured by the MSE on labeled set before and after adding this prediction as training data to the current regressor. Similarly Sindhwani et al. (2005b); Brefeld et al. (2006) perform multi-view regression, where a regularization term depends on the disagreement among regressors on different views.

Cortes and Mohri (2006) propose a simple yet efficient transductive regression model. On top of a standard ridge regression model, an addition term is applied to each unlabeled point x_u. This additional regularization term makes the prediction $f(x_u)$ close to a heuristic prediction y_u^*, which is computed by a weighted average of the labels of labeled points in a neighborhood of x_u. A generalization error bound is also given.

11.5 Active Learning and Semi-supervised Learning

Active learning and semi-supervised learning face the same issue, i.e. that labeled data is scarce and hard to obtain. It is quite natural to combine active learning and semi-supervised learning to address this issue from both ends.

McCallum and Nigam (1998b) use EM with unlabeled data integrated into the active learning algorithm. Muslea et al. (2002) propose CO-EMT which combines multi-view (e.g. co-training) learning with active learning. Zhou et al. (2004c); Zhou et al. (2006b) apply semi-supervised learning together with active learning to content-based image retrieval.

Many active learning algorithms naively select as query the point with maximum label ambiguity (entropy), or least confidence, or maximum disagreement between multiple learners. Zhu et al. (2003b) show that these are not necessarily the right things to do, if one is interested in classification error. They show that one can select active learning queries that minimize the (estimated) generalization error, in a graph-based semi-supervised learning framework.

11.6 Nonlinear Dimensionality Reduction

The goal of nonlinear dimensionality reduction is to find a faithful low dimensional mapping of the high dimensional data. As such it belongs to unsupervised learning. However the way it discovers low dimensional manifold within a high dimensional space is closely related to spectral graph semi-supervised learning. Representative methods include Isomap (Tenenbaum et al., 2000), locally linear embedding (LLE) (Roweis & Saul, 2000) (Saul & Roweis, 2003), Hessian LLE (Donoho & Grimes, 2003), Laplacian eigenmaps (Belkin & Niyogi, 2003), and semidefinite embedding (SDE) (Weinberger & Saul, 2004) (Weinberger et al., 2004) (Weinberger et al., 2005).

If one has some labeled data, for example in the form of the target low-dimensional representation for a few data points, the dimensionality reduction problem becomes semi-supervised. One approach for this setting is presented in (Yang et al., 2006).

11.7 Learning a Distance Metric

Many learning algorithms depend, either explicitly or implicitly, on a distance metric on X. We use the term metric here loosely to mean a measure of distance or (dis)similarity between two data points. The default distance in the feature space may not be optimal, especially when the data forms a lower dimensional manifold in the feature vector space. With a large amount of U, it is possible to detect such manifold structure and its associated metric. The graph-based methods above are based on this principle. We review some other methods next.

The simplest example in text classification might be Latent Semantic Indexing (LSI, a.k.a. Latent Semantic Analysis LSA, Principal Component Analysis PCA, or sometimes Singular Value Decomposition SVD). This technique defines a linear subspace, such that the variance of the data, when projected to the subspace, is maximumly preserved. LSI is widely used in text classification, where the original space for X is usually tens of thousands dimensional, while people believe meaningful text documents reside in a much lower dimensional space. Zelikovitz and Hirsh (2001) and Cristianini et al. (2001) both use U, in this case unlabeled documents, to augment the term-by-document matrix of L. LSI is performed on the augmented matrix. This representation induces a new distance metric. By the property of LSI, words that co-occur very often in the same documents are merged into a single dimension of the new space. In the extreme this allows two documents with no common words to be 'close' to each other, via chains of co-occur word pairs in other documents.

Oliveira et al. (2005) propose a simple procedure for semi-supervised learning: First one runs PCA on $L \cup U$ (ignoring the labels). The result is a linear subspace that is constructed with more data points if one uses only L in PCA. In the next step, only L is mapped onto the subspace, and an SVM is learned. The method is useful when class separation is linear and along the principal component directions, and unlabeled helps by reducing the variance in estimating such directions.

Probabilistic Latent Semantic Analysis (PLSA) (Hofmann, 1999) is an important improvement over LSI. Each word in a document is generated by a 'topic' (a multinomial, i.e. unigram). Different words in the document may be generated by different topics. Each document in turn has a fixed topic proportion (a multinomial on a higher level). However there is no link between the topic proportions in different documents.

Latent Dirichlet Allocation (LDA) (Blei et al., 2003) is one step further. It assumes the topic proportion of each document is drawn from a Dirichlet distribution. With variational approximation, each document is represented by a posterior Dirichlet over the topics. This is a much lower dimensional representation. Griffiths et al. (2005) extend LDA model to 'HMM-LDA' which uses both short-term syntactic and long-term topical dependencies, as an effort to integrate semantics and syntax. Li and McCallum (2005) apply the HMM-LDA model to obtain word clusters, as a rudimentary way for semi-supervised learning on sequences.

Some algorithms derive a metric entirely from the density of U. These are motivated by unsupervised clustering and based on the intuition that data points in the same high density 'clump' should be close in the new metric. For instance, if U is generated from a single Gaussian, then the Mahalanobis distance induced by the covariance matrix is such a metric. Tipping (1999) generalizes the Mahalanobis distance by fitting U with a mixture of Gaussian, and define a Riemannian manifold with metric at x being the weighted average of individual component inverse covariance. The distance between x_1 and x_2 is computed along the straight line (in Euclidean space) between the two points. Rattray (2000) further generalizes the metric so that it only depends on the change in log probabilities of the density, not on a particular Gaussian mixture assumption. And the distance is computed along a curve that minimizes the distance. The new metric is invariant to linear transformation of the features, and connected regions of relatively homogeneous density in U will be close to each other. Such metric is attractive, yet it depends on the homogeneity of the initial Euclidean space. Their application in semi-supervised learning needs further investigation. Sajama and Orlitsky (2005) analyze the lower and upper bounds on estimating data-density-based distance. There are two sources of error: one stems from the fact that the true density $p(x)$ is not known, the second is that for practical reasons one typically build a grid on the data points, instead of a regular grid in R^d. The authors separate these two kinds of errors (computational and estimation), and analyze them independently. It sheds light on the complexity of density-based distance, independent of the specific method one uses. It also sheds some light on approximation errors when using neighborhood graphs on data points, which is used widely in semi-supervised learning and non-linear dimensionality reduction, etc. Understanding this dichotomy is helpful when trying to improve methods for semi-supervised learning.

We caution the reader that the metrics proposed above are based on unsupervised techniques. They all identify a lower dimensional manifold within which the data reside. However the data manifold may or may not correlate with a particular classification task. For example, in LSI the new metric emphasizes words with prominent count variances, but ignores words with small variances. If the classification task is subtle and depends on a few words with small counts, LSI might wipe out the salient words all together. Therefore the success of these methods is hard to guarantee without putting some restrictions on the kind of classification tasks. It would be interesting to include L into the metric learning process.

In a separate line of work, Baxter (1997) proves that there is a unique optimal metric for classification if we use 1-nearest-neighbor. The metric, named Canonical Distortion Measure (CDM), defines a distance $d(x_1, x_2)$ as the expected loss if we classify x_1 with x_2's label. The distance measure proposed in (Yianilos, 1995) can be viewed as a special case. Yianilos assume a Gaussian mixture model has been learned from U, such that a class correspond to a component, but the correspondence is unknown. In this case CDM $d(x_1, x_2) = p(x_1, x_2 \text{from same component})$ and can be computed analytically. Now that a metric has been learned from U, we can find within L the 1-nearest-neighbor of a new data point x, and classify x with the nearest neighbor's label. It will be interesting to compare this scheme with EM based semi-supervised learning, where L is used to label mixture components.

Weston et al. (2004) propose the neighborhood mismatch kernel and the bagged mismatch kernel. More precisely both are *kernel transformation* that modifies an input kernel. In the neighborhood method, one defines the neighborhood of a point as points close enough according to certain similarity measure (note this is *not* the measure induced by the input kernel). The output kernel between point i, j is the average of pairwise kernel entries between i's neighbors and j's neighbors. In bagged method, if a clustering algorithm thinks they tend to be in the same cluster (note again this is a different measure than the input kernel), the corresponding entry in the input kernel is boosted.

11.8 Inferring Label Sampling Mechanisms

Most semi-supervised learning methods assume L and U are both *i.i.d.* from the underlying distribution. However as (Rosset et al., 2005) points out that is not always the case. For example y can be the binary label whether a customer is satisfied, obtained through a survey. It is conceivable survey participation (and thus labeled data) depends on the satisfaction y.

Let s_i be the binary missing indicator for y_i. The authors model $p(s|x, y)$ with a parametric family. The goal is to estimate $p(s|x, y)$ which is the label sampling mechanism. This is done by computing the expectation of an arbitrary function $g(x)$ in two ways: on $L \cup U$ as $1/n \sum_{i=1}^{n} g(x_i)$, and on L only as $1/n \sum_{i \in L} g(x_i)/p(s_i = 1|x_i, y_i)$. By equating the two $p(s|x, y)$ can be estimated. The intuition is that the expectation on L requires weighting the labeled samples inversely proportional to the labeling probability, to compensate for ignoring the unlabeled data.

11.9 Metric-Based Model Selection

Metric-based model selection (Schuurmans & Southey, 2001) is a method to detect hypotheses inconsistency with unlabeled data. We may have two hypotheses which are consistent on L, for example they all have zero training set error. However they may be inconsistent on the much larger U. If so we should reject at least one of them, e.g. the more complex one if we employ Occam's razor.

The key observation is that a distance metric is defined in the hypothesis space H. One such metric is the number of different classifications two hypotheses make under the data distribution $p(x)$: $d_p(h_1, h_2) = E_p[h_1(x) \neq h_2(x)]$. It is easy to verify that the metric satisfies the three metric properties. Now consider the true classification function h^* and two hypotheses h_1, h_2. Since the metric satisfies the triangle inequality (the third property), we have

$$d_p(h_1, h_2) \leq d_p(h_1, h^*) + d_p(h^*, h_2)$$

Under the premise that labels in L is noiseless, let's assume we can approximate $d_p(h_1, h^*)$ and $d_p(h^*, h_2)$ by h_1 and h_2's training set error rates $d_L(h_1, h^*)$ and $d_L(h_2, h^*)$, and approximate $d_p(h_1, h_2)$ by the difference h_1 and h_2 make on a large amount of unlabeled data U: $d_U(h_1, h_2)$. We get

$$d_U(h_1, h_2) \leq d_L(h_1, h^*) + d_L(h^*, h_2)$$

which can be verified directly. If the inequality does not hold, at least one of the assumptions is wrong. If $|U|$ is large enough and $U \overset{\text{iid}}{\sim} p(x)$, $d_U(h_1, h_2)$ will be a good estimate of $d_p(h_1, h_2)$. This leaves us with the conclusion that at least one of the training errors does not reflect its true error. If both training errors are close to zero, we would know that at least one model is overfitting. An Occam's razor type of argument then can be used to select the model with less complexity. Such use of unlabeled data is very general and can be applied to almost any learning algorithms. However it only selects among hypotheses; it does not generate new hypothesis based on unlabeled data.

The co-validation method (Madani et al., 2005) also uses unlabeled data for model selection and active learning. Kaariainen (2005) uses the metric to derive a generalization error bound, see Section 9.

11.10 Multi-Instance Learning

In multi-instance learning the training set consists of labeled bags, each consisting of many unlabeled instances. A bag is positively labeled if it contains at least one positive instance, and negatively labeled if all instances in it are negative. Zhou and Xu (2007) show that under the i.i.d. instance assumption, multi-instance learning is a special case of semi-supervised learning, and can be solved with a special semi-supervised support vector machine (MissSVM).

12 Scalability Issues of Semi-Supervised Learning Methods

Current semi-supervised learning methods have not yet handled large amount of data. The complexity of many elegant graph-based methods is close to $O(n^3)$. Speed-up improvements have been proposed (Mahdaviani et al. 2005; Delalleau et al. 2005; Zhu and Lafferty 2005; Yu et al. 2005; Garcke and Griebel 2005; and more), their effectiveness has yet to be proven on real large problems. Some of them are discussed in Section 6.3.

Figure 7 compares the experimental dataset sizes in many representative semi-supervised learning papers. The unlabeled dataset size in these papers are evidently not large. Ironically huge amount of unlabeled data should have been the optimal operation environment for semi-supervised learning. More research efforts are needed to address the scalability issue.

Recent advances include (Sindhwani & Keerthi, 2006) and (Tsang & Kwok, 2006).

13 Do Humans do Semi-Supervised Learning?

Now let us turn our attention from *machine* learning to *human* learning. It is possible that understanding of the human cognitive model will lead to novel machine learning approaches (Langley, 2006; Mitchell, 2006). We ask the question: Do humans do semi-supervised learning? My hypothesis is yes. We humans accumulate ‘unlabeled’ input data, which we use (often unconsciously) to help building

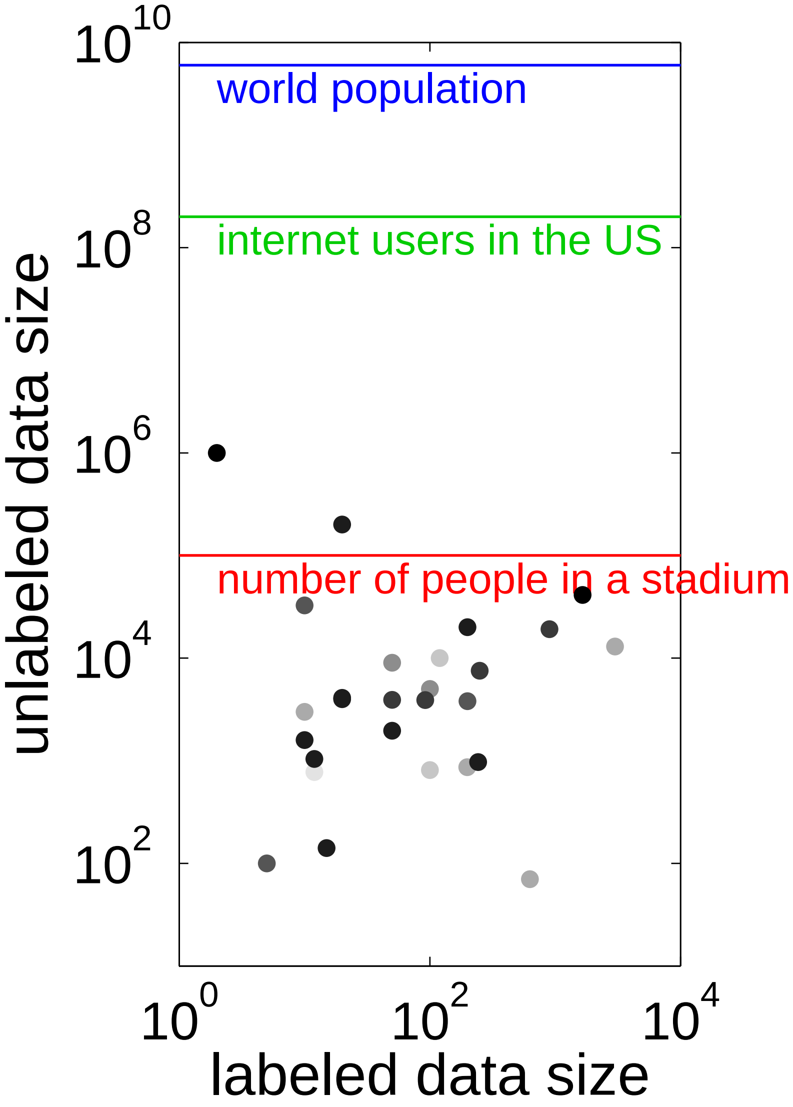


Figure 7: As recently as 2005, semi-supervised learning methods have not addressed large-scale problems. Shown above are the largest dataset size (labeled and unlabeled portion respectively) used in representative semi-supervised learning papers. Each dot is a paper, with darkness indicating publication year (darkest: 2005, lightest: 1998). Most papers only used hundreds of labeled points and tens of thousands of unlabeled points. Also shown are some interesting large numbers for comparison. Note the log-log scale.

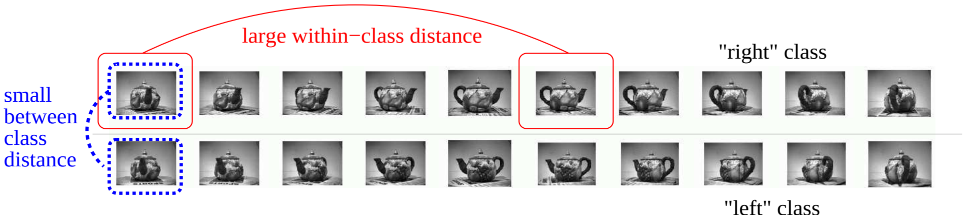


Figure 8: Classify teapot images by its spout orientation. Some images within the same class are quite different, while some images from different classes are similar.

the connection between 'labels' and input once labeled data is provided. I present some evidence below.

13.1 Visual Object Recognition with Temporal Association

The appearance of an object usually changes greatly when viewed from different angles. In the case of faces, the difference between the same face from two view points can be much larger than the difference between two faces from the same angle. Human observers nonetheless can connect the correct faces. It has been suggested that temporal correlation serves as the glue, as summarized by (Sinha et al., 2006) (Result 14). It seems when we observe an object with changing angles, we link the images as 'containing the same object' by the virtue that the images are close in time. Wallis and Bülthoff (2001) created artificial image sequences where a frontal face is morphed into the profile face of a different person. When observers are shown such sequences during training, their ability to match frontal and profile faces was impaired during test, due to the wrong links. The authors further argue that the object has to have similar location in the images to establish the link.

The idea of spatio-temporal link is directly related to graph-based semi-supervised learning. Consider the Teapot dataset used in (Zhu & Lafferty, 2005) (originally from (Weinberger et al., 2004)), with images of a teapot viewed from different angles. Now suppose we want to classify an image by whether its spout points to the left or right. As Figure 8 shows there are large within-class distances and small between-class distances. However the similarity between adjacent images (which comes from temporal relation) allow a graph to be constructed for semi-supervised learning. In another work, Balcan et al. (2005a) construct a graph on webcam images using temporal links (as well as color, face similarity links) for semi-supervised learning.

13.2 Infant Word-Meaning Mapping

17-month old infants were shown to be able to associate a word with a visual object better if they have heard the word many times before (Graf Estes et al., 2006). If the word was not heard before, the infant's ability to associate it with the object was weaker. If we view the sound of the word as unlabeled data, and the object as the label, we can propose a model where an infant builds up clusters of familiar-sounding words, which are easily labeled as a whole. This is similar to semi-supervised learning with mixture models (Nigam et al., 2000) or clusters (Dara et al., 2002; Demiriz et al., 1999).

13.3 Human Categorization Experiments

Perhaps the first attempt to observe semi-supervised learning in humans is described in (Stromsten, 2002) (Chapter 3), who uses drawings of artificial fish to show that human categorization behavior can be influenced by the presence of unlabeled examples. However, the experiment uses a single positive labeled example and no negative labeled examples, making it a one-class setting similar to novelty detection or quantile estimation instead of binary classification. In addition, the fish stimulus is a familiar real-world concept which might induce prior bias.

Zhu et al. (2007) show that human binary classification behavior conforms well to a generative model (Gaussian Mixture Models) for semi-supervised learning. In particular, they set up the data such that the decision boundaries derived from labeled data only vs. labeled and unlabeled data are different under the semi-supervised machine learning model. They then observe similar decision boundary differences in a human behavioral experiment. The stimuli are novel 3-D shapes which do not correspond to real-world objects, thus avoiding prior bias. They also observe that people's reaction time (time between a stimulus is display and a key is pressed to classify it) peaks around the decision boundary, and the reaction time peak also changes accordingly with and without unlabeled data. The Gaussian mixture model, trained with the EM algorithm, fits the human behavior nicely both in terms of classification and reaction time.

Acknowledgment

I thank John Lafferty, Zoubin Ghahramani, Tommi Jaakkola, Ronald Rosenfeld, Maria Florina Balcan, Kai Yu, Sajama, Matthias Seeger, Yunpeng Xu, Olivier Chapelle, Zhi-Hua Zhou, and all other colleagues who discussed the literature with me.

References

Altun, Y., McAllester, D., & Belkin, M. (2005). Maximum margin semi-supervised learning for structured variables. *Advances in Neural Information Processing Systems (NIPS) 18*.

Ando, R., & Zhang, T. (2005). A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, *6*, 1817–1853.

Argyriou, A. (2004). Efficient approximation methods for harmonic semi-supervised learning. Master's thesis, University College London.

Balcan, M.-F., & Blum, A. (2005). A PAC-style model for learning from labeled and unlabeled data. *COLT 2005*.

Balcan, M.-F., & Blum, A. (2006). An augmented pac model for semi-supervised learning. In O. Chapelle, B. Schölkopf and A. Zien (Eds.), *Semi-supervised learning*. MIT Press.

Balcan, M.-F., Blum, A., Choi, P. P., Lafferty, J., Pantano, B., Rwebangira, M. R., & Zhu, X. (2005a). Person identification in webcam images: An application of semi-supervised learning. *ICML 2005 Workshop on Learning with Partially Classified Training Data*.

Balcan, M.-F., Blum, A., & Yang, K. (2005b). Co-training and expansion: Towards bridging theory and practice. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Baluja, S. (1998). Probabilistic modeling for face orientation discrimination: Learning from labeled and unlabeled data. *Neural Information Processing Systems*.

Baxter, J. (1997). The canonical distortion measure for vector quantization and function approximation. *Proc. 14th International Conference on Machine Learning* (pp. 39–47). Morgan Kaufmann.

Belkin, M., Matveeva, I., & Niyogi, P. (2004a). Regularization and semi-supervised learning on large graphs. *COLT*.

Belkin, M., & Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, *15*, 1373–1396.

Belkin, M., Niyogi, P., & Sindhwani, V. (2004b). *Manifold regularization: A geometric framework for learning from examples* (Technical Report TR-2004-06). University of Chicago.

Belkin, M., Niyogi, P., & Sindhwani, V. (2005). On manifold regularization. *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTAT 2005)*.

Bennett, K., & Demiriz, A. (1999). Semi-supervised support vector machines. *Advances in Neural Information Processing Systems*, *11*, 368–374.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Blum, A., & Chawla, S. (2001). Learning from labeled and unlabeled data using graph mincuts. *Proc. 18th International Conf. on Machine Learning*.

Blum, A., Lafferty, J., Rwebangira, M., & Reddy, R. (2004). Semi-supervised learning using randomized mincuts. *ICML-04, 21st International Conference on Machine Learning*.

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. *COLT: Proceedings of the Workshop on Computational Learning Theory*.

Bousquet, O., Chapelle, O., & Hein, M. (2004). Measure based regularization. *Advances in Neural Information Processing Systems 16.*.

Brefeld, U., Büscher, C., & Scheffer, T. (2005). Multiview discriminative sequential learning. *European Conference on Machine Learning (ECML)*.

Brefeld, U., Gaertner, T., Scheffer, T., & Wrobel, S. (2006). Efficient co-regularized least squares regression. *ICML06, 23rd International Conference on Machine Learning*. Pittsburgh, USA.

Brefeld, U., & Scheffer, T. (2006). Semi-supervised learning for structured output variables. *ICML06, 23rd International Conference on Machine Learning*. Pittsburgh, USA.

Burges, C. J., & Platt, J. C. (2005). Semi-supervised learning with conditional harmonic mixing. In O. Chapelle, B. Schölkopf and A. Zien (Eds.), *Semi-supervised learning*. Cambridge, MA: MIT Press.

Callison-Burch, C., Talbot, D., & Osborne, M. (2004). Statistical machine translation with word- and sentence-aligned parallel corpora. *Proceedings of the ACL*.

Carreira-Perpinan, M. A., & Zemel, R. S. (2005). Proximity graphs for clustering and manifold learning. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Castelli, V., & Cover, T. (1995). The exponential value of labeled samples. *Pattern Recognition Letters, 16*, 105–111.

Castelli, V., & Cover, T. (1996). The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. *IEEE Transactions on Information Theory, 42*, 2101–2117.

Chapelle, O., Chi, M., & Zien, A. (2006a). A continuation method for semi-supervised SVMs. *ICML06, 23rd International Conference on Machine Learning*. Pittsburgh, USA.

Chapelle, O., Sindhwani, V., & Keerthi, S. S. (2006b). Branch and bound for semi-supervised support vector machines. *Advances in Neural Information Processing Systems (NIPS)*.

Chapelle, O., Weston, J., & Schölkopf, B. (2002). Cluster kernels for semi-supervised learning. *Advances in Neural Information Processing Systems, 15*.

Chapelle, O., & Zien, A. (2005). Semi-supervised classification by low density separation. *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTAT 2005)*.

Chapelle, O., Zien, A., & Schölkopf, B. (Eds.). (2006c). *Semi-supervised learning*. MIT Press.

Chawla, N. V., & Karakoulas, G. (2005). Learning from labeled and unlabeled data: An empirical study across techniques and domains. *Journal of Artificial Intelligence Research, 23*, 331–366.

Chu, W., & Ghahramani, Z. (2004). *Gaussian processes for ordinal regression* (Technical Report). University College London.

Chu, W., Sindhwani, V., Ghahramani, Z., & Keerthi, S. S. (2006). Relational learning with gaussian processes. *Advances in NIPS*.

Chung, F. R. K. (1997). *Spectral graph theory, regional conference series in mathematics, no. 92*. American Mathematical Society.

Collins, M., & Singer, Y. (1999). Unsupervised models for named entity classification. *EMNLP/VLC-99*.

Collobert, R., Weston, J., & Bottou, L. (2006). Trading convexity for scalability. *ICML06, 23rd International Conference on Machine Learning*. Pittsburgh, USA.

Corduneanu, A., & Jaakkola, T. (2001). *Stable mixing of complete and incomplete information* (Technical Report AIM-2001-030). MIT AI Memo.

Corduneanu, A., & Jaakkola, T. (2003). On information regularization. *Nineteenth Conference on Uncertainty in Artificial Intelligence (UAI03)*.

Corduneanu, A., & Jaakkola, T. S. (2005). Distributed information regularization on graphs. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Cortes, C., & Mohri, M. (2006). On transductive regression. *Advances in Neural Information Processing Systems (NIPS) 19*.

Cozman, F., Cohen, I., & Cirelo, M. (2003). Semi-supervised learning of mixture models. *ICML-03, 20th International Conference on Machine Learning*.

Cristianini, N., Shawe-Taylor, J., & Lodhi, H. (2001). Latent semantic kernels. *Proc. 18th International Conf. on Machine Learning*.

Culp, M., & Michailidis, G. (2007). An iterative algorithm for extending learners to a semisupervised setting. *The 2007 Joint Statistical Meetings (JSM)*.

Dara, R., Kremer, S., & Stacey, D. (2002). Clsutering unlabeled data with SOMs improves classification of labeled real-world data. *Proceedings of the World Congress on Computational Intelligence (WCCI)*.

Dasgupta, S., Littman, M. L., & McAllester, D. (2001). PAC generalization bounds for co-training. *Advances in Neural Information Processing Systems (NIPS)*.

De Bie, T., & Cristianini, N. (2004). Convex methods for transduction. In S. Thrun, L. Saul and B. Schölkopf (Eds.), *Advances in neural information processing systems 16*. Cambridge, MA: MIT Press.

De Bie, T., & Cristianini, N. (2006a). Fast SDP relaxations of graph cut clustering, transduction, and other combinatorial problems. *Journal of Machine Learning Research*, *7*, 1409–1436.

De Bie, T., & Cristianini, N. (2006b). Semi-supervised learning using semi-definite programming. In O. Chapelle, B. Schoëlkopf and A. Zien (Eds.), *Semi-supervised learning*. Cambridge-Massachussets: MIT Press.

de Sa, V. R. (1993). Learning classification with unlabeled data. *Advances in Neural Information Processing Systems (NIPS).*

Delalleau, O., Bengio, Y., & Roux, N. L. (2005). Efficient non-parametric function induction in semi-supervised learning. *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTAT 2005)*.

Demirez, A., & Bennett, K. (2000). Optimization approaches to semisupervised learning. In M. Ferris, O. Mangasarian and J. Pang (Eds.), *Applications and algorithms of complementarity*. Boston: Kluwer Academic Publishers.

Demiriz, A., Bennett, K., & Embrechts, M. (1999). Semi-supervised clustering using genetic algorithms. *Proceedings of Artificial Neural Networks in Engineering*.

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*.

Denis, F., Gilleron, R., & Tommasi, M. (2002). Text classification from positive and unlabeled examples. *The 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems(IPMU).*

Donoho, D. L., & Grimes, C. E. (2003). Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Arts and Sciences*, *100*, 5591–5596.

Elworthy, D. (1994). Does Baum-Welch re-estimation help taggers? *Proceedings of the 4th Conference on Applied Natural Language Processing.*

Farquhar, J. D., Hardoon, D. R., Meng, H., Shawe-Taylor, J., & Szedmak, S. (2006). Two view learning: SVM-2K, theory and practice. In *Advances in neural information processing systems (nips)*.

Fowlkes, C., Belongie, S., Chung, F., & Malik, J. (2004). Spectral grouping using the Nyström method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*, 214–225.

Fujino, A., Ueda, N., & Saito, K. (2005). A hybrid generative/discriminative approach to semi-supervised classifier design. *AAAI-05, The Twentieth National Conference on Artificial Intelligence*.

Fung, G., & Mangasarian, O. (1999). *Semi-supervised support vector machines for unlabeled data classification* (Technical Report 99-05). Data Mining Institute, University of Wisconsin Madison.

Garcke, J., & Griebel, M. (2005). Semi-supervised learning with sparse grids. *Proc. of the 22nd ICML Workshop on Learning with Partially Classified Training Data*. Bonn, Germany.

Getz, G., Shental, N., & Domany, E. (2005). Semi-supervised learning – a statistical physics approach. *Proc. of the 22nd ICML Workshop on Learning with Partially Classified Training Data*. Bonn, Germany.

Goldberg, A., & Zhu, X. (2006). Seeing stars when there aren't many stars: Graph-based semi-supervised learning for sentiment categorization. *HLT-NAACL 2006 Workshop on Textgraphs: Graph-based Algorithms for Natural Language Processing*. New York, NY.

Goldberg, A., Zhu, X., & Wright, S. (2007). Dissimilarity in graph-based semi-supervised classification. *Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Goldman, S., & Zhou, Y. (2000). Enhancing supervised learning with unlabeled data. *Proc. 17th International Conf. on Machine Learning* (pp. 327–334). Morgan Kaufmann, San Francisco, CA.

Grady, L., & Funka-Lea, G. (2004). Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. *ECCV 2004 workshop*.

Graf Estes, K., Evans, J. L., Alibali, M. W., & Saffran, J. R. (2006). Can infants map meaning to newly segmented words? Statistical segmentation and word learning. *Psychological Science*.

Grandvalet, Y., & Bengio, Y. (2005). Semi-supervised learning by entropy minimization. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Griffiths, T. L., Steyvers, M., Blei, D. M., & Tenenbaum, J. B. (2005). Integrating topics and syntax. *NIPS 17*.

Grira, N., Crucianu, M., & Boujemaa, N. (2004). Unsupervised and semi-supervised clustering: a brief survey. in 'A Review of Machine Learning Techniques for Processing Multimedia Content', Report of the MUSCLE European Network of Excellence (FP6).

Haffari, G., & Sarkar, A. (2007). Analysis of semi-supervised learning with the Yarowsky algorithm. *23rd Conference on Uncertainty in Artificial Intelligence (UAI)*.

Hein, M., & Maier, M. (2006). Manifold denoising. *Advances in Neural Information Processing Systems (NIPS) 19.*

Hofmann, T. (1999). Probabilistic latent semantic analysis. *Proc. of Uncertainty in Artificial Intelligence, UAI'99*. Stockholm.

Holub, A., Welling, M., & Perona, P. (2005). Exploiting unlabelled data for hybrid object classification. *NIPS 2005 Workshop in Inter-Class Transfer*.

Jaakkola, T., & Haussler, D. (1998). Exploiting generative models in discriminative classifiers. *Advances in Neural Information Processing Systems 11*.

Jaakkola, T., Meila, M., & Jebara, T. (1999). Maximum entropy discrimination. *Neural Information Processing Systems, 12*, *12*.

Joachims, T. (1999). Transductive inference for text classification using support vector machines. *Proc. 16th International Conf. on Machine Learning* (pp. 200–209). Morgan Kaufmann, San Francisco, CA.

Joachims, T. (2003). Transductive learning via spectral graph partitioning. *Proceedings of ICML-03, 20th International Conference on Machine Learning*.

Johnson, R., & Zhang, T. (2007). Two-view feature generation model for semi-supervised learning. *The 24th International Conference on Machine Learning*.

Jones, R. (2005). *Learning to extract entities from labeled and unlabeled text* (Technical Report CMU-LTI-05-191). Carnegie Mellon University. Doctoral Dissertation.

Kaariainen, M. (2005). Generalization error bounds using unlabeled data. *COLT 2005*.

Kapoor, A., Qi, Y., Ahn, H., & Picard, R. (2005). Hyperparameter and kernel learning for graph based semi-supervised classification. *Advances in NIPS.*

Kemp, C., Griffiths, T., Stromsten, S., & Tenenbaum, J. (2003). Semi-supervised learning with trees. *Advances in Neural Information Processing System 16.*

Kondor, R. I., & Lafferty, J. (2002). Diffusion kernels on graphs and other discrete input spaces. *Proc. 19th International Conf. on Machine Learning*.

Krishnapuram, B., Williams, D., Xue, Y., Hartemink, A., Carin, L., & Figueiredo, M. (2005). On semi-supervised classification. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Lafferty, J., Zhu, X., & Liu, Y. (2004). Kernel conditional random fields: Representation and clique selection. *The 21st International Conference on Machine Learning (ICML)*.

Langley, P. (2006). *Intelligent behavior in humans and machines* (Technical Report). Computational Learning Laboratory, CSLI, Stanford University.

Lawrence, N. D., & Jordan, M. I. (2005). Semi-supervised learning via Gaussian processes. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Lee, C.-H., Wang, S., Jiao, F., Schuurmans, D., & Greiner, R. (2006). Learning to model spatial dependency: Semi-supervised discriminative random fields. *Advances in Neural Information Processing Systems (NIPS) 19*.

Lee, W. S., & Liu, B. (2003). Learning with positive and unlabeled examples using weighted logistic regression. *Proceedings of the Twentieth International Conference on Machine Learning (ICML)*.

Leskes, B. (2005). The value of agreement, a new boosting algorithm. *COLT 2005*.

Levin, A., Lischinski, D., & Weiss, Y. (2004). Colorization using optimization. *ACM Transactions on Graphics*.

Li, W., & McCallum, A. (2005). Semi-supervised sequence modeling with syntactic topic models. *AAAI-05, The Twentieth National Conference on Artificial Intelligence*.

Liu, B., Lee, W. S., Yu, P. S., & Li, X. (2002). Partially supervised classification of text documents. *Proceedings of the Nineteenth International Conference on Machine Learning (ICML)*.

Lu, Q., & Getoor, L. (2003). Link-based classification using labeled and unlabeled data. *ICML 2003 workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*.

Madani, O., Pennock, D. M., & Flake, G. W. (2005). Co-validation: Using model disagreement to validate classification algorithms. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Maeireizo, B., Litman, D., & Hwa, R. (2004). Co-training for predicting emotions with spoken dialogue data. *The Companion Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

Mahdaviani, M., de Freitas, N., Fraser, B., & Hamze, F. (2005). Fast computational methods for visually guided robots. *The 2005 International Conference on Robotics and Automation (ICRA)*.

Mann, G. S., & McCallum, A. (2007). Simple, robust, scalable semi-supervised learning via expectation regularization. *The 24th International Conference on Machine Learning*.

McCallum, A., & Nigam, K. (1998a). A comparison of event models for naive bayes text classification. *AAAI-98 Workshop on Learning for Text Categorization*.

McCallum, A. K., & Nigam, K. (1998b). Employing EM in pool-based active learning for text classification. *Proceedings of ICML-98, 15th International Conference on Machine Learning* (pp. 350–358). Madison, US: Morgan Kaufmann Publishers, San Francisco, US.

Miller, D., & Uyar, H. (1997). A mixture of experts classifier with learning based on both labelled and unlabelled data. *Advances in NIPS 9* (pp. 571–577).

Mitchell, T. (1999). The role of unlabeled data in supervised learning. *Proceedings of the Sixth International Colloquium on Cognitive Science*. San Sebastian, Spain.

Mitchell, T. (2006). *The discipline of machine learning* (Technical Report CMU-ML-06-108). Carnegie Mellon University.

Muslea, I., Minton, S., & Knoblock, C. (2002). Active + semi-supervised learning = robust multi-view learning. *Proceedings of ICML-02, 19th International Conference on Machine Learning* (pp. 435–442).

Narayanan, H., Belkin, M., & Niyogi, P. (2006). On the relation between low density separation, spectral clustering and graph cuts. *Advances in Neural Information Processing Systems (NIPS) 19*.

Ng, A., Jordan, M., & Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems, 14*.

Nigam, K. (2001). *Using unlabeled data to improve text classification* (Technical Report CMU-CS-01-126). Carnegie Mellon University. Doctoral Dissertation.

Nigam, K., & Ghani, R. (2000). Analyzing the effectiveness and applicability of co-training. *Ninth International Conference on Information and Knowledge Management* (pp. 86–93).

Nigam, K., McCallum, A. K., Thrun, S., & Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine Learning*, *39*, 103–134.

Niu, Z.-Y., Ji, D.-H., & Tan, C.-L. (2005). Word sense disambiguation using label propagation based semi-supervised learning. *Proceedings of the ACL*.

Oliveira, C. S., Cozman, F. G., & Cohen, I. (2005). Splitting the unsupervised and supervised components of semi-supervised learning. *Proc. of the 22nd ICML Workshop on Learning with Partially Classified Training Data*. Bonn, Germany.

Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *Proceedings of the Association for Computational Linguistics* (pp. 271–278).

Pham, T. P., Ng, H. T., & Lee, W. S. (2005). Word sense disambiguation with semi-supervised learning. *AAAI-05, The Twentieth National Conference on Artificial Intelligence*.

Rabiner, L. (1989). A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*, 257–285.

Raina, R., Battle, A., Lee, H., Packer, B., & Ng, A. Y. (2007). Self-taught learning: Transfer learning from unlabeled data. *The 24th International Conference on Machine Learning*.

Ratsaby, J., & Venkatesh, S. (1995). Learning from a mixture of labeled and unlabeled examples with parametric side information. *Proceedings of the Eighth Annual Conference on Computational Learning Theory*, 412–417.

Rattray, M. (2000). A model-based distance for clustering. *Proc. of International Joint Conference on Neural Networks.*

Riloff, E., Wiebe, J., & Wilson, T. (2003). Learning subjective nouns using extraction pattern bootstrapping. *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-2003).*

Rosenberg, C., Hebert, M., & Schneiderman, H. (2005). Semi-supervised self-training of object detection models. *Seventh IEEE Workshop on Applications of Computer Vision*.

Rosset, S., Zhu, J., Zou, H., & Hastie, T. (2005). A method for inferring label sampling mechanisms in semi-supervised learning. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, *290*, 2323–2326.

Sajama, & Orlitsky, A. (2005). Estimating and computing density based distance metrics. *ICML05, 22nd International Conference on Machine Learning*. Bonn, Germany.

Saul, L. K., & Roweis, S. T. (2003). Think globally, fit locally: unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research*, *4*, 119–155.

Schuurmans, D., & Southey, F. (2001). Metric-based methods for adaptive model selection and regularization. *Machine Learning, Special Issue on New Methods for Model Selection and Model Combination*, *48*, 51–84.

Seeger, M. (2001). *Learning with labeled and unlabeled data* (Technical Report). University of Edinburgh.

Shahshahani, B., & Landgrebe, D. (1994). The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon. *IEEE Trans. On Geoscience and Remote Sensing*, *32*, 1087–1095.

Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*, 888–905.

Sindhwani, V., Keerthi, S., & Chapelle, O. (2006). Deterministic annealing for semi-supervised kernel machines. *ICML06, 23rd International Conference on Machine Learning*. Pittsburgh, USA.

Sindhwani, V., & Keerthi, S. S. (2006). Large scale semisupervised linear SVMs. *SIGIR 2006*.

Sindhwani, V., Niyogi, P., & Belkin, M. (2005a). Beyond the point cloud: from transductive to semi-supervised learning. *ICML05, 22nd International Conference on Machine Learning*.

Sindhwani, V., Niyogi, P., & Belkin, M. (2005b). A co-regularized approach to semi-supervised learning with multiple views. *Proc. of the 22nd ICML Workshop on Learning with Multiple Views*.

Sindhwani, V., Niyogi, P., Belkin, M., & Keerthi, S. (2005c). Linear manifold regularization for large scale semi-supervised learning. *Proc. of the 22nd ICML Workshop on Learning with Partially Classified Training Data*.

Sinha, P., Balas, B., Ostrovsky, Y., & Russell, R. (2006). Face recognition by humans: 20 results all computer vision researchers should know about. *(under review)*.

Smola, A., & Kondor, R. (2003). Kernels and regularization on graphs. *Conference on Learning Theory, COLT/KW*.

Stromsten, S. B. (2002). *Classification learning from both classified and unclassified examples*. Doctoral dissertation, Stanford University.

Szummer, M., & Jaakkola, T. (2001). Partially labeled classification with Markov random walks. *Advances in Neural Information Processing Systems, 14*.

Szummer, M., & Jaakkola, T. (2002). Information regularization with partially labeled data. *Advances in Neural Information Processing Systems, 15*.

Taskar, B., Guestrin, C., & Koller, D. (2003). Max-margin Markov networks. *NIPS'03*.

Teh, Y. W., & Roweis, S. (2002). Automatic alignment of local representations. *Advances in NIPS*.

Tenenbaum, J. B., de Silva, V., , & Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, *290*, 2319–2323.

Tipping, M. (1999). Deriving cluster analytic distance functions from Gaussian mixture models.

Tong, W., & Jin, R. (2007). Semi-supervised learning by mixed label propagation. *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence (AAAI)*.

Tsang, I., & Kwok, J. (2006). Large-scale sparsified manifold regularization. *Advances in Neural Information Processing Systems (NIPS) 19*.

Vapnik, V. (1998). *Statistical learning theory*. Wiley-Interscience.

von Luxburg, U., Belkin, M., & Bousquet, O. (2004). *Consistency of spectral clustering* (Technical Report TR-134). Max Planck Institute for Biological Cybernetics.

von Luxburg, U., Bousquet, O., & Belkin, M. (2005). Limits of spectral clustering. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Wallis, G., & Bülthoff, H. (2001). Effects of temporal association on recognition memory. *Proceedings of the National Academy of Sciences*, *98*, 4800–4804.

Wang, F., & Zhang, C. (2006). Label propagation through linear neighborhoods. *ICML06, 23rd International Conference on Machine Learning*. Pittsburgh, USA.

Weinberger, K. Q., Packer, B. D., & Saul, L. K. (2005). Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization. *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTAT 2005)*.

Weinberger, K. Q., & Saul, L. K. (2004). Unsupervised learning of image manifolds by semidefinite programming. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 988–995).

Weinberger, K. Q., Sha, F., & Saul, L. K. (2004). Learning a kernel matrix for nonlinear dimensionality reduction. *Proceedings of ICML-04* (pp. 839–846).

Weiss, Y. (1999). Segmentation using eigenvectors: A unifying view. *ICCV (2)* (pp. 975–982).

Weston, J., Collobert, R., Sinz, F., Bottou, L., & Vapnik, V. (2006). Inference with the universum. *ICML06, 23rd International Conference on Machine Learning*. Pittsburgh, USA.

Weston, J., Leslie, C., Zhou, D., Elisseeff, A., & Noble, W. S. (2004). Semi-supervised protein classification using cluster kernels. In S. Thrun, L. Saul and B. Schölkopf (Eds.), *Advances in neural information processing systems 16*. Cambridge, MA: MIT Press.

Wu, M., & Schölkopf, B. (2007). Transductive classification via local learning regularization. *Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Xu, L., & Schuurmans, D. (2005). Unsupervised and semi-supervised multi-class support vector machines. *AAAI-05, The Twentieth National Conference on Artificial Intelligence*.

Yang, X., Fu, H., Zha, H., & Barlow, J. (2006). Semi-supervised nonlinear dimensionality reduction. *ICML-06, 23nd International Conference on Machine Learning*.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics* (pp. 189–196).

Yianilos, P. (1995). *Metric learning via normal mixtures* (Technical Report). NEC Research Institute.

Yu, K., Tresp, V., & Zhou, D. (2004). *Semi-supervised induction with basis functions* (Technical Report 141). Max Planck Institute for Biological Cybernetics, Tübingen, Germany.

Yu, K., Yu, S., & Tresp, V. (2005). Blockwise supervised inference on large graphs. *Proc. of the 22nd ICML Workshop on Learning with Partially Classified Training Data*. Bonn, Germany.

Zelikovitz, S., & Hirsh, H. (2001). Improving text classification with LSI using background knowledge. *IJCAI01 Workshop Notes on Text Learning: Beyond Supervision*.

Zhang, T., & Ando, R. (2006). Analysis of spectral kernel design based semi-supervised learning. In Y. Weiss, B. Schölkopf and J. Platt (Eds.), *Advances in neural information processing systems 18*. Cambridge, MA: MIT Press.

Zhang, T., & Oles, F. J. (2000). A probability analysis on the value of unlabeled data for classification problems. *Proc. 17th International Conf. on Machine Learning* (pp. 1191–1198). Morgan Kaufmann, San Francisco, CA.

Zhang, X., & Lee, W. S. (2006). Hyperparameter learning for graph based semi-supervised learning algorithms. *Advances in Neural Information Processing Systems (NIPS) 19*.

Zhou, D., Bousquet, O., Lal, T., Weston, J., & Schl̈kopf, B. (2004a). Learning with local and global consistency. *Advances in Neural Information Processing System 16*.

Zhou, D., Huang, J., & Schoelkopf, B. (2006a). Learning with hypergraphs: Clustering, classification, and embedding. *Advances in Neural Information Processing Systems (NIPS) 19*.

Zhou, D., Huang, J., & Schölkopf, B. (2005a). Learning from labeled and unlabeled data on a directed graph. *ICML05, 22nd International Conference on Machine Learning*. Bonn, Germany.

Zhou, D., Schölkopf, B., & Hofmann, T. (2005b). Semi-supervised learning on directed graphs. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems 17*. Cambridge, MA: MIT Press.

Zhou, D., Weston, J., Gretton, A., Bousquet, O., & Schlkopf, B. (2004b). Ranking on data manifolds. *Advances in Neural Information Processing System 16*.

Zhou, Y., & Goldman, S. (2004). Democratic co-learing. *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2004)*.

Zhou, Z.-H., Chen, K.-J., & Dai, H.-B. (2006b). Enhancing relevance feedback in image retrieval using unlabeled data. *ACM Transactions on Information Systems*, *24*, 219–244.

Zhou, Z.-H., Chen, K.-J., & Jiang, Y. (2004c). Exploiting unlabeled data in content-based image retrieval. *Proceedings of ECML-04, 15th European Conference on Machine Learning*. Italy.

Zhou, Z.-H., & Li, M. (2005a). Semi-supervised regression with co-training. *International Joint Conference on Artificial Intelligence (IJCAI)*.

Zhou, Z.-H., & Li, M. (2005b). Tri-training: exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge and Data Engineering*, *17*, 1529–1541.

Zhou, Z.-H., & Xu, J.-M. (2007). On the relation between multi-instance learning and semi-supervised learning. *The 24th International Conference on Machine Learning*.

Zhou, Z.-H., Zhan, D.-C., & Yang, Q. (2007). Semi-supervised learning with very few labeled training examples. *Twenty-Second AAAI Conference on Artificial Intelligence (AAAI-07)*.

Zhu, X. (2005). *Semi-supervised learning with graphs*. Doctoral dissertation, Carnegie Mellon University. CMU-LTI-05-192.

Zhu, X., & Ghahramani, Z. (2002). *Towards semi-supervised classification with Markov random fields* (Technical Report CMU-CALD-02-106). Carnegie Mellon University.

Zhu, X., Ghahramani, Z., & Lafferty, J. (2003a). Semi-supervised learning using Gaussian fields and harmonic functions. *The 20th International Conference on Machine Learning (ICML)*.

Zhu, X., Kandola, J., Ghahramani, Z., & Lafferty, J. (2005). Nonparametric transforms of graph kernels for semi-supervised learning. In L. K. Saul, Y. Weiss and L. Bottou (Eds.), *Advances in neural information processing systems (nips) 17*. Cambridge, MA: MIT Press.

Zhu, X., & Lafferty, J. (2005). Harmonic mixtures: combining mixture models and graph-based methods for inductive and scalable semi-supervised learning. *The 22nd International Conference on Machine Learning (ICML)*. ACM Press.

Zhu, X., Lafferty, J., & Ghahramani, Z. (2003b). Combining active learning and semi-supervised learning using Gaussian fields and harmonic functions. *ICML 2003 workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*.

Zhu, X., Lafferty, J., & Ghahramani, Z. (2003c). *Semi-supervised learning: From Gaussian fields to Gaussian processes* (Technical Report CMU-CS-03-175). Carnegie Mellon University.

Zhu, X., Rogers, T., Qian, R., & Kalish, C. (2007). Humans perform semi-supervised classification too. *Twenty-Second AAAI Conference on Artificial Intelligence (AAAI-07)*.