Technical Report
TTI-TR-2009-1

Zero-Shot Domain Adaptation: A Multi-View Approach

John Blitzer
University of California, Berkeley
Dean P. Foster
University of Pennsylvania
Sham M. Kakade
Toyota Technological Institute at Chicago

ABSTRACT

Domain adaptation algorithms attempt to address situations where our training (source) data distribution and test (target) data distribution differ, potentially by a substantial amount. For example, in a natural language processing task there may be many important phrases in our target genre which are required for low target error but do not occur in our source training set or even have support under the source domain's distribution.

This work provides a domain adaptation algorithm, which (provably) permits *zero-shot* learning — by this, we mean learning an accurate classifier on our target domain with only labeled data from our source domains (and *no* labeled data on the target domain). Furthermore, we give finite sample error bounds, showing how this zero-shot learning is possible even in the aforementioned NLP example. The key intuition we formalize is how to use these novel target-specific features via their correlation with those features that are present in both the source and target domains (this learning may be done with unlabeled data). Our experiments demonstrate the robust success of our algorithm for a variety of domain adaptation tasks on product review rating prediction across multiple product types.

1. Introduction

The supervised learning paradigm of training and testing on identical distributions has provided a powerful abstraction for developing and analyzing learning algorithms. In many natural applications, though, we train our algorithm on a source distribution, but we desire high performance on target distributions which differ from that source [Ben-David et al., 2007, Huang et al., 2007, Bickel et al., 2007, Dai et al., 2007, Blitzer et al., 2008, Mansour et al., 2009].

For example: in speech recognition, we seek to adapt models trained for one (or a few) speakers to speakers with a variety of different accents [Legetter and Woodland, 1995]; computational biologists desire to apply statistical gene annotation tools to newly-sequenced genomes that differ significantly from the training data for these tools [Liu et al., 2008]; natural language processing models are typically transferred across genres [McClosky et al., 2006, Daume, 2007]; and in web search ranking, we seek to adapt English ranking models for use in other languages [Chen et al., 2008].

This is essentially the problem of domain adaptation — we seek an algorithm which is trained on one (or more) source domains but yet gracefully adapts when it is deployed on new target domains. One of the challenges here is that there may be predictive features in these target domains which do not even have support under the source domain's distribution, and it may not be possible to have low error in the target domain without using such features.

In this work, we design and analyze algorithms for domain adaptation, which (provably) permit *zero-shot* learning — by zero-shot learning, we mean transferring an accurate classifier from (one or more) source domains to a high accuracy classifier on the target domain (without *any* labeled data on the target domain, though we are permitted to use unlabeled data from the target domain). In particular, we show how this is possible even in the setting where the source and target distributions differ substantially, such as in the aforementioned case where crucial predictive features have no support under the source distribution.

Let us provide an intuitive example as to why such learning may be possible. Suppose that we desire to build a predictor of sentiment for consumer product reviews (as in our Experiments section) — say our two domains are reviews of consumer electronics and DVDs. There are many predictive words/bi-grams (features) like *excellent* and *horrible* that are shared between the two domains, but there are also many predictive phrases like *broke_quickly* or *plot_twist* that are specific to a particular domain. By using unlabeled data, we can discover how these domain specific features correlate with a set of common features that are shared between the domains. Based on this correlational structure, we may hope that our adaptation algorithm can exploit target specific features (e.g. use the phrase *plot_twist* for target prediction on the DVD domain when trained with only labeled data on the consumer electronics domain).

This intuition forms the basis for the structural correspondence learning (SCL) algorithm of Blitzer et al. [2007], which we view as an special case of this work. SCL uses unlabeled data from both the source and target domains to construct new "cross-domain" features that have functional dependence on domain specific features from *both* domains. By putting weight on such cross-domain features using only source domain data, we are effectively building predictors which use target specific features.

While this intuition is appealing, there is no apriori reason why exploiting such correlations should be possible in general. In this work, we formalize an assumption under which this intuition is applicable. The assumption is that we have we have a dimensionality reduction method for each domain such that after we have projected the input X, we have not lost predictive power of the output variable Y. Intuitively, it is this projection which ends up intertwining features on the target domain, so that novel target features are potentially coupled with those target features which are shared with the source domain. Learning such a dimensionality reduction scheme is plausible based on recent work on multi-view learning [Ando and Zhang, 2007, Kakade and Foster, 2007, Foster et al., 2008].

Under this assumption, we provide a simple domain adaptation algorithm, which is able to exploit novel target specific features due to how these features relate to certain shared features (those present in both domains). We provide finite sample error bounds (on the target domain) when trained only with source domain data. These bounds depend on a certain effective dimension characterizing how the source and target domains are related. Here, fast rates of convergence are possible, even if novel target features are required for low target error, so long as such features are related to the shared features in a manner we make precise.

Finally, we provide experimental results on the publicly available product review rating data set described in Blitzer et al. [2007]. We show that across the board our method performs significantly better than baselines which do not exploit correlations derived from unlabeled data.

1.1. Related Work

While there is a body of work that has theoretically considered this problem where training and test distributions differ [Huang et al., 2007, Ben-David et al., 2007, Cortes et al., 2008], this work says little when the target and source distributions may significantly differ in the manner we described. Our point is not to minimize this body of work but to point out that our stronger results for zero-shot learning stem from our stronger structural assumptions (based on domain based dimensionality reduction), which we believe to be applicable in many settings. It should be clear that the transfer we describe is not possible without certain structural assumptions.

We briefly mention that our algorithms and generalization results extend naturally to the setting where we have multiple source domains. They thus apply to another setting from the domain adaptation literature where we have small amounts of labeled target domain data [Daume, 2007, Blitzer et al., 2008]. Finally, we note the connection to the closely-related setting of multi-task learning [Baxter, 2000, Crammer et al., 2007, Arygriou et al., 2007]. This setting differs from ours in that it typically assumes the same underlying domain together with many different but related prediction tasks. Their goal is to do well on all tasks simultaneously.

2. The Setting

We assume our input $X \in \mathcal{X}$, where $\mathcal{X}$ is vector space, and our output $Y \in \mathbb{R}$. We have a set of domains and for each domain $D = d$, we have a joint distribution $\Pr[X, Y | D = d]$.

Assumption 1. *(Identical Tasks) Assume there exists a linear map on $\mathcal{X}$, denoted by the vector $\beta^\top$, such that for all domains d:*

$$\mathbb{E}[Y|X, D = d] = \beta^\top X$$

We do not view this assumption as especially restrictive, since we are envisioning a setting where $\mathcal{X}$ is a rich feature space.

Using samples from domain d, we can estimate β in those directions in which X varies (on domain d). To make this precise, define the *principal subspace* for a domain d as follows:

Definition We say $\mathcal{X}_d$ is the *principal subspace*, if it is lowest dimensional subspace of $\mathcal{X}$ such that $X \in \mathcal{X}_d$ with probability 1.

Equivalently, this subspace is the subspace spanned by the principal components of the covariance matrix of X on domain d, $\mathbb{E}[XX^\top | D = d]$ (where the principal components are those eigenvectors with non-zero eigenvalues).

Recall, that M is a projection operator if M is a linear and if M is idempotent, i.e. $M^2 x = Mx$ (for all x in the vector space). If P_d is a projection operator onto $\mathcal{X}_d$, then it follows that:

$$\mathbb{E}[Y|X, D = d] = \beta^\top (P_d X)$$

To see this, note that $P_d X = X$ with probability one on domain d. Hence, trivially, the projection $P_d X$ looses no predictive power of Y.

With the previous assumption alone, using samples from domain d, we can only estimate β on this subspace (i.e. we can only estimate $\beta^\top P_d$). The following assumption is that for each domain d we have knowledge of a projection operator Π_d such that $\Pi_d X$ does *not* lose any predictive power of Y.

Assumption 2. *(Dimensionality Reduction) For each domain d, there exists both a projection operator Π_d which maps $\mathcal{X}$ onto a subspace $\mathcal{D}_d \subset \mathcal{X}_d$ and a linear map, $\beta_d^\top$, on $\mathcal{D}_d$, such that*

$$\mathbb{E}[Y|X, D = d] = \beta_d^\top (\Pi_d X) \, .$$

Note that both $\Pi_d X$ and β_d can be specified by $\dim_d$ *numbers, where* $\dim_d$ *is the dimension of $\mathcal{D}_d$.*

Implicitly, we are assuming these projections can be learned with unlabeled data on domain d, which we discuss in the next section. It is these projections that allow us to relate novel features on a new target domain to those features which may be present in both the source and target domains.

Although $\beta_d^\top \Pi_d$ is optimal for domain d, it will not in general equal $\beta^\top$. However, since both $\beta_d^\top \Pi_d$ and $\beta^\top$ are optimal predictors on $D = d$, they must agree on $\mathcal{X}_d$. Specifically, for all $X \in \mathcal{X}_d$, we must have $\beta_d^\top \Pi_d X = \beta^\top X$, which implies:

$$\beta_d^\top \Pi_d P_d = \beta^\top P_d \, . \tag{1}$$

(since $P_d X \in \mathcal{X}_d$ for all $X \in \mathcal{X}$, these two mappings must be equal). This relationship is useful later.

We should also note that the while our analysis assumes our assumptions hold exactly, under a perturbation analysis one can show that our algorithms are robust and that our results decay gracefully.

2.1. Multi-View Dimensionality Reduction

Our setting is agnostic as to how to obtain the projections Π_d which satisfy Assumption 2, but we take the time here to briefly address the multi-view dimensionality reduction framework we use in practice to find Π_d. Here (with two views), we write $X = (X^{(1)}, X^{(2)})$, where $X^{(1)}$ and $X^{(2)}$ are two "views" of the data (sometimes in a rather abstract sense). The goal is to implicitly learn about the output Y via the relationship between $X^{(1)}$ and $X^{(2)}$.

The work in Foster et al. [2008] (see also Ando and Zhang [2007], Kakade and Foster [2007]) provides conditions under which we can reduce the dimensionality of X (via a learned projection) without losing predictive power of Y — this is possible under either a *conditional independence* assumption or a *redundancy* assumption between the two views. Here, the projections are learned via Canonical Correlation Analysis (CCA) between $X^{(1)}$ and $X^{(2)}$ using only unlabeled data. In particular, this work shows that (under either assumption) the best linear predictor of Y using only ΠX is equivalent to the best linear predictor with X (which implies our Assumption 2 since the best linear predictor is the conditional mean).

3. Transfer Learning

We now present algorithms (and analysis) for learning predictors on one or more target domains using training data from one or more source domains (these source domains may actually include a subset of the target domains themselves).

We begin in the simplest case where we have full knowledge of $\beta_s^\top$ for source domain s (so $\beta_s^\top \Pi_s X$ is the conditional mean of Y), and we specify how to transfer this exact $\beta_s^\top$ to an estimate $\hat{\beta}_t^\top$ for target domain t. We characterize under what conditions this estimator transfers exactly to $\beta_t^\top$ — these conditions are rather mild in many natural settings. In the next subsection, we provide an algorithm (and generalization bound) for learning an estimate $\hat{\beta}_t^\top$ of $\beta_t^\top$ using training data on a single source domain s. In the final subsection, we present a more general algorithm (and analysis) for estimating $\beta_t^\top$ (on one or more targets t) using training data on multiple source domains.

3.1. Perfect Transfer

Suppose we have (perfect) knowledge of $\beta_s^\top$. What knowledge does this impart on $\beta_t^\top$? Intuitively, the relation between $\beta_s^\top$ and $\beta_t^\top$ must be linked through the subspace of $\mathcal{X}$ which is "shared" between the domains. Let us define this notion precisely.

Definition For two domains s and t, define $\mathcal{X}_{s,t} = \mathcal{X}_s \cap \mathcal{X}_t$ (the intersection of the principal subspaces) which is itself a subspace. We say that a subspace $\mathcal{X}_{\text{shared}}$ is a *shared subspace* between domains s and t if $\mathcal{X}_{\text{shared}} \subset \mathcal{X}_{s,t}$. Clearly, $\mathcal{X}_{s,t}$ is the largest shared subspace.

Example 1. *(Shared Words) Let coordinate $X_i \in \{0,1\}$ indicate the presence or absence of word i and let e_i be the unit vector in the i-th direction. Let $\mathcal{I}_s$ be a subset of coordinates (words) in domain s such that for each $i \in \mathcal{I}_s$, X_i is not perfectly correlated with any other direction in X on domain s (i.e. X_i is not not perfectly correlated with any linear transformation of the other coordinates, X_{-i}). Let $\mathcal{I}_t$ be such a set for domain t. Then the index set $\mathcal{I}_{s,t} := \mathcal{I}_s \cap \mathcal{I}_t$ corresponds to a set of "shared words". Since the span of $\{e_i : i \in \mathcal{I}_s\}$ is a subspace of $\mathcal{X}_s$ (similarly for $\mathcal{I}_t$), we have that the span of $\{e_i : i \in \mathcal{I}_{s,t}\}$ is a shared subspace between domains s and t.*

With this notion in hand, we are able to precisely characterize how $\beta_s^\top$ and $\beta_t^\top$ are related.

Lemma 3. *(Agreement on Shared Subspace) If $P_{s,t}$ is a projection onto any shared subspace $\mathcal{X}_{shared}$, then:*

$$\beta_s^\top \Pi_s P_{s,t} = \beta_t^\top \Pi_t P_{s,t}$$

Proof. By properties of projections, for all $X \in \mathcal{X}_s$, $P_s X = X$ and for all $X \in \mathcal{X}_t$, $P_t X = X$. Since $P_{s,t}X$ is in both $\mathcal{X}_s$ and $\mathcal{X}_t$, we have that $P_s P_{s,t} = P_{s,t}$ and $P_t P_{s,t} = P_{s,t}$. By Equation 1, we have $\beta_s^\top \Pi_s P_s = \beta^\top P_s$, and, by left multiplication by $P_{s,t}$, we have $\beta_s^\top \Pi_s P_{s,t} = \beta^\top P_{s,t}$. Also, by Equation 1, $\beta_t^\top \Pi_t P_t = \beta^\top P_t$, and, analogously, $\beta_t^\top \Pi_t P_{s,t} = \beta^\top P_{s,t}$, which completes the proof. □

This implies that an estimator $\hat{\beta}_t^\top$ of $\beta_t^\top$ (using $\beta_s^\top$) should satisfy the following *Transfer Constraint*:

$$\hat{\beta}_t^\top \Pi_t P_{s,t} = \beta_s^\top \Pi_s P_{s,t} \qquad (2)$$

As we point out later, in Remark 1, if $P_{s,t}$ is a projection onto the largest shared subspace, $\mathcal{X}_{s,t}$ (rather than any shared subspace), then the Transfer Constraint fully characterizes our knowledge of $\beta_t^\top$.

Let us now characterize how accurate a solution, $\hat{\beta}_t^\top$, to the Transfer Constraint is. In particular, we specify when perfect transfer occurs. Note that $\hat{\beta}_t^\top \Pi_t$ is what is relevant for prediction on domain t and requires only $\dim_t = \text{rank}(\Pi_t)$ numbers to specify (see Assumption 2).

Theorem 4. *Let $P_{s,t}$ be a projection onto any shared subspace. We have:*

- *(Transferred Knowledge) If $\hat{\beta}_t^\top$ is a solution to the Transfer Constraint (Equation 2), then residual* *error $\hat{\beta}_t^\top - \beta_t^\top$ satisfies the following constraint:*

$$(\hat{\beta}_t^\top - \beta_t^\top)\Pi_t P_{s,t} = 0 \tag{3}$$

Hence, $\hat{\beta}_t^\top \Pi_t$ is correctly specified on a subspace of dimensionality $\mathrm{rank}(\Pi_t P_{s,t})$.

- *(Perfect Transfer) If $\mathrm{rank}(\Pi_t P_{s,t}) = \dim_t$, then any solution $\hat{\beta}_t^\top$ satisfies $\hat{\beta}_t^\top \Pi_t = \beta_t^\top \Pi_t$, so we have that $\hat{\beta}_t^\top$ is optimal.*

We view this perfect transfer condition as being a rather mild non-degeneracy condition in many settings. The requirement is that every direction (row) in Π_t must be related (i.e. have non-zero inner product) with *some* shared direction (in $\mathcal{X}_{\text{shared}}$). Clearly, this condition depends on our (learned) projection operators Π_t, but in many natural settings, learning such Π_t, which couple new target specific features with shared features in this manner, may be relatively easy to do with unlabeled data (such as in our experiments).

Proof. Equation 3 follows since both $\hat{\beta}_t^\top$ and $\beta_t^\top$ satisfy the Transfer Constraint (by Lemma 3). The remaining claims follow by noting that, by properties of projection operators, we have $(\hat{\beta}_t^\top \Pi_t - \beta_t^\top \Pi_t)\Pi_t P_{s,t} = 0$ so that $\hat{\beta}_t^\top \Pi_t$ (which has $\dim_t$ free parameters) is constrained in $\mathrm{rank}(\Pi_t P_{s,t})$ of them. □

Our final remark (without proof) points out that the Transfer Constraint captures all knowledge of $\beta_t^\top$.

Remark 1. *(Completeness) Let $P_{s,t}$ be the projection onto the largest shared subspace, $\mathcal{X}_{s,t}$. Let $\hat{\beta}_t$ be any solution to the Transfer Constraint. It is possible to construct a $\tilde{\beta}$ and a joint distribution $\widetilde{\Pr}[X,Y|D=t]$ (with the same marginal distribution on X as $\Pr[X|D=t]$) such that for $d \in \{s,t\}$:*

$$\begin{aligned} \mathbb{E}[Y|X, D=d] &= \tilde{\beta}^\top X \\ \mathbb{E}[Y|X, D=s] &= \beta_s^\top(\Pi_s X) \\ \mathbb{E}[Y|X, D=t] &= \hat{\beta}_t^\top(\Pi_t X) \end{aligned}$$

where the expectations are with respect to $\Pr[X,Y|D=s]$ or $\widetilde{\Pr}[X,Y|D=t]$. Now note that Assumptions 1 and 2 are satisfied with respect to the above parameters and distributions, and the Transfer Constraint (under these modifications) is unaltered (as $\mathcal{X}_{s,t}$ is identical). Hence, without additional assumptions, we cannot further restrict the solution space of the Transfer Constraint (else Lemma 3 will be violated).

3.2. Training with a Single Source

Say we have labeled training data $T = \{(x,y)\}$ on the source domain s of size $|T| = n$. Here, for each (x,y), we have that that y is sampled from $\Pr[Y|X=x, D=s]$, but we do not consider the inputs x as random (i.e. our results hold for this particular fixed set of inputs).

We are interested in obtaining an estimator $\hat{\beta}_t$ with low ℓ_2 error on some domain t, defined as:

$$\mathrm{Risk}_t(\hat{\beta}_t) = \mathbb{E}[(\hat{\beta}_t^\top \Pi_t X - \beta_t^\top \Pi_t X)^2 | D=t]$$

Let us also denote the covariance functions as:

$$\widehat{\Sigma}_s = \frac{1}{n}\sum_{x\in T}(\Pi_s x)(\Pi_s x)^\top,\ \Sigma_t = \mathbb{E}[(\Pi_t X)(\Pi_t X)^\top | D=t]$$

Throughout this section we represent the linear maps $\beta_s^\top$ and $\beta_t^\top$ as $\dim_s$ and $\dim_t$ vectors (which live in their appropriate spaces) and the covariance functions as $\dim_s \times \dim_s$ and $\dim_t \times \dim_t$ matrices. Our results do not depend on the choice of coordinates used in this representation.

We now specify an algorithm for estimating β_t. First, note that empirical risk minimizer of β_s (with respect to the square loss on the training set T) is:

$$\hat{\beta}_s = \widehat{\Sigma}_s^{-1}\left(\frac{1}{n}\sum_{(x,y)\in T} y(\Pi_s x)\right) \tag{4}$$

where we have implicitly assumed that $\widehat{\Sigma}_s$ is invertible.

Now let us specify an estimate of $\hat{\beta}_t$ motivated by the Transfer Constraint (Equation 2). With respect to a projection $P_{s,t}$ onto a shared subspace, define

$$M_s = \Pi_s P_{s,t},\ M_t = \Pi_t P_{s,t},\ M_{s\to t} = M_s(M_t)^+ \ .$$

where A^+ is the Moore-Penrose pseudoinverse [1]. Note that in most practical cases (such as those in our experiments) we do not expect M_t to be invertible as the number of rows, $\dim_t$, will typically be much smaller than the number of columns (effectively determined by the common subspace dimensionality), so the pseudoinverse must be used. By Theorem 4, we know that $\beta_s^\top M_{s\to t}$ is an optimal estimator of $\beta_t^\top$ if M_t has full row rank (i.e. the rank of M_t equals the number of rows). This suggests the following estimator of β_t:

$$\hat{\beta}_t^\top = \hat{\beta}_s^\top M_{s\to t} \tag{5}$$

Now we provide a bound on the generalization error when using this estimator. In this theorem, we also

[1] Recall that if $A = UDV^\top$ is the "thin" singular value decomposition of A (so D is $\mathrm{rank}(A) \times \mathrm{rank}(A)$ invertible matrix), then $A^+ = VD^{-1}U^\top$.

assume that prefect transfer is possible, i.e. that M_t has full row rank.

Corollary 5. *(Generalization) Suppose Assumptions 1 and 2 hold. Also assume that: M_t has full row rank; $\widehat{\Sigma}_s$ is invertible; and the conditional variance of Y is bounded by 1, i.e. with probability one, $\mathrm{Var}(Y|X, D = s) \leq 1$.*

For the estimator $\hat{\beta}_s$ (specified in Equation 5), we have

$$\mathbb{E}\left[\mathrm{Risk}_t(\hat{\beta}_t)\right] \leq \frac{\mathrm{trace}(M_{s\to t}^\top \widehat{\Sigma}_s^{-1} M_{s\to t}\Sigma_t)}{n}$$

where n is the training set size, and the expectation is only with respect to the labels Y on the training set. Furthermore, if $\mathrm{Var}(Y|X, D = s) = 1$ with probability one, then the above holds with equality.

Let us interpret this result. The transfer operator $M_{s\to t}$ essentially relates the covariance form on training data, $\widehat{\Sigma}_s$, to the relevant covariance form on the target, Σ_t (it is this latter form for which we desire $\hat{\beta}_t$ to be accurately estimated under). Note that in the special case where $s = t$ (and if $\widehat{\Sigma}_s = \Sigma_t$, say with a sufficiently large sample), then $M_{s\to t}$ would be the identity and the trace would reduce to $\mathrm{dim_t}$ — resulting in the usual rate of $\frac{\mathrm{dim_t}}{n}$ for regression in a $\mathrm{dim_t}$ dimensional space. Hence, we can view $\mathrm{trace}(M_{s\to t}^\top \widehat{\Sigma}_s^{-1} M_{s\to t}\Sigma_t)$ as the "effective dimensionality" for learning with samples from source s. This number could be large if there are directions required to fit for β_t which are not effectively observed in the training data (as determined by $\widehat{\Sigma}_s$ and $M_{s\to t}$).

The proof is (essentially) a corollary of our multi-source generalization Theorem 6, provided in the next subsection.

Proof. (sketch) This result follows from Theorem 6 if we make the further requirement that $M_{s\to t}$ has full row rank (which is a condition stipulated in Theorem 6). To see this, using properties of the pseudo-inverse (with this extra condition), one can show that the estimator in Equation 9, is equivalent to the estimator used here (Equation 5) and that the bound in Theorem 6 reduces to this bound. Technically, this Corollary also holds without this additional row rank condition on $M_{s\to t}$, but the proof is not provided here (the proof is analogous to that of Theorem 6). □

3.3. Training with Multiple Sources

Now say our training data is from a set $\mathcal{S}$ of source domains, and, again, we desire to transfer to one or more target domains. Let T_s be our training set for source domain $s \in \mathcal{S}$ and let $n = \sum_s |T_s|$ (the cumulative training set size). The set of source domains could potentially include the target domain (so, as a special case, this setting includes the case where we have labeled data on both our source domains and target domain). We now specify an algorithm for estimating β_t. Computing an estimator for a different target domain t' is efficient, since, as we shall see, we need only store certain sufficient statistics for each set T_s.

Let us motivate how we construct such an estimator by considering a certain risk minimization problem. First, note that the sum cumulative error of the estimators $\{\hat{\beta}_s\}_{s\in\mathcal{S}}$ over all training sets is:

$$\sum_{s\in\mathcal{S}} \sum_{(x,y)\in T_s} (y - \hat{\beta}_s^\top \Pi_s x)^2 \tag{6}$$

Naively, we could minimize this to find the empirical risk minimizing estimates of $\{\beta_s\}$. However, we are interested in an estimator of β_t.

By the Theorem 4, we know that $\beta_s^\top M_{s\to t}$ is an optimal estimate of $\beta_t^\top$ if M_t has full row rank. Define:

$$M_{t\to s} = M_{s\to t}^+$$

so we can view $\beta_t^\top M_{t\to s}$ as an estimator of $\beta_s^\top$ (with equality if $M_{s\to t}$ has full row rank). This observation leads us to consider the following cumulative loss:

$$\sum_{s\in\mathcal{S}} \sum_{(x,y)\in T_s} (y - \hat{\beta}_t^\top M_{t\to s}\Pi_s x)^2 \tag{7}$$

where have just substituted $\hat{\beta}_t^\top M_{t\to s}$ in place of $\hat{\beta}_s^\top$ in the previous cumulative loss (Equation 6).

We can write the empirical minimizer as follows: let

$$A_s = \sum_{x\in T_s} (\Pi_s x)(\Pi_s x)^\top, \quad B_s = \sum_{(x,y)\in T_s} y(\Pi_s x)$$

and define:

$$\widehat{\Sigma}_{\mathcal{S}} = \frac{1}{n}\sum_{s\in\mathcal{S}} M_{t\to s} A_s M_{t\to s}^\top \ . \tag{8}$$

It is straightforward to show that the minimizer is:

$$\hat{\beta}_t = \widehat{\Sigma}_{\mathcal{S}}^+ \left(\frac{1}{n}\sum_{s\in\mathcal{S}} M_{t\to s} B_s\right) \tag{9}$$

In the special case of having a single source, i.e. $\mathcal{S} = \{s\}$, if $M_{s\to t}$ has full row rank, then this estimator reduces to the previous estimator $\hat{\beta}_t$ in Equation 5 (shown using properties of the pseudo-inverse).

Furthermore, note that A_s and B_s are sufficient statistics for the training datasets T_s, so that for any new target domain t' we only need to have knowledge of these sufficient statistics to compute $\hat{\beta}_{t'}$. Hence, the algorithm efficiently transfers to any new domain.

Theorem 6. *(Multi-Source Generalization) Suppose Assumptions 1 and 2 hold. Also assume that: M_t has full row rank; $M_{s\to t}$ has full row rank for all $s \in \mathcal{S}$, $\widehat{\Sigma}_{\mathcal{S}}$ is invertible; and that with probability one,* $\mathrm{Var}(Y|X, D = s) \leq 1$ *(for all $s \in \mathcal{S}$).*

For the estimator $\hat{\beta}_t$ (specified in Equation 9), we have

$$\mathbb{E}\left[\mathrm{Risk}_t(\hat{\beta}_t)\right] \leq \frac{\mathrm{trace}(\widehat{\Sigma}_{\mathcal{S}}^{-1}\Sigma_t)}{n}$$

where n is the cumulative training set size and the expectation is with respect to the labels Y on the all training sets. Furthermore, if $\mathrm{Var}(Y|X, D = s) = 1$ *(with probability one), then the above holds with equality.*

As discussed earlier, we can view $\mathrm{trace}(\widehat{\Sigma}_{\mathcal{S}}^{-1}\Sigma_t)$ as the *effective dimensionality* of this multi-source transfer. Here, $\widehat{\Sigma}_{\mathcal{S}}$ determines how the observed covariance form from the multiple sources (after they have been pushed through the transfer, as specified by Equation 8) relate to the desired covariance form Σ_t.

Proof. First, we prove that $\hat{\beta}_t$ is an unbiased estimate of β_t. By assumption on M_t and Theorem 4, $\beta_t^\top = \beta_s^\top M_{s\to t}$, which implies $\beta_s^\top = \beta_t^\top M_{t\to s}$ (where the latter equation follows from the full row rank assumption on $M_{s\to t}$). Hence, $\mathbb{E}[Y|X, D = s] = \beta_s^\top \Pi_s X = \beta_t^\top M_{t\to s}\Pi_s X$, so for $(x, y) \in T_s$, we have

$$\mathbb{E}[y\Pi_s x] = \mathbb{E}(\Pi_s x)(\Pi_s x)^\top]M_{t\to s}^\top \beta_t$$

This implies that $\mathbb{E}[M_{t\to s}B_s] = M_{t\to s}A_s M_{t\to s}^\top \beta_t$ and so $\frac{1}{n}\sum_s \mathbb{E}[M_{t\to s}B_s] = \widehat{\Sigma}_{\mathcal{S}}\beta_t$. Hence, we have that $\mathbb{E}[\hat{\beta}_t] = \widehat{\Sigma}_{\mathcal{S}}^{-1}\widehat{\Sigma}_{\mathcal{S}}\beta_t = \beta_t$ (by assumption on $\widehat{\Sigma}_{\mathcal{S}}$).

Using this and Assumption 2, one can show:

$$\mathbb{E}[(\hat{\beta}_t - \beta_t)(\hat{\beta}_t - \beta_t)^\top] \leq \widehat{\Sigma}_{\mathcal{S}}^{-1}/n \qquad (10)$$

(where the inequality is on positive definite matrices). This holds with equality if $\mathrm{Var}(Y|X, D = s) = 1$.

Using that $\mathrm{trace}(AB) = \mathrm{trace}(BA)$, we can write the risk as follows:

$$\begin{aligned}
& \mathbb{E}[\mathrm{Risk}_t(\hat{\beta}_t)] \\
= \;& \mathbb{E}[(\hat{\beta}_t - \beta_t)^\top \Sigma_t(\hat{\beta}_t - \beta_t)] \\
= \;& \mathrm{trace}(\mathbb{E}[(\hat{\beta}_t - \beta_t)^\top \Sigma_t(\hat{\beta}_t - \beta_t)]) \\
= \;& \mathrm{trace}(\Sigma_t^{1/2}\mathbb{E}[(\hat{\beta}_t - \beta_t)(\hat{\beta}_t - \beta_t)^\top]\Sigma_t^{1/2}) \\
\leq \;& \mathrm{trace}(\Sigma_t^{1/2}\widehat{\Sigma}_{\mathcal{S}}^{-1}\Sigma_t^{1/2})/n
\end{aligned}$$

and the first claim follows. Furthermore, if the conditional variance is unity with probability one, then all the previous inequalities become equalities. □

4. Experiments

We now evaluate the algorithms described in Section 3. Our first set of experiments illustrate how zero-shot learning is possible, when we transfer from one source to one target (using the estimator specified in Equation 5). Our second set of experiments is in the multi-source setting, where we demonstrate the performance of our multi-source estimator (Equation 9).

4.1. Data and Setup

We use the publicly available sentiment dataset from [Blitzer et al., 2007] [2], which consist of reviews for four different categories of products from Amazon: books, DVDs, electronics, and kitchen appliances. There are roughly 5000 reviews from each domain, and each review is labeled with a 1,2,4, or 5 star rating (there are no 3-star reviews in this dataset). Our goal is to predict the rating using the text of the review.

Following Blitzer et al. [2007], we use unigrams and bigrams as features. We have approximately 10^5 features across all domains (after discarding those features which occur in only one document). Of these, approximately 28,000 are shared. The others are specific to particular domains and will be of no use in the original space for at least one adaptation setting, although the algorithms from Section 3 can exploit them.

The first step to applying our algorithms in this setting is to construct Π_d for each domain such that it (approximately) satisfies Assumption 2. As we mentioned in Section 2.1, our procedure is based on Canonical Correlation Analysis. While we would like to run CCA individually for each domain, our unlabeled data set is too small for this procedure to be stable. Because of this, we take an intermediate approach similar to that of Ando and Zhang [2007] and Blitzer et al. [2007]. Since CCA is essentially predicting one view with the other view, we first compute a single matrix W which is a prediction weight vector for 500 words which are shared (using all other words). This gives us a 500-dimensional representation for each X, namely WX. We now construct Π_d using a CCA on a random split of this lower dimensional WX on each domain.

The next step is to specify the shared space $\mathcal{X}_{\text{shared}}$. In our procedure, this is quite simple: $\mathcal{X}_{\text{shared}}$ is just the range of W, so $P_{s,t}$ is onto this space. With Π_d and $\mathcal{X}_{\text{shared}}$ in hand, we can now directly find $\hat{\beta}_t$ using Equations 5 and 9.

[2] `http://www.cis.upenn.edu/~mdredze/datasets/sentiment/`

(A) Zero-shot transfer

Targ	Books		DVD		Electronics		Kitchen	
Src	Base	**Tran**	Base	**Tran**	Base	**Tran**	Base	**Tran**
Books	1.50	**1.47**	1.70	**1.57**	2.15	**1.60**	2.0	**1.52**
DVD	1.69	**1.51**	1.61	**1.55**	2.13	**1.55**	1.97	**1.49**
Electronics	1.99	**1.66**	1.97	**1.66**	1.42	**1.40**	1.50	**1.39**
Kitchen	2.05	**1.63**	1.90	**1.66**	1.53	**1.43**	1.36	**1.35**

(B) Multi-source transfer

Targ	Books		Electronics	
Src	Base	**Tran**	Base	**Tran**
B+D	1.52	**1.51**	2.03	**1.46**
B+K	1.56	**1.53**	1.63	**1.42**
D+E	1.72	**1.51**	1.52	**1.43**
E+K	1.96	**1.53**	1.45	**1.42**

Table 1. **(A)** Results on zero-shot transfer across all 16 pairs of domains. Each row corresponds to a source domain and each group of two columns corresponds to a target domain. *Base* is the word and bigram baseline (trained as a (regularized) linear predictor in the high dimensional space). *Tran* runs the algorithm in Equation 5. The red numbers on the diagonal are gold standard results which involve training on 2000 points of in-domain data, so we cannot expect to perform significantly better than this (the Tran number for this column is slightly better as it was trained using the projection of X rather than using regularization in the high dimensional space). Finally, we note that the $\mathbf{0}$ vector has averaged squared error of 2.5 in every domain. **(B)** Results on multi-source transfer for several source combinations and the target domains of books and electronics. *B+D* indicates combining training data from books and DVDs. We always use 1000 training samples from each source. *Base* trains a combined linear predictor using both sources in the high-dimensional representation. *Tran* uses our multi-source method (Equation 9).

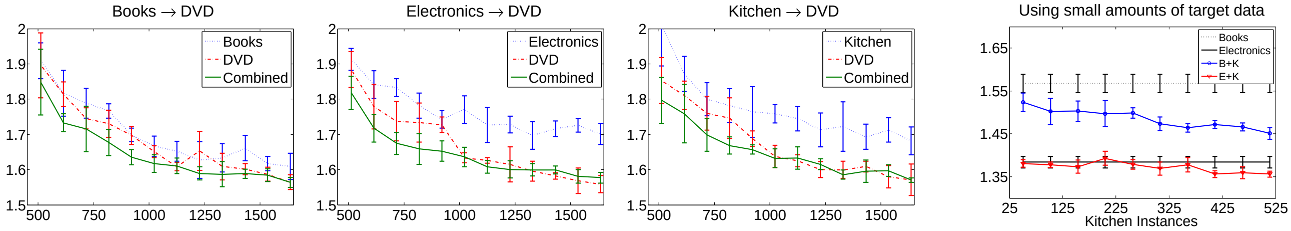

Figure 1. **Left 3 plots:** Source vs Target Data. All plots show test error rate (with DVD target domain) vs. training data size. Every curve we show here uses the canonical representation Π_d for each domain. The dotted (blue) curve trained with source data; the dashed (red) curve is trained with the target data; and the solid (green) curve is trained data which is half from the source domain and half from target domain. **Right plot:** We fix the number of source (books or electronics) instances to 2000 and show curves for increasing number of target (kitchen) instances.

4.2. Zero-shot Domain Adaptation

Table 1A illustrates our zero shot learning experiments when trained on a single source, with 2000 sample points, and tested on a single target (see Table 1 caption for details). We note that our four domains fall into roughly two groups. Books and DVD reviews share similar positive and negative vocabulary such as *riveting* and *boring*. Similarly electronics and kitchen appliances share many similar terms like *broken* or *reliable*. Transfer within these groups tends to be significantly easier than between groups. Our adaptation algorithm always outperforms the high-dimensional baseline. For example, from books to electronics, we achieve a 26% relative reduction in error. Even for domains which are significantly closer, we can achieve a huge reduction in error relative to the gold standard. For example, when transferring from kitchen appliances to electronics, the transfer predictor is as good as the gold standard.

4.3. Multi-source transfer

Multi-source transfer encompasses a much broader and challenging set of problems than single-source, and our experiments only explore a subset of these aspects.

Multiple large labeled source domains. Table 1B provides results when training on two sources, with 1000 points on each source (see Table 1 caption for details). Due to space constraints we only give results for two targets, but the results are similar for other combinations. Again, our results are consistently and significantly better. For some of these cases our target domain is one of our source domains and here, as expected, our improvement is not as substantial. We also find that even when both sources are quite different from the target, we perform significantly better than 2000 instances from either of the sources alone (For example, books and DVDs to electronics).

Learning Curves for Source vs Target Data. Here we empirically explore the question, "How much faster does having target data reduce the error vs having source data?" We illustrate this for the target domain of DVDs in Figure 1 (left three plots. See caption for details.). As expected, the source curve is typically worse (but not by much). However, the solid (green) shows that when half the training data is from the source and half is from the target, we can perform as well as (and sometimes better than) an equivalent amount of target data alone.

Learning from small amounts of target data. Of particular interest to natural language processing and information retrieval is the setting in which we have much less labeled target data when compared to source data [Daume, 2007, Chen et al., 2008]. Figure 1 (right) shows that we can make significant improvements to our model using as few as 50 target domain instances. As we might expect, this technique only makes sense for distant source domains (like books to kitchen), since transferring from closer domains is already nearly optimal.

References

R. Ando and T. Zhang. Two-view feature generation model for semi-supervised learning. In *ICML*, 2007.

A. Arygriou, C. Micchelli, M. Pontil, and Y. Yang. A spectral regularization framework for multi-task structure learning. In *NIPS*, 2007.

J. Baxter. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12:149–226, 2000.

S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. In *NIPS*, 2007.

S. Bickel, M. Brückner, and T. Scheffer. Discriminative learning for differing training and test distributions. In *ICML*, 2007.

J. Blitzer, M. Dredze, and F. Pereira. Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification. In *ACL*, 2007.

J. Blitzer, K. Crammer, A. Kulesza, and F. Pereira. Learning bounds for domain adaptation. In *NIPS*, 2008.

K. Chen, R. Liu, C.K. Wong, G. Sun, L. Heck, and B. Tseng. Trada: tree based ranking function adaptation. In *CIKM*, 2008.

C. Cortes, M. Mohri, M. Riley, and A. Rostamizadeh. Sample selection bias correction theory. In *ALT*, 2008.

K. Crammer, M. Kearns, and J. Wortman. Learning from multiple sources. In *NIPS*, 2007.

W. Dai, Q. Yang, G. Xue, and Y. Yu. Boosting for transfer learning. In *ICML*, 2007.

H. Daume. Frustratingly easy domain adaptation. In *ACL*, 2007.

D. Foster, S. Kakade, and T. Zhang. Multi-view dimensionality reduction via canonical correlation analysis. Technical Report TR-2008-4, TTI-Chicago, 2008.

J. Huang, A. Smola, A. Gretton, K. Borgwardt, and B. Schoelkopf. Correcting sample selection bias by unlabeled data. In *NIPS*, 2007.

S. Kakade and D. Foster. Multi-view regression via canonical correlation analysis. In *COLT*, 2007.

C. Legetter and P. Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models. *Computer Speech and Language*, 9:171–185, 1995.

Q. Liu, A. Mackey, D. Roos, and F. Pereira. Evigan: a hidden variable model for integrating gene evidence for eukaryotic gene prediction. *Bioinformatics*, 5:597–605, 2008.

Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain adaptation with multiple sources. In *NIPS*, 2009.

D. McClosky, E. Charniak, and M. Johnson. Reranking and self-training for parser adaptation. In *COLING-ACL 2006*, 2006.