

Gene Functional classification by Semi-supervised Learning from Heterogeneous Data

Tao Li* Shenghuo Zhu
Department of Computer Science
University of Rochester
Rochester, NY 14620
{taoli,zsh}@cs.rochester.edu

Qi Li
Department of CIS
University of Delaware
Newark, DE 19716
qili@cis.udel.edu

Mitsunori Ogihara
Department of Computer Science
University of Rochester
Rochester, NY 14620
ogihara@cs.rochester.edu

ABSTRACT

Gene function discovery is an important and interesting problem in computational analysis of microarray data. In this paper, we investigate the use of a semi-supervised learning algorithm for inferring gene functional classifications from heterogeneous data set consisting of DNA microarray expression measurements and phylogenetic profiles from whole-genome sequence comparisons. The semi-supervised learning approach aims at minimizing the disagreement between individual models built from each separate information source by employing a **co-updating** method and making use of both labeled and unlabeled data. Our results suggest that the semi-supervised approach could be used for gene functional classification. The data sets and the program code used for the experiments can be accessed from our webpage¹.

Keywords: Gene functional classification, semi-supervised, Support Vector Machine(SVM), heterogeneous, minimize disagreement

1. INTRODUCTION

Gene function discovery is an important problem in microarray data analysis. It can help us understand the molecular machinery of the cell. Initial analysis of microarray data for gene function discovery focused on clustering algorithms, such as hierarchical clustering [9] and self-organizing maps [19]. These unsupervised algorithms work under the assumption that genes with similar expression pattern may have similar function. Besides the microarray gene expression data, the sequencing projects provide a complementary view for exploring the molecular machinery. The availability of complete genomic sequence of human and other species provides a tremendous opportunity for understanding the functions of genes [15]. Phylogenetic profile is another data type for gene function discovery and it is derived from a comparison between

*The email address of the contacting author is taoli@cs.rochester.edu. This work was done while the author was doing summer research intern at Xerox Wilson Research Center.

¹<http://www.cs.rochester.edu/u/taoli/bio>.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

a given gene and a collection of complete genomes. Each profile characterizes the evolutionary history of a given gene. Two genes with similar phylogenetic profiles are likely to have similar functions, under the assumption that their similar pattern of inheritance across species is the result of a functional link.

There are many works on gene function discovery on microarray expression data and sequence data separately, however, few works that considered combining these heterogeneous information together. [15] is one of the few works on combining different types of data for gene function discovery. It presented a supervised method using Support Vector Machine(SVM) to learn gene functional classifications from a heterogeneous data set consisting of microarray expression data and phylogenetic profiles. It showed that this combination is most successful when the SVM operates in a feature space that is explicitly heterogeneous. An SVM with an explicitly heterogeneous kernel function by first computing separate kernels for each data type and then summing the results is constructed as follows:

$$K_{combined}(X, Y) = K(X_e, Y_e) + K(X_p, Y_p),$$

where the subscripts denote gene expression and phylogenetic profile data, and the local kernel function inside each data type is as follows:

$$K(X, Y) = \left(\frac{X \cdot Y}{\sqrt{X \cdot X} \sqrt{Y \cdot Y}} + 1 \right)^3. \quad (1)$$

The heterogeneous kernel incorporates prior knowledge by accounting for higher-order correlations among features of one data type but ignoring higher-order correlations across data types. Although it provided a promising view of the analysis of combining heterogeneous data, several questions/directions still need to be further investigated such as: How to fully take advantage of these heterogeneous information? How to efficiently utilize heterogeneous data especially when some category information is unavailable? etc.

The problem of learning from multiple information sources has been extensively studied in machine learning and computer vision literature where it is called as multi-modal learning². Generally there are two types of multi-modal learning: feature level integration and semantic integration [23]. The feature integration combines the information at the feature level and performs learning in the joint feature space. The correlation structure between different sources can be discovered via learning. The semantic integration, on the other hand, first builds individual models based on separate information sources and then combines these models via some processes say, mutual information maximization [3]. In this context,

²In this paper, we interchangeably use modal, component and information source.

the combination of heterogeneous kernel used in [15] is also a semantic integration.

The ability of using labeled and unlabeled data are useful for gene functional classification in two different ways. As mentioned before, there are cases that some gene category information is not available. So we have some labeled data and some unlabeled data. The heterogeneous data types could let us take advantage of both the labeled and unlabeled data. Even in the cases where all the gene category information is available, most learning methods use cross-validation technique to build prediction models where some data samples are hold out. These hold-out samples are not used for training. With information from heterogeneous sources, these hold-out samples could also be used for training by viewing them as unlabeled data, i.e., with no category information.

In this paper, we will present a **co-updating** approach for semi-supervised learning from heterogeneous data types. **Co-updating** approach is a kind of semantic integration. The reason we prefer semantic integration is four-fold. First, although the structure in the joint feature space is often more informative than that available to each of the individual sources, feature integration tends not to generalize well. The model complexity, computation intensity, and training difficulty typically are other problems associated with feature integration [23]. Second, learning in the joint space is not able to marginalize over the missing sources and requires future patterns for classification containing all feature dimensions [8]. While we intend to learn from a joint feature space, we still need to be able to analyze and act on the information from a single source. For example, we sometimes would like to predict the functional classifications solely based on the DNA microarray information. Feature integration is thus not suitable. Third, we would like to have a way to utilize both labeled and unlabeled data since there are scenarios that some gene category information is available and while the other is unavailable. Fourth, the semantic integration appears to have biological and physical plausibility. It is well known that the cerebral cortex competently classifies unimodal stimuli while keeping the different modalities largely separate. Also, McGurk effect showed that although the information from different sensory modalities is combined in determining human's perception, the combination is often not subject to conscious control [12].

Several algorithms have been proposed to combine information from labeled and unlabeled data include using Expectation Maximization [2, 11] and generative models [14]. Blum and Mitchell described the co-training setting [4] where the features in the problem domain are naturally divided into two disjoint sets (or in other words, two-modal) and showed that, under certain assumptions, PAC-like guarantee on learning with labeled and unlabeled data holds. They also present an algorithm for web pages classification using both labeled and unlabeled data. Roth and Zelenko developed a theory for learning scenarios where multiple learner co-exist but there are mutual compatibility constraints on their outcomes [17]. Co-training has also been used in information extraction [16], word sense disambiguation [24] and name entity classification [7]. These applications demonstrated that using labeled data could lead to the performance improvement over learning solely based on labeled data. Nigam and Ghani [13] showed that when an independent and redundant feature split exists, co-training algorithms outperform other algorithms using unlabeled data.

When we have two different information sources for gene functional classification, we generally fall into the co-training setting. In this paper, we first give a theoretical justification on minimizing the disagreement between two individual models could lead to the improvement of the classification accuracy of individual models. Based on the proof, we then develop a co-updating approach which

tries to minimizing the disagreement between the individual models. Finally we report our experiment results. Since our learning algorithm takes into account both labeled and unlabeled samples, we refer it as semi-supervised learning algorithm. The paper is organized as follows: Section 2 presents the theoretical proof that minimizing the disagreement would lead to performance improvement, Section 3 describes our co-updating approach and Section 4 shows and discusses experimental results. Finally Section 5 concludes and discusses future research directions.

2. MINIMIZING DISAGREEMENT

In this section, we show that, theoretically, minimizing the disagreement between two individual models could lead to the improvement of the classification accuracy of individual models. In this paper, we focus on binary classification problems and we use 0, 1 to label the two classes respectively³. Suppose we have an instance space $X = (X_1, X_2)$ where X_1 and X_2 are from different observations. Let D be the distribution over X . If f is the target function over D , then for any example $x = (x_1, x_2)$ we would have $f(x_1, x_2) = f_1(x_1) = f_2(x_2)$ where f_1 and f_2 are the target functions over X_1 and X_2 respectively.

Suppose we build hypotheses f'_1 on X_1 and f'_2 on X_2 . We assume

$$\begin{aligned} P(f'_1(x_1) = f'_2(x_2)) &= P(f'_1(x_1) = f'_2(x_2) | f_2(x_2) = f'_2(x_2)) \\ P(f'_1(x_1) \neq f'_2(x_2)) &= P(f'_1(x_1) \neq f'_2(x_2) | f_2(x_2) \neq f'_2(x_2)) \end{aligned}$$

The assumption can be interpreted as whether f'_1 and f'_2 agree is independent of the accuracy of f'_2 (or f'_1). This seems to be a somewhat plausible starting point. One popular assumption used[4] is that x_1 and x_2 are conditional independent given the labels, i.e., $P(x_1 = x'_1 | x_2 = x'_2) = P(x_1 = x'_1 | f_2(x_2) = f_2(x'_2))$. Now we use f'_1 to classify new instances $x = (x_1, x_2)$, let a be the accuracy of f'_2 . The misclassification error is $P(f'_1(x_1) \neq f(x_1, x_2))$. Note that $P(f'_1(x_1) \neq f(x_1, x_2)) = P(f'_1(x_1) \neq f_2(x_2)) = P(f'_1(x_1) \neq f_1(x_1))$.

$$\begin{aligned} \text{Hence } P(f'_1(x_1) \neq f_2(x_2)) &= 1 - P(f'_1(x_1) = f_2(x_2)) \\ &= 1 - \{aP(f'_1(x_1) = f'_2(x_2)) + (1 - a)P(f'_1(x_1) \neq f'_2(x_2))\} \\ &= 1 - \{aP(f'_1(x_1) = f'_2(x_2)) + P(f'_1(x_1) \neq f'_2(x_2)) \\ &\quad - aP(f'_1(x_1) \neq f'_2(x_2))\} \\ &= 1 - a\{P(f'_1(x_1) = f'_2(x_2)) - P(f'_1(x_1) \neq f'_2(x_2))\} \\ &\quad - P(f'_1(x_1) \neq f'_2(x_2)) \end{aligned}$$

Now define a function $f(a, x, y) = 1 - a(x - y) - y$ where $x + y = 1$. Hence

$$\begin{aligned} g(x, a) &= f(a, x, y) = x - a(x - 1 + x) = x - 2ax + a, \\ dg/dx &= 1 - 2a < 0 \quad \text{if } a > 1/2, \\ dg/da &= 1 - 2x < 0 \quad \text{if } x > 1/2. \end{aligned}$$

So if both $P(f'_1(x_1) = f'_2(x_2))$ and a are greater than 1/2, then increasing a and $P(f'_1(x_1) = f'_2(x_2))$ would decrease the misclassification error of the model using the first component of the instances. Similarly, when using f'_2 to classify new instances $x = (x_1, x_2)$ and letting b be the accuracy of f'_1 , we obtain that if both $P(f'_1(x_1) = f'_2(x_2))$ and b are greater than 1/2, then increasing a and $P(f'_1(x_1) = f'_2(x_2))$ would decrease the misclassification error of the model using the second component of the instances. In other words, minimizing the disagreement, i.e., increasing $P(f'_1(x_1) = f'_2(x_2))$, would decrease the misclassification error.

³For multi-class classification problem, several approaches can be used to reduce it to binary ones [1].

3. CO-UPDATING APPROACH

Based on the proof in Section 2, we propose a co-updating approach to minimize the disagreement. The co-updating approach differs from the popular co-training algorithm presented in [4] and the coBoosting algorithm described in [7]. The popular co-training algorithm incrementally incorporate the unlabeled data into learning and the coBoosting adaptively update the distribution of the training samples. The basic idea of co-updating is as follows: The labeled samples are first used to get weak hypotheses f_1^0 on X_1 and f_2^0 on X_2 . For every unlabeled sample (x_1, x_2) , we use $f_2^0(x_2)$ as the noisy label for x_1 to update f_1^0 and use $f_1^0(x_1)$ as noisy label for x_2 to update f_2^0 . This process is then repeated until some stop criterion is met. The co-updating approach is described as follows: The intuition behind the co-updating approach is that we stochas-

Algorithm 1 Co-updating

Input: A collection of m labeled and n unlabeled data
 α, β — default 0.15
 T — default 10

Output: Two classifiers that predict class labels for new instances based on different information sources

- 1: Build f_1^0 from the first component of labeled samples
- 2: Build f_2^0 from the second component of labeled samples
- 3: Loop for T times
- 4: Use f_1^{i-1} get the labels of all the samples based on their first component, l_1^{i-1}
- 5: Use f_2^{i-1} get the labels of all the samples based on their second component, l_2^{i-1}
- 6: Update l_1^{i-1} , i.e., for $k := 1$ to $m + n$
- 7: $r_1 = \text{random}(0, 1)$ (a random number in $(0, 1)$)
- 8: if $(r_1 < \alpha)$, $l_1^i(k) = l_2^{i-1}(k)$
- 9: Update l_2^{i-1} , i.e., for $j := 1$ to $m + n$
- 10: $r_2 = \text{random}(0, 1)$ (a random number in $(0, 1)$)
- 11: if $(r_2 < \beta)$, $l_2^i(j) = l_1^{i-1}(j)$
- 12: Utilize the first component of all the samples, viewing l_1^i as labels, to update f_1^{i-1} and get f_1^i
- 13: Utilize the second component of all the samples, viewing l_2^i as labels, to update f_2^{i-1} and get f_2^i
- 14: Output f_1^T, f_2^T

tically update the prediction of one modal using the information from the other modal. If the initial classifiers are better than random guess ($a, b > \frac{1}{2}$) and the agreement of the predictions from the two modal is independent of each modal's prediction accuracy, then on average the co-updating approach, trying to minimize the disagreement, should progress toward higher performance.

Let γ_1, γ_2 be two random vectors decided by r_1, r_2 in the co-updating process, then

$$\begin{aligned} |l_1^i - l_2^i| &= |l_1^{i-1} + \gamma_1(l_2^{i-1} - l_1^{i-1}) - l_2^{i-1} - \gamma_2(l_1^{i-1} - l_2^{i-1})| \\ &= |(l_1^{i-1} - l_2^{i-1}) - (\gamma_1 + \gamma_2)(l_2^{i-1} - l_1^{i-1})| \\ &= (1 - \gamma_1 - \gamma_2)|l_1^{i-1} - l_2^{i-1}|. \end{aligned}$$

So the co-updating approach is trying to increase the disagreement. Let L be the true labels for the samples. Note that

$$\begin{aligned} |l_1^i - L| &= |l_1^{i-1} + \gamma_1(l_2^{i-1} - l_1^{i-1}) - L| \\ &= |(1 - \gamma_1)(l_1^{i-1} - L) + \gamma_1(l_2^{i-1} - L)| \\ &\leq (1 - \gamma_1)|l_1^{i-1} - L| + \gamma_1|l_2^{i-1} - L|. \end{aligned}$$

Hence, we might be able to increase a or b while reducing the disagreement. So the update rule concurs with our previous proof.

4. RESULTS AND ANALYSIS

4.1 Data Description

The experiments carried out here use two types of genomic data. The first data set consists of a set of 79-dimension gene expression vectors for 2465 yeast genes and it derives from a collection of DNA microarray hybridization experiments where each sample represents the logarithm of the ratio of expression levels of a particular gene under two different experimental conditions. The second data set is a set of 24-element vectors for 2465 yeast genes and it derives from phylogenetic profiles. Classification experiments are carried out using gene functional categories from the Munich Information Center for Protein Sequences Comprehensive Yeast Genome Database(CYGD)⁴. The database contains several hundred functional classes whose definitions come from biochemical and genetic studies of gene function. Our experiments here use the five most learnable classes with heterogeneous data types described in [15]. The five classes are listed in Table 1.

Class Name	Size
amino acid transporters	22
ribosomal proteins	173
sugar and carbohydrate transporters	32
deoxyribonucleotide metabolism	9
mitochondrial organization	296

Table 1: Five Classes. Each row contains the name of the class and the size of the class.

For convenience, in the rest of the paper, we call the five classes as class I, II, III, IV and V respectively. We use SVM as base classifiers since it is state-of-art and has been successfully and widely used in classification [5, 10]. Basically speaking, SVM is a mechanism which tries to classify data points in the input space by mapping them into a higher dimension feature space (using kernel function) and then finding the separating hyperplane in the feature space with the largest margin. More details on SVM can be found in [6, 21]. In our experiments, we use the kernel function described in Equation 1. This kernel function takes into account pairwise and tertiary correlations among the measurements and was reported to be very efficient [5, 15]. We also use the default parameter settings in the **co-update** procedure.

4.2 Results on Balanced Distribution

Generally building models when one class is rare can be quite difficult because there are often many unstated assumptions [25]. It is conventional wisdom that classifiers built using all the data tend to perform worse on the minority class than on the majority class since the class *priors* in the natural distribution are biased strongly in favor of the majority class and the minority class has much fewer training and test samples [22]. It has been shown [22] that when the area of ROC curve is used as the performance measure, the optimal distribution generally contains between 50% and 90% minority-class examples and the general strategy is always allocating about half of the training examples to the minority class. Although the balanced distribution will not always yield optimal distribution, it will generally lead to results which are no worse than, and often superior to those which use the natural class distribution [22]. So our first set of experiments are carried on the balanced distributions of the five classes. We use all the minority-class samples and the balanced distributions are generated by randomly selecting an equal number of major-class samples.

⁴<http://mips.gsf.de/proj/yeast/CYGD/db/index.html>.

We then conduct experiments using our co-updating approach. Although all of our samples are labeled, for the purpose of illustrating the usefulness of our co-updating approach, we use about 50% of the selected samples as labeled samples while the remaining are regarded as unlabeled. The distributions of the samples are described in Table 2. Due to the inside randomness of the co-updating approach, the accuracy is calculated by averaging 10 runs. The results are shown in Table 3–Table 7. The results are calculated by testing on all the selected samples. Each element of the table is a 4-tuple (TP, TN, FP, FN) where TP represents the number of true positives, TN is the number of true negatives, FP is the number of false positives and FN is the number of false negatives. We represent the result as a 4-tuple since the accuracy is not an approximate measure when the class distribution is highly skewed [22]. The cost function adopted in [5, 15] also suffers from the fact of lacking theoretical or experimental verification. So here in our experiment, we use the 4-tuple to try to give the readers a more systematic and comprehensive view of the performance. In each table, column A shows the result of the case when SVM is obtained by training only on labeled samples, column B shows the result using co-updating approach and column C shows the result when SVM is obtained by training on all the selected samples (by treating all of them as labeled samples)⁵. In these tables, column C shows that, SVM correctly classify all the selected samples when all of them are being used for training. In other words, the close test of SVM gives perfect results. It can be drawn from these tables that the co-updating approach utilizes the unlabeled samples to improve the learning performance on most counts: The results of column B are better than those of column A in Table 3, Table 4 and Table 7 and the result of column B is the same as that of column A in Table 6. The results in Table 5 is a little bit mixed. While column B has more correctly classified positive and negative samples than column A for expression data, it has few correctly classified negative samples than column A for phylogenetic profiles. The reason may be that for class III, the result of phylogenetic profiles in column A are much better than the result of expression data in column A. In other words, the base classifier from one data type are consistently better than the other, then the co-updating approach might degrade the performance for the modal with a better base classifier.

4.3 Results on Whole Data Set

We also did another set of experiments on the whole data set. In the set of experiments, we use all the data samples. Similar to Section 4.2, for the purpose of illustrating the usefulness of our co-updating approach, we use about 50% of the selected samples as labeled samples while the remaining are regarded as unlabeled. We only reported some experiments here to save space and they are shown in Table 8 and Table 9. Similar results are observed for other experiments and they are available from our website. Each element of the result table is a 4-tuple (TP, TN, FP, FN). In each table, column A shows the result of the case when SVM is obtained by training only on labeled samples, column B shows the result using co-updating approach and column C shows the result when SVM is obtained by training on all the samples (by treating all the samples as labeled samples). It can be drawn from these tables that the co-updating approach utilize the unlabeled samples to improve the learning performance.

5. CONCLUSION AND DISCUSSION

⁵The test result obtained in this case is usually called the close test.

Class Name	TS	PS	NS	PSL	PSU	NSL	NSU
Class I	80	22	58	11	11	29	29
Class II	360	173	187	86	87	94	93
Class III	100	32	68	16	16	34	34
Class IV	40	9	31	5	4	16	15
Class V	600	296	304	148	148	152	152

Table 2: The distributions of the samples used with co-updating approach. All the entries are the number of samples, TS=total, PS=positive, NS=negative, PSL=positively labeled, PSU=unlabeled positive, NSL=negatively labeled and NSU=unlabeled negative.

	A	B	C
Expression	(18,58,4,0)	(19,59,1,1)	(22,58,0,0)
Phylogeny	(20,58,2,0)	(21,58,1,0)	(22,58,0,0)

Table 3: Experimental results of class I. For expression data, column B has more correctly classified positive and negative samples than A. For phylogeny, column B has more correctly classified positive samples than A.

	A	B	C
Expression	(172,176,1,11)	(173,179,0,8)	(173,187,0,0)
Phylogeny	(173,175,0,12)	(173,177,0,10)	(173,187,0,0)

Table 4: Experimental results of class II. For expression data, column B has more correctly classified positive and negative samples than A. For phylogeny, column B has more correctly classified negative samples than A.

	A	B	C
Expression	(26,60,6,8)	(27,64,5,4)	(32,68,0,0)
Phylogeny	(29,67,3,1)	(29,65,3,3)	(32,68,0,0)

Table 5: Experimental results of class III. For expression data, column B has more correctly classified positive and negative samples than A. For phylogeny, column B has few correctly classified negative samples than A.

	A	B	C
Expression	(9,66,0,5)	(9,66,0,5)	(9,71,0,0)
Phylogeny	(9,66,0,6)	(9,66,0,6)	(9,71,0,0)

Table 6: Experimental results of class IV. The result of column B and column A are the same.

	A	B	C
Expression	(288,284,8,20)	(289,286,7,18)	(296,304,0,0)
Phylogeny	(281,286,15,18)	(287,288,9,16)	(296,304,0,0)

Table 7: Experimental results of class V. For expression data, column B has more correctly classified positive and negative samples than A. For phylogeny, column B has more correctly classified positive and negative samples than A.

	A	B	C
Expression	(13,2443,9,0)	(13,2438,9,5)	(12,2432,10,11)
Phylogeny	(13,2436,9,7)	(13,2436,9,7)	(12,2432,10,11)

Table 8: Experimental results of class I on whole data set. For expression data, column B has more correctly classified negative samples than A. For phylogeny, column B has the same result as A.

	A	B	C
Expression	(191,2144,105,25)	(253,2154,43,15)	(292,2163,4,6)
Phylogeny	(225,2136,71,33)	(248,2149,48,20)	(294,2168,2,1)

Table 9: Experimental results of class V on whole data set. For expression data, column B has more correctly classified positive and negative samples than A. For phylogeny, column B has more correctly classified positive and negative samples than A.

In this paper, we present a co-updating approach for gene functional classification from multiple information sources. The co-updating approach tries to minimize the disagreement between the individual models and makes use of both labeled and unlabeled data. The co-updating approach could also be applied to multi-modal learning tasks such as word learning [18] and object recognition [20] since spatio-temporal and cross modal coherence is a powerful constraint in sensory data of the physical world. We could apply the co-updating approach to utilize the approximately co-incident information of different modalities in these tasks.

Our experiments reported here suggest that co-updating approach could be able to utilize the unlabeled data to improve the classification performance. In the case where one data type has a consistent better base classifier than the other data type, the co-updating might degrade the performance of the data type with a better classifier. So the desired situation for co-updating approach is that different data types should have approximately the same prediction power. This indeed confirms the widely used assumption in multi-modal research. Our experiments also verify that SVM is the state-of-art supervised learning method: it gives perfect classification results in most of our experiments.

There are several natural avenues for future research. First, one obvious research direction is to include multiple biological data types for gene functional classification and to extend the co-updating approach to multiple data sources. Second, many feature selection techniques could be incorporated into co-updating approach. Third, as we mentioned earlier, there does not exist a good performance measure for gene functional classification. So to come up with a theoretical or experimental justified performance measure is also an important task.

Acknowledgments

The project is supported in part by NIH Grants 5-P41-RR09283, RO1-AG18231, and P30-AG18254 and by NSF Grants EIA-0080124, EIA-0205061, and DUE-9980943.

6. REFERENCES

- [1] E. L. Allwein, R. E. Schapire, and Y. Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. In *Proc. 17th International Conf. on Machine Learning*, pages 9–16. Morgan Kaufmann, San Francisco, CA, 2000.
- [2] A.P.Dempster, N. Laird, and D.B.Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1, 38 1977.
- [3] S. Becker. Mutual information maximization: Models of cortical self-organization. *Network: Computation in Neural Systems*, 7(1):7–31, February 1996.
- [4] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT: Proceedings of the Workshop on Computational Learning Theory*, Morgan Kaufmann Publishers, 1998.
- [5] M. P. S. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. Sugnet, T. S. Furey, J. Manuel Ares, and D. Haussler. Knowledge-based analysis of microarray gene expression data using support vector machines. In *Proc. of the National Academy of Science*, volume 97, pages 262–267, 2000.
- [6] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
- [7] M. Collins and Y. Singer. Unsupervised models for named entity classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.
- [8] V. De Sa and D. Ballard. Category learning through multi-modality sensing. *Neural Computation*, 10(5):1097–1117, 1998.
- [9] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. In *Proc. of the National Academy of Sciences of USA*, volume 95, 1998.
- [10] T. S. Furey, N. Christianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.
- [11] Z. Ghahramani and M. I. Jordan. Supervised learning from incomplete data via an EM approach. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6, pages 120–127. Morgan Kaufmann Publishers, Inc., 1994.
- [12] H. McGurk and MacDonald. Hearing lips and seeing voices. *Nature*, 264:746–748, 1976.
- [13] K. Nigam and R. Ghani. Analyzing the effectiveness and applicability of co-training. In *CIKM*, pages 86–93, 2000.
- [14] K. Nigam, A. K. McCallum, S. Thrun, and T. M. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134, 2000.
- [15] J. C. Paul Pavlidis, Jason Weston and W. N. Grundy. Gene functional classification from heterogeneous data. In *Proceedings of the Fifth International Conference on Computational Molecular Biology*, 2001.
- [16] E. Riloff and R. Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *AAAI/IAAI*, pages 474–479, 1999.
- [17] D. Roth and D. Zelenko. Toward a theory of learning coherent concepts. In *AAAI/IAAI*, pages 639–644, 2000.
- [18] D. Roy. Learning from multimodal observations. In *IEEE International Conference on Multimedia and Expo (I)*, pages 579–582, 2000.
- [19] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E. Lander, and T. Golub. Interpreting patterns of gene expression with self-organizing maps. In *Proc. of the National Academy of Sciences of USA*, volume 96, 1999.
- [20] M. K. Tanenhaus, S.-K. M. J., E. K.M., and S. J.E. Integration of visual and linguistic information in spoken language comprehension. *Science*, 268:1632–1634, 1995.
- [21] V. N. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.
- [22] G. Weiss and F. Provost. The effect of class distribution on classifier learning: An empirical study. Technical Report ML-TR 44, Rutgers University, 2001.
- [23] L. Wu, S. L. Oviatt, and P. R. Cohen. Multimodal integration - a statistical view. *IEEE Transactions on Multimedia*, 1(4):334–341, 1999.
- [24] D. Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 189–196, 1995.
- [25] B. Zadrozny and C. Elkan. Learning and making decisions when costs and probabilities are both unknown. In *Knowledge Discovery and Data Mining*, pages 204–213, 2001.