Accelerating Active Learning with Transfer Learning

David Kale and Yan Liu
Department of Computer Science
University of Southern California
Los Angeles, CA 90007
Email: {dkale,yanliu.cs}@usc.edu

***Abstract*—Active learning, transfer learning, and related techniques are unified by a core theme: efficient and effective use of available data. Active learning offers scalable solutions for building effective supervised learning models while minimizing annotation effort. Transfer learning utilizes existing labeled data from one task to help learn related tasks for which limited labeled data are available. There has been limited research, however, on how to combine these two techniques. In this paper, we present a simple and principled transfer active learning framework that leverages pre-existing labeled data from related tasks to improve the performance of an active learner. We derive an intuitive bound on the generalization error for the classifiers learned by this algorithm that provides insight into the algorithm's behavior and the problem in general. We provide experimental results using several well-known transfer learning data sets that confirm our theoretical analysis. What is more, our results suggest that this approach represents a promising solution to a specific weakness of active learning algorithms: cold starts with zero labeled data.**

I. Introduction

In the age of the "data tsunami," we are confronted with a central challenge: how do we efficiently and effectively learn from massive amounts of data? Supervised learning remains the dominant learning paradigm for many practical problems, and many supervised learning problems can be formulated as classification. Learning a classifier requires class labels, which can be difficult or expensive to acquire in large quantities. In response to this dilemma, researchers have developed *active learning*. An active learner is given access to an (often human) *oracle* that can label data, a limited *budget* to spend on acquiring labels, and the freedom to choose which observations to label [1]. The goal of active learning is to build an effective classifier with as few label queries as possible.

Recent theoretical breakthroughs have produced active learning algorithms that are practical and have strong statistical consistency and unbiased sampling guarantees [2]. Nevertheless, there remain significant barriers to wider adoption of active learning. One challenge that has both practical and theoretical implications is the *cold start* phenonemon. Active learning requires a good classifier to generate useful label queries; training a good classifier requires labeled data. If the active learner begins with zero labeled data, then it must query labels at random until it has enough to train a good classifier. Thus, early in the query process or when the labeling budget is small, active learning offers little or no advantage over passive learning [3]. What is more, classifier performance (e.g., test set error) often improves slowly as a function of the number of label queries. The cold start problem has not been studied in earnest, although a number of approaches (e.g., cluster-based active learning) offer potential remedies [4].

Another promising solution to the cold start problem is *transfer learning*. The intuition behind transfer learning is that learning a new task should be easier if we transfer knowledge from previously learned tasks [5]. Related (or *source*) tasks often take the form of labeled data sets that are "similar" to our *target* task data. Examples include product reviews from different categories [6] or clinical trials data from different hospitals [7]. In these settings, straightforward supervised learning (train a model on source data, then apply it to target task) often produces models that perform poorly. However, with a proper transfer learning framework, we can use source data to improve our ability to learn the new task, especially when little or no labeled target data is available.

This suggests a strategy for addressing the cold start problem in active learning: *use transfer learning to initialize the active learner using data from a related task*. In this way, the active learner begins with a classifier to guide early label queries, eliminating the need to query at random. If the transfer from the source task is effective, then the active learner should begin with a good classifier and require many fewer target label queries to improve it. This would mitigate the cold start problem. If transfer learning produces a poor classifier, the active learner may be forced to query many more target labels in order to recover. In this way, we can understand transfer learning as providing an initial *bias* to the active learner. A good framework for combining transfer and active learning should provide a way to measure the impact of the transfer-based bias on the active learner's behavior and performance.

In this paper, we describe a simple, principled approach to transfer-initialized active learning, based on two relatively new frameworks for transfer learning [8] and active learning [2]. This approach is easy to implement and efficient, and it permits a theoretical analysis that provides insight into the interaction between these two learning paradigms. We derive a bound on the generalization error that relates target task performance to the similarity between source and target tasks. We identify a trade-off between potential sources of error that can be exploited to produce effective transfer active learners. We present experimental results that confirm our theory and show that this approach accelerates active learning. We conclude by identifying the most fruitful directions for future research.

II. Related Work

To our knowledge, there has been only a handful of papers, most of them quite recent, exploring the combination of transfer learning and active learning [9], [10]. [11] combines uncertainty region sampling with several transfer learning concepts, including the use of a *domain separation* classifier trained to distinguish between unlabeled source and target samples. The authors provide convincing empirical results on a number of standard transfer learning tasks, as well as a simple analysis

of label complexity and error rates. [12] describes a novel active transfer learning framework that combines sample reweighting with *batch mode active learning*, which chooses all of its label queries simultaneously. What makes this approach especially interesting is that it uses a different set of criteria to select queries: diversity among labeled samples and distributional similarity between labeled and unlabeled target data. Their empirical results indicate that this approach can be used to build effective classifiers with a small number of target label queries. Unfortunately, most of these approaches are heuristic in nature and lack guarantees for consistency and sampling bias. The most notable exception is [7], which presents a theoretically rigorous Bayesian framework for active transfer learning, based on *prior-dependent learning*. Assuming a prior distribution over *target concepts* (i.e., classifiers) greatly accelerates active learning, and the authors show that the prior is identifiable from a finite number of labeled examples in sequential multitask settings. The empirical effectiveness of this approach remains an open question.

III. Methodology

Our approach to *transfer active learning* combines two principled learning frameworks. For transfer learning, we use a *convex combination of source and target empirical risks* [8]. For active learning, we use the *importance weighted consistent active learner* (IWAL CAL) algorithm [2]. We provide a brief overview of each and then describe how to combine them in order to address the cold start problem.

A. Transfer learning framework

Formally, we define a *task* or *domain* as a distribution $\mathcal{D}$ on a set of points $\mathcal{X}$ paired with a labeling function $f : \mathcal{X} \mapsto \mathcal{Y}$, where $\mathcal{Y} = \{\pm 1\}$. In transfer learning, we seek to transfer knowledge from a *source* domain $\langle \mathcal{D}_{\mathrm{S}}, f_{\mathrm{S}} \rangle$ to a *target* domain $\langle \mathcal{D}_{\mathrm{T}}, f_{\mathrm{T}} \rangle$ of interest. When learning we search over a *hypothesis* space $\mathcal{H}$ for a function $h : \mathcal{X} \mapsto \mathcal{Y}$ that does a good job of predicting the true $f(\boldsymbol{x})$ for any point $\boldsymbol{x} \in \mathcal{X}$. We measure the quality of a hypothesis h by its *risk*, relative to a domain (e.g., the target domain T): $\epsilon_{\mathrm{T}}(h, f_{\mathrm{T}}) = \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}_{\mathrm{T}}} [\mathbb{1}\{h(\boldsymbol{x}) \neq f_{\mathrm{T}}(\boldsymbol{x})\}]$ where $\mathbb{1}$ is the indicator function. The *empirical risk* of a hypothesis, relative to a finite sample $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, is defined as $\hat{\epsilon}_{\mathrm{T}}(h, f_{\mathrm{T}}, (\boldsymbol{x})_{1:n}) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{h(\boldsymbol{x}_i) \neq f_{\mathrm{T}}(\boldsymbol{x}_i)\}$ where $(\boldsymbol{x})_{1:n}$ is a notational convenience. When it is clear from context, we will use shorthand, such as $\epsilon_{\mathrm{T}}(h)$ and $\hat{\epsilon}_{\mathrm{T}}(h, n)$.

Our goal is to choose a hypothesis to minimize the target risk ($h^* = \arg\min_{h \in \mathcal{H}} \epsilon_{\mathrm{T}}(h)$), though this is impossible in practice. Instead we minimize a weighted sum of empirical risks $\hat{\epsilon}_\alpha(h) = \alpha \hat{\epsilon}_{\mathrm{T}}(h, f, (\boldsymbol{x})_{1:n}) + (1-\alpha) \hat{\epsilon}_{\mathrm{S}}(h, f, (\boldsymbol{x})_{1:m})$ with scalar weight $\alpha \in [0, 1]$. We assume access to $m > 0$ labeled source examples and $n \geq 0$ labeled target examples.

This approach to transfer learning is attractive because of its simplicity and elegant theoretical properties. [8] derives an upper bound on the target generalization error of $\hat{h} = \arg\min_{h \in \mathcal{H}} \hat{\epsilon}_\alpha(h)$, the classifier that minimizes the combined empirical risk. This bound includes two particularly interesting terms that quantify the similarity between domains. The first is a hypothesis-dependent measure of the similarity between the source and target data distributions $\mathcal{D}_{\mathrm{S}}$ and $\mathcal{D}_{\mathrm{T}}$. Even if the domains share the same labeling function (i.e., $f_{\mathrm{S}} = f_{\mathrm{T}}$), training examples with different distributions may produce different classifiers. We define the $d_{\mathcal{H}}$ *distance* between two distributions:

$$d_{\mathcal{H}}(\mathcal{D}_{\mathrm{S}}, \mathcal{D}_{\mathrm{T}}) = 2 \sup_{h \in \mathcal{H}} |P_{\mathcal{D}_{\mathrm{S}}}\{\mathcal{A}_h\} - P_{\mathcal{D}_{\mathrm{T}}}\{\mathcal{A}_h\}|$$

where $\mathcal{A}_h = \{\boldsymbol{x} : \boldsymbol{x} \in \mathcal{X} \text{ where } h(\boldsymbol{x}) = +1\}$. This is the maximum possible difference between probability masses assigned by our domains to a set $\mathcal{A}_h$ of points classified as $+1$ by any hypothesis $h \in \mathcal{H}$. Now let $\mathcal{H}\Delta\mathcal{H} = \{g : g(\boldsymbol{x}) = +1 \text{ if } h(\boldsymbol{x}) \neq h'(\boldsymbol{x}) \text{ for given } h, h' \in \mathcal{H}\}$ be the *symmetric difference hypothesis space*. Additionally, let $\epsilon_{\mathrm{S}}(h, h')$ be the disagreement between two hypotheses $h, h' \in \mathcal{H}$ about the labels of points drawn from $\mathcal{D}_{\mathrm{S}}$ (likewise for ϵ_{T} and $\mathcal{D}_{\mathrm{T}}$). Then we can define a distance $d_{\mathcal{H}\Delta\mathcal{H}}$ for which the following inequality holds for all $h, h' \in \mathcal{H}$:

$$|\epsilon_{\mathrm{S}}(h, h') - \epsilon_{\mathrm{T}}(h, h')| \leq \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_{\mathrm{S}}, \mathcal{D}_{\mathrm{T}})$$

This distance places an upper bound on the difference between source label and target label disagreement between any two hypotheses $h, h' \in \mathcal{H}$. $d_{\mathcal{H}\Delta\mathcal{H}}$ has two useful properties: first, for any $\mathcal{H}$ with finite VC dimension, it can be computed from finite unlabeled samples $\mathcal{U}_{\mathrm{S}} \sim \mathcal{D}_{\mathrm{S}}$ and $\mathcal{U}_{\mathrm{T}} \sim \mathcal{D}_{\mathrm{T}}$ [13]. Second, it can be approximated using a *domain separator hypothesis*, i.e., a classifier from $\mathcal{H}$ trained to separate $\mathcal{U}_{\mathrm{S}}$ and $\mathcal{U}_{\mathrm{T}}$ [14].

The second term of interest is the *combined source and target risk*: $\epsilon^*_{\mathrm{ST}} = \min_{h \in \mathcal{H}} \epsilon_{\mathrm{S}}(h) + \epsilon_{\mathrm{T}}(h)$. This can be thought of as a general measure of the similarity between the source and target domains. A small ϵ^*_{ST} implies the existence a hypothesis $h \in \mathcal{H}$ that simultaneously minimizes source and target risk, which in turn implies minimal differences between data distributions and labeling functions. This corresponds to the traditional transfer learning assumption that domains are "sufficiently similar" [8]. We assume that ϵ^*_{ST} is negligible but acknowledge that this may not be true in real applications.

B. Active learning framework

IWAL CAL is an importance weighted *mellow* active learner designed for online learning settings: rather than choosing from a pool, it waits as points "arrive" in streaming fashion and queries each label with some probability. When a point's label is queried, it is assigned an *importance weight* inversely proportional to its query probability. Importance weights correct for bias that accrues during selective sampling. After seeing t points, we choose a classifier $\bar{h}_t$ to minimize the *importance weighted empirical risk* $\bar{\epsilon}(h, t) = \bar{\epsilon}(h, f, (\boldsymbol{x}, w)_{1:t}) = \frac{1}{t} \sum_{i=1}^{t} w_i \mathbb{1}\{h(\boldsymbol{x}_i) \neq f(\boldsymbol{x}_i)\}$. [2] proves that this is an unbiased estimate of the true risk and provides a nice deviation bound for it. When compared with aggressive active learners (e.g., uncertainty region sampling), mellow active learners often exhibit a slower rate of improvement in performance as a function of the number of queries. However, they have sounder theoretical properties and are more conducive to analysis [3].

Formally, for the tth unlabeled point $\boldsymbol{x}_t$, IWAL CAL queries the label $y_t = f(\boldsymbol{x}_t)$ with probability p_t computed using a *rejection threshold* function $p((\boldsymbol{x}, q, w)_{1:t-1}, \boldsymbol{x}_t)$. Here q_i is a binary indicator of whether the ith label was queried, and $w_i = 1/q_i$ is the importance weight (bounded from above since $p_i > 0$). $(\boldsymbol{x}, q, w)_{1:t}$ is a notational convenience for $\{(\boldsymbol{x}_1, q_1, w_1), \ldots, (\boldsymbol{x}_t, q_t, w_t)\}$. We can now redefine $\bar{\epsilon}$ over all points seen: $\bar{\epsilon}(h, t) = \bar{\epsilon}(h, f, (\boldsymbol{x}, q, p)_{1:t}) = \frac{1}{t} \sum_{i=1}^{t} \frac{q_i}{p_i} \mathbb{1}\{h(\boldsymbol{x}_i) \neq f(\boldsymbol{x}_i)\}$. Unlabeled points have $q_i = 0$ and so are ignored in the error. This risk estimator is unbiased; notice that $\mathbb{E}[Q_i / p_i] = \mathbb{E}[Q_i] / p_i = p_i / p_i = 1$ [2].

After seeing $t-1$ samples, IWAL CAL uses them to implicitly maintain a space $\mathcal{H}_{t-1}$ of candidate hypotheses that with high probability contains h^*, the optimal classifier in $\mathcal{H}$. The probability p_t of querying the label for $\boldsymbol{x}_t$ is inversely proportional to the level of disagreement in $\mathcal{H}_{t-1}$. The difference between the importance weighted empirical risk $\bar{\epsilon}(h, t-1)$ and the true risk $\epsilon(h)$ is bounded, giving us a method to compute p_t. Let $\bar{h}_{t-1} = \arg\min_{h\in\mathcal{H}} \ \bar{\epsilon}(h, t-1)$ be the hypothesis that minimizes the importance weighted empirical error. Next let $\bar{h}'_{t-1}$ be the hypothesis that minimizes this error but disagrees with $\bar{h}_{t-1}$ on $\boldsymbol{x}_t$'s label: $\bar{h}'_{t-1}(\boldsymbol{x}_t) \neq \bar{h}_{t-1}(\boldsymbol{x}_t)$. Then $G_t = \bar{\epsilon}(\bar{h}_{t-1}, t-1) - \bar{\epsilon}(\bar{h}'_{t-1}, t-1)$ is an estimate of the disagreement within $\mathcal{H}_{t-t}$ about the label of $\boldsymbol{x}_t$. If G_t exceeds the upper bound on disagreement, then $\bar{h}_{t-1}$ likely agrees with h^* on $\boldsymbol{x}_t$ and so it is probably unnecessary to query $\boldsymbol{x}_t$'s label. Thus, the label for $\boldsymbol{x}_t$ is queried with probability $p_t \approx \min\left\{1, \left(1/G_t^2 + 1/G_t\right) C_0 \log(t)/(t-1)\right\}$ where $C_0 = O(\log(|\mathcal{H}|/\delta))$ [2].

C. Transfer active learning

We now assume that an IWAL CAL active learner has access to m labeled points from the source domain and has seen t points from the target domain. We can define a new weighted empirical risk over these $m+t$ points, where the weights depend on m and t, the α parameter, and the IWAL CAL importance weights q_i/p_i:

Definition 1. *We define a **combined weighted empirical risk** for transfer-accelerated active learning as*

$$\bar{\epsilon}_\alpha(h, m, t) \triangleq \alpha\bar{\epsilon}_{\mathrm{T}}(h) + (1-\alpha)\hat{\epsilon}_{\mathrm{S}}(h)$$

or equivalently

$$\bar{\epsilon}_\alpha(h, m, t) \triangleq \frac{1}{m+t}\sum_{i=1}^{m+t} w_i \mathbb{1}\{h(\boldsymbol{x}_i) \neq f(\boldsymbol{x}_i)\}$$

where

$$w_i = \begin{cases} \frac{(1-\alpha)(m+t)}{m} & i \leq m \quad \text{(source)} \\ \frac{\alpha(m+t)}{tp_i} & i > m,\ q_i = 1 \quad \text{(labeled target)} \\ 0 & i > m,\ q_i = 0 \quad \text{(unlabeled target)} \end{cases}$$

These can be shown to be equivalent with a simple derivation. The first form is easier to analyze, allowing us to leverage the results from [2] for IWAL CAL and from [8] for transfer learning. The second form is easier to implement as it permits us to use any supervised learning routine that accepts individually weighted training data. **Algorithm 1** shows pseudocode for our Transfer IWAL CAL (TIWAL CAL) algorithm. It uses the combined weighted empirical risk in *Steps 3-6* of the algorithm. $\mathrm{Gbound}_t = \alpha\left(\sqrt{\frac{C_0 \log t}{t-1}} + \frac{C_0 \log t}{t-1}\right) + (1-\alpha)\sqrt{\frac{C_0 \log 2}{2m}}$ is the upper bound on the disagreement within $\mathcal{H}_{t-1}$. To obtain p_t, we solve the quadratic equation $G_t = \alpha\left(\frac{c_1}{\sqrt{p_t}} - c_1 + 1\right)\sqrt{\frac{C_0 \log t}{t-1}} + \alpha\left(\frac{c_2}{p_t} - c_2 + 1\right)\frac{C_0 \log t}{t-1} + (1-\alpha)\sqrt{\frac{C_0 \log 2}{2m}}$. The constants C_0, c_1, and c_2 can be treated as tunable parameters but are defined for analysis as follows: $C_0 = O(\log(|\mathcal{H}|/\delta)$, $c_1 = 5+2\sqrt{2}$, and $c_2 = 5$. This algorithm uses labeled source data to provide the active learner with a transfer-based bias that can improved by labeling target data. In the next section, we show that TIWAL CAL's behavior and performance depend upon the similarity between source and target domains and the value of α.

Algorithm 1 Transfer IWAL CAL

1: **for** $t = 1, 2, \ldots$ until target samples exhausted **do**
2: Receive unlabeled $\boldsymbol{x}_t$
3: Compute weights $(w)_{1:(t-1)}$ as in **Definition 1**.
4: Choose $\bar{h}_{t-1} = \arg\min_{h\in\mathcal{H}} \bar{\epsilon}_\alpha(h, t-1)$
5: Choose $\bar{h}'_{t-1} = \arg\min_{h\in\mathcal{H}: h(\boldsymbol{x}_t)\neq\bar{h}_{t-1}(\boldsymbol{x}_t)} \bar{\epsilon}_\alpha(h, t-1)$
6: Set $G_t = \bar{\epsilon}_\alpha(\bar{h}'_{t-1}, t-1) - \bar{\epsilon}_\alpha(\bar{h}_{t-1}, t-1)$
7: **if** $G_t \leq \mathrm{Gbound}_t$ **then**
8: Set $p_t = 1$
9: **else**
10: Solve for p_t (see below)
11: **end if**
12: Sample $q_t \sim \mathrm{Bernoulli}(p_t)$
13: **if** $q_t = 1$ **then**
14: Query label y_t
15: Set $w_t = 1/p_t$
16: **end if**
17: **end for**
18: **return** $\bar{h}_t = \arg\min_{h\in\mathcal{H}} \bar{\epsilon}_\alpha(h, t)$, $(\boldsymbol{x}, y, q, 1/p)_{1:t}$

D. Deviation and generalization bounds

We provide two useful bounds that codify the above intuition and guide the application of our algorithm. **Lemma 1** places an upper bound on the *deviation* of the combined weighted empirical risk $\bar{\epsilon}_\alpha(h, m, t)$ from the true combined risk $\epsilon_\alpha(h, m, t)$. This directly motivates *Steps 6-10* of **Algorithm 1**, in which we compute Gbound_t and p_t and decide whether to query $\boldsymbol{x}_t$'s label.

Lemma 1. *With probability at least $1-\delta$, the following holds for all $t \geq 1$ and all $h \in \mathcal{H}$:*

$$\begin{aligned}|(\bar{\epsilon}_\alpha(h, m, t) \ &-\bar{\epsilon}_\alpha(h^*_\alpha, m, t)) - (\epsilon_\alpha(h) - \epsilon_\alpha(h^*_\alpha))| \\ &\leq \alpha\left(\sqrt{\frac{\varepsilon_t}{p_{\min,t}(h)}} + \frac{\varepsilon_t}{p_{\min,t}(h)}\right) + (1-\alpha)\sqrt{\varepsilon_{\mathrm{S}}}\end{aligned}$$

where $\varepsilon_t = O\left(\log(t|\mathcal{H}|/\delta)/t\right)$, $\varepsilon_{\mathrm{S}} = O\left(\log(2|\mathcal{H}|/\delta)/(2m)\right)$, and $p_{\min,t}(h)$ is the minimum query probability assigned to a target point about whose label h and h^* disagree.

The proof of this lemma involves decomposition of the combined empirical risk, followed by an application of the triangle inequality and Hoeffding's inequality.[1] **Theorem 1** places an upper bound on the target risk of $\bar{h}_t$.

Theorem 1. *For $\bar{h}_t = \arg\min_{h\in\mathcal{H}} \bar{\epsilon}_\alpha(h, m, t)$, the following holds with probability at least $1-\delta$:*

$$\begin{aligned}\epsilon_{\mathrm{T}}(\bar{h}_t) \leq \epsilon_{\mathrm{T}}(h^*_{\mathrm{T}}) &+ \alpha\left(\sqrt{\frac{2C_0\log(t+1)}{t}} + \frac{2C_0\log(t+1)}{t}\right) \\ &+ 2(1-\alpha)\left(\sqrt{\frac{C_0\log 2}{2m}} + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_{\mathrm{S}}, \mathcal{D}_{\mathrm{T}}) + \epsilon^*_{\mathrm{ST}}\right)\end{aligned}$$

The proof proceeds along lines similar to that of *Theorem 2* from the appendix of [8].[1]

[1] Complete proofs are available at `http://www-scf.usc.edu/~dkale/active-transfer/`.

Implication: This is an interesting and intuitive bound that trades off two sources of error via the parameter α. The active learning error term (with coefficient α) decreases as t grows large but may be significant early on. The transfer learning error term (with coefficient $1-\alpha$) depends primarily upon the number of source points m and upon the dissimilarity of our source and target domains, which is determined primarily by the $d_{\mathcal{H}\Delta\mathcal{H}}$ distance since we assume ϵ^*_{ST} is small enough to be ignored. The transfer learning error can be viewed as constant; it does not depend directly on t or change as we query labels. If the domains are substantially different and transfer learning error is high, an injudicious choice of α can introduce a large and constant negative bias, from which the active learner may never recover and which may increase the overall number of queries made. If the domains are sufficiently similar and transfer error is small, then a careful choice of α should significantly improve performance early on in the query process and reduce the overall number of queries. [8] gives a detailed analysis of choices of α that also applies here. A rule of thumb is that lower values of α should work well when the domains are similar and we have a lot of source data. Otherwise, we should use a higher value of α or even consider plain active learning.

IV. Experiments

We compare IWAL CAL and TIWAL CAL using two publicly available transfer learning data sets.[2] For each, we choose a target domain and divide it into a test set and two training sets. The first is treated as unlabeled to start and is used for active learning. The second is treated as a labeled source domain. We also choose two additional labeled source domains. We then compare the test set error and query rates of TIWAL CAL against IWAL CAL.

The details of our data sets are shown in the table in **Figure 3**. These include approximate $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ distances using a linear domain separator and hinge loss. Our base learner is a linear model with hinge loss and L2-regularization. For the free parameters in TIWAL CAL, we follow [2] by setting $c_1 = c_2 = 1$ and dropping the $\log(t)$ terms when computing Gbound_t and p_t. We use a heuristic to learn the constrained hypothesis $\bar{h}'_t$: set the instance weight for $\boldsymbol{x}_t$ to be equal to the sum of the weights for the rest of the training data. The above changes result in an approximation to our the abstract algorithm that works in practice. We use $C_0 = 0.25$ for IWAL CAL and $25 \leq C_0 \leq 100$ for TIWAL CAL.

A. Data

20 Newsgroups: Our first data set is the *20 Newsgroups* data set. We create a handful of "category versus category" classification tasks. Our target domain is a subset of `rec.sport.baseball vs. talk.politics.misc` (`BvP`). Our source domains include a second subset of `rec.sport.baseball vs. talk.politics.misc` (`BvP2`), `rec.sport.hockey vs. talk.religion.misc` (`HvR`), and `rec.autos vs. soc.religion.christian` (`AvC`). The original *20 Newsgroups* data has 61,188 word counts, which we convert to log term frequency. We then reduce the number of features by keeping only the 250 words with the top term-frequency inverse document-frequency (TF-IDF) scores across all categories. This is an efficient way to choose a small number of interesting features without using labels [15]. It also changed the $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ distance between domains, making for interesting experiments.

Sentiment: Our second data set is the *sentiment classification* data set [6]. We use the preprocessed binary ("positive" vs. "negative" review) version, with a subset of `kitchen` as our target domain and a second subset of `kitchen` (`kitchen2`), `dvd`, and `electronics` as our source domains. The preprocessed version of the data includes 1,110,352 unigram and bigram count features. As with *20 Newsgroups*, we convert these to log counts and then keep only the 1000 features with the top TF-IDF scores across all domains.

B. Results

Typical results are shown in **Figures 1** and **2**. The first thing to observe is that basic IWAL CAL falls prey to the cold start phenomenon. For *sentiment*, IWAL CAL requires nearly 400 queries to reach error of 0.20 or less and thousands of queries before it reaches the same performance as fully supervised learning. On the easier *20 Newsgroups* data set, IWAL CAL still needs 200 queries to achieve the supervised-level performance (error of 0.10). The results for TIWAL CAL are consistent with our analysis: the transfer learning bias drastically improves test set error early in the query process (with early error rates near optimal) and reduces the overall number of queries by as much as 50%. Further, TIWAL CAL often converges to nearly the same error rate as IWAL CAL, suggesting little or no negative bias. The exceptions to this pattern are the `AvC` and `dvd` source domains. Both yield less early improvement in test set error, and `dvd` actually increases the overall number of queries. These results are explained by our theory: each has a relatively high $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ distance with its respective target domain. Nonetheless, even for these sources, there is still an early 30-40% reduction in error, while the "penalty" on future test set error is relatively small (TIWAL CAL converges to a 5-10% higher error).

V. Discussion

Researchers are increasingly interested in how to introduce useful biases into active learning without compromising consistency guarantees.[3] This will allow active learners to produce good classifiers faster, mitigating the cold start problem. In this paper, we presented a principled framework that addresses cold starts by using transfer learning to leverage data from related tasks. Our framework is straightforward to analyze and apply. We proved a generalization bound that provides intuition into the problem and helps trade off different sources of error. We demonstrated empirical results that suggest this approach significantly improves classifier performance early in the query process and reduces the overall number of target label queries. In other words, we can accelerate active learning with transfer learning. Our work establishes a sound foundation that will facilitate future research on this topic and empower practitioners to apply these ideas to real world problems.

Our empirical results are modest; their primary virtue is consistency with our theoretical analysis. Clearly further experimentation and evaluation are warranted. In particular, it sometimes appears as though transfer learning is doing most of the work and that active learning plays little role other than to

[2] Code and data to reproduce our experiments can be found at `https://github.com/uscmelady/active-transfer`.

[3] See "The End of the Beginning of Active Learning" by Daniel Hsu and John Langford at `http://hunch.net/?p=1800`.

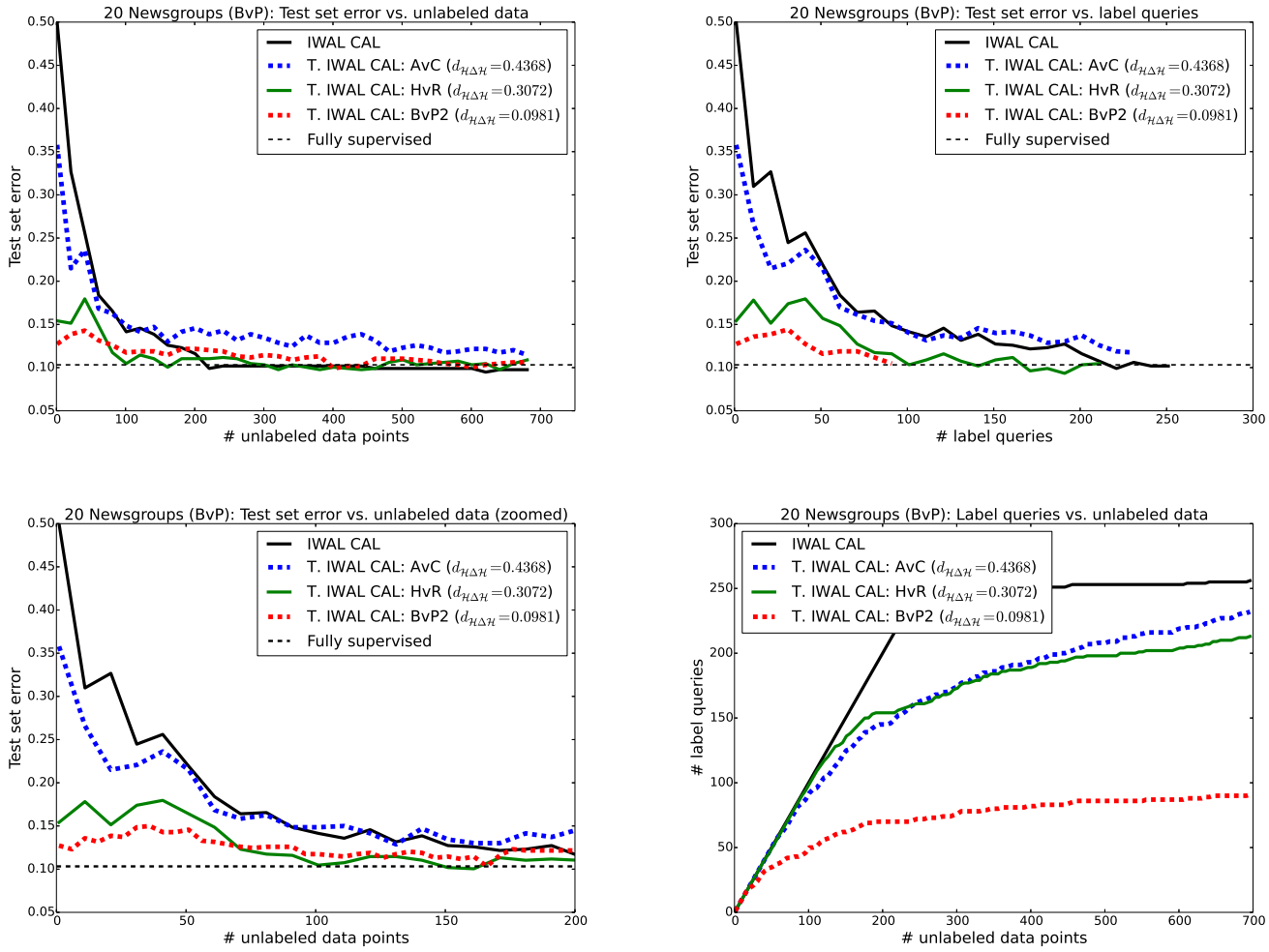

Fig. 1. `20 Newsgroups` results. The lefthand plots show test set error versus number of points seen by the active learner (the bottom is zoomed). The upper right shows test set error versus number of queries. The bottom right plot shows the query rate (number of queries versus number of points seen).

manage the label query budget. This is due to the conservative nature of IWAL CAL; the results in [2] are similarly modest. The natural question is whether this framework can be used with more aggressive approaches to active learning, such as uncertainty region sampling, while avoiding the usual sampling bias problems. We are pursuing this line of work.

One unsatisfying property of our framework is the persistent nature of the transfer-based bias. This introduces a constant source of error into our generalization bound and may prevent us from learning an optimal classifier, even with a large number of labeled target examples. Intuitively, with enough target data, we should de-emphasize the source data when training classifiers. One simple strategy that we are investigating is to gradually increase α (the weight on target risk) as we query more target labels. Another promising approach would be to combine active learning with an adaptive transfer learning framework that re-weights or transforms the source data to reduce the difference between domains [16].

ACKNOWLEDGMENTS

We would like to thank Sanjoy Dasgupta, Byron Wallace, and Taha Bahadori for many illuminating discussions and for their insightful feedback. This work was partially supported by the U.S. Defense Advanced Research Projects Agency (DARPA) under the Social Media in Strategic Com- munication (SMISC) program, Agreement Number W911NF-12-1-0034 and a Yahoo! Faculty Award.

REFERENCES

[1] B. Settles, *Active Learning*, ser. Synthesis Lectures on Artificial Intelligence and Machine Learning Series. Morgan & Claypool Publishers, Jun. 2012, vol. 6, no. 1.

[2] A. Beygelzimer, J. Langford, D. Hsu, and Z. Tong, "Agnostic Active Learning Without Constraints," in *Advances in Neural Information Processing Systems 23*, 2011, pp. 199–207.

[3] S. Dasgupta, "Two Faces of Active Learning," *Theoretical Computer Science*, vol. 412, no. 19, pp. 1767–1781, 2011.

[4] X. Zhu, J. Lafferty, and Z. Ghahramani, "Combining Active Learning and Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions," in *ICML 2003 Workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*, 2003, pp. 58–65.

[5] S. Thrun, "Is Learning the Nth Thing Any Easier Than Learning The First?" in *Advances in Neural Information Processing Systems 8*, 1996, pp. 640–646.

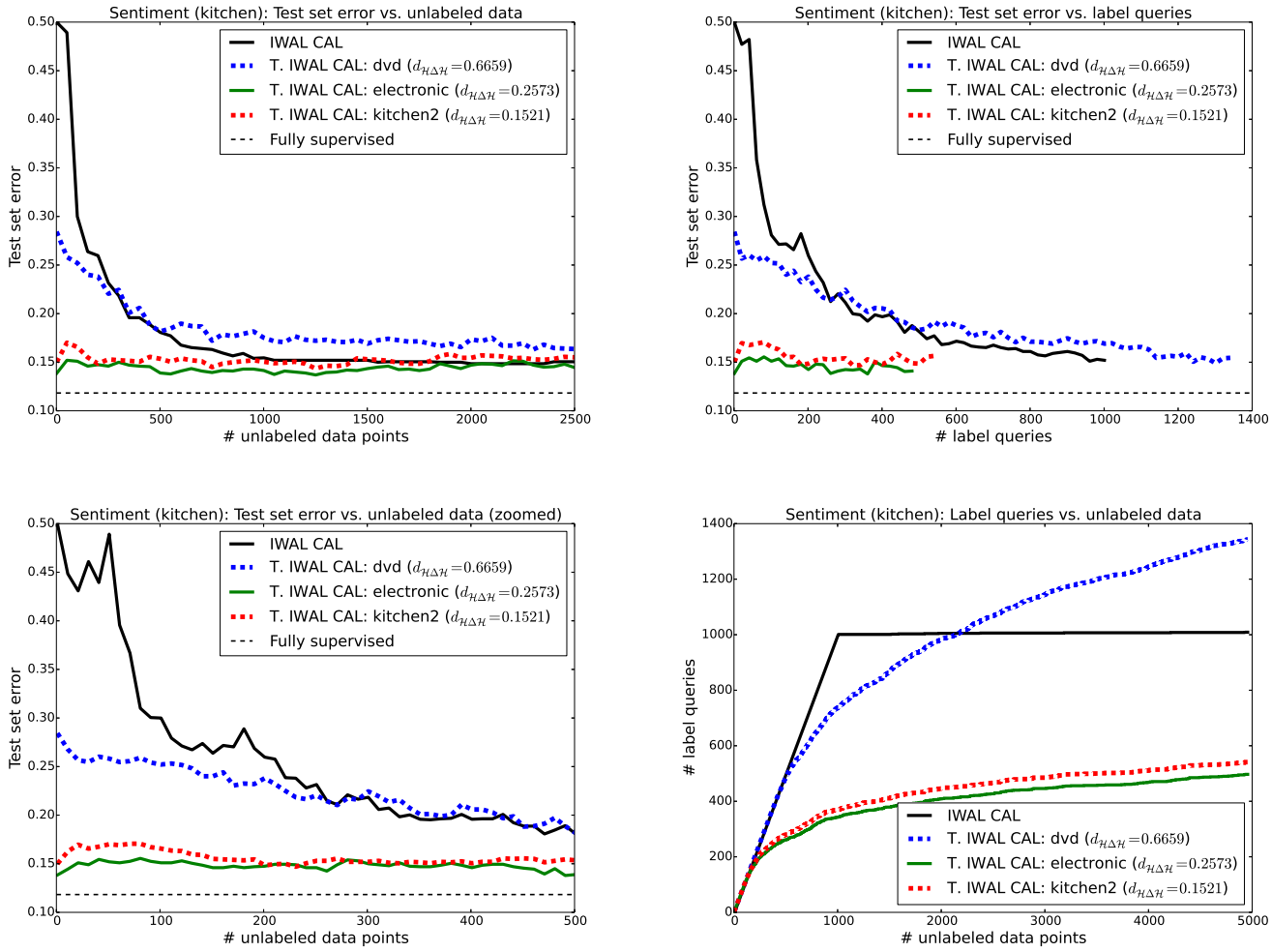

Fig. 2. `sentiment` results. The lefthand plots show test set error versus number of points seen by the active learner (the bottom is zoomed). The upper right shows test set error versus number of queries. The bottom right plot shows the query rate (number of queries versus number of points seen).

Source	m	Target	$\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$	α
`BvP2`	360	`BvP`	0.0981	0.3
`HvR`	974	`BvP`	0.3072	0.6
`AvC`	1191	`BvP`	0.4368	0.9
`kitchen2`	1001	`kitchen`	0.1521	0.3
`electronics`	5760	`kitchen`	0.2573	0.3
`dvd`	4189	`kitchen`	0.6659	0.9

Fig. 3. Summary of experiments, including number of labeled source examples and approximate $d_{\mathcal{H}\Delta\mathcal{H}}$ distances, and values of α.

[6] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, Bollywood, Boomboxes and Blenders: Domain Adaptation for Sentiment Classification," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, 2007.

[7] L. Yang, S. Hanneke, and J. Carbonell, "Two Faces of Active Learning," *Machine Learning*, vol. 90, no. 2, pp. 1–28, 2012.

[8] J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman, "Learning Bounds for Domain Adaptation," in *Advances in Neural Information Processing Systems 21*, 2008, pp. 1–12.

[9] Y. S. Chan and H. T. Ng, "Domain Adaptation with Active Learning for Word Sense Disambiguation," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 2007.

[10] X. Shi, W. Fan, and J. Ren, "Actively Transfer Domain Knowledge," in *Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases*, 2008, pp. 342–357.

[11] A. Saha, P. Rai, H. Daumé, S. Venkatasubramanian, and S. L. DuVall, "Active Supervised Domain Adaptation," in *Proceedings of the 2011 European Conference on Machine Learning and Knowledge Discovery in Databases*, 2011, pp. 97–112.

[12] R. Chattopadhyay, S. M. Ave, W. Fan, and I. Davidson, "Joint Transfer and Batch-mode Active Learning," in *Proceedings of the 30th Annual International Conference on Machine Learning*, 2013.

[13] D. Kifer, S. Ben-David, and J. Gehrke, "Detecting Change in Data Streams," in *Proceedings of the 30th international Conference on Very Large Databases*, 2004, pp. 180–191.

[14] S. Ben-David, J. Blitzer, K. Crammer, and O. Pereira, "Analysis of Representations for Domain Adaptation," in *Advances in Neural Information Processing Systems 19*, 2007, pp. 137–144.

[15] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Proceedings of the 14th International Conference on Machine Learning*, 1997, pp. 412–420.

[16] M. T. Bahadori, Y. Liu, and D. Zhang, "Learning with Minimum Supervision: A General Framework for Transductive Transfer Learning," in *Proceedings of the 11th IEEE International Conference on Data Mining*, 2011, pp. 61–70.