# Source Free Transfer Learning for Text Classification

**Zhongqi Lu**[†], **Yin Zhu**[†], **Sinno Jialin Pan**[‡], **Evan Wei Xiang**[§], **Yujing Wang**[♮] and **Qiang Yang**[†♯]
[†]Hong Kong University of Science and Technology, Hong Kong
[‡]Institute for Infocomm Research, Singapore 138632
[§]Baidu Inc., China
[♮]Microsoft Research Asia, Beijing, China
[♯]Huawei Noah's Ark Lab, Hong Kong
[†]{zluab, yinz, qyang}@cse.ust.hk, [‡]jspan@i2r.a-star.edu.sg, [§]xiangwei@baidu.com, [♮]yujwang@microsoft.com

## Abstract

Transfer learning uses relevant auxiliary data to help the learning task in a target domain where labeled data is usually insufficient to train an accurate model. Given appropriate auxiliary data, researchers have proposed many transfer learning models. How to find such auxiliary data, however, is of little research so far. In this paper, we focus on the problem of auxiliary data retrieval, and propose a transfer learning framework that effectively selects helpful auxiliary data from an open knowledge space (e.g. the World Wide Web). Because there is no need of manually selecting auxiliary data for different target domain tasks, we call our framework Source Free Transfer Learning (SFTL). For each target domain task, SFTL framework iteratively queries for the helpful auxiliary data based on the learned model and then updates the model using the retrieved auxiliary data. We highlight the automatic constructions of queries and the robustness of the SFTL framework. Our experiments on 20NewsGroup dataset and a Google search snippets dataset suggest that the framework is capable of achieving comparable performance to those state-of-the-art methods with dedicated selections of auxiliary data.

## Introduction

Because of the high cost of human labelling, the training data for a classification task is usually hard to obtain. In order to tackle the training data insufficiency problem, frontier researches try to transfer the knowledge from some auxiliary data to help the learning in the target domain task. This learning paradigm is known as Transfer Learning (Pan and Yang 2010). The performance of most transfer learning methods highly depends on the availability of appropriate auxiliary data. Previous research in transfer learning chooses the auxiliary data by human instinct, or by empirically trying several different auxiliary datasets. This auxiliary data selection process is indeed non-trivial. For example, in the experiments of (Dai et al. 2007), the authors deliberately select auxiliary documents that have the same top category as those in the target domain tasks. However in practice, how to define "the same top category" and how to find such data may be harder than the subsequent knowledge transfer itself.



Our work tries to free the auxiliary data selection process. In contrast with previous transfer learning techniques, Source Free Transfer Learning (SFTL) is not limited to use pre-defined auxiliary data to help a target domain task. We aim to iteratively query an open knowledge space, such as the World Wide Web, to retrieve auxiliary data to help *any* target domain task. And in this paper, we focus on applying SFTL to text classification problems.

To automatically retrieve auxiliary data for transfer learning, a straightforward approach may consists of two steps: 1) to retrieve relevant knowledge based on the target domain data, and 2) to use the retrieved relevant knowledge as auxiliary data and perform existing transfer learning techniques. However based on our observations, both steps are not trivial. On one hand, it may be insufficient to only define the relevance between the auxiliary data and target domain training data, because some auxiliary data could be related to both classes of target domain classification tasks, which harms the learning of target domain models. On the other hand, the retrieved auxiliary data is usually not labeled. And even if it is labeled, the auxiliary data usually do not share the same label set with the target domain data (Xiang et al. 2011). This restriction fails most of the transfer learning frameworks. Therefore, the straightforward approach mainly suffers from the following two problems: 1) some relevant knowledge may be harmful, and 2) existing transfer learning techniques cannot directly make use of the auxiliary data from an open knowledge space.

When solving a problem, we, as humans, first try to see if we already have the necessary knowledge. If such knowledge is not yet learned in the past but we find clues from existing knowledge, we try to retrieve it from other open sources, such as the World Wide Web, books and so on. Note that those sources are not pre-defined and are obtained during the learning process. Inspired by the natural learning process of human beings, we propose our learning framework. We train the model based on both the training and retrieved auxiliary data in an open knowledge space in an iterative manner. For each iteration, we first learn a model with the retrieved auxiliary and target domain training data, and then test it on the target domain validation data. We query for additional auxiliary data to train an enhanced model for the next iteration till the model is satisfiable.

In this framework, there are two key issues: 1) what aux-

iliary data to query, and 2) how to use the retrieved auxiliary data to help the target domain task. To address the first issue, we inspect our model to find out the words that are more likely to discriminate the target domain classes. Those words are then extended via a word-to-word relevance matrix to form the queries. To tackle the second issue, we introduce a graph-based data-dependent prior or regularizer to the logistic regression model. The retrieved auxiliary data is used to regularize model learning of the target domain task. In addition, to prevent SFTL from overfitting when the target domain training data is scarce, we iteratively query for auxiliary data and update the model.

## Related Work

### Transfer Learning

Pan and Yang (2010) surveyed the field of transfer learning and pointed out three main research issues: when to transfer, what to transfer and how to transfer. Our framework mainly tackles the "what to transfer" and "how to transfer" issues. TrAdaBoost (Dai et al. 2007) is one of the most related work to our proposed framework, which performs selection on the auxiliary data via a boosting-style algorithm. However, to ensure the success of knowledge transfer, the auxiliary data used in TrAdaBoost needs to be carefully chosen, so that at least part of the auxiliary data follows the same generative distribution as that of the target domain data. Besides, with the increasing size of the auxiliary data, TrAdaBoost suffers in terms of computation cost. Recently, a source-selection-free transfer learning framework (Xiang et al. 2011) is proposed to free the users from selecting auxiliary domain data. However, the method is not yet "source free", because it still needs to pre-define a large set of auxiliary data, although such auxiliary data is not task-specific. Besides, it requires the auxiliary data to be labeled, which is a rigorous restriction in many real-world scenarios. Lu et al. (Lu et al. 2013) proposed to selectively use the source domain data, whose selection process could be used to guide querying the auxiliary data in our proposed SFTL framework.

### Semi-supervised Learning

Similar to semi-supervised learning (Zhu 2006), we also use both the labeled and unlabeled data. However we do not assume that all the unlabeled data is with the same label set as that of the supervised learning task. In our framework, the labels of the unlabeled data can be different to those of the labeled data in the target domain task.

### Self-taught Learning

Self-taught Learning (Raina et al. 2007) is proposed to first use a very large number of unlabeled images to construct a high-level feature space, then project the labeled samples onto the high-level feature space and train the classification models. The authors made a strong assumption that the high-level features summarized from a very large set of randomly acquired data could form a good representation of the target domain data. Our work is different with Self-taught Learning mainly in two ways: First, we do not rely on the construction of the high-level features. Instead, we are interested in the interpretable features, such as using the words as the features. Second, our framework does not need to precompute over the whole knowledge base. In other words, we do not perform the expensive computations on every samples in the knowledge base.

### Active Learning

In Active Learning (Settles 2009), a pool of unlabeled data is assumed to be available to query for labels with budget. Therefore, one can use similarity between unlabeled and labeled data, or distance from unlabeled data point to the margin of classification model, etc, as the criterion for query unlabeled data to be labeled. However, in SFTL, we consider the whole Web as an infinite and distributed pool of unlabeled data, which indeed is *not* available to process before hand. In other words, there is no observed unlabeled data at the beginning when we selectively retrieve auxiliary data. Therefore, we propose to generate *keywords* based on the weights of the words (i.e.,features) to retrieve unlabeled auxiliary documents. Besides, even though for unsupervised active learning methods (Settles 2009) which do not use label information to select unlabeled data, there is an assumption that unlabeled data share the same label space with labeled data, which is not necessary in SFTL.

### Encyclopedic Knowledge

Some previous work has been proposed to use auxiliary knowledge, such as encyclopedia, to help text classification. In (Gabrilovich and Markovitch 2006; Egozi, Markovitch, and Gabrilovich 2011), researchers proposed to augment the keyword-based text representation of documents via knowledge from encyclopedia. Wang et al. (2007) further proposed to use structural knowledge of encyclopedia. In summary, more previous work using encyclopedic knowledge is to enrich the feature representations of text data. While a good feature representation is helpful for learning text-based tasks, the generation of feature representation will often bring much noise (Gabrilovich and Markovitch 2005). As an alternative use of the auxiliary knowledge, such as encyclopedia or the World Wide Web, we simply use the occurrences of words to represent documents, and focus on developing a transfer learning approach to leverage the helpful auxiliary knowledge to learn models for the target tasks.

## Source Free Transfer Learning

### Problem Settings

We assume that an instance has $Z$ features, and to cope with our learning framework in text classification, we define that each feature corresponds to a word in a dictionary. The value of a feature is the occurrences of a corresponding word in the corresponding document. We denote an instance by $\mathbf{x}$ and the value of the $k^{th}$ feature as $x_k$. Furthermore, we use $w_k, k \in \{1, \ldots, Z\}$, to denote the weight of the $k^{th}$ feature, and $f_i$ to denote a linear function of an instance $\mathbf{x_i}$,

$$f_i = f(\mathbf{x_i}) = y_i = \mathbf{w}^T \mathbf{x_i}. \quad (1)$$

We assume that the probability of an instance $\mathbf{x_i}$ being drawn from the positive class is $p(y = +1 \mid \mathbf{x_i}) = g(f_i)$, where $g(z) = \frac{1}{1+e^{-z}}$.

In our problem, we are given a labeled training set $D_L = \{(\mathbf{x_1}, y_1), \ldots, (\mathbf{x_n}, y_n)\}$ in the target domain. The goal is to learn a binary classifier for the target domain, where the labels are $+1$ or $-1$. In addition, an open knowledge space, such as the World Wide Web, is available, but no auxiliary data is pre-defined. In the following, we propose to automatically query for some unlabeled instances $D_U = \{\mathbf{x_{n+1}}, \ldots, \mathbf{x_{n+m}}\}$ from the open knowledge space as auxiliary data. Note that $D_U$ is iteratively incremented. We first discuss how to locate the helpful auxiliary data to augment $D_U$ based on the current $\mathbf{w}$, then propose a model to update $\mathbf{w}$ with the augmented $D_U$.

## Locate the Helpful Auxiliary Data

Given an open knowledge space, such as the whole Internet, it would be most desirable to *selectively* retrieve *helpful* information on the target domain task. We propose to first generate some keywords based on the learned model, then construct queries based on these keywords, finally query an open knowledge space for the auxiliary data.

**Keywords Generation** In our learning framework, we generate keywords by examining the coefficients $\mathbf{w}$ of the linear model in 1. Note that the absolute value of a coefficient $w_k$ is proportional to the rate of changes in $y$ (the dependent variable) as $x_k$ (the input feature value) changes. If the absolute value of $w_k$ becomes larger, $x_k$ will have a more significant effect on $y$, which means that $x_k$ is a more discriminative feature. Therefore, we propose to generate the keywords, $K$, whose corresponding coefficients are of larger absolute values.

**Keywords Filtering and Expansion** In the case when the number of training samples is small, the coefficients $\{w_k\}$ may overfit the training data, because some common words in the documents of one class may never appear in the documents of another class by coincidence. To prevent the learning model from overfitting, we propose to perform filtering to remove some empirically common words, like "to", "they", etc. Moreover, to make queries be informative, we propose to perform keywords expansion based on the target domain labeled data. Following the automatic thesaurus generation process (Manning, Raghavan, and Schütze 2008), we compute a co-occurrence thesaurus based on the word-word similarity. Given a word-document matrix $A$, where each cell $A_{t,d}$ represents the term frequency-inverse document frequency (tf-idf) for the word $t$ and the document $d$, the similarity is defined as

$$C = A \times A^T,$$

where $C_{u,v}$ is a similarity score between the words $u$ and $v$, with larger number being better. With the set of keywords $K$ generated in the previous section, we can find the most related words via the similarity matrix $C$ to expand queries. For example, the keyword "church" could be expanded to a query ["church", "religious"]. The intuition behind this keywords expansion is that because we obtain $A$ based on the labeled training data, the queries, which are formed by the expanded keywords, are more likely to retrieve task-relevant documents that are helpful for the target classification task. Finally, by expanding each of the keywords into a query, we construct a set of queries $Q$.

## Learn from Auxiliary Data

Using the queries constructed in the previous section, we can query an open knowledge space for auxiliary data. Since searching is not the focus of our work, we omit the details here. We assume that the retrieved auxiliary data is much relevant to the queries, yet contains some noise. In order to adopt the helpful information and at the same time filter out the noise, we consider two factors when learning the model: 1) *similar* instances are more likely to be of similar labels, and 2) $f_i$ should be discriminative on each labeled training data in the target domain. In the following sections, we first discuss how to measure the similarity $I_{ij}$ between two instances $i$ and $j$. And we then introduce a graph-based data-dependent prior for optimization in detail. Finally, we discuss the model robustness.

**Definition of Similarity** $I_{ij}$ To measure the similarity of two documents, we propose to consider the semantic of the documents and the frequencies of their words. Following the work in (Hofmann 2000), to measure the similarity between two documents, we derive a Fisher kernel function (Jaakkola, Haussler, and others 1999) from the Probabilistic Latent Semantic Analysis (PLSA) model (Hofmann 1999). Given a document $d_i$, and a collection of the words $\{c_n\}$, we define the log-probability of $d_i$ by the probability of all the word occurrences in $d_i$, which is normalized by the document length

$$l(d_i) = \sum_n [\hat{P}(c_n \mid d_i) log \sum_k P(c_n \mid z_k) P(z_k \mid d_i)], \quad (2)$$

where $z_k$ is the latent features, $\hat{P}(c_n \mid d_i) = \frac{COUNT(d_i, c_n)}{\sum_m COUNT(d_i, c_m)}$. Note that by defining $l(d_i)$ in (2), $l(d_i)$ is directly correlated to the Kullback-Leibler divergence between the empirical distribution $\hat{P}(c_n \mid d_i)$ and the distribution derived from PLSA.

In order to derive the Fisher Kernel, we compute the Fisher information and Fisher scores. By the definition, the Fisher score $u(d_i; \theta)$ is set to be the gradient of $l(d_i)$ with respect to $\theta$. For simplicity, we make the same assumption as in (Hofmann 2000) that the Fisher information matrix approximates the identity matrix. Above all, the Fisher Kernel of two documents $d_i$ and $d_j$ with respect to a set of parameters $\theta$ is given by

$$\mathcal{K}(d_i, d_j) = \langle u(d_i; \theta), u(d_j; \theta) \rangle, \quad (3)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product operator. Note that we have two types of parameters, i.e. $\{P(z_k)\}$'s and $\{P(c_n \mid z_k)\}$'s. Due to the limits of spaces, we omit the detailed derivation of the gradients, and present the results. The similarity measure with respect to the parameters $\{P(z_k)\}$'s is given by

$$\mathcal{K}_1(d_i, d_j) = \sum_k \frac{P(z_k \mid d_i) P(z_k \mid d_j)}{P(z_k)}. \quad (4)$$

And the similarity measure with respect to the parameters $\{P(c_n \mid z_k)\}$'s is given by

$$\mathcal{K}_2(d_i, d_j) = \sum_n [\hat{P}(c_n \mid d_i)\hat{P}(c_n \mid d_j) \sum_k \frac{P(z_k \mid d_i, c_n)P(z_k \mid d_j, c_n)}{P(c_n \mid z_k)}],$$

where $P(z_k \mid d_i)$, $P(z_k)$, $P(z_k \mid d_i, c_n)$ and $P(c_n \mid z_k)$ are obtained from the estimation of PLSA in (Hofmann 1999).

The $\mathcal{K}_1$ kernel computes a "semantic" overlap between the two documents, while the $\mathcal{K}_2$ kernel handles the empirical word distributions of the two documents. We sum the outputs of the both measures to produce an overall similarity $I_{ij}$.

**Graph-based Data-dependent Prior** We consider each document as a vertex to form an undirected graph $(V, E)$. Each edge of the graph, connecting vertices $i$ and $j$, is given a weight $I_{ij} = I_{ji} \geq 0$, which represents the similarity between the documents $i$ and $j$. As we have discussed this similarity $I_{ij}$ in the previous section, we propose to measure how well the learned $f_{(\cdot)}$ captures the clustering property of the graph by the following quantity:

$$\sum_i \sum_j I_{ij}(f_i - f_j)^2, \tag{5}$$

where larger value indicates better $f_{(\cdot)}$. This technique is also known as the graph-based data-dependent prior (Krishnapuram et al. 2004). Note that although (5) is a measure of effectiveness for the discriminative function $f_{(\cdot)}$, the above quantity is independent of the class labels. This is desirable to us because in our framework, while learning the discrimination function using target domain labeled data, we also retrieve numerous unlabeled and relevant instances $D_U$ from an open knowledge space as auxiliary data to help the classification tasks.

Above all, in addition to the logistic function as described in logistic regression (Menard 2001), we propose to augment the log-likelihood with the prior in (5) as follows,

$$\mathcal{L}(\mathbf{w}) = \sum_{i=1}^{n} g(y_i f_i) + \frac{\lambda}{2} \sum_{i=1}^{n} \sum_{j=1}^{m+n} I_{ij}(f_i - f_j)^2, \tag{6}$$

where $I_{ij}$ represents the non-negative measure of similarity between instances $\mathbf{x_i}$ and $\mathbf{x_j}$ as described in the previous section, and $\lambda$ is the trade-off parameter that controls the effect of unlabeled auxiliary data. This parameter is task-dependent. Given a particular task, a simple way to search the optimal value of $\lambda$ is to evaluate the model to be learned on a small validation subset of the target task. Since this is a common practice for parameter selection, we do not discuss the details in this paper.

Note that in (6), we do not care the similarity between the retrieved auxiliary data, because it does not affect the specific target task. To estimate $\mathbf{w}$, we maximize the likelihood in (6). The detailed procedures for optimization are discussed in the following section.

**Learn the Regression Coefficients w** As we have defined above, $f$ is a linear function of $\mathbf{x}$, and $f_i = f(x_i) = \mathbf{w}^T \mathbf{x_i}$. We adopt gradient descent over (6) to update the weight vector $w$. The gradient of the log-likelihood with respect to the $k^{th}$ weight vector $w_k$ can be written as

$$\frac{\partial L}{\partial w_k} = \sum_{i=1}^{n} (y_i x_{ik} g(y_i f_i)) + \sum_{i=1}^{n} \sum_{j=1}^{m+n} D_{ijk}, \tag{7}$$

where

$$D_{ijk} = \lambda I_{ij}(f_i x_{ik} + f_j x_{jk} - f_i x_{jk} - f_j x_{ik}).$$

Moreover, the second order derivative of (6) is:

$$\frac{\partial L}{\partial w_k \partial w_k} = \sum_{i=1}^{n} (x_{ik}^2 g(y_i f_i)) + \lambda \sum_{i=1}^{n} \sum_{j=1}^{m+n} I_{ij}(x_{ik} - x_{jk})^2, \tag{8}$$

which is non-negative for any inputs.

Above all, we have the following weight update rule:

$$w_k^{(t+1)} = w_k^t + \epsilon \frac{\partial L}{\partial w_k} \tag{9}$$

where $\epsilon$ is the empirical learning rate, which is non-negative.

**Robustness to Noise**

**Proposition 1.** *The noisy auxiliary knowledge $\mathbf{x_{noise}} \in D_U$ will not harm the discrimination of the classifier in (6).*

*Proof.* We define the noisy knowledge $\mathbf{x_{noise}}$ that contains documents which either have little semantic overlap, or have large difference in terms of empirical word distributions with the target domain labeled data $D_L$. Following the definition of teh similarity matrix $I$, we have $I_{(i,noise)} \approx 0$, where $i \in \{1, \ldots, n\}$. Therefore the noisy instances do not affect the gradient in (7). This completes the proof. □

In an extreme case when all the retrieved documents are unrelated to the target classification task, the SFTL framework is reduced to learn a predictive model only with the target domain labeled data. From this point of view, the framework is robust: the irrelevant documents retrieved from an open knowledge space do not lead to a worse result.

## Validate on Target Data

We validate the model to be learned on a validation set $V = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ of the target domain. With $\mathbf{w}$ obtained from training, the performance is measured by

$$p = \sum_{i=1}^{n} \mathbf{w}^T x_i y_i, \tag{10}$$

where $(x_i, y_i) \in V$. Ideally, the performance keeps rising when new auxiliary unlabeled data are retrieved and added to the training process. We propose to set an empirical bound, where the change of performance approaches an infinite small number.

At this point, we have discussed an iteration of the SFTL framework. In summary, as presented in Algorithm 1, in an iteration, we query for auxiliary data based on the coefficients $\mathbf{w}$ of the current model, and then update the model with the retrieved auxiliary data $D_U$. The above query-update cycle repeats for several iterations until an empirical validation error bound is reached.

**Algorithm 1** Source Free Transfer Learning.

**Input**: labeled training set $D_L$, labeled validation set $V$, $search\ engine$: $S(Q)$, where $Q$ represents the queries, $bound$ of performance change on the validation set.

**Initialize:** Initialize coefficient $w_k : w_k \leftarrow \frac{1}{Z}$, the set of auxiliary data $D_U \leftarrow \emptyset$

**while** performance gain $(p_i - p_{i-1}) \leq bound$ **do**

**Step 1:** Learn the coefficients **w** by minimizing Eq. 6

**Step 2:** Calculate the performance $p_i$ on $V$ by Eq. 10

**Step 3:** Obtain the queries $Q$ as described in Section *Locate the Helpful Auxiliary Data*

**Step 4:** Retrieve the auxiliary data from an open knowledge space by $S(Q)$ and add them to $D_U$

**end while**

**Output:** The coefficients **w**.

# Experiments

## Experimental Settings

There are two main procedures in our proposed SFTL framework. The first is to query for the helpful knowledge, and the second is to learn a classifier based on the auxiliary data. To retrieve helpful knowledge for experiments, we feed queries to a commonly used search engine, such as Ask.com[1]. Empirically, given queries, the search engine satisfies our assumptions on retrieved documents. And to train a classifier, we follow the method as described in (6).

We perform text classification tasks on two datasets: the 20Newsgroups dataset[2] (**20NG**) and a Google snippets dataset (**GOOGLE**) (Phan, Nguyen, and Horiguchi 2008).

## Baselines

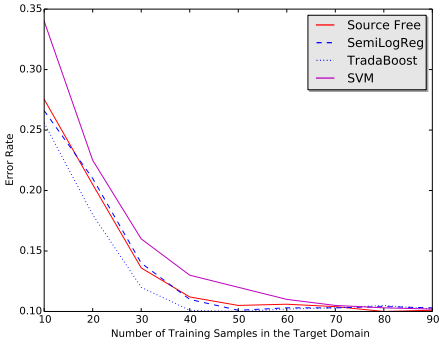
We compare our framework with the following methods for text classification:

- **SVM.** We use the Linear support vector machine (Vapnik 1999) as a supervised learning baseline and demonstrate the improvements of the SFTL framework over it.
- **SemiLogReg.** The unlabeled data are obtained from the original labeled training data set by omitting the labels of the instances and these sampled instances are not used for the labeled training data for the semi-supervised learning. Therefore, the unlabeled data are considered to be very helpful to the classification. The performance of the semi-supervised method SemiLogReg serves as an *empirical upper bound* for SFTL.
- **TrAdaBoost.** TrAdaBoost (Dai et al. 2007) is a boosting based transfer learning framework for text classification. The auxiliary data are hand-selected from the original labeled training data set to ensure the helpfulness for the target domain tasks. Our SFTL framework aims to achieve similar accuracy to TrAdaBoost, yet without any pre-defined auxiliary data.

[1]http://www.ask.com

[2]http://people.csail.mit.edu/jrennie/20Newsgroups

Figure 1: Change the number of training samples. The number of auxiliary samples (if any) is limited to 50.




## Change the Number of Training Samples

We first investigate the effectiveness of the afore-mentioned methods in handling the lack of training samples. The experiments are conducted on the **20NG** dataset. In the experiments, we varied the number of training samples from 10 to 90 for each of the tasks. The number of auxiliary samples is limited to 50 for the semi-supervised learning method (SemiLogReg), transfer learning method (TrAdaBoost) and our SFTL. For both SemiLogReg and TrAdaBoost, the 50 auxiliary samples are from the same two sub-groups as the training data in the **20NG** dataset. In other words, those auxiliary samples for the baselines are *very relevant* to the target task and are guaranteed to help in the target domain tasks. For SFTL, we use the top 50 automatically retrieved samples. The performance comparison is shown in Figure 1. Comparing to the supervised learning methods without auxiliary data, such as SVM, our SFTL performs significantly better when the number of training samples is small (less than 20 for each task on the **20NG** dataset). On the other hand, our SFTL achieves similar error rates with the SemiLogReg and the TrAdaBoost, which are considered to be the optimal models because they use the best quality auxiliary data that come from the same distribution as the training data in the target domain.

## Performance Comparison

We compare performances on both **20NG** and **GOOGLE** datasets. In our experiments, we randomly choose only 10 target domain training samples for each task. For both SemiLogReg and TrAdaBoost methods, we choose about 80 samples from the original training datasets as the auxiliary data for each task. Our SFTL method aims to achieve similar error rates with SemiLogReg or TrAdaBoost methods, yet with no pre-defined auxiliary data. The comparison is shown in Figure 2. Because when the number of training data is only 10, supervised learning methods, such as SVM, could not learn a robust model. With the samples from original training data set as auxiliary data, both semi-supervised learning methods, such as SemiLogReg, and the transfer learning methods, such as the TrAdaBoost, approach the upper bound of accuracy. Our SFTL is significant better than

Figure 2: Performance comparison on two datasets.

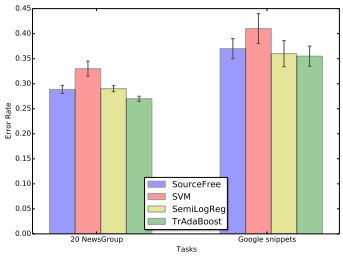
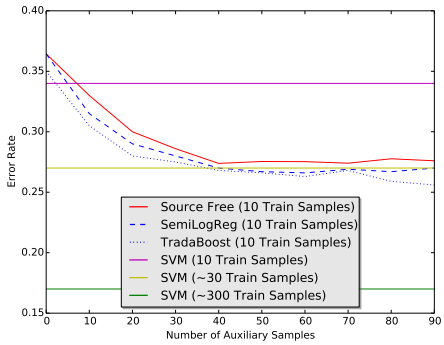



Figure 3: Performance gain with the help of increasing auxiliary data.




SVM, and achieves similar results as the empirical upper bound, i.e., SemiLogReg and TrAdaBoost.

## Help from Auxiliary Data

We explore the performance gain with the continuous adding of auxiliary data. The experiments are conducted on the **20NG** dataset. In Figure 3, we present the error rate changes upon the increasing of auxiliary data. First, our SFTL achieves similar error rate curves with the semi-supervised method and the transfer learning method, though the SFTL model does not ask for well prepared auxiliary data. This implies that our method is capable to target the helpful data from the open knowledge space. Second, we notice that the performance of SFTL is improved significantly when the first few auxiliary samples are included. When the number of the auxiliary samples exceeds certain value, the accuracy of SFTL stops growing. Comparing to the two ideal cases, where both SemiLogReg and TrAdaBoost use well prepared auxiliary data, we conclude that SFTL almost reaches the upper bound of accuracy. Third, we increase the number of training data for the supervised learning. When the number of training samples is increased up to 30, the supervised learning methods, such as SVM, have the similar performance with SFTL, which uses only 10 training samples.

## Quality of Queries

When training a Source Free model, we iteratively generate queries to search for the auxiliary data, based on the learned Source Free model. The quality of the search queries is essential to the quality of retrieved auxiliary data, and subsequently affects the accuracy of the Source Free model.

Table 1: Samples of Queries for a 20Newsgroups Task.

| Tasks | Queries | Acc SVM | Acc SourceFree |
|---|---|---|---|
| alt.atheism VS comp.graphics (Iteration 1) | [people, help], [see, art], [sort, hand], [support, stop], [csd, info] | 63.56 % | 68.14 % |
| alt.atheism VS comp.graphics (Iteration 2) | [people, help], [see, art], [sort, hand], [religious, church], [issue, political] | 63.56 % | 70.52 % |
| ..... | | | |
| alt.atheism VS comp.graphics (Iteration 5) | [see, art], [religious, church], [issue, political], [support, stop], [graphics, look] | 63.56 % | 72.04 % |

In Table 1, we show the queries for one of the **20NG** tasks at some of the iterations of the query-update cycle. For each iteration, we augment the number of auxiliary samples by about 20. First, we notice that as the iteration goes, the queries make more sense, which corresponds to the increasing of classification accuracy. Second, recall that in the Section *Query Generation*, we choose the keywords corresponding to the $w$ with large absolute values. Intuitively we are looking for those words that are strong indicators of class labels, although the semantic meanings of the class are unknown to the model. This is effective. For example in Table 1, in order to distinguish "alt.atheism"-related documents with "comp.graphics"-related documents, the chosen words like "religious", "graphics" etc. would be good indicators of the class label. And the queries, like "[religious, church]", are expected to get helpful auxiliary samples. By retrieving the documents, which are related to those discriminative queries, we largely improved the classification accuracy comparing to the SVM baseline.

## Conclusion

We have proposed a Source Free Transfer Learning (SFTL) framework, which automatically selects helpful auxiliary data from an open knowledge space. In the SFTL framework, the auxiliary data retrieval and the model training are iteratively guided by each other. This automation of auxiliary data selection for any target tasks is a breakthrough of the current transfer learning methods, whose appropriate auxiliary data are usually selected by human instinct and experience. Our experiments on two datasets, i.e. 20NewsGroup and Google Search Snippets, show that SFTL can achieve the performances of other transfer learning or semi-supervised learning methods with dedicated selection of auxiliary data. With less human intervention yet the same performance, the SFTL framework is therefore more practical in the real world classification problems. In the future, we plan to extend the SFTL framework to the classification problems other than text categorization.

## Acknowledgments

We thank the support of China National 973 project 2014CB340304, National Natural Science Foundation of China 61309011 and Hong Kong RGC Projects 621013, 620812 and 621211.

## References

Dai, W.; Yang, Q.; Xue, G.; and Yu, Y. 2007. Boosting for transfer learning. In *Proceedings of the 24th international conference on Machine learning*, 193–200. ACM.

Egozi, O.; Markovitch, S.; and Gabrilovich, E. 2011. Concept-based information retrieval using explicit semantic analysis. *ACM Transactions on Information Systems (TOIS)* 29(2):8.

Gabrilovich, E., and Markovitch, S. 2005. Feature generation for text categorization using world knowledge. In *IJCAI*, volume 5, 1048–1053.

Gabrilovich, E., and Markovitch, S. 2006. Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge. In *AAAI*, volume 6, 1301–1306.

Hofmann, T. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 50–57. ACM.

Hofmann, T. 2000. Learning the similarity of documents: An information-geometric approach to document retrieval and categorization.

Jaakkola, T.; Haussler, D.; et al. 1999. Exploiting generative models in discriminative classifiers. *Advances in neural information processing systems* 487–493.

Krishnapuram, B.; Williams, D.; Xue, Y.; Hartemink, A. J.; Carin, L.; and Figueiredo, M. A. T. 2004. On semi-supervised classification. In *Advances in Neural Information Processing Systems*.

Lu, Z.; Pan, W.; Xiang, E. W.; Yang, Q.; Zhao, L.; and Zhong, E. 2013. Selective transfer learning for cross domain recommendation. In *Proceedings of the 13th SIAM International Conference on Data Mining*, 641–649.

Manning, C. D.; Raghavan, P.; and Schütze, H. 2008. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge.

Menard, S. 2001. *Applied logistic regression analysis*, volume 106. Sage Publications, Incorporated.

Pan, S. J., and Yang, Q. 2010. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* 22(10):1345–1359.

Phan, X. H.; Nguyen, M. L.; and Horiguchi, S. 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In Huai, J.; Chen, R.; Hon, H.-W.; Liu, Y.; Ma, W.-Y.; Tomkins, A.; and Zhang, X., eds., *WWW*, 91–100. ACM.

Raina, R.; Battle, A.; Lee, H.; Packer, B.; and Ng, A. Y. 2007. Self-taught learning: transfer learning from unlabeled data. In *Proceedings of the 24th international conference on Machine learning*, 759–766. ACM.

Settles, B. 2009. Active learning literature survey. Computer sciences technical report, University of Wisconsin–Madison.

Vapnik, V. 1999. *The nature of statistical learning theory*. springer.

Wang, P.; Hu, J.; Zeng, H.-J.; Chen, L.; and Chen, Z. 2007. Improving text classification by using encyclopedia knowledge. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, 332–341. IEEE.

Xiang, E. W.; Pan, S. J.; Pan, W.; Su, J.; and Yang, Q. 2011. Source-selection-free transfer learning. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Three*, 2355–2360. AAAI Press.

Zhu, X. 2006. Semi-supervised learning literature survey. *Computer Science, University of Wisconsin-Madison* 2:3.