Journal of Artificial Intelligence Research 27 (2006) 25-53 Submitted 10/05; published 9/06

Generative Prior Knowledge for Discriminative Classification

Arkady Epshteyn AEPSHTEY@UIUC.EDU
Gerald DeJong DEJONG@UIUC.EDU
Department of Computer Science
University of Illinois at Urbana-Champaign
201 N. Goodwin
Urbana, IL, 61801 USA

Abstract

We present a novel framework for integrating prior knowledge into discriminative classifiers. Our framework allows discriminative classifiers such as Support Vector Machines (SVMs) to utilize prior knowledge specified in the generative setting. The dual objective of fitting the data and respecting prior knowledge is formulated as a bilevel program, which is solved (approximately) via iterative application of second-order cone programming. To test our approach, we consider the problem of using WordNet (a semantic database of English language) to improve low-sample classification accuracy of newsgroup categorization. WordNet is viewed as an approximate, but readily available source of background knowledge, and our framework is capable of utilizing it in a flexible way.

1. Introduction

While SVM (Vapnik, 1995) classification accuracy on many classification tasks is often competitive with that of human subjects, the number of training examples required to achieve this accuracy is prohibitively large for some domains. Intelligent user interfaces, for example, must adopt to the behavior of an individual user after a limited amount of interaction in order to be useful. Medical systems diagnosing rare diseases have to generalize well after seeing very few examples. Any natural language processing task that performs processing at the level of n-grams or phrases (which is frequent in translation systems) cannot expect to see the same sequence of words a sufficient number of times even in large training corpora. Moreover, supervised classification methods rely on manually labeled data, which can be expensive to obtain. Thus, it is important to improve classification performance on very small datasets. Most classifiers are not competitive with humans in their ability to generalize after seeing very few examples. Various techniques have been proposed to address this problem, such as active learning (Tong & Koller, 2000b; Campbell, Cristianini, & Smola, 2000), hybrid generative-discriminative classification (Raina, Shen, Ng, & McCallum, 2003), learning-to-learn by extracting common information from related learning tasks (Thrun, 1995; Baxter, 2000; Fink, 2004), and using prior knowledge.

In this work, we concentrate on improving small-sample classification accuracy with prior knowledge. While prior knowledge has proven useful for classification (Scholkopf, Simard, Vapnik, & Smola, 2002; Wu & Srihari, 2004; Fung, Mangasarian, & Shavlik, 2002; Epshteyn & DeJong, 2005; Sun & DeJong, 2005), it is notoriously hard to apply in practice because there is a mismatch between the form of prior knowledge that can be employed by classification algorithms (either prior probabilities or explicit constraints on the hypothesis space of the classifier) and the domain theories articulated by human experts. This is unfortunate because various ontologies and domain theories are available in abundance, but considerable amount of manual effort is required to incorporate existing prior knowledge into the native learning bias of the chosen algorithm. What would it take to apply an existing domain theory automatically to a classification task for which it was not specifically designed? In this work, we take the first steps towards answering this question.

In our experiments, such a domain theory is exemplified by WordNet, a linguistic database of semantic connections among English words (Miller, 1990). We apply WordNet to a standard benchmark task of newsgroup categorization. Conceptually, a generative model describes how the world works, while a discriminative model is inextricably linked to a specific classification task. Thus, there is reason to believe that a generative interpretation of a domain theory would seem to be more natural and generalize better across different classification tasks. In Section 2 we present empirical evidence that this is, indeed, the case with WordNet in the context of newsgroup classification. For this reason, we interpret the domain theory in the generative setting. However, many successful learning algorithms (such as support vector machines) are discriminative. We present a framework which allows the use of generative prior in the discriminative classification setting.

Our algorithm assumes that the generative distribution of the data is given in the Bayesian framework: $Prob(data|model)$ and the prior $Prob'(model)$ are known. However, instead of performing Bayesian model averaging, we assume that a single model M^* has been selected a-priori, and the observed data is a manifestation of that model (i.e., it is drawn according to $Prob(data|M^*)$). The goal of the learning algorithm is to estimate M^*. This estimation is performed as a two-player sequential game of full information. The bottom (generative) player chooses the Bayes-optimal discriminator function $f(M)$ for the probability distribution $Prob(data|model = M)$ (*without taking the training data into account*) given the model M. The model M is chosen by the top (discriminative) player in such a way that its prior probability of occurring, given by $Prob'(M)$, is high, *and* it forces the bottom player to minimize the training-set error of its Bayes-optimal discriminator $f(M)$. This estimation procedure gives rise to a bilevel program. We show that, while the problem is known to be NP-hard, its approximation can be solved efficiently by iterative application of second-order cone programming.

The only remaining issue is how to construct the generative prior $Prob'(model)$ automatically from the domain theory. We describe how to solve this problem in Section 2, where we also argue that the generative setting is appropriate for capturing expert knowledge, employing WordNet as an illustrative example. In Section 3, we give the necessary preliminary information and important known facts and definitions. Our framework for incorporating generative prior into discriminative classification is described in detail in Section 4. We demonstrate the efficacy of our approach experimentally by presenting the results of using WordNet for newsgroup classification in Section 5. A theoretical explanation of the improved generalization ability of our discriminative classifier constrained by generative prior knowledge appears in Section 6. Section 7 describes related work. Section 8 concludes the paper and outlines directions for future research.

2. Generative vs. Discriminative Interpretation of Domain Knowledge

WordNet can be viewed as a network, with nodes representing words and links representing relationships between two words (such as synonyms, hypernyms (is-a), meronyms (part-of), etc.). An important property of WordNet is that of semantic distance - the length (in links) of the shortest path between any two words. Semantic distance approximately captures the degree of semantic relatedness of two words. We set up an experiment to evaluate the usefulness of WordNet for the task of newsgroup categorization. Each posting was represented by a bag-of-words, with each binary feature representing the presence of the corresponding word. The evaluation was done on pairwise classification tasks in the following two settings:

1. The generative framework assumes that each posting $x = [x^1, .., x^n]$ is generated by a distinct probability distribution for each newsgroup. The simplest version of a Linear Discriminan Analysis (LDA) classifier posits that $x|(y = -1) \sim N(\mu_1, I)$ and $x|(y = 1) \sim N(\mu_2, I)$ for posting x given label $y \in \{-1, 1\}$, where $I \in \mathbb{R}^{(n \times n)}$ is the identity matrix. Classification is done by assigning the most probable label to x: $y(x) = 1 \Leftrightarrow Prob(x|1) > Prob(x|-1)$. It is well-known (e.g. see Duda, Hart, & Stork, 2001) that this decision rule is equivalent to the one given by the hyperplane $(\mu_2 - \mu_1)^T x - \frac{1}{2}(\mu_2^T \mu_2 - \mu_1^T \mu_1) > 0$. The means $\widehat{\mu_i} = [\widehat{\mu_i^1}, .., \widehat{\mu_i^n}]$ are estimated via maximum likelihood from the training data $[x_1, y_1], .., [x_m, y_m]$[1].

2. The discriminative SVM classifier sets the separating hyperplane to directly minimize the number of errors on the training data:
   $[\widehat{w} = [\widehat{w^1}, .., \widehat{w^n}], \widehat{b}] = \arg\min_{w,b} \|w\| \,\text{s.t.}\, y_i(w^T x_i + b) \geq 1, i = 1, .., m.$

Our experiment was conducted in the learning-to-learn framework (Thrun, 1995; Baxter, 2000; Fink, 2004). In the first stage, each classifier was trained using training data from the *training task* (e.g., for classifying postings into the newsgroups 'atheism' and 'guns'). In the second stage, the classifier was generalized using WordNet's semantic information. In the third stage, the generalized classifier was applied to a different, *test task* (e.g., for classifying postings for the newsgroups 'atheism' vs. 'mideast') *without seeing any data from this new classification task*. The only way for a classifier to generalize in this setting is to use the original sample to acquire information about WordNet, and then exploit this information to help it label examples from the test sample. In learning how to perform this task, the system also learns how to utilize the classification knowledge implicit in WordNet.

We now describe the second and third stages for the two classifiers in more detail:

1. It is intuitive to interpret information embedded in WordNet as follows: if the title of the newsgroup is 'guns', then all the words with the same semantic distance to 'gun' (e.g., 'artillery', 'shooter', and 'ordnance' with the distance of two) provide a similar degree of classification information. To quantify this intuition, let $l_{i,train} = [l_{i,train}^1, .., l_{i,train}^j, .., l_{i,train}^n]$ be the vector of semantic distances in WordNet between each feature word j and the label of each training task newsgroup $i \in \{1, 2\}$. Define

1. The standard LDA classifier assumes that $x|(y = -1) \sim N(\mu_1, \Sigma)$ and $x|(y = -1) \sim N(\mu_2, \Sigma)$ and estimates the covariance matrix Σ as well as the means μ_1, μ_2 from the training data. In our experiments, we take $\Sigma = I$.

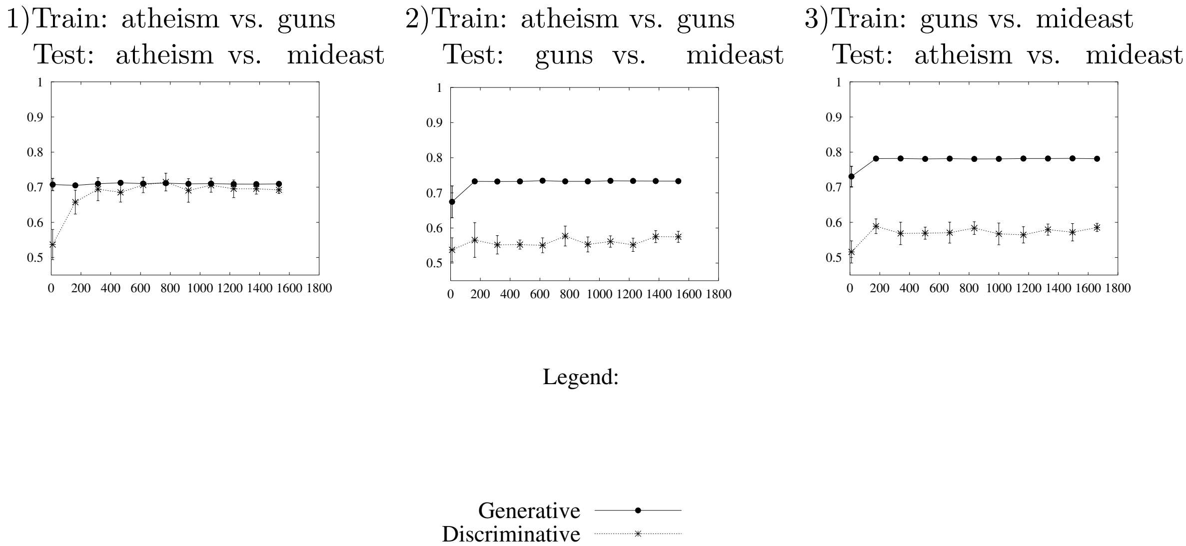

Figure 2.1: Test set accuracy as a percentage versus the number of training points for 3 different classification experiments. For each classification task, a random test set is chosen from the full set of articles in 20 different ways. Error bars are based on 95% confidence intervals.

$\chi_i(v) \triangleq \frac{\sum_{j:l^j_{i,train}=v} \widehat{\mu^j_i}}{|j:l^j_{i,train}=v|}$, $i = 1, 2$, where $|\cdot|$ denotes cardinality of a set. χ_i compresses information in $\widehat{\mu_i}$ based on the assumption that words equidistant from the newsgroup label are equally likely to appear in a posting from that newsgroup. To test the performance of this compressed classifier on a new task with semantic distances given by $l_{i,test}$, the generative distributions are reconstructed via $\mu^j_i := \chi_i(l^j_{i,test})$. Notice that if the classifier is trained and tested on the same task, applying the function χ_i is equivalent to averaging the components of the means of the generative distribution corresponding to the equivalence classes of words equidistant from the label. If the classifier is tested on a different classification task, the reconstruction process reassigns the averages based on the semantic distances to the new labels.

2. It is less intuitive to interpret WordNet in a discriminative setting. One possible interpretation is that coefficients w^j of the separating hyperplane are governed by semantic distances to labels, as captured by the compression function $\chi'(v, u) \triangleq \frac{\sum_{j:l^j_{1,train}=v, l^j_{2,train}=u} \widehat{w^j}}{|j:l^j_{1,train}=v, l^j_{2,train}=u|}$ and reconstructed via $w^j := \chi'(l^j_{1,test}, l^j_{2,test})$.

Note that both the LDA generative classifier and the SVM discriminative classifier have the same hypothesis space of separating hyperplanes. The resulting test set classification accuracy for each classifier for a few classification tasks from the 20-newsgroup dataset

(Blake & Merz, 1998) is presented in Figure 2.1. The x-axis of each graph represents the size of the training task sample, and the y-axis - the classifier's performance on the test classification task. The generative classifier consistently outperforms the discriminative classifier. It converges much faster, and on two out of three tasks the discriminative classifier is not able to use prior knowledge nearly as effectively as the generative classifier even after seeing 90% of all of the available training data. The generative classifier is also more consistent in its performance - note that its error bars are much smaller than those of the discriminative classifier. The results clearly show the potential of using background knowledge as a vehicle for sharing information between tasks. But the effective sharing is contingent on an appropriate task decomposition, here supplied by the tuned generative model.

The evidence in Figure 2.1 seemingly contradicts the conventional wisdom that discriminative training outperforms generative for sufficiently large training samples. However, our experiment evaluates the two frameworks in the context of using an ontology to transfer information between learning tasks. This was never done before. The experiment demonstrates that the interpretation of semantic distance in WordNet is more intuitive in the generative classification setting, probably because it better reflects the human intuitions behind WordNet.

However, our goal is not just to construct a classifier that performs well without seeing *any* examples of the test classification task. We also want a classifier that improves its behavior as it sees new labeled data from the test classification task. This presents us with a problem: one of the best-performing classifiers (and certainly the best on the text classification task according to the study by Joachims, 1998) is SVM, a discriminative classifier. Therefore, in the rest of this work, we focus on incorporating generative prior knowledge into the discriminative classification framework of support vector machines.

3. Preliminaries

It has been observed that constraints on the probability measure of a half-space can be captured by second-order cone constraints for Gaussian distributions (see, e.g., the tutorial by Lobo, Vandenberghe, Boyd, & Lebret, 1998). This allows for efficient processing of such constraints within the framework of second-order cone programming (SOCP). We intend to model prior knowledge with elliptical distributions, a family of probability distributions which generalizes Gaussians. In what follows, we give a brief overview of second-order cone programming and its relationship to constraints imposed on the Gaussian probability distribution. We also note that it is possible to extend the argument presented by Lobo et al. (1998) to elliptical distributions.

Second-order cone program is a mathematical program of the form:

$$\min_x v^T x \tag{3.1}$$

$$\text{s.t.} \ \|A_i x + b_i\| \le c_i^T x + d_i, \ i = 1, ..., N \tag{3.2}$$

where $x \in \mathbb{R}^n$ is the optimization variable and $v \in \mathbb{R}^n$, $A_i \in \mathbb{R}^{(k_i \text{x} n)}$, $b_i \in \mathbb{R}^{k_i}$, $c_i \in \mathbb{R}^n$, $d_i \in \mathbb{R}$ are problem parameters ($\|\cdot\|$ represents the usual L_2-norm in this paper). SOCPs can be solved efficiently with interior-point methods, as described by Lobo et al. (1998) in a tutorial which contains an excellent overview of the theory and applications of SOCP.

We use the elliptical distribution to model distribution of the data a-priori. Elliptical distributions are distributions with ellipsoidally-shaped equiprobable contours. The density function of the n-variate elliptical distribution has the form $f_{\mu,\Sigma,g}(x) = c(\det\Sigma)^{-1}g((x-\mu)^T\Sigma^{-1}(x-\mu))$, where $x \in \mathbb{R}^n$ is the random variable, $\mu \in \mathbb{R}^n$ is the location parameter, $\Sigma \in \mathbb{R}^{(nxn)}$ is a positive definite $(n \times n)$-matrix representing the scale parameter, function $g(\cdot)$ is the density generator, and c is the normalizing constant. We will use the notation $X \sim E(\mu, \Sigma, g)$ to denote that the random variable X has an elliptical distribution with parameters μ, Σ, g. Choosing appropriate density generator functions g, the Gaussian distribution, the Student-t distribution, the Cauchy distribution, the Laplace distribution, and the logistic distribution can be seen as special cases of the elliptical distribution. Using an elliptical distribution relaxes the restrictive assumptions the user has to make when imposing a Gaussian prior, while keeping many desirable properties of Gaussians, such as:

1. If $X \sim E(\mu, \Sigma, g)$, $A \in \mathbb{R}^{(k\times n)}$, and $B \in \mathbb{R}^k$, then $AX + B \sim E(A\mu + B, A\Sigma A^T, g)$
2. If $X \sim E(\mu, \Sigma, g)$, then $E(X) = \mu$.
3. If $X \sim E(\mu, \Sigma, g)$, then $Var(X) = \alpha_g\Sigma$, where α_g is a constant that depends on the density generator g.

The following proposition shows that for elliptical distributions, the constraint $P(w^Tx+b \geq 0) \leq \eta$ (i.e., the probability that X takes values in the half-space $\{w^Tx + b \geq 0\}$ is less than η) is equivalent to a second-order cone constraint for $\eta \leq \frac{1}{2}$:

Proposition 3.1. *If $X \sim E(\mu, \Sigma, g)$, $Prob(w^Tx+b \geq 0) \leq \eta \leq \frac{1}{2}$ is equivalent to $-(w^T\mu + b)/\beta_{g,\eta} \geq \left\|\Sigma^{1/2}w\right\|$, where $\beta_{g,\eta}$ is a constant which only depends on g and η.*

Proof. The proof is identical to the one given by Lobo (1998) and Lanckriet et al. (2001) for Gaussian distributions and is provided here for completeness:

$$\text{Assume } Prob(w^Tx + b \geq 0) \leq \eta. \tag{3.3}$$

Let $u = w^Tx+b$. Let $\overline{u}$ denote the mean of u, and σ denote its variance. Then the constraint 3.3 can be written as

$$Prob(\frac{u-\overline{u}}{\sqrt{\sigma}} \geq -\frac{\overline{u}}{\sqrt{\sigma}}) \leq \eta. \tag{3.4}$$

By the properties of elliptical distributions, $\overline{u} = w^T\mu + b$, $\sigma = \sqrt{\alpha_g}\left\|\Sigma^{1/2}w\right\|$, and $\frac{u-\overline{u}}{\sqrt{\sigma}} \sim E(0, 1, g)$. Thus, statement 3.4 above can be expressed as $Prob_{X\sim E(0,1,g)}(X \geq -\frac{w^T\mu+b}{\sqrt{\alpha_g}\left\|\Sigma^{1/2}w\right\|}) \leq \eta$, which is equivalent to $-\frac{w^T\mu+b}{\sqrt{\alpha_g}\left\|\Sigma^{1/2}w\right\|} \geq \Phi^{-1}(\eta)$, where $\Phi(z) = Prob_{X\sim E(0,1,g)}(X \geq z)$. The proposition follows with $\beta_{g,\eta} = \sqrt{\alpha_g}\Phi^{-1}(\eta)$. $\square$

Proposition 3.2. *For any monotonically decreasing g, $Prob_{X\sim E(\mu,\Sigma,g)}(x) \geq \delta$ is equivalent to $\left\|\Sigma^{-1/2}(x-\mu)\right\| \leq \varphi_{g,c,\Sigma}$, where $\varphi_{g,c,\Sigma,\delta} = g^{-1}(\frac{\delta|\Sigma|}{c})$ is a constant which only depends on g, c, Σ, δ.*

Proof. Follows directly from the definition of $Prob_{X\sim E(\mu,\Sigma,g)}(x)$. $\square$

4. Generative Prior via Bilevel Programming

We deal with the binary classification task: the classifier is a function $f(x)$ which maps instances $x \in \mathbb{R}^n$ to labels $y \in \{-1, 1\}$. In the generative setting, the probability densities $Prob(x|y = -1; \mu_1)$ and $Prob(x|y = 1; \mu_2)$ parameterized by $\mu = [\mu_1, \mu_2]$ are provided (or estimated from the data), along with the prior probabilities on class labels $\Pi(y = -1)$ and $\Pi(y = 1)$, and the Bayes optimal decision rule is given by the classifier

$$f(x|\mu) = sign(Prob(x|y = -1; \mu_1)\Pi(y = -1) - Prob(x|y = 1; \mu_2)\Pi(y = 1)),$$

where $sign(x) := 1$ if $x \geq 0$ and -1 otherwise. In LDA, for instance, the parameters μ_1 and μ_2 are the means of the two Gaussian distributions generating the data given each label.

Informally, our approach to incorporating prior knowledge is straightforward: we assume a two-level hierarchical generative probability distribution model. The low-level probability distribution of the data given the label $Prob(x|y; \mu)$ is parameterized by μ, which, in turn, has a known probability distribution $Prob'(\mu)$. The goal of the classifier is to estimate the values of the parameter vector μ from the training set of labeled points $[x_1, y_1]...[x_m, y_m]$. This estimation is performed as a two-player sequential game of full information. The bottom (generative) player, given μ, selects the Bayes optimal decision rule $f(x|\mu)$. The top (discriminative) player selects the value of μ which has a high probability of occurring (according to $Prob'(\mu)$) *and* which will force the bottom player to select the decision rule which minimizes the discriminative error on the training set. We now give a more formal specification of this training problem and formulate it as a bilevel program. Some of the assumptions are subsequently relaxed to enforce both tractability and flexibility.

We use an elliptical distribution $E(\mu_1, \Sigma_1, g)$ to model $X|y = -1$, and another elliptical distribution $E(\mu_2, \Sigma_2, g)$ to model $X|y = 1$. If the parameters $\mu_i, \Sigma_i, i = 1, 2$ are known, the Bayes optimal decision rule *restricted to the class of linear classifiers*[2] of the form $f_{w,b}(x) = sign(w^T x + b)$ is given by $f(x)$ which minimizes the probability of error among all linear discriminants: $Prob(error) = Prob(w^T x + b \geq 0|y = 1)\Pi(y = 1) + Prob(w^T x + b \leq 0|y = -1)\Pi(y = -1) = \frac{1}{2}(Prob_{X\sim E(\mu_1,\Sigma_1,g)}(w^T x + b \geq 0) + Prob_{X\sim E(\mu_2,\Sigma_2,g)}(w^T x + b \leq 0))$, assuming equal prior probabilities for both classes. We now model the uncertainty in the means of the elliptical distributions $\mu_i, i = 1, 2$ by imposing elliptical prior distributions on the locations of the means: $\mu_i \sim E(t_i, \Omega_i, g), i = 1, 2$. In addition, to ensure the optimization problem is well-defined, we maximize the margin of the hyperplane subject to the imposed generative probability constraints:

$$\min_{\mu_1,\mu_2} \|w\| \tag{4.1}$$

$$\text{s.t.} y_i(w^T x_i + b) \geq 1,\ i = 1, .., m \tag{4.2}$$

$$Prob_{\mu_i\sim E(t_i,\Omega_i,g)}(\mu_i) \geq \delta,\ i = 1, 2 \tag{4.3}$$

$$[w, b] \text{ solves } \min_{w,b}[Prob_{X\sim E(\mu_1,\Sigma_1,g)}(w^T x + b \geq 0) + Prob_{X\sim E(\mu_2,\Sigma_2,g)}(w^T x + b \leq 0)] \tag{4.4}$$

This is a bilevel mathematical program (i.e., an optimization problem in which the constraint region is implicitly defined by another optimization problem), which is strongly

2. A decision rule *restricted to some class of classifiers H* is optimal if its probability of error is no larger than that of any other classifier *in H* (Tong & Koller, 2000a).

NP-hard even when all the constraints and both objectives are linear (Hansen, Jaumard, & Savard, 1992). However, we show that it is possible to solve a reasonable approximation of this problem efficiently with several iterations of second-order cone programming. First, we relax the second-level minimization (4.4) by breaking it up into two constraints: $Prob_{X\sim E(\mu_1,\Sigma_1,g)}(w^T x + b \geq 0) \leq \eta$ and $Prob_{X\sim E(\mu_2,\Sigma_2,g)}(w^T x + b \leq 0) \leq \eta$. Thus, instead of looking for the Bayes optimal decision boundary, the algorithm looks for a decision boundary with low probability of error, where low error is quantified by the choice of η.

Propositions 3.1 and 3.2 enable us to rewrite the optimization problem resulting from this relaxation as follows :

$$\min_{\mu_1,\mu_2,w,b} \|w\| \tag{4.5}$$

$$\text{s.t.} y_i(w^T x_i + b) \geq 1,\ i = 1,..,m \tag{4.6}$$

$$Prob_{\mu_i\sim E(t_i,\Omega_i,g)}(\mu_i) \geq \delta,\ i = 1,2 \ \Leftrightarrow\ \left\|\Omega_i^{-1/2}(\mu_i - t_i)\right\| \leq \varphi, i = 1,2 \tag{4.7}$$

$$Prob_{X\sim E(\mu_1,\Sigma_1,g)}(w^T x + b \geq 0) \leq \eta \ \Leftrightarrow\ -\frac{w^T \mu_1 + b}{\left\|\Sigma_1^{1/2} w\right\|} \geq \beta \tag{4.8}$$

$$Prob_{X\sim E(\mu_2,\Sigma_2,g)}(w^T x + b \leq 0) \leq \eta \ \Leftrightarrow\ \frac{w^T \mu_2 + b}{\left\|\Sigma_2^{1/2} w\right\|} \geq \beta \tag{4.9}$$

Notice that the form of this program does not depend on the generator function g of the elliptical distribution - only constants β and φ depend on it. φ defines how far the system is willing to deviate from the prior in its choice of a generative model, and β bounds the tail probabilities of error (Type I and Type II) which the system will tolerate assuming its chosen generative model is correct. These constants depend both on the specific generator g and the amount of error the user is willing to tolerate. In our experiments, we select the values of these constants to optimize performance. Unless the user wants to control the probability bounds through these constants, it is sufficient to assume a-priori only that probability distributions (both prior and hyper-prior) are elliptical, without making any further commitments.

Our algorithm solves the above problem by repeating the following two steps:

1. Fix the top-level optimization parameters μ_1 and μ_2. This step combines the objectives of maximizing the margin of the classifier on the training data and ensuring that the decision boundary is (approximately) Bayes optimal with respect to the given generative probability densities specified by the μ_1, μ_2.

2. Fix the bottom-level optimization parameters w, b. Expand the feasible region of the program in step 1 as a function of μ_1, μ_2. This step fixes the decision boundary and pushes the means of the generative distribution as far away from the boundary as the constraint (4.7) will allow.

The steps are repeated until convergence (in practice, convergence is detected when the optimization parameters do not change appreciably from one iteration to the next). Each step of the algorithm can be formulated as a second-order cone program:

Step 1. Fix μ_1 and μ_2. Removing unnecessary constraints from the mathematical program above and pushing the objective into constraints, we get the following SOCP:

$$\min_{w,b} \rho \tag{4.10}$$

$$\text{s.t.} \rho \geq \|w\| \tag{4.11}$$

$$y_i(w^T x_i + b) \geq 1,\ i = 1, .., m \tag{4.12}$$

$$-\frac{w^T \mu_1 + b}{\left\|\Sigma_1^{1/2} w\right\|} \geq \beta \tag{4.13}$$

$$\frac{w^T \mu_2 + b}{\left\|\Sigma_2^{1/2} w\right\|} \geq \beta \tag{4.14}$$

Step 2. Fix w, b and expand the span of the feasible region, as measured by $\frac{w^T\mu_2+b}{\left\|\Sigma_2^{1/2}w\right\|} - \frac{w^T\mu_1+b}{\left\|\Sigma_1^{1/2}w\right\|}$. Removing unnecessary constraints, we get:

$$\max_{\mu_1,\mu_2} \frac{w^T \mu_2 + b}{\left\|\Sigma_2^{1/2} w\right\|} - \frac{w^T \mu_1 + b}{\left\|\Sigma_1^{1/2} w\right\|} \tag{4.15}$$

$$\text{s.t.} \left\|\Omega_i^{-1/2}(\mu_i - t_i)\right\| \leq \varphi,\ i = 1, 2 \tag{4.16}$$

The behavior of the algorithm is illustrated in Figure 4.1.

The following theorems state that the algorithm converges.

Theorem 4.1. *Suppose that the algorithm produces a sequence of iterates* $\left\{\mu_1^{(t)}, \mu_2^{(t)}, w^{(t)}, b^{(t)}\right\}_{t=0}^{\infty}$, *and the quality of each iterate is evaluated by its margin* $\left\|w^{(t)}\right\|$. *This evaluation function converges.*

Proof. Let $\mu_1^{(t)}, \mu_2^{(t)}$ be the values of the prior location parameters, and $w_1^{(t)}, b_1^{(t)}$ be the minimum error hyperplane the algorithm finds at the end of the t-th step. At the end of the $(t+1)$-st step, $w_1^{(t+1)}, b_1^{(t+1)}$ is still in the feasible region of the t-th step SOCP. This is true because the function $f\left(\frac{(w^{(t)})^T\mu_2+b^{(t)}}{\left\|\Sigma_2^{1/2}w^{(t)}\right\|}, -\frac{(w^{(t)})^T\mu_1+b^{(t)}}{\left\|\Sigma_1^{1/2}w^{(t)}\right\|}\right) = \frac{(w^{(t)})^T\mu_2+b^{(t)}}{\left\|\Sigma_2^{1/2}w^{(t)}\right\|} - \frac{(w^{(t)})^T\mu_1+b^{(t)}}{\left\|\Sigma_1^{1/2}w^{(t)}\right\|}$ is monotonically increasing in each one of its arguments when the other argument is fixed, and fixing μ_1 (or μ_2) fixes exactly one argument. If the solution $\mu_1^{(t+1)}, \mu_2^{(t+1)}$ at the end of the $(t+1)$-st step were such that $\frac{(w^{(t)})^T\mu_2^{(t+1)}+b^{(t)}}{\left\|\Sigma_2^{1/2}w^{(t)}\right\|} < \beta$, then f could be increased by fixing $\mu_1^{(t+1)}$ and using the value of $\mu_2^{(t)}$ from the beginning of the step which ensures that $\frac{(w^{(t)})^T\mu_2^{(t)}+b^{(t)}}{\left\|\Sigma_2^{1/2}w^{(t)}\right\|} \geq \beta$, which contradicts the observation that f is maximized at the end of the second step. The same contradiction is reached if $-\frac{(w^{(t)})^T\mu_1^{(t+1)}+b^{(t)}}{\left\|\Sigma_1^{1/2}w^{(t)}\right\|} < \beta$. Since the minimum error hyperplane from the previous iteration is in the feasible region at the start of the next iteration, the objective $\left\|w^{(t)}\right\|$ must decrease monotonically from one iteration to the next. Since it is bounded below by zero, the algorithm converges. □

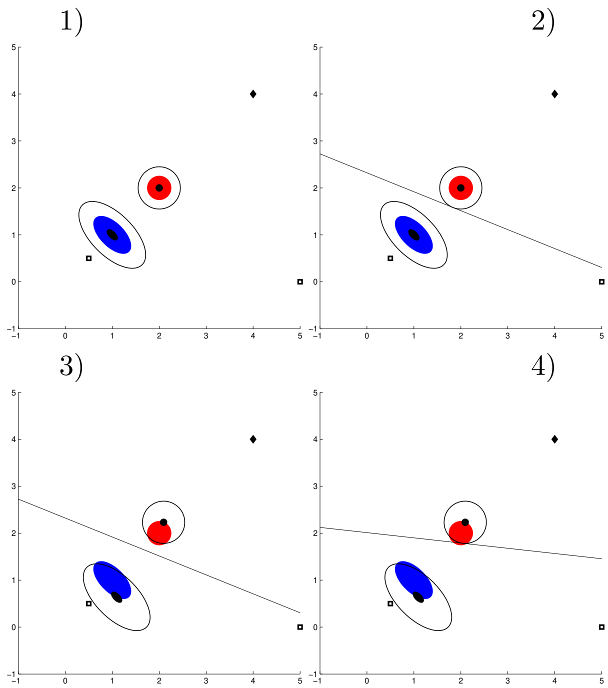

Figure 4.1: Steps of the iterative (hard-margin) SOCP procedure:
(The region where the hyperprior probability is larger than δ is shaded for each prior distribution. The covariance matrices are represented by equiprobable elliptical contours. In this example, the covariance matrices of the hyperprior and the prior distributions are multiples of each other. Data points from two different classes are represented by diamonds and squares.)

1. Data, prior, and hyperprior before the algorithm is executed.

2. Hyperplane discriminator at the end of step 1, iteration 1

3. Priors at the end of step 2, iteration 1

4. Hyperplane discriminator at the end of step 2, iteration 2

The algorithm converges at the end of step 2 for this problem (step 3 does not move the hyperplane).

In addition to the convergence of the objective function, the accumulation points of the sequence of iterates can be characterized by the following theorem:

Theorem 4.2. *The accumulation points of the sequence* $\left\{\mu_1^{(t)}, \mu_2^{(t)}, w^{(t)}, b^{(t)}\right\}$ *(i.e., limiting points of its convergent subsequences) have no feasible descent directions for the original optimization problem given by (4.5)-(4.9).*

Proof. See Appendix A. $\square$

If a point has no feasible descent directions, then any sufficiently small step along any directional vector will either increase the objective function, leave it unchanged, or take the algorithm outside of the feasible region. The set of points with no feasible descent directions is a subset of the set of local minima. Hence, convergence to such a point is a somewhat weaker result than convergence to a local minimum.

In practice, we observed rapid convergence usually within 2-4 iterations.

Finally, we may want to relax the strict assumptions of the correctness of the prior/linear separability of the data by introducing slack variables into the optimization problem above. This results in the following program:

$$\min_{\mu_1,\mu_2,w,b,\xi_i,\zeta_1,\zeta_2,\nu_1,\nu_2} \|w\| + C_1 \sum_{i=1}^{m} \xi_i + C_2(\zeta_1+\zeta_2) + C_3(\nu_1+\nu_2) \tag{4.17}$$

$$\text{s.t.} y_i(w^T x_i + b) \geq 1 - \xi_i,\ i = 1,..,m \tag{4.18}$$

$$\left\|\Omega_i^{-1/2}(\mu_i - t_i)\right\| \leq \varphi + \nu_i, i = 1,2 \tag{4.19}$$

$$-\frac{w^T\mu_1 + b}{\beta} \geq \left\|\Sigma_1^{1/2} w\right\| - \zeta_1 \tag{4.20}$$

$$\frac{w^T\mu_2 + b}{\beta} \geq \left\|\Sigma_2^{1/2} w\right\| - \zeta_2 \tag{4.21}$$

$$\xi_i \geq 0,\ i = 1,..,m \tag{4.22}$$

$$\nu_i \geq 0,\ i = 1,2 \tag{4.23}$$

$$\zeta_i \geq 0,\ i = 1,2 \tag{4.24}$$

As before, this problem can be solved with the two-step iterative SOCP procedure. Imposing the generative prior with soft constraints ensures that, as the amount of training data increases, the data overwhelms the prior and the algorithm converges to the maximum-margin separating hyperplane.

5. Experiments

The experiments were designed both to demonstrate the usefulness of the proposed approach for incorporation of generative prior into discriminative classification, and to address a broader question by showing that it is possible to use an existing domain theory to aid in a classification task for which it was not specifically designed. In order to construct the generative prior, the generative LDA classifier was trained on the data from the training classification task to estimate the Gaussian location parameters $\widehat{\mu_i}, i = 1,2$, as described in Section 2. The compression function $\chi_i(v)$ is subsequently computed (also as described in Section 2), and is used to set the hyperprior parameters via $\mu_i^j := \chi_i(l_{i,test}^j), i = 1,2$. In order to apply a domain theory effectively to the task for which it was not specifically designed, the algorithm must be able to estimate its confidence in the decomposition of the domain theory with respect to this new learning task. In order to model the uncertainty in applicability of WordNet to newsgroup categorization, our system estimated its confidence in homogeneity of equivalence classes of semantic distances by computing the variance of each

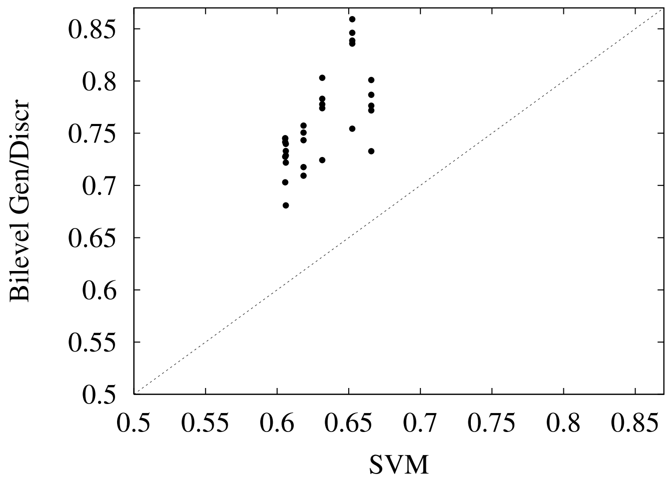

Figure 5.1: Performance of the bilevel discriminative classifier constrained by generative prior knowledge versus performance of SVM. Each point represents a unique pair of training/test tasks, with 0.5% of the test task data used for training. The results are averaged over 100 experiments.

random variable $\chi_i(v)$ as follows: $\sigma_i(v) \triangleq \frac{\sum_{j:l^j_{i,train=v}} (\widehat{\mu_i^j} - \chi_i(v))^2}{|j:l^j_{i,tran}=v|}$. The hyperprior confidence matrices $\Omega_i, i = 1, 2$ were then reconstructed with respect to the test task semantic distances $l_{i,test}, i = 1, 2$ as follows: $[\Omega_i]_{j,k} := \left\{ \begin{array}{c} \sigma_i(l^j_{i,test}), k = j \\ 0, k \neq j \end{array} \right.$. Identity matrices were used as covariance matrices of the lower-level prior: $\Sigma_1 = \Sigma_2 := I$. The rest of the parameters were set as follows: $\beta := 0.2, \varphi := 0.01$, $C_1 = C_2 := 1$, $C_3 := \infty$. These constants were chosen manually to optimize performance on Experiment 1 (for the training task: atheism vs. guns, test task: guns vs. mideast, see Figure 5.2) without observing any data from any other classification tasks.

The resulting classifier was evaluated in different experimental setups (with different pairs of newsgroups chosen for the training and the test tasks) to justify the following claims:

1. The bilevel generative/discriminative classifier with WordNet-derived prior knowledge has good low-sample performance, showing both the feasibility of automatically interpreting the knowledge embedded in WordNet and the efficacy of the proposed algorithm.

2. The bilevel classifier's performance improves with increasing training sample size.

3. Integrating generative prior into the discriminative classification framework results in better performance than integrating the same prior directly into the generative framework via Bayes' rule.

4. The bilevel classifier outperforms a state-of-the-art discriminative multitask classifier proposed by Evgeniou and Pontil (2004) by taking advantage of the WordNet domain theory.

In order to evaluate the low-sample performance of the proposed classifier, four newsgroups from the 20-newsgroup dataset were selected for experiments: *atheism, guns, middle east*, and *auto*. Using these categories, thirty experimental setups were created for all the possible ways of assigning newsgroups to training and test tasks (with a pair of newsgroups assigned to each task, under the constraint that the training and test pairs cannot be identical)[3]. In each experiment, we compared the following two classifiers:

1. Our bilevel generative-discriminative classifier with the knowledge transfer functions $\chi_i(v), \sigma_i(v), i = 1, 2$ learned from the labeled training data provided for the training task (using 90% of all the available data for that task). The resulting prior was subsequently introduced into the discriminative classification framework via our approximate bilevel programming approach

2. A vanilla SVM classifier which minimizes the regularized empirical risk:

$$\min_{w,b,\xi_i} \sum_{i=1}^{m} \xi_i + C_1 \|w\|^2 \tag{5.1}$$

$$\text{s.t.} y_i(w^T x_i + b) \geq 1 - \xi_i, i = 1, .., m \tag{5.2}$$

Both classifiers were trained on 0.5% of all the available data from the *test* classification task[4], and evaluated on the remaining 99.5% of the test task data. The results, averaged over one hundred randomly selected datasets, are presented in Figure 5.1, which shows the plot of the accuracy of the bilevel generative/discriminative classifier versus the accuracy of the SVM classifier, evaluated in each of the thirty experimental setups. All the points lie above the 45^o line, indicating improvement in performance due to incorporation of prior knowledge via the bilevel programming framework. The amount of improvement ranges from 10% to 30%, with all of the improvements being statistically significant at the 5% level.

The next experiment was conducted to evaluate the effect of increasing training data (from the test task) on the performance of the system. For this experiment, we selected three newsgroups (*atheism, guns,* and *middle east*) and generated six experimental setups based on all the possible ways of splitting these newsgroups into unique training/test pairs. In addition to the classifiers 1 and 2 above, the following classifiers were evaluated:

3. A state-of-the art multi-task classifier designed by Evgeniou and Pontil (2004). The classifier learns a set of related classification functions $f_t(x) = w_t^T x + b_t$ for classification tasks $t \in \{\text{training task, test task}\}$ given $m(t)$ data points $[x_{1t}, y_{1t}], .., [x_{m(t)t}, y_{m(t)t}]$

3. Newsgroup articles were preprocessed by removing words which could not be interpreted as nouns by WordNet. This preprocessing ensured that only one part of WordNet domain theory was exercised and resulted in virtually no reduction in classification accuracy.
4. SeDuMi software (Sturm, 1999) was used to solve the iterative SOCP programs.

for each task t by minimizing the regularized empirical risk:

$$\min_{w_0, w_t, b_t, \xi_{it}} \sum_t \sum_{i=1}^{m(t)} \xi_{it} + \frac{C_1}{C_2} \sum_t \|w_t - w_0\|^2 + C_1 \|w_0\|^2 \tag{5.3}$$

$$\text{s.t. } y_{it}(w_t^T x_{it} + b_t) \geq 1 - \xi_{it}, i = 1, .., m(t), \forall t \tag{5.4}$$

$$\xi_{it} \geq 0, i = 1, .., m(t), \forall t \tag{5.5}$$

The regularization constraint captures a tradeoff between final models w_t being close to the average model w_0 and having a large margin on the training data. 90% of the training task data was made available to the classifier. Constant $C_1 := 1$ was chosen, and $C_2 := 1000$ was selected from the set $\{.1, .5, 1, 2, 10, 1000, 10^5, 10^{10}\}$ to optimize the classifier's performance on Experiment 1 (for the training task: atheism vs. guns, test task: guns vs. mideast, see Figure 5.2) after observing .05% of the test task data (in addition to the training task data).

4. The LDA classifier described in Section 2 trained on 90% of the *test* task data. Since this classifier is the same as the bottom-level generative classifier used in the bilevel algorithm, its performance gives an upper bound on the performance of the bottom-level classifier trained in a generative fashion.

Figure 5.2 shows performance of classifiers 1-3 as a function of the size of the training data from the test task (evaluation was done on the remaining test-task data). The results are averaged over one hundred randomly selected datasets. The performance of the bilevel classifier improves with increasing training data both because the discriminative portion of the classifier aims to minimize the training error and because the generative prior is imposed with soft constraints. As expected, the performance curves of the classifiers converge as the amount of available training data increases. Even though the constants used in the mathematical program were selected in a single experimental setup, the classifier's performance is reasonable for a wide range of data sets across different experimental setups, with the possible exception of Experiment 4 (training task: guns vs. mideast, testing task: atheism vs. mideast), where the means of the constructed elliptical priors are much closer to each other than in the other experiments. Thus, the prior is imposed with greater confidence than is warranted, adversely affecting the classifier's performance.

The multi-task classifier 3 outperforms the vanilla SVM by generalizing from data points across classification tasks. However, it does not take advantage of prior knowledge, while our classifier does. The gain in performance of the bilevel generative/discriminative classifier is due to the fact that the relationship between the classification tasks is captured much better by WordNet than by simple linear averaging of weight vectors.

Because of the constants involved in both the bilevel classifier and the generative classifiers with Bayesian priors, it is hard to do a fair comparison between classifiers constrained by generative priors in these two frameworks. Instead, the generatively trained classifier 4 gives an empirical upper bound on the performance achievable by the bottom-level classifier trained generatively on the test task data. The accuracy of this classifier is shown as as a horizontal in the plots in Figure 5.2. Since discriminative classification is known to be superior to generative classification for this problem, the SVM classifier outperforms the

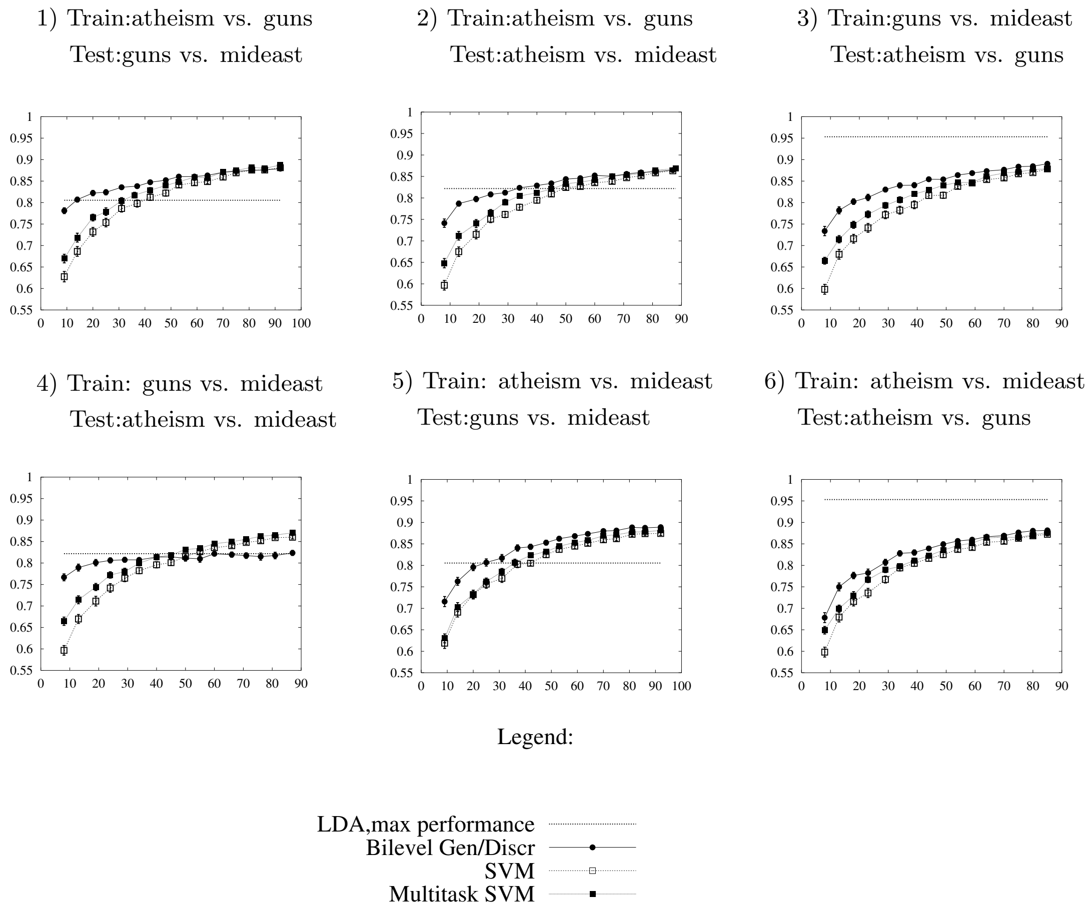

Figure 5.2: Test set accuracy as a percentage versus number of test task training points for two classifiers (SVM and Bilevel Gen/Discr) tested on six different classification tasks. For each classification experiment, the data set was split randomly into training and test sets in 100 different ways. The error bars based on 95% confidence intervals.

generative classifier given enough data in four out of six experimental setups. What is more interesting, is that, for a range of training sample sizes, the bilevel classifier constrained by the generative prior outperforms both the SVM trained on the same sample and the generative classifier trained on a much larger sample in these four setups. This means that, unless prior knowledge outweighs the effect of learning, it cannot enable the LDA classifier to compete with our bilevel classifier on those problems.

Finally, a set of experiments was performed to determine the effect of varying mathematical program parameters β and φ on the generalization error. Each parameter was varied over a set of values, with the rest of the parameters held fixed (β was increased up to its maximum feasible value). The evaluation was done in the setup of Experiment 1 (for

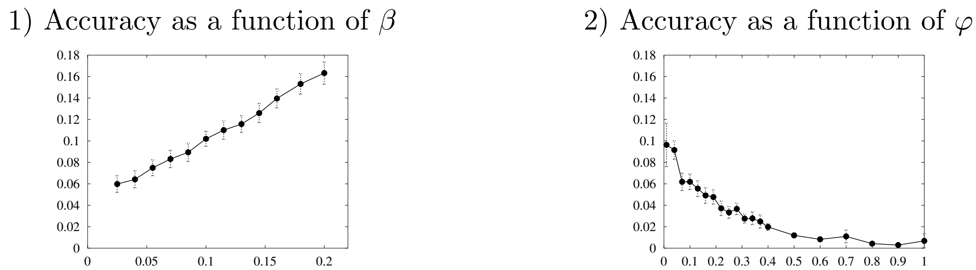

Figure 5.3: Plots of test set accuracy as percentage versus mathematical program parameter values. For each classification task, a random training set of size 9 was chosen from the full set of test task articles in 100 different ways. Error bars are based on 95% confidence intervals. All the experiments were performed on the training task: atheism vs. guns, test task: guns vs. mideast.

the training task:atheism vs. guns, test task: guns vs. mideast), with the training set size of 9 points. The results are presented in Figure 5.3. Increasing the value of β is equivalent to requiring a hyperplane separator to have smaller error given the prior. Decreasing the value of φ is equivalent to increasing the confidence in the hyperprior. Both of these actions tighten the constraints (i.e., decrease the feasible region). With good prior knowledge, this should have the effect of improving generalization performance for small training samples since the prior is imposed with higher confidence. This is precisely what we observe in the plots of Figure 5.3.

6. Generalization Performance

Why does the algorithm generalize well for low sample sizes? In this section, we derive a theorem which demonstrates that the convergence rate of the generalization error of the constrained generative-discriminative classifier depends on the parameters of the mathematical program and not just the margin, as would be expected in the case of large-margin classification without the prior. In particular, we show that as the certainty of the generative prior knowledge increases, the upper bound on the generalization error of the classifier constrained by the prior decreases. By increasing certainty of the prior, we mean that either the hyper-prior becomes more peaked (i.e., the confidence in the locations of the prior means increases) or the desired upper bounds on the Type I and Type II probabilities of error of the classifier decrease (i.e., the requirement that the lower-level discriminative player choose the restricted Bayes-optimal hyperplane is more strictly enforced).

The argument proceeds by bounding the fat-shattering dimension of the classifier constrained by prior knowledge. The fat-shattering dimension of a large margin classifier is given by the following definition (Taylor & Bartlett, 1998):

Definition 6.1. *A set of points $S = \{x^1...x^m\}$ is γ-shattered by a set of functions F mapping from a domain X to $\mathbb{R}$ if there are real numbers $r^1, ..., r^m$ such that, for each $b \in \{-1, 1\}^m$, there is a function f_b in F with $b(f_b(x^i) - r^i) \geq \gamma$, $i = 1..m$. We say* *that* $r^1, ..., r^m$ *witness the shattering. Then the fat-shattering dimension of* F *is a function* $fat_F(\gamma)$ *that maps* γ *to the cardinality of the largest* γ*-shattered set* S*.*

Specifically, we consider the class of functions

$$F = \{x \to w^T x : \|x\| \leq R, \|w\| = 1, \tag{6.1}$$
$$\frac{w^T(-\mu_1)}{\left\|\Sigma_1^{1/2} w\right\|} \geq \beta, \left\|\Omega_1^{-1/2}(\mu_1 - t_1)\right\| \leq \varphi, \frac{w^T \mu_2}{\left\|\Sigma_2^{1/2} w\right\|} \geq \beta, \left\|\Omega_2^{-1/2}(\mu_2 - t_2)\right\| \leq \varphi\}.$$

The following theorem bounds the fat-shattering dimension of our classifier:

Theorem 6.2. *Let* F *be the class of a-priori constrained functions defined by (6.1), and let* $\lambda_{min}(P)$ *and* $\lambda_{max}(P)$ *denote the minimum and maximum eigenvalues of matrix* P*, respectively. If a set of points* S *is* γ*-shattered by* F*, then* $|S| \leq \frac{4R^2(\alpha^2(1-\alpha^2))}{\gamma^2}$*, where* $\alpha = \max(\alpha_1, \alpha_2)$ *with* $\alpha_1 = \min(\frac{\lambda_{min}(\Sigma_1)\beta}{\|\mu_2\|}, \frac{\|t_2\|^2 - (\lambda_{max}(\Omega_2)\varphi)^2}{\|t_2\|(\lambda_{max}(\Omega_2)\varphi)^2 + \|t_2\|)})$ *and* $\alpha_2 = \min(\frac{\lambda_{min}(\Sigma_1)\beta}{\|\mu_1\|}, \frac{\|t_1\|^2 - (\lambda_{max}(\Omega_1)\varphi)^2}{\|t_1\|((\lambda_{max}(\Omega_1)\varphi)^2 + \|t_1\|)})$*, assuming that* $\beta \geq 0$*,* $\|t_i\| \geq \|t_i - \mu_i\|$*, and* $\alpha_i \geq \frac{1}{\sqrt{2}}, i = 1, 2$*.*

Proof. See Appendix B. □

We have the following corollary which follows directly from Taylor and Bartlett's (1998) Theorem 1.5 and bounds the classifier's generalization error based on its fat-shattering dimension:

Corollary 6.3. *Let* G *be a class of real-valued functions. Then, with probability at least* $1 - \delta$ *over* m *independently generated examples* z*, if a classifier* $h = sgn(g) \in sgn(G)$ *has margin at least* γ *on all the examples in* z*, then the error of* h *is no more than* $\frac{2}{m}(d * log(\frac{8em}{d})log(32m) + log(\frac{8m}{\delta}))$ *where* $d_G = fat_G(\frac{\gamma}{16})$*. If* $G = F$ *is the class of functions defined by (6.1), then* $d_F \leq \frac{265R^2(4(\alpha^2(1-\alpha^2)))}{\gamma^2}$*. If* $G = F'$ *is the usual class of large margin classifiers (without the prior), then the result in (Taylor & Bartlett, 1998) shows that* $d_{F'} \leq \frac{265R^2}{\gamma^2}$*.*

Notice that both bounds depend on $\frac{R^2}{\gamma^2}$. However, the bound of the classifier constrained by the generative prior also depends on β and φ through the term $4(\alpha^2(1-\alpha^2))$. In particular, as β increases, tightening the constraints, the bound decreases, ensuring, as expected, quicker convergence of the generalization error. Similarly, decreasing φ also tightens the constraints and decreases the upper bound on the generalization error. For $\alpha > \frac{1}{\sqrt{2}}$, the factor $4(\alpha^2(1-\alpha^2))$ is less than 1 and the upper bound on the fat-shattering dimension d_F is tighter than the usual bound in the no-prior case on $d_{F'}$.

Since β controls the amount of deviation of the decision boundary from the Bayes-optimal hyperplane and φ depends on the variance of the hyper-prior distribution, tightening of these constraints corresponds to increasing our confidence in the prior. Note that a high value β represents high level of user confidence in the generative elliptical model. Also note that there are two ways of increasing the tightness of the hyperprior constraint (4.7) - one is through the user-defined parameter φ, the other is through the automatically estimated covariance matrices $\Omega_i, i = 1, 2$. These matrices estimate the extent to which the equivalence classes defined by WordNet create an appropriate decomposition of the domain theory for the newsgroup categorization task. Thus, tight constraint (4.7) represents both high level of user confidence in the means of the generative classification model (estimated from WordNet) and a good correspondence between the partition of the words imposed by the semantic distance of WordNet and the elliptical generative model of the data. As φ approaches zero and β approaches its highest feasible value, the solution of the bilevel mathematical program reduces to the restricted Bayes optimal decision boundary computed solely from the generative prior distributions, without using the data.

Hence, we have shown that, as the prior is imposed with increasing level of confidence (which means that the elliptical generative model is deemed good, or the estimates of its means are good, which in turn implies that the domain theory is well-suited for the classification task at hand), the convergence rate of the generalization error of the classifier increases. Intuitively, this is precisely the desired effect of increased confidence in the prior since the benefit derived from the training data is outweighed by the benefit derived from prior knowledge. For low data samples, this should result in improved accuracy assuming the domain theory is good, which is what the plots in Figure 5.3 show.

7. Related Work

There are a number of approaches to combining generative and discriminative models. Several of these focus on deriving discriminative classifiers from generative distributions (Tong & Koller, 2000a; Tipping, 2001) or on learning the parameters of generative classifiers via discriminative training methods (Greiner & Zhou, 2002; Roos, Wettig, Grunwald, Myllymaki, & Tirri, 2005). The closest in spirit to our approach is the Maximum Entropy Discrimination framework (Jebara, 2004; Jaakkola, Meila, & Jebara, 1999), which performs discriminative estimation of parameters of a generative model, taking into account the constraints of fitting the data and respecting the prior. One important difference with our framework is that, in estimating these parameters, maximum entropy discrimination minimizes the distance between the generative model and the prior, subject to satisfying the discriminative constraint that the training data be classified correctly with a given margin. Our framework, on the other hand, maximizes the margin on the training data subject to the constraint that the generative model is not too far from the prior. This emphasis on maximizing the margin allows us to derive a-priori bounds on the generalization error of our classifier based on the confidence in the prior which are not (yet) available for the maximum entropy framework. Another difference is that our approach performs classification via a single generative model, while maximum entropy discrimination averages over a set of generative models weighted by their probabilities. This is similar to the distinction between maximum-a-posteriori and Bayesian estimation and has repercussions for tractability. Maximum entropy discrimination, however, is more general than our framework in a sense of allowing a richer set of behaviors based on different priors.

Ng et al. (2003, 2001) explore the relative advantages of discriminative and generative classification and propose a hybrid approach which improves classification accuracy for both low-sample and high-sample scenarios. Collins (2002) proposes to use the Viterbi algorithm for HMMs for inferencing (which is based on generative assumptions), combined with a discriminative learning algorithm for HMM parameter estimation. These research directions are orthogonal to our work since they do not explicitly consider the question of integration of prior knowledge into the learning problem.

In the context of support vector classification, various forms of prior knowledge have been explored. Scholkopf et al. (2002) demonstrate how to integrate prior knowledge about invariance under transformations and importance of local structure into the kernel function. Fung et al. (2002) use domain knowledge in form of labeled polyhedral sets to augment the training data. Wu and Srihari (2004) allow domain experts to specify their confidence in the example's label, varying the effect of each example on the separating hyperplane proportionately to its confidence. Epshteyn and DeJong (2005) explore the effects of rotational constraints on the normal of the separating hyperplane. Sun and DeJong (2005) propose an algorithm which uses domain knowledge (such as WordNet) to identify relevant features of examples and incorporate resulting information in form of soft constraints on the hypothesis space of SVM classifier. Mangasarian et al. (2004) suggest the use of prior knowledge for support vector regression. In all of these approaches, prior knowledge takes the form of explicit constraints on the hypothesis space of the large-margin classifier. In this work, the emphasis is on generating such constraints automatically from domain knowledge interpreted in the generative setting. As we demonstrate with our WordNet application, generative interpretation of background knowledge is very intuitive for natural language processing problems.

Second-order cone constraints have been applied extensively to model probability constraints in robust convex optimization (Lobo et al., 1998; Bhattacharyya, Pannagadatta, & Smola, 2004) and constraints on the distribution of the data in minimax machines (Lanckriet et al., 2001; Huang, King, Lyu, & Chan, 2004). Our work, as far as we know, is the first one which models prior knowledge with such constraints. The resulting optimization problem and its connection with Bayes optimal classification is very different from the approaches mentioned above.

Our work is also related to empirical Bayes estimation (Carlin & Louis, 2000). In empirical Bayes estimation, the hyper-prior parameters of the generative model are estimated using statistical estimation methods (usually maximum likelihood or method of moments) through the marginal distribution of the data, while our approach learns those parameters discriminatively using the training data.

8. Conclusions and Future Work.

Since many sources of domain knowledge (such as WordNet) are readily available, we believe that significant benefit can be achieved by developing algorithms for automatically applying their information to new classification problems. In this paper, we argued that the generative paradigm for interpreting background knowledge is preferable to the discriminative interpretation, and presented a novel algorithm which enables discriminative classifiers to utilize generative prior knowledge. Our algorithm was evaluated in the context of a complete system which, faced with the newsgroup classification task, was able to estimate the parameters needed to construct the generative prior from the domain theory, and use this construction to achieve improved performance on new newsgroup classification tasks.

In this work, we restricted our hypothesis class to that of linear classifiers. Extending the form of the prior distribution to distributions other than elliptical and/or looking for

Bayes-optimal classifiers restricted to a more expressive class than that of linear separators may result in improvement in classification accuracy for non linearly-separable domains. However, it is not obvious how to approximate this more expressive form of prior knowledge with convex constraints. The kernel trick may be helpful in handling nonlinear problems, assuming that it is possible to represent the optimization problem exclusively in terms of dot products of the data points and constraints. This is an important issue which requires further study.

We have demonstrated that interpreting domain theory in the generative setting is intuitive and produces good empirical results. However, there are usually multiple ways of interpreting a domain theory. In WordNet, for instance, semantic distance between words is only one measure of information contained in the domain theory. Other, more complicated, interpretations might, for example, take into account types of links on the path between the words (hypernyms, synonyms, meronyms, etc.) and exploit commonsense observations about WordNet such as words that are closer to the category label are more likely to be informative than words farther away. Comparing multiple ways of constructing the generative prior from the domain theory and, ultimately, selecting one of these interpretations automatically is a fruitful direction for further research.

Acknowledgments

The authors thank the anonymous reviewers for valuable suggestions on improving the paper. This material is based upon work supported in part by the National Science Foundation under Award NSF IIS 04-13161 and in part by the Information Processing Technology Office of the Defense Advanced Research Projects Agency under award HR0011-05-1-0040. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation or the Defense Advanced Research Projects Agency.

Appendix A. Convergence of the Generative/Discriminative Algorithm

Let the map $H : Z \to Z$ determine an algorithm that, given a point $\mu^{(0)}$, generates a sequence $\{\mu^{(t)}\}_{t=0}^{\infty}$ of iterates through the iteration $\mu^{(t+1)} = H(\mu^{(t)})$. The iterative algorithm in Section 4 generates a sequence of iterates $\mu^{(t)} = [\mu_1^{(t)}, \mu_2^{(t)}] \in Z$ by applying the following map H:

$$H = H_2 \circ H_1 : \tag{A.1}$$

$$\text{In step 1, } H_1([\mu_1, \mu_2]) = \arg \min_{[w,b] \in U([\mu_1,\mu_2])} \|w\|, \tag{A.2}$$

$$\text{with the set } U([\mu_1, \mu_2]) \text{ defined by constraints:} \tag{A.3}$$

$$y_i(w^T x_i + b) - 1 \geq 0, i = 1, .., m \tag{A.4}$$

$$c_{-1}(w, b; \mu_1, \Sigma_1) - \beta \geq 0 \tag{A.5}$$

$$c_1(w, b; \mu_2, \Sigma_2) - \beta \geq 0 \tag{A.6}$$

with the conic constraints $c_s(w, b; \mu, \Sigma) \triangleq s\left(\frac{w^T\mu+b}{\left\|\Sigma^{1/2}w\right\|}\right)$.

$$\text{In step 2, } H_2(w,b) = \arg\min_{(\mu_1,\mu_2)\in V} -(c_{-1}(w,b;\mu_1,\Sigma_1) + c_1(w,b;\mu_2,\Sigma_2)) \tag{A.7}$$

with the set V given by the constraints

$$\varphi - o(\mu_1;\Omega_1,t_1) \geq 0 \tag{A.8}$$

$$\varphi - o(\mu_2;\Omega_2,t_2) \geq 0 \tag{A.9}$$

with $o(\mu;\Omega,t) \triangleq \left\|\Omega^{-1/2}(\mu - t)\right\|$.

Notice that H_1 and H_2 are functions because the minima for optimization problems (4.10)-(4.14) and (4.15)-(4.16) are unique. This is the case because Step 1 optimizes a strictly convex function on a convex set, and Step 2 optimizes a linear non-constant function on a strictly convex set.

Convergence of the objective function $\psi(\mu^{(t)}) \triangleq \min_{[w,b]\in U([\mu_1^{(t)},\mu_2^{(t)}])} \|w\|$ of the algorithm was shown in Theorem 4.1. Let Γ denote the set of points on which the map H does not change the value of the objective function, i.e. $\mu^* \in \Gamma \Leftrightarrow \psi(H(\mu^*)) = \psi(\mu^*)$. We will show that every accumulation point of $\{\mu^{(t)}\}$ lies in Γ. We will also show that every point $[\mu_1^*,\mu_2^*] \in \Gamma$ augmented with $[w^*,b^*] = H_1([\mu_1^*,\mu_2^*])$ is a point with no feasible descent directions for the optimization problem (4.5)-(4.9), which can be equivalently expressed as:

$$\min_{\mu_1,\mu_2,w,b} \|w\| \,\text{s.t.} [\mu_1,\mu_2] \in V; [w,b] \in U([\mu_1,\mu_2]) \tag{A.10}$$

In order to formally state our result, we need a few concepts from the duality theory. Let a constrained optimization problem be given by

$$\min_x f(x) \,\text{s.t.}\, c_i(x) \geq 0, i = 1,..,k \tag{A.11}$$

The following conditions, known as Karush-Kuhn-Tucker(KKT) conditions are necessary for x^* to be a local minimum:

Proposition A.1. *If x^* is a local minimum of (A.11), then $\exists \lambda_1,..,\lambda_k$ such that*

1. $\nabla f(x^*) = \sum_{i=1}^k \lambda_i \nabla c_i(x^*)$
2. $\lambda_i \geq 0$ *for* $\forall i \in \{1,..,k\}$
3. $c_i(x^*) \geq 0$ *for* $\forall i \in \{1,..,k\}$
4. $\lambda_i c_i(x^*) = 0$ *for* $\forall i \in \{1,..,k\}$

$\lambda_1,..,\lambda_k$ are known as Lagrange multipliers of constraints $c_1,..,c_k$.

The following well-known result states that KKT conditions are sufficient for x^* to be a point with no feasible descent directions:

Proposition A.2. *If $\exists \lambda_1,..,\lambda_k$ such that the following conditions are satisfied at x^*:*

1. $\nabla f(x^*) = \sum_{i=1}^k \lambda_i \nabla c_i(x^*)$

2. $\lambda_i \geq 0$ *for* $\forall i \in \{1, .., k\}$

then x^* *has no feasible descent directions in the problem (A.11)*

Proof. *(sketch)* We reproduce the proof given in a textbook by Fletcher (1987). The proposition is true because for any feasible direction vector s, $s^T \nabla c_i(x) \geq 0$ for $\forall x$ and for $\forall i \in \{1, .., k\}$. Hence, $s^T \nabla f(x^*) = \sum_{i=1}^{k} \lambda_i s^T \nabla c_i(x^*) \geq 0$, so s is not a descent direction. □

The following lemma characterizes the points in the set Γ:

Lemma A.3. *Let* $\mu^* \in \Gamma$, *and let* $[w^*, b^*] = H_1(\mu^*)$ *be the optimizer of* $\psi(\mu^*)$, *and let* $\boldsymbol{\lambda}^* = [\lambda^*_{(A.4),1}, .., \lambda^*_{(A.4),m}, \lambda^*_{(A.5)}, \lambda^*_{(A.6)}]$ *be a set of Lagrange multipliers corresponding to the constraints for the solution* $[w^*, b^*]$. *Define* $\mu' = H(\mu^*)$, *and let* $[w', b']$ *be the optimizer of* $\psi(\mu')$. *If* $\mu'_2 \neq \mu^*_2$, *then* $\lambda^*_{(A.6)} = 0$ *for some* $\boldsymbol{\lambda}^*$. *If* $\mu'_1 \neq \mu^*_1$, *then* $\lambda^*_{(A.5)} = 0$ *for some* $\boldsymbol{\lambda}^*$. *If both* $\mu'_1 \neq \mu^*_1$ *and* $\mu'_2 \neq \mu^*_2$, *then* $\lambda^*_{(A.6)} = \lambda^*_{(A.5)} = 0$ *for some* $\boldsymbol{\lambda}^*$.

Proof. Consider the case when

$$\mu'_2 \neq \mu^*_2 \tag{A.12}$$

and

$$\mu'_1 = \mu^*_1 \tag{A.13}$$

Since $\mu^* \in \Gamma$, $\|w'\| = \|w^*\|$. Let $\boldsymbol{\lambda}'$ be a set of Lagrange multipliers corresponding to the constraints for the solution $[w', b']$. Since w^* is still feasible for the optimization problem given by $\psi(\mu')$ (by the argument in Theorem 4.1) and the minimum of this problem is unique, this can only happen if

$$[w', b'] = [w^*, b^*]. \tag{A.14}$$

Then $[w^*, b^*]$ and $\boldsymbol{\lambda}'$ must satisfy KKT conditions for $\psi(\mu')$. (A.12) implies that $c_1(w^*; \mu'_2, \Sigma_2) > c_1(w^*; \mu^*_2, \Sigma_2) \geq \beta$ by the same argument as in Theorem 4.1, which means that, by KKT condition (4) for $\psi(\mu')$,

$$\lambda'_{(\text{A.6})} = 0. \tag{A.15}$$

Therefore, by KKT condition (1) for $\psi(\mu')$ and (A.15), at $[w, b, \mu_1, \mu_2] = [w^* = w', b^* = b', \mu^*_1 = \mu'_1, \mu^*_2]$

$$\begin{bmatrix} \frac{\partial \|w\|}{\partial w} \\ \frac{\partial \|w\|}{\partial b} \end{bmatrix} = \sum_{i=1}^{m} \lambda'_{(\text{A.4}),i} \begin{bmatrix} y_i x_i \\ y_i \end{bmatrix} + \lambda'_{(\text{A.5})} \begin{bmatrix} \frac{\partial c_{-1}(w, b^*; \mu^*_1, \Sigma_1)}{\partial w} \\ \frac{\partial c_{-1}(w^*, b; \mu^*_1, \Sigma_1)}{\partial b} \end{bmatrix} + \lambda'_{(\text{A.6})} \begin{bmatrix} \frac{\partial c_1(w, b^*; \mu^*_2, \Sigma_2)}{\partial w} \\ \frac{\partial c_1(w^*, b; \mu^*_2, \Sigma_2)}{\partial b} \end{bmatrix},$$

which means that KKT conditions (1),(2) for the optimization problem $\psi(\mu^*)$ are satisfied at the point $[w^*, b^*]$ with $\boldsymbol{\lambda}^* = \boldsymbol{\lambda}'$. KKT condition (3) is satisfied by feasibility of $[w^*, b^*]$ and KKT condition (4) is satisfied by the same condition for $\psi(\mu')$ and observations (A.13), (A.14), and (A.15).

The proofs for the other two cases ($\mu'_2 = \mu^*_2, \mu'_1 \neq \mu^*_1$ and $\mu'_2 \neq \mu^*_2, \mu'_1 \neq \mu^*_1$) are analogous. □

The following theorem states that the points in Γ are KKT points (i.e., points at which KKT conditions are satisfied) for the optimization problem given by (A.10).

Theorem A.4. *If $\mu^* \in \Gamma$ and let $[w^*, b^*] = H_1(\mu^*)$, then $[w^*, b^*, \mu_1^*, \mu_2^*]$ is a KKT point for the optimization problem given by (A.10).*

Proof. Let $\mu' = H(\mu^*)$. Just like in Lemma A.3, we only consider the case

$$\mu_2' \neq \mu_2^*, \tag{A.16}$$

$$\mu_1' = \mu_1^* \Rightarrow \lambda^*_{(\text{A.6})} = 0 \text{ (by Lemma A.3).} \tag{A.17}$$

(the proofs for the other two cases are similar).

By KKT conditions for $H_2(w^*, b^*)$, at $\mu_1 = \mu_1'$

$$-\frac{\partial c_{-1}(w^*, b^*; \mu_1, \Sigma_1)}{\partial \mu_1} = \lambda'_{\text{A.8}} \frac{\partial(-o(\mu_1; \Omega_1, t))}{\partial \mu_1} \text{ for some } \lambda'_{\text{A.8}} \geq 0. \tag{A.18}$$

By KKT conditions for $H_1(\mu^*)$ and (A.17), at $[w, b] = [w^*, b^*]$

$$\begin{bmatrix} \frac{\partial\|w\|}{\partial w} \\ \frac{\partial\|w\|}{\partial b} \end{bmatrix} = \sum_{i=1}^{m} \lambda^*_{(\text{A.4}),i} \begin{bmatrix} y_i x_i \\ y_i \end{bmatrix} + \lambda^*_{(\text{A.5})} \begin{bmatrix} \frac{\partial c_{-1}(w, b^*; \mu_1^*, \Sigma_1)}{\partial w} \\ \frac{\partial c_{-1}(w^*, b; \mu_1^*, \Sigma_1)}{\partial b} \end{bmatrix} \text{ for some } \begin{bmatrix} \lambda^*_{(\text{A.4}),1} \\ .. \\ \lambda^*_{(\text{A.4}),m} \\ \lambda^*_{(\text{A.5})} \end{bmatrix} \succeq 0. \tag{A.19}$$

By (A.16),(A.17),(A.18), and (A.19), at $[w, b, \mu_1, \mu_2] = [w^*, b^*, \mu_1^* = \mu_1', \mu_2^*]$

$$\begin{bmatrix} \frac{\partial\|w\|}{\partial w} \\ \frac{\partial\|w\|}{\partial b} \\ \frac{\partial\|w\|}{\partial \mu_1} \\ \frac{\partial\|w\|}{\partial \mu_2} \end{bmatrix} = \begin{bmatrix} \frac{\partial\|w\|}{\partial w} \\ 0 \\ 0 \\ 0 \end{bmatrix} = \sum_{i=1}^{m} \lambda^*_{(\text{A.4}),i} \begin{bmatrix} y_i x_i \\ y_i \\ 0 \\ 0 \end{bmatrix} + \lambda^*_{(\text{A.5})} \begin{bmatrix} \frac{\partial c_{-1}(w, b^*; \mu_1^*, \Sigma_1)}{\partial w} \\ \frac{\partial c_{-1}(w^*, b; \mu_1^*, \Sigma_1)}{\partial b} \\ \frac{\partial c_{-1}(w^*, b^*; \mu_1, \Sigma_1)}{\partial \mu_1} \\ 0 \end{bmatrix} +$$

$$\lambda'_{\text{A.8}} \lambda^*_{(\text{A.5})} \begin{bmatrix} 0 \\ 0 \\ \frac{\partial(-o(\mu_1; \Omega_1, t))}{\partial \mu_1} \\ 0 \end{bmatrix} + \lambda^*_{(\text{A.6})} \begin{bmatrix} \frac{\partial c_1(w, b^*; \mu_2^*, \Sigma_2)}{\partial w} \\ \frac{\partial c_1(w^*, b; \mu_2^*, \Sigma_2)}{\partial b} \\ 0 \\ \frac{\partial c_1(w^*, b^*; \mu_2, \Sigma_2)}{\partial \mu_2} \end{bmatrix} + \lambda^*_{(\text{A.6})} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{\partial(-o(\mu_2; \Omega_2, t))}{\partial \mu_2} \end{bmatrix},$$

which means that KKT conditions (1),(2) for the optimization problem (A.10) are satisfied at the point $[w^*, b^*, \mu_1^*, \mu_2^*]$ with $\boldsymbol{\lambda}'' = [\lambda^*_{(\text{A.4}),1}, .., \lambda^*_{(\text{A.4}),m}, \lambda^*_{(\text{A.5})}, \lambda^*_{(\text{A.6})}, \lambda'_{\text{A.8}} \lambda^*_{(\text{A.5})}, \lambda^*_{(\text{A.6})}]$. $\boldsymbol{\lambda}''$ also satisfies KKT conditions (3),(4) by assumption (A.17) and the KKT conditions for H_1 and H_2. $\square$

In order to prove convergence properties of the iterates $\mu^{(t)}$, we use the following theorem due to Zangwill (1969):

Theorem A.5. *Let the map $H : Z \to Z$ determine an iterative algorithm via $\mu^{(t+1)} = H(\mu^{(t)})$, let $\psi(\mu)$ denote the objective function, and let Γ be the set of points on which the map H does not change the value of the objective function, i.e. $\mu \in \Gamma \Leftrightarrow \psi(H(\mu)) = \psi(\mu)$. Suppose*

1. *H is uniformly compact on Z, i.e. there is a compact subset $Z_0 \subseteq Z$ such that $H(\mu) \in Z_0$ for $\forall \mu \in Z$.*

2. *H is strictly monotonic on $Z - \Gamma$, i.e. $\psi(H(\mu)) < \psi(\mu)$.*

3. *H is closed on $Z - \Gamma$, i.e. if $w_i \to w$ and $H(w_i) \to \xi$, then $\xi = H(w)$.*

Then the accumulation points of the sequence of $\mu^{(t)}$ lie in Γ.

The following proposition shows that minimization of a continuous function on a feasible set which is a continuous map of the function's argument forms a closed function.

Proposition A.6. *Given*

1. *a real-valued continuous function f on $A \times B$,*

2. *a point-to-set map $U : A \to 2^B$ continuous with respect to the Hausdorff metric:*[5] *$dist(X, Y) \triangleq \max(d(X, Y), d(Y, X))$, where $d(X, Y) \triangleq \max_{x \in X} \min_{y \in Y} \|x - y\|$,*

define the function $F : A \to B$ by

$$F(a) = \arg\min_{b' \in U(a)} f(a, b') = \{b : f(a, b) < f(a, b') \text{ for } \forall b' \in U(a)\},$$

assuming the minimum exists and is unique. Then, the function F is closed at a.

Proof. This proof is a minor modification of the one given by Gunawardana and Byrne (2005). Let $\{a^{(t)}\}$ be a sequence in A such that

$$a^{(t)} \to a, F(a^{(t)}) \to b \tag{A.20}$$

The function F is closed at a if $F(a) = b$. Suppose this is not the case, i.e. $b \neq F(a) = \arg\min_{b' \in U(a)} f(a, b')$. Therefore,

$$\exists \hat{b} = \arg\min_{b' \in U(a)} f(b') \text{ such that } f(a, b) > f(a, \hat{b}) \tag{A.21}$$

By continuity of $f(\cdot, \cdot)$ and (A.20),

$$f(a^{(t)}, F(a^{(t)})) \to f(a, b) \tag{A.22}$$

By continuity of $U(\cdot)$ and (A.20),

$$dist(U(a^{(t)}), U(a)) \to 0 \Rightarrow \exists \hat{b}^{(t)} \to \hat{b} \text{ and } \hat{b}^{(t)} \in U(a^t), \text{ for } \forall t. \tag{A.23}$$

(A.22), (A.23), and (A.21) imply that

$$\exists K \text{ such that } f(a^{(t)}, F(a^{(t)})) > f(a^{(t)}, \hat{b}^{(t)}), \text{ for } \forall t > K \tag{A.24}$$

which is a contradiction since by assumption, $F(a^{(t)}) = \arg\min_{b' \in U(a^t)} f(b')$ and by (A.24), $\hat{b}^{(t)} \in U(a^{(t)})$. ∎

5. A point-to-set map $U(a)$ maps a point a to a set of points. $U(a)$ is continuous with respect to a distance metric $dist$ iff $a^{(t)} \to a$ implies $dist(U(a^{(t)}), U(a)) \to 0$.

Proposition A.7. *The function H defined by (A.1)-(A.7) is closed.*

Proof. Let $\{\mu^{(t)}\}$ be a sequence such that $\mu^{(t)} \to \mu^*$. Since all the iterates $\mu^{(t)}$ lie in the closed feasible region bounded by constraints (4.6)-(4.9) and the boundary of $U(\mu)$ is piecewise linear in μ, the boundary of $U(\mu^{(t)})$ converges uniformly to the boundary of $U(\mu^*)$ as $\mu^{(t)} \to \mu^*$, which implies that the Hausdorff distance between the boundaries converges to zero. Since the Hausdorff distance between convex sets is equal to the Hausdorff distance between their boundaries, $dist(U(\mu^{(t)}), U(\mu^*))$ also converges to zero. Hence, proposition A.6 implies that H_1 is closed. The same proposition implies that H_2 is closed. A composition of closed functions is closed, hence H is closed. □

We now prove the main result of this Section:

Theorem 4.2. *Let H be the function defined by (A.1)-(A.7) which determines the generative/discriminative algorithm via $\mu^{(t+1)} = H(\mu^{(t)})$. Then accumulation points μ^* of the sequence $\mu^{(t)}$ augmented with $[w^*, b^*] = H_1(\mu^*)$ have no feasible descent directions for the original optimization problem given by (4.5)-(4.9).*

Proof. The proof is by verifying that H satisfies the properties of Theorem A.5. Closedness of H was shown in Proposition A.7. Strict monotonicity of $\psi(\mu^{(t)})$ was shown in Theorem 4.1. Since all the iterates $\mu^{(t)}$ are in the closed feasible region bounded by constraints (4.6)-(4.9), H is uniformly compact on Z. Since all the accumulation points μ^* lie in Γ, they are KKT points of the original optimization problem by Theorem A.4, and, therefore, have no feasible descent directions by Proposition A.2. □

Appendix B. Generalization of the Generative/Discriminative Classifier

We need a few auxiliary results before proving Theorem 6.2. The first proposition bounds the angle of rotation between two vectors w_1, w_2 and the distance between them if the angle of rotation between each of these vectors and some reference vector v is sufficiently small:

Proposition B.1. *Let $\|w_1\| = \|w_2\| = \|v\| = 1$. If $w_1^T v \geq \alpha \geq 0$ and $w_2^T v \geq \alpha \geq 0$, then*

1. $w_1^T w_2 \geq 2\alpha^2 - 1$
2. $\|w_1 - w_2\| \leq 2\sqrt{(1-\alpha^2)}$

Proof.

1. By the triangle inequality, $\arccos(w_1^T w_2) \leq \arccos(w_1^T v) + \arccos(w_2^T v) \leq 2\arccos(\alpha)$ (since the angle between two vectors is a distance measure). Taking cosines of both sides and using trigonometric equalities yields $w_1^T w_2 \geq 2\alpha^2 - 1$.
2. Expand $\|w_1 - w_2\|^2 = \|w_1\|^2 + \|w_2\|^2 - 2w_1^T w_2 = 2(1 - w_1^T w_2)$. Since $w_1^T w_2 \geq 2\alpha^2 - 1$ from part 1, $\|w_1 - w_2\|^2 \leq 4(1-\alpha^2)$.

□

The next proposition bounds the angle of rotation between two vectors t and μ if they are not too far away from each other as measured by the L_2-norm distance:

Proposition B.2. *Let* $\|t\| = \nu$, $\|\mu - t\| \leq \tau$. *Then* $\frac{t^T\mu}{\|t\|\|\mu\|} \geq \frac{\nu^2-\tau^2}{\nu(\nu+\tau)}$.

Proof. Expanding $\|\mu - t\|^2 = \|t\|^2 + \|\mu\|^2 - 2t^t\mu$ and using $\|\mu - t\|^2 \leq \tau^2$, we get $\frac{t^T\mu}{\|t\|\|\mu\|} \geq \frac{1}{2}(\frac{\|t\|}{\|\mu\|} + \frac{\|\mu\|}{\|t\|} - \frac{\tau^2}{\|t\|\|\mu\|})$. We now use the triangle inequality $\nu - \tau \leq \|t\| - \|\mu - t\| \leq \|\mu\| \leq \|t\| + \|\mu - t\| \leq \nu + \tau$ and simplify. □

The following proposition will be used to bound the angle of rotation between the normal w of the separating hyperplane and the mean vector t of the hyper-prior distribution:

Proposition B.3. *Let* $\frac{w^T\mu}{\|w\|\|\mu\|} \geq \beta \geq 0$ *and* $\|\mu - t\| \leq \varphi \leq \|t\|$. *Then* $\frac{w^Tt}{\|w\|\|t\|} \geq (2\alpha^2 - 1)$, *where* $\alpha = \min(\beta, \frac{\|t\|^2-\varphi^2}{\|t\|(\varphi+\|t\|)})$.

Proof. Follows directly from Propositions B.1 (part 1) and B.2. □

We now prove Theorem 6.2, which relies on parts of the well-known proof of the fat-shattering dimension bound for large margin classifiers derived by Taylor and Bartlett (1998).

Theorem 6.2. *Let F be the class of a-priori constrained functions defined by 6.1, and let $\lambda_{min}(P)$ and $\lambda_{max}(P)$ denote the minimum and maximum eigenvalues of matrix P, respectively. If a set of points S is γ-shattered by F, then $|S| \leq \frac{4R^2(\alpha^2(1-\alpha^2))}{\gamma^2}$, where $\alpha = \max(\alpha_1, \alpha_2)$ with $\alpha_1 = \min(\frac{\lambda_{min}(\Sigma_1)\beta}{\|\mu_2\|}, \frac{\|t_2\|^2-(\lambda_{max}(\Omega_2)\varphi)^2}{\|t_2\|(\lambda_{max}(\Omega_2)\varphi)^2+\|t_2\|)})$ and $\alpha_2 = \min(\frac{\lambda_{min}(\Sigma_1)\beta}{\|\mu_1\|}, \frac{\|t_1\|^2-(\lambda_{max}(\Omega_1)\varphi)^2}{\|t_1\|((\lambda_{max}(\Omega_1)\varphi)^2+\|t_1\|)})$, assuming that $\beta \geq 0$, $\|t_i\| \geq \|t_i - \mu_i\|$, and $\alpha_i \geq \frac{1}{\sqrt{2}}, i = 1, 2$.*

Proof. First, we use the inequality $\lambda_{min}(P)\,\|w\| \leq \left\|P^{1/2}w\right\| \leq \lambda_{max}(P)\,\|w\|$ to relax the constraints

$$\frac{w^T\mu_2}{\left\|\Sigma_2^{1/2}w\right\|} \geq \beta \Rightarrow \frac{w^T\mu_2}{\|w\|} \geq \lambda_{min}(\Sigma_2)\beta \tag{B.1}$$

$$\left\|\Omega_2^{-1/2}(\mu_2 - t_2)\right\| \leq \varphi \Rightarrow \|\mu_2 - t_2\| \leq \frac{\varphi}{\lambda_{min}(\Omega_2^{-1})} = \varphi\lambda_{max}(\Omega_2). \tag{B.2}$$

The constraints imposed by the second prior $\frac{-w^T\mu_1}{\left\|\Sigma_2^{1/2}w\right\|} \geq \beta$, $\left\|\Omega_1^{-1/2}(\mu_1 - t_1)\right\| \leq \varphi$ are relaxed in a similar fashion to produce:

$$\frac{w^T(-\mu_1)}{\|w\|} \geq \lambda_{min}(\Sigma_1)\beta \tag{B.3}$$

$$\|\mu_1 - t_1\| \leq \varphi\lambda_{max}(\Omega_1) \tag{B.4}$$

Now, we show that if the assumptions made in the statement of the theorem hold, then every subset $S_o \subseteq S$ satisfies $\|\sum S_o - \sum(S - S_0)\| \leq \frac{4R^2(\alpha^2(1-\alpha^2)}{\gamma^2}$.

Assume that S is γ-shattered by F. The argument used by Taylor and Bartlett (1998) in Lemma 1.2 shows that, by the definition of fat-shattering, there exists a vector w_1 such that

$$w_1(\sum S_o - \sum(S - S_0)) \geq |S|\,\gamma. \tag{B.5}$$

Similarly (reversing the labeling of S_0 and $S_1 - S_0$), there exists a vector w_2 such that

$$w_2(\sum(S - S_0) - \sum S_o) \geq |S|\,\gamma. \tag{B.6}$$

Hence, $(w_1 - w_2)(\sum S_o - \sum(S - S_0)) \geq 2\,|S|\,\gamma$, which, by Cauchy-Schwartz inequality, implies that

$$\|w_1 - w_2\| \geq \frac{2\,|S|\,\gamma}{\|\sum S_o - \sum(S - S_0)\|} \tag{B.7}$$

The constraints on the classifier represented in B.1 and B.2 imply by Proposition B.3 that $\frac{w_1^T t_2}{\|w_1\|\|t_2\|} \geq (2\alpha_1^2 - 1)$ and $\frac{w_2^T t_2}{\|w_2\|\|t_2\|} \geq (2\alpha_2^2 - 1)$. Now, applying Proposition B.1 (part 2) and simplifying, we get

$$\|w_1 - w_2\| \leq 4\sqrt{\alpha_1^2(1 - \alpha_1^2)}. \tag{B.8}$$

Applying the same analysis to the constraints B.3 and B.4, we get

$$\|w_1 - w_2\| \leq 4\sqrt{\alpha_2^2(1 - \alpha_2^2)}. \tag{B.9}$$

Combining B.7, B.8, and B.9, we get

$$\left\|\sum S_o - \sum(S - S_0)\right\| \geq \frac{|S|\,\gamma}{2\sqrt{\alpha^2(1 - \alpha^2)}} \tag{B.10}$$

with α as defined in the statement of the theorem.

Taylor and Bartlett's (1998) Lemma 1.3 proves, using the probabilistic method, that some $S_o \subseteq S$ satisfies

$$\left\|\sum S_o - \sum(S - S_0)\right\| \leq \sqrt{|S|}R. \tag{B.11}$$

Combining B.10 and B.11 yields $|S| \leq \frac{4R^2(\alpha^2(1-\alpha^2))}{\gamma^2}$. □

References

Baxter, J. (2000). A model of inductive bias learning. *Journal of Artificial Intelligence Research*, *12*, 149–198.

Bhattacharyya, C., Pannagadatta, K. S., & Smola, A. (2004). A second order cone programming formulation for classifying missing data. In *NIPS*.

Blake, C., & Merz, C. (1998). 20 newsgroups database, http://people.csail.mit.edu/people/jrennie/20newsgroups/..

Campbell, C., Cristianini, N., & Smola, A. (2000). Query learning with large margin classifiers. In *Proceedings of The Seventeenth International Conference on Machine Learning*.

Carlin, B., & Louis, T. (2000). *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman and Hall.

Collins, M. (2002). Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of 2002 Conference on Empirical Methods in Natural Language Processing*.

Duda, R., Hart, P., & Stork, D. (2001). *Pattern Classification*. John Wiley. 2nd edition.

Epshteyn, A., & DeJong, G. (2005). Rotational prior knowledge for svms. In *Proceedings of the Sixteenth European Conference on Machine Learning*.

Evgeniou, T., & Pontil, M. (2004). Regularized multi-task learning. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Fink, M. (2004). Object classification from a single example utilizing class relevance metrics. In *Advances in Neural Information Processing Systems*.

Fletcher, R. (1987). *Practical Methods of Optimization*. John Wiley and Sons, West Sussex, England.

Fung, G., Mangasarian, O., & Shavlik, J. (2002). Knowledge-based support vector machine classifiers. In *Advances in Neural Information Processing Systems*.

Greiner, R., & Zhou, W. (2002). Structural extension to logistic regression: Discriminative parameter learning of belief net classifiers. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence*.

Gunawardana, A., & Byrne, W. (2005). Convergence theorems for generalized alternating minimization procedures. *Journal of Machine Learning Research*, *6*, 2049–2073.

Hansen, P., Jaumard, B., & Savard, G. (1992). New branch-and-bound rules for linear bilevel programming. *SIAM Journal on Scientific and Statistical Computing*, *13*, 1194–1217.

Huang, K., King, I., Lyu, M. R., & Chan, L. (2004). The minimum error minimax probability machine. *Journal of Machine Learning Research*, *5*, 1253–1286.

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

Jebara, T. (2004). *Machine Learning: Discriminative and Generative*. Kluwer Academic Publishers.

Joachims, T. (1998). Text categorization with support vector machines: learning with many relevant features. In *Proceedings of the Tenth European Conference on Machine Learning*.

Lanckriet, G. R. G., Ghaoui, L. E., Bhattacharyya, C., & Jordan, M. I. (2001). Minimax probability machine. In *Advances in Neural Information Processing Systems*.

Lobo, M. S., Vandenberghe, L., Boyd, S., & Lebret, H. (1998). Applications of second-order cone programming. *Linear Algebra and its Applications*, *284*(1–3), 193–228.

Mangasarian, O., Shavlik, J., & Wild, E. (2004). Knowledge-based kernel approximation. *Journal of Machine Learning Research*.

Miller, G. (1990). WordNet: an online lexical database. *International Journal of Lexicography*, *3*(4).

Ng, A. Y., & Jordan, M. I. (2001). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing Systems*.

Raina, R., Shen, Y., Ng, A. Y., & McCallum, A. (2003). Classification with hybrid generative/discriminative models. In *Advances in Neural Information Processing Systems*.

Roos, T., Wettig, H., Grunwald, P., Myllymaki, P., & Tirri, H. (2005). On discriminative bayesian network classifiers and logistic regression. *Machine Learning*, *59*, 267–296.

Scholkopf, B., Simard, P., Vapnik, V., & Smola, A. (2002). Prior knowledge in support vector kernels. *Advances in kernel methods - support vector learning*.

Sturm, J. F. (1999). Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, *11*, 625–653.

Sun, Q., & DeJong, G. (2005). Explanation-augmented svm: an approach to incorporating domain knowledge into svm learning. In *Proceedings of The Twenty Second International Conference on Machine Learning*.

Taylor, J. S., & Bartlett, P. (1998). Generalization performance of support vector machines and other pattern classifiers. In *Advances in kernel methods: support vector learning*.

Thrun, S. (1995). Is learning the n-th thing any easier than learning the first?. In *Advances in Neural Information Processing Systems*.

Tipping, M. E. (2001). Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, *1*, 211–244.

Tong, S., & Koller, D. (2000a). Restricted bayes optimal classifiers. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*.

Tong, S., & Koller, D. (2000b). Support vector machine active learning with applications to text classification. In *Proceedings of The Seventeenth International Conference on Machine Learning*.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag.

Wu, X., & Srihari, R. (2004). Incorporating prior knowledge with weighted margin support vector machines. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Zangwill, W. (1969). Convergence conditions for nonlinear programming algorithms. *Management Science*, *16*, 1–13.