The Constrained Weight Space SVM: Learning with Ranked Features

Kevin Small[†] KEVIN.SMALL@GMAIL.COM
Byron C. Wallace[†‡] BYRON.WALLACE@GMAIL.COM
Carla E. Brodley[‡] BRODLEY@CS.TUFTS.EDU
Thomas A. Trikalinos[†] TTRIKALIN@GMAIL.COM
[†]ICHRPS, Tufts Medical Center, Boston, MA
[‡]Department of Computer Science, Tufts University, Medford, MA

Abstract

Applying supervised learning methods to new classification tasks requires domain experts to label sufficient training data for the classifier to achieve acceptable performance. It is desirable to mitigate this annotation effort. To this end, a pertinent observation is that instance labels are often an indirect form of supervision; it may be more efficient to impart domain knowledge directly to the model in the form of *labeled features*. We present a novel classification model for exploiting such domain knowledge which we call the *Constrained Weight Space SVM* (CW-SVM). In addition to exploiting binary labeled features, our approach allows domain experts to provide *ranked* features, and, more generally, to express arbitrary expected relationships between sets of features. Our empirical results show that the CW-SVM outperforms both baseline supervised learning strategies and previously proposed methods for learning with labeled features.

1. Introduction

Supervised learning for classification entails inducing a classifier from labeled training data that generalizes well to unseen data with respect to a specified evaluation metric. To achieve satisfactory performance for a particular task, machine learning practitioners must typically select an appropriate learning algorithm, encode available domain knowledge (i.e., engineer features and set hyper-parameters), and procure sufficient training data. If the resulting performance is deemed unsatisfactory, additional effort must be spent improving design choices made during at least one of these three steps. In our experience, alternating between different state-of-the-art classification algorithms is the least fruitful alternative of the three.

This leaves the practitioner with feature engineering and acquiring additional training data as the primary means for improving classifier performance. Recent research has examined methods for reducing annotation costs, such as active (Settles, 2009) and semi-supervised (Chapelle et al., 2010) learning. Alternatively, while feature engineering is more task specific, it is widely known that a well designed representation can make the learning problem significantly easier (Fawcett & Utgoff, 1992). Motivated by this observation, an emerging and potentially powerful strategy is to design learning algorithms that facilitate domain expert encoding of beliefs regarding relationships among class labels and specific features, herein referred to as *labeled features* (Liu et al., 2004).

The key benefit of labeled features is that by allowing the domain expert to more directly bias the hypothesis space, the amount of labeled data required to achieve good generalization can be significantly reduced. To this end, we present a method that explicitly encodes feature label information by using *weight constraints* in the induction of linear classifiers. For example, when considering the task of classifying movie reviews as *positive* or *negative* based on the text of the review (Pang & Lee, 2004), we may wish to encode that the weight associated with *terrific* should be more positive than that associated with *terrible*.

An instructive example that motivates our work is the task of classifying biomedical texts as *relevant* or *irrel-*

Appearing in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

evant with respect to a specific clinical question (e.g., "Does β-blockers medication cause mortality in patients who have suffered a recent heart attack?"). In this case, a PubMed[1] search often returns many thousands of abstracts, of which only a few tens are actually relevant. Typical supervised learning would require a physician to label hundreds of abstracts in order to induce a model capable of accurately classifying the remainder. However, imparting domain knowledge in the form of labeled terms to the model provides a direct form of supervision stronger than instance labels alone. In the example above, *β-blocker* is very indicative of relevance, *humans* weakly so, and *rats* is indicative of irrelevance. Indeed, it would likely require a substantial number of (rare) *relevant* training instances to learn that the token *humans* is positively correlated with relevance.

The main contribution of this paper is a novel formulation for exploiting expert-provided labeled features during classifier induction. Specifically, we extend the support vector machine (SVM) model (Cortes & Vapnik, 1995) by adding additional constraints to reflect this domain knowledge. While methods for learning with labeled features have been recently proposed elsewhere (e.g., (Zaidan et al., 2007; Druck et al., 2008), which we discuss at length in Section 5), our method is unique in two ways. First, it emphasizes direct encoding of expert beliefs in the form of weight constraints. Second, we are able to exploit *ranked* labeled features (e.g., while *great* and *good* are both indicative of a *positive* movie review, the former is *more* indicative of this than the latter). Such a ranking is natural in many domains, and as we shall see, exploiting this ranking can improve classifier performance.

The remainder of the paper is organized as follows. First, we briefly review the standard SVM, which we build upon in Section 3 to illustrate two specific formulations of our proposed *constrained weight space SVM* (CW-SVM) that exclusively support pairwise weight constraints (PWCs). In Section 3.2 we give a general formulation of the CW-SVM, of which PWC-based formulations are special cases. We conclude the presentation of the CW-SVM in Section 3.3, providing a concrete instantiation of the general case using function-based constraints (FBCs). We provide experimental results over a sentiment analysis task and two biomedical citation screening tasks in Section 4 – providing an empirical comparison with existing methods that learn with labeled features. We discuss other related work and offer concluding remarks in Sections 5 and 6, respectively.

2. Preliminaries

We focus on learning binary linear classifiers of the form $f(\mathbf{x}) = sgn(\mathbf{w} \cdot \mathbf{x} + b)$ where $\mathbf{x} \in \{0,1\}^d$ is a d-dimensional feature vector, $\mathbf{w} \in \mathbb{R}^d$ is a d-dimensional weight vector, and $b \in \mathbb{R}$ is a learned threshold (i.e., bias element). Following conventional notation, let $y \in \{-1, 1\}$ denote the label associated with an item. Given a set of m training instances $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$, the goal is to inductively learn classifier parameters $\{\mathbf{w}, b\}$ that generalize well to unseen data.

We build upon the C parameterization for soft margin SVM classifiers (Cortes & Vapnik, 1995). Defining $\boldsymbol{\xi} \in [0, \infty]^m$ as a slack variable vector to minimize instance-wise hinge loss and C as a tradeoff parameter between misclassification error and regularization, recall that the C-SVM formulation is given by

$$\begin{array}{lll} \underset{\mathbf{w},b}{\operatorname{argmin}} & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^m \xi_i & \\ s.t. & y_i\left(\mathbf{w}\cdot\mathbf{x}_i + b\right) \geq 1 - \xi_i & \forall i = 1 \ldots m \\ & \xi_i \geq 0 & \forall i = 1 \ldots m. \end{array}$$

3. The Constrained Weight Space SVM

A domain expert may know that particular feature values are correlated with specific classes. In the aforementioned movie sentiment analysis task, it is sensible that the word *terrible* should have a lower weight than the word *terrific* (i.e., $w_{terrific} > w_{terrible}$). We augment C-SVMs to exploit such information by biasing the optimization procedure toward returning weight vectors in the hypothesis space that satisfy these constraints. More specifically, our method directly encodes expert feature knowledge through the definition of weight constraint sets, $p \in \mathcal{P}$, each comprising a set of binary relationships $\{\alpha, \beta\}_{\alpha,\beta\in p}$ that describe the relative weight values (e.g., $w_\alpha \geq w_\beta$).

Generally, we call this model the constrained weight space SVM (CW-SVM). In the remainder of this section, we describe a sequence of CW-SVM instantiations. We begin with the relatively straightforward but powerful approach of allowing the expert to specify a single set of independent *pairwise constraints* (PWCs), as this is the simplest case. We then proceed by generalizing the CW-SVM framework, allowing for the incorporation of *function-based constraints* (FBCs). It should be noted that in all of the proposed variants only a small number of features need to be labeled to achieve performance gains over baseline strategies, leaving the remaining weights associated with unlabeled features unconstrained but influenced by their value in relation to the explicitly constrained weights.

[1] PubMed is a repository of biomedical literature.

3.1. Pairwise Weight Constraints

The simplest instance of explicit weight constraints are *pairwise constraints* (PWCs). In this case, we assume only that the domain expert has specified pairs $\{\alpha, \beta\}$ of labeled features such that the weight associated with α should have greater value than the weight associated with β. Once specified, a scaling parameter $\rho_{\alpha,\beta}$ is associated with each PWC such that the distance between the two weights (e.g., $w_\alpha - w_\beta$) is maximized in coordination with the existing C-SVM parameterization. Considering Figure 1, an example PWC is that $w_{terrific} > w_{lively}$, where the weight *ordering* is specified and $\rho_{terrific,lively}$ is to be learned from the data. We now describe two CW-SVM formulations that exclusively utilize PWCs: *feature polarity* and *ranked features*.

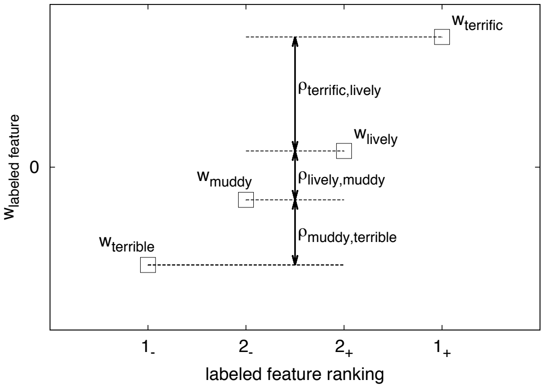

Figure 1. Weight bias induced by pairwise constraints.

3.1.1. Feature Polarity

In the *feature polarity* setting, we assume that the expert provides a set of positive labeled terms $\boldsymbol{\alpha}$ and a set of negative labeled terms $\boldsymbol{\beta}$. In this case, we generate $|\boldsymbol{\alpha}||\boldsymbol{\beta}|$ constraints and reward hypotheses where $w_\alpha > w_\beta$,[2] giving rise to the optimization:

$$\begin{aligned} \underset{\mathbf{w},b}{\text{argmin}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C_1 \sum_{i=1}^{m} \xi_i - C_2 \sum_{\alpha,\beta} \rho_{\alpha,\beta} && (1) \\ s.t. \quad & y_i \left(\mathbf{w} \cdot \mathbf{x}_i + b\right) \geq 1 - \xi_i \quad \forall i = 1 \ldots m \\ & w_\alpha - w_\beta \geq \rho_{\alpha,\beta} \qquad\quad \forall \alpha, \beta && (2) \\ & \tau_- \leq w_\alpha, w_\beta \leq \tau_+ \qquad \forall \alpha, \beta && (3) \\ & \xi_i \geq 0 \qquad\qquad\qquad \forall i = 1 \ldots m. \end{aligned}$$

In this case, we augment the C-SVM optimization problem by encoding a preference to separate the weights of features with known polarity, using the defined PWCs of Equation 2 and rewarding this separation in the objective function of Equation 1, bounded by the box constraints τ_-, τ_+ of Equation 3.

3.1.2. Ranked Features

In the preceding section we described a method for incorporating labeled features with respect only to class polarity. We now introduce machinery to exploit *ranked features*. For example, while *terrific, lively* may be associated with a positive movie review and *muddy, terrific* with a negative review, an expert may want to specify that they believe $w_{terrific} \geq w_{lively} \geq w_{muddy} \geq w_{terrible}$. It is straightforward to derive a PWC formalism to include ranked features. Specifically, if we define $\alpha \succ \beta$ to indicate that $w_\alpha \geq w_\beta$ such that the rankings for α and β are adjacent and α is "more positive" than β, the following optimization problem captures ranked features:

$$\begin{aligned} \underset{\mathbf{w},b}{\text{argmin}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C_1 \sum_{i=1}^{m} \xi_i - C_2 \sum_{\alpha,\beta:\alpha\succ\beta} \rho_{\alpha,\beta} && (4) \\ s.t. \quad & y_i \left(\mathbf{w} \cdot \mathbf{x}_i + b\right) \geq 1 - \xi_i \quad \forall i = 1 \ldots m \\ & w_\alpha - w_\beta \geq \rho_{\alpha,\beta} \qquad\quad \forall \alpha, \beta : \alpha \succ \beta && (5) \\ & \tau_- \leq w_\alpha, w_\beta \leq \tau_+ \qquad \forall \alpha, \beta \\ & \xi_i \geq 0 \qquad\qquad\qquad \forall i = 1 \ldots m \end{aligned}$$

Note that the "most weakly" positive labeled features are considered adjacent to the "most weakly" negative labeled features; in our above example w_{lively} would be considered adjacent to w_{muddy}. We augment the C-SVM optimization problem in order to encourage separation between features with adjacent rankings using the pairwise weight constraints of Equation 5 and rewarding separation in the objective function of Equation 4 subject to the same box constraints. Note also that the feature polarity formulation described in the previous section is a special case of ranked features where there are only two possible rankings.

3.2. CW-SVM: A General Formulation

As developed thus far, PWC formulations reward correct parameter "orderings" (with respect to *a priori* expert beliefs), but do not provide a means for encoding beliefs regarding the relative distances between the provided sets of ranked weights. For some tasks experts may wish to express an intuition such as "The terms *horrible* and *awful* are exponentially more indicative that a movie review is negative than are the terms *convoluted* and *long*." We now present a general formulation of the CW-SVM that allows the expert to formally express such domain knowledge.

[2] Note that that this can be equivalently accomplished with PWC which constrain positive (negative) feature weights to be greater (less) than the decision threshold.

First, we define ranked feature sets where $r_p(x)$ denotes the expert defined rank associated with each labeled feature such that $r_p(x) > 0$ indicates a ranking associated with the positive class and $r_p(x) < 0$ is associated with the negative class. We encode the rankings numerically as follows: the terms belonging to the *most* positive set map to rank 1; terms in the second most positive set to rank 2, etc. The same holds for negatively ranked terms, only the values are negated to encode polarity. In our running example from Figure 1, $r_p(\text{lively}) = 2$, $r_p(\text{terrific}) = 1$, $r_p(\text{muddy}) = -2$ and $r_p(\text{terrible}) = -1$.

Next we define a function g_p over ranks $r(\alpha)$ and $r(\beta)$ to provide a scalar expressing the expected difference in weight values of their sets' respective members. For example, consider Figure 2, where we are shaping both the positive and negative ranked features with separate exponential functions.

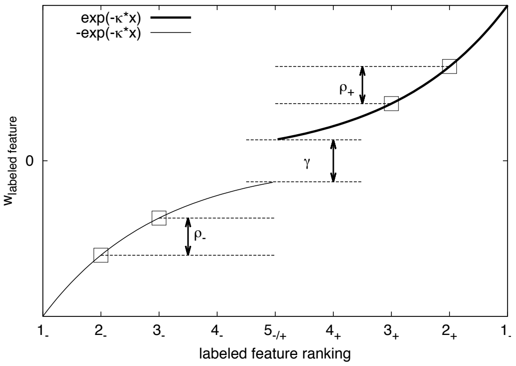

Figure 2. Weight space bias induced by function-based constraints

In this case, all of the weights associated with positively (negatively) ranked features are shaped along an exponential function where the distance between parameters is scaled by ρ_+ and ρ_- respectively. In general, there can be many such functions for different sets of features, although this will likely be a small number of functional families in practice (e.g., linear, exponential, sigmoidal, etc.). Formally, this line of reasoning results in the following optimization procedure, which is the general CW-SVM formulation:

$$\begin{aligned} \underset{\mathbf{w},b}{\text{argmin}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C_1 \sum_{i=1}^{m} \xi_i - \sum_{c \in \mathcal{C}} C_p \cdot \rho_p && (6) \\ s.t. \quad & y_i\left(\mathbf{w} \cdot \mathbf{x}_i + b\right) \geq 1 - \xi_i \quad \forall i = 1 \ldots m \\ & w_\alpha - w_\beta \geq \rho_p \cdot g_p(r(\alpha), r(\beta)) \\ & \qquad \forall \alpha, \beta, p : p \in \mathcal{P}; \alpha, \beta \in p && (7) \\ & \tau_- \leq w_\alpha, w_\beta \leq \tau_+ \qquad \forall \alpha, \beta \\ & \xi_i \geq 0 \qquad \forall i = 1 \ldots m \end{aligned}$$

Provided with the feature constraint sets $\mathcal{P}$, the optimization procedure balances the minimization of the magnitude of $\mathbf{w}$ and the minimization of training error (as in C-SVM), while attempting to maximize the relative influence the the constraint information through the scaling vector $\boldsymbol{\rho} \in \mathbb{R}^{|\mathcal{P}|}$ (the influence of these terms is influenced through their respective C parameters). Thus, while the expert-defined g_p determines the shape of the constraining function, the scale of the relative separation is still learned from data. The influence of the scaling parameters associated with each p is determined by the parameter C_p (which is set using expert knowledge or cross-validation over the training data). Once the quadratic program (QP) is specified, existing QP packages can be used to solve the optimization problem.[3] Using this formulation, an expert can define several sets of parameter constraints and functions that define beliefs about their relationships. In the next section, we describe a particular instantiation of the CW-SVM that includes *function-based constraints* (FBC).

3.3. Function-based Constraints

The PWC formulations of Section 3.1 are specific instantiations of the general CW-SVM where there exists an independent function $g_p(r(\alpha), r(\beta)) = 1$ for each pairwise constraint (i.e., there is one parameter constraint in each parameter constraint set). However, there are situations where the expert may wish to provide the classifier information such as "$w_{terrific}$ is much more positive than w_{good} while w_{good} is slightly more positive than w_{lively}." This is shown in Figure 2, where the aforementioned weights are biased to fit along the function $f(w) = e^{-\kappa \cdot r(w)}$ (where κ is a constant). In this case, we would define $g(r(\alpha), r(\beta)) = e^{-\kappa \cdot r(\alpha)} - e^{-\kappa \cdot r(\beta)}$ and constrain all of the positive ranked parameters to the shape of this function (therefore learning the scaling parameter ρ associated with a specified g). Here the expert would group these parameter constraints into a parameter constraint set p and specify a function to express relationships between labeled features in this set. By allowing the expert to specify this additional information, and thus inducing a stronger bias on the parameter space than PWC, we can further reduce the labeled data requirements, as demonstrated by our empirical results.

We now introduce a particular instantiation of CW-SVM where all of the positively labeled features are used to generate one parameter constraint set (which are related to each other by a single shaping function) and all of the negatively labeled features are used to

[3] We use CVXOPT (Dahl & Vandenberghe, 2004).

generate a second parameter constraint set (which are related to each other by a second single shaping function). Finally, we define PWCs along the *polarity border* to enforce a notion of margin among the labeled features. This results in the following optimization problem:

$$
\begin{aligned}
\underset{\mathbf{w},b}{\operatorname{argmin}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C_1 \sum_{i=1}^{m} \xi_i - C_2 \sum_{\substack{\max(r(\alpha)) \\ \min(r(\beta))}} \rho_{\alpha,\beta} \\
& \quad -C_3 \cdot \rho_1 - C_4 \cdot \rho_2 && (8) \\
s.t. \quad & y_i\left(\mathbf{w} \cdot \mathbf{x}_i + b\right) \geq 1 - \xi_i \qquad \forall i = 1 \ldots m \\
& w_\alpha - w_\beta \geq \rho_{\alpha,\beta} \\
& \qquad \forall \alpha, \beta : \max(r(\alpha)), \min(r(\beta)) && (9) \\
& w_\alpha - w_\beta \geq \rho_1 \cdot g_1(r(\alpha), r(\beta)) \\
& \qquad \forall \alpha, \beta : \alpha \succ \beta, r(\alpha) > 0, r(\beta) > 0 && (10) \\
& w_\beta - w_\alpha \geq \rho_2 \cdot g_2(r(\beta), r(\alpha)) \\
& \qquad \forall \alpha, \beta : \alpha \succ \beta, r(\alpha) < 0, r(\beta) < 0 && (11) \\
& \tau_- \leq w_\alpha, w_\beta \leq \tau_+ \qquad \forall \alpha, \beta \\
& \xi_i \geq 0 \qquad \forall i = 1, \ldots, m
\end{aligned}
$$

This form is very general because there are infinitely many possible shaping functions which can be used to define FBCs. However, as previously stated, there only a small number of functional families are useful in practice – making FBC formulations feasible for expert specification in the common cases.

4. Experiments

For our experimental evaluation, we consider two text classification tasks: biomedical citation screening (Wallace et al., 2010) and classifying the sentiment of movie reviews (Pang & Lee, 2004). In each case, we compare against appropriate baselines (i.e., without labeled features) and existing strategies that exploit labeled features. We note that, to our knowledge, this is also the first empirical comparison of these particular methods for learning with labeled features.

4.1. Methods

Annotator rationales. Zaidan et al. (2007) proposed the *annotator rationales* framework as a means of incorporating 'explanations' into the training algorithm. This is done by having the expert mark the text (features) that most influenced their labeling decision. To then exploit provided rationales, several *contrast examples* are generated for each instance, which intuitively are examples assumed to be negative due to the forced absence of a particular rationale. The SVM algorithm is correspondingly modified with *contrast constraints* to encourage the model to find weights that are consistent with the expert-provided rationales. While our approach requires a small set of labeled terms as opposed to rationales for each instance (which doctors are not anxious to supply when conducting reviews), we compared our CW-SVM to the rationales approach over the sentiment analysis task of Section 4.3 using the methodology described in (Zaidan et al., 2007).

Pooling multinomials. Melville et al. (2009) proposed the *pooling multinomials* model, which extends the standard Naive Bayes model for text classification. In particular, they compute posterior estimates of a document belonging to a given class using both the standard Naive Bayes model and a generative *background* model that incorporates labeled features (terms), which they refer to as the *lexical* model. The basic strategy in deriving their lexical model is to assign probabilities to the labeled terms reflecting their polarity, or class association. For technical details of their lexical model, see (Melville et al., 2009).

The outputs of the two models are then combined via linear pooling (i.e., the estimated probabilities of class membership are linearly combined with weights reflecting the accuracy of the respective models as estimated via cross-validation). In particular, each model m (Naive Bayes, lexical) has an associated weight α_m computed as follows: $\alpha_m = log\frac{1-err_m}{err_m}$ where err_m is the error rate of model m. Because of our emphasis on recall in the citation screening scenario (reflected by evaluation via F_2), we modify their approach slightly for these datasets such that the two models are combined according a weighted error; in particular, we use $err_m = \frac{fpr_m + \beta fnr_m}{1+\beta}$, where fpr_m and fnr_m are the false positive and false negative rates, respectively.[4] This modification improves performance on the screening task datasets when compared to the method published in their paper which optimizes for accuracy (which we utilize for the movies dataset).

GEC. The *generalized expectation criteria* (GEC) framework prescribes a general (mathematically) semi-supervised way of exploiting labeled features during learning (Druck et al., 2008). Specifically, we use the feature labeling variant (GE-FL), a method of optimizing discriminative probabilistic models subject to (soft) constraints over predictions on unlabeled instances, which in this case reflect *a priori* assumptions about feature-label distributions. It should be noted that GE-FL is intended for different scenarios than CW-SVM, namely when there exists an abundance of labeled features, but only a few (if any) instance labels. Nonetheless, even though GE-FL uses exclusively la-

[4] We set β to 10, reflecting intuition.

beled features (i.e., the instance labels are ignored), we include an empirical comparison for completeness. We used the Mallet (McCallum, 2002) implementation of the GE-FL framework (Druck et al., 2008).

CW-SVM. For the CW-SVM, we compared results using the following variants:

- **Polarity** - The PWC formalism wherein weights associated with positively labeled terms are constrained to be greater than all weights associated with negatively labeled terms (Section 3.1.1).
- **Ranked** - The PWC formalism in which adjacently ranked features correspond to associated constraints in weight space (Section 3.1.2).
- **FBCs** - The formulation of Section 3.3 where parameters are constrained to fit along a specified function. We consider **Linear** $\{g_1(r(\alpha), r(\beta)) = r(\alpha) - r(\beta), g_2(r(\beta), r(\alpha)) = r(\beta) - r(\alpha)\}$ and **Exp**onential $\{g_1(r(\alpha), r(\beta)) = e^{-\kappa \cdot r(\alpha)} - e^{-\kappa \cdot r(\beta)}, g_2(r(\beta), r(\alpha)) = e^{-\kappa \cdot r(\beta)} - e^{-\kappa \cdot r(\alpha)}\}$ cases.

4.2. Biomedical Citation Screening

Systematic reviews have become an increasingly important aspect of evidence-based healthcare practice. An important step in conducting a systematic review is *citation screening*, in which reviewers (usually physicians) search for literature relevant to their clinical question. This is done via a search designed to achieve high recall, because missing relevant literature may compromise the scientific validity of the review. Reviewers typically screen between 2,000 and 5,000 citations for a given review, of which approximately 200 to 1,000 are deemed potentially relevant (Wallace et al., 2010). In this task there is therefore both class imbalance ('relevant' articles comprise only ~10% of the corpus on average) and asymmetric misclassification costs (false negatives are costlier than false positives). Due to these observations, we use a recall-centric metric for our evaluation for these datasets. In particular, we use a weighted harmonic mean which values recall twice as much as precision (i.e., $F_2 = \frac{5 \cdot precision \cdot recall}{4 \cdot precision + recall}$).

For the citation screening datasets, we use a bag-of-words (BOW) representation, ignoring word capitalization and removing words found in the PubMed stoplist. During each experiment, we perform five-fold cross-validation, setting C_1 for each fold via two-fold cross-validation on the available training data for that fold (covering the search space $C_1 = 2^{\{-10,\ldots,3\}}$). Once C_1 is determined for the baseline SVM, we use the resulting $\mathbf{w}$ to inform $\tau_{\{+,-\}}$ such that $\tau_+ = 2 \cdot \max_{w' \in \mathbf{w}_{SVM}} w', \tau_- = 2 \cdot \min_{w' \in \mathbf{w}_{SVM}} w'$ and perform the same search over C_2, C_3, C_4 (as appropriate). In each case, we undersample the negative data such that we are learning from a balanced dataset.[5]

The first citation screening dataset is *Proton Beam*, comprising 4748 documents – 243 of which are labeled as *relevant*. A clinician involved in the review provided 70 *positive* terms divided into 6 ranked term classes and 11 *negative* terms divided into 3 ranked term classes (independent of any interaction with this work). For *Proton Beam*, we conducted five experiments in which we induce a classifier over $\{50, 100, 150, 200, 243\}$ relevant and 4505 irrelevant documents. The results for this experiment are shown in Figure 3.

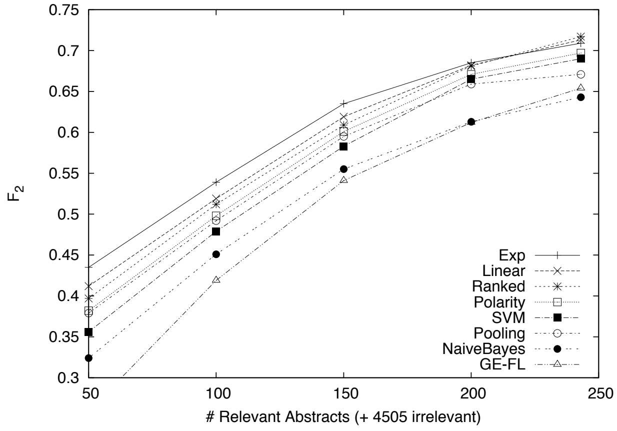

Figure 3. Empirical Results on Proton Beam Review

Unsurprisingly, Naive Bayes fares poorly compared to the other supervised models. However, the pooling multinomials model (Melville et al., 2009) does relatively well, outperforming the standard SVM model, at least at the first three evaluation points, demonstrating the utility of labeled features. All four of the CW-SVM models outperform the other strategies, particularly at the start of the learning curve (i.e., when fewer labeled instances are available). This makes sense, as biasing the learner with (prior) domain knowledge in the absence of sufficient training data seems likely to improve performance. GE-FL is generally outperformed by the directly supervised methods, but does beat Naive Bayes when provided with sufficient unlabeled data.

The second citation screening dataset, *COPD*, comprises 1606 documents, 196 of which were found to be *relevant*. In this case, we have 15 positive terms di-

[5] Random undersampling of the majority class has been shown to mitigate class imbalance (Van Hulse et al., 2007).

vided into 3 ranked term classes and 7 negative terms divided into 2 ranked classes (again derived independently). For *COPD*, we conducted five experiments where we learn a classifier from $\{40, 80, 120, 160, 196\}$ relevant examples and 1410 irrelevant documents. The results for this experiment are shown in Figure 4.

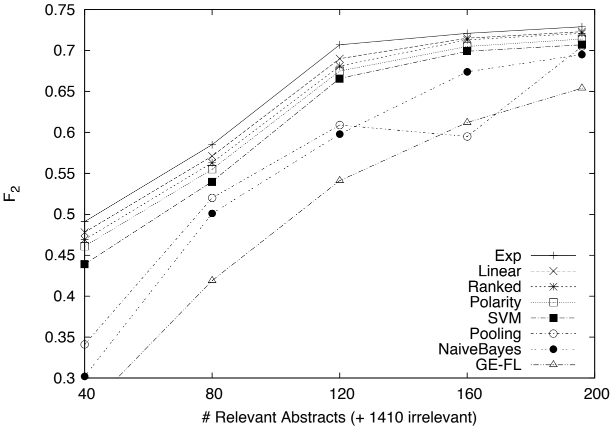

Figure 4. Empirical Results on COPD Review

Naive Bayes again performs poorly on the *COPD* dataset. Interestingly, the pooling multinomials does not perform as well here as in the *Proton Beam* data. Although not entirely clear, it may be attributable to the small number of labeled features for this dataset – which is supported by GE-FL being significantly outperformed by directly supervised methods. We again observe that the CW-SVM outperforms all other methods, particularly when provided with less data.

4.3. Sentiment Analysis

We now present results over the *movies* dataset (Pang & Lee, 2004), in which the task is to classify movie reviews as *positive* or *negative*. There are 2000 movie reviews in this corpus, 1000 of which are *positive* and 1000 of which are *negative*. For this dataset, we have rationales provided by Zaidan et al. (2007) and follow the data encoding, training and testing procedures described therein. To derive labeled features, we used an information-gain metric to rank terms with respect to their discriminative power (using the instance labels to effectively simulate an oracle, as has been done elsewhere (Druck et al., 2008)). We created three classes of each polarity: 30 positive terms total (10 per positive class) and 45 negative terms (15 per negative class), using the same strategy as previously to set $C_1, \ldots, C_4$.

Both standard Naive Bayes and linear pooling perform poorly in this case.[6] All of the other strategies that exploit labeled features (our CW-SVM and the rationales approach) outperform the baseline SVM induced over instance labels alone, again highlighting the utility of labeled features. Our CW-SVM, however, outperforms the already strong rationales approach. GE-FL is outperformed by directly supervised methods, although performing quite well (particularly for smaller datasets) considering it doesn't use the instance labels.

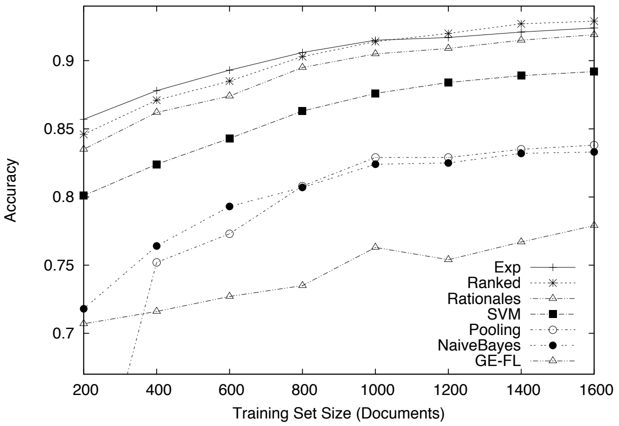

Figure 5. Empirical Results on Movies Dataset

5. Related Work

There have been several recent investigations into learning with alternative forms of supervision (i.e., labeled features). Most similar to our work are the three algorithms used for comparison in Section 4.1: Zaidan et al.'s *annotator rationales* approach (2007), Druck et al.'s GE-FL (2008), and the *pooling multinomials* model developed by Melville et al. (2009). (The former is similar to the Explanation-Augmented SVM previously proposed by Sun et al. (2005)). We also note that in more recent work, Zaidan et al. (2008) formalized annotator rationales in a generative probabilistic framework, though the underlying intuition remains fundamentally the same. As previously stated, GE-FL is more suitable for scenarios where labeled features are the primary form of supervision, whereas CW-SVM is more applicable in cases where supervised learning is *augmented* with additional feature information. Another vein of work is Knowledge-Based SVMs (Fung et al., 2002) where advice is specified in the form [*if* ANTECEDENT *then* CONSEQUENT]. While ostensibly applicable, this framework is really intended for more complex logical ANTECEDENT statements as opposed to simple labeled features.

[6] Our linear pooling results agree with those of (Melville et al., 2009), though our implementation of standard Naive Bayes outperforms theirs, for reasons unclear to us.

Our method of exploiting labeled features differs in a few key ways from the aforementioned approaches. First, we integrate the parameter constraints directly into the optimization procedure, as opposed to doing this implicitly via *contrastive* instances as in (Zaidan & Eisner, 2008). Moreover, our approach is an augmentation of the SVM algorithm, generally held to be the state-of-the-art in text classification (Joachims, 1998) in contrast to (Melville et al., 2009).[7] Furthermore, unlike the Generalized Expectation Feature Learning (GE-FL) (Druck et al., 2008), we do not assume that the expert is capable of providing a large set of labeled features (or feature-class distributions), which we believe is too restrictive for many applications.

Perhaps the most distinguishing feature of the CW-SVM is that, unlike previous methods, which can exploit only feature-class associations, CW-SVM allows for the direct incorporation of *ranked* features, allowing domain experts to impart knowledge regarding groupings of terms with varying degrees of polarity. As we saw in the experimental results, such rankings can boost classifier performance.

6. Conclusions

We have presented the CW-SVM, a novel, flexible method for directly incorporating labeled features in classifier induction. Our method needs only a small number of labeled features to outperform the baseline SVM. We presented strong empirical results, demonstrating that the CW-SVM outperforms existing methods that learn with labeled feature information over two biomedical abstract screening datasets and a sentiment analysis task.

Acknowledgments

We thank the reviewers for their invaluable comments. This work was supported by grant R01HS018494.

References

Chapelle, O., Schölkopf, B., and Zien, A. (eds.). *Semi-Supervised Learning.* MIT Press, 2010.

Cortes, C. and Vapnik, V. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

Dahl, J. and Vandenberghe, L. CXVOPT - python software for convex optimization, 2004. URL `http://abel.ee.ucla.edu/cvxopt`.

Druck, G., Mann, G., and McCallum, A. Learning from labeled features using generalization expectation criteria. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 595–602, 2008.

Fawcett, T. E. and Utgoff, P. E. Automatic feature generation for problem solving systems. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 144–153, 1992.

Fung, G., Mangasarian, O. L., and Shavlik, J. W. Knowledge-based support vector machine classifiers. In *The Conference on Advances in Neural Information Processing Systems (NIPS)*, pp. 521–528, 2002.

Joachims, T. Text categorization with support vector machines: Learning with many relevant features. *Proceedings of the European Conference on Machine Learning (ECML)*, pp. 137–142, 1998.

Liu, B., Li, X., Lee, W. S., and Yu, P. S. Text classification by labeling words. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 425–430, 2004.

McCallum, A. MALLET: A machine learning for language toolkit, 2002. URL `http://mallet.cs.umass.edu`.

Melville, P., Gryc, W., and Lawrence, R. D. Sentiment analysis of blogs by combining lexical knowledge with text classification. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 1275–1284, 2009.

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

Settles, B. Active learning literature survey. Technical Report 1648, University of Wisconsin, 2009.

Sun, Q. and DeJong, G. Explanation-augmented SVM: an approach to incorporating domain knowledge into SVM learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 864–871, 2005.

Van Hulse, J., Khoshgoftaar, T.M., and Napolitano, A. Experimental perspectives on learning from imbalanced data. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 935–942, 2007.

Wallace, B. C., Small, K., Brodley, C. E., and Trikalinos, T. A. Active learning for biomedical citation screening. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 173–182. ACM, 2010.

Zaidan, O., Eisner, J., and Piatko, C. Using "annotator rationales" to improve machine learning for text categorization. In *Proceedings of the Annual Meeting of the North American Association of Computational Linguistics (NAACL)*, pp. 260–267, 2007.

Zaidan, O. F. and Eisner, J. Modeling annotators: A generative approach to learning from annotator rationales. In *Proceedings of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*, pp. 31–40, 2008.

[7] Indeed, as observed on the movies task, standard SVM is capable of beating the pooling multinomials model.