Abstract

We present a novel approach for domain adaptation based on feature grouping and re-weighting. Our algorithm operates by creating an ensemble of multiple classifiers, where each classifier is trained on one particular feature group. Faced with the distribution change involved in domain change, different feature groups exhibit different cross-domain prediction abilities. Herein, ensemble models provide us the flexibility of tuning the weights of corresponding classifiers in order to adapt to the new domain. Our approach is supported by a solid theoretical analysis based on the expressiveness of ensemble classifiers, which allows trading-off errors across source and target domains. Moreover, experimental results on sentiment classification and spam detection show that our approach not only outperforms the baseline method, but is also superior to other state-of-the-art methods.

1 Introduction

Discriminative learning models work effectively when the training and testing examples are drawn from the same distribution. However, in several real-world applications, it is often highly desirable to train a classifier from one source domain, and apply it to a similar but different target domain, where the annotated data is unavailable or expensive to create. One example of this scenario is to learn a text categorizer from a large collection of labeled newswire articles, but use it to process regular Web documents. In this domain adaptation setting, the goal is to leverage the data available in the source domain to improve the accuracy of the model when testing on target domain examples.

As observed by previous studies, unfortunately, naively applying classifiers to a different domain often leads to considerable performance degradation [Daumé III, 2007; Jiang and Zhai, 2007]. Consequently, a number of approaches have been proposed recently to address the problem of domain adaptation. Some methods focus on re-weighting training instances from different domains [Jiang and Zhai, 2007; Bickel et al., 2009], while others modify the feature space to capture domain-specific and domain-invariant aspects [Blitzer et al., 2006; Daumé III, 2007; Finkel and Manning, 2009; Jiang and Zhai, 2006]. Despite the fact that these methods adapt seemingly very different strategies, the shared rationale behind is to bring the empirical source distribution closer to the target domain, and thus increase the accuracy of the classifier when evaluated on the target domain data.

In this paper, we present a novel ensemble-based approach for domain adaptation, based on feature re-weighting. Our method builds on the observation that when moving from the source domain to the target domain, different features may have different levels of distributional change. For instance, in email spam detection, the distribution of content-based features changes heavily over time as it is very easy for spammers to revise the subject and body of a spam message. In contrast, the distribution of features based on user-preference or sender-IP does not change as much. This phenomenon can be easily observed empirically, which we demonstrate in Sec. 4.2. Intuitively, a classifier trained on distributionally stable features is likely to behave more consistently than a classifier based on an unstable feature group. Therefore, in addition to the expressiveness of its feature space in predicting the label, the importance of each classifier should also depend on the distributional stability of its features. Given a set of feature groups that capture this notion, where the grouping can be decided by domain knowledge or statistics derived from unlabeled data, we first train individual classifiers separately using only the corresponding group of features. The final model is a weighted ensemble of individual classifiers, where the weights are tuned based on the performance of the ensemble on a small amount of labeled target data. Compared to the existing approaches, our method is unique in that it considers the cross-domain behavior of different feature groups directly in terms of their classification accuracy. Afterwards, instead of creating an instance distribution close to the target domain, it adjusts the influence of features on the final classifier by tuning their weights.

Our approach is supported both by a solid theoretical analysis and a strong empirical validation. We present a generalization bound for the style of ensembles mentioned above based on the \(d_{H}\)-distance framework proposed by Ben-David et al. [2007]. We show that ensembles provide an additional degree of freedom in the form of tunable weights of individ-
The idea behind domain adaptation is to leverage groups: $X \subseteq \{0,1\}^p$ is sampled and let $\mathcal{Y} = \{-1,1\}$ be the output space. Suppose $D_S(x,y)$ and $D_T(x,y)$ are two distributions over $\mathcal{X} \times \mathcal{Y}$. Let $L_S$ and $L_T$ be labeled data samples from $D_S$ and $D_T$, respectively. In the setting of domain adaptation, we assume that $|L_S| \gg |L_T|$. Moreover, we assume that some unlabeled data from the target domain, $U_T \sim D_T(x)$, is available.

Let $\Delta : \mathbb{R} \times \mathbb{R} \to \{0,1\}$ be the standard zero-one loss function: $\Delta(z_1, z_2)$ is zero if $z_1 z_2 \geq 0$, one otherwise. The idea behind domain adaptation is to leverage $L_S$ and $U_T$, in addition to $L_T$, to learn a hypothesis $h_t : \mathcal{X} \to \mathcal{Y}$ with low expected risk on target, defined by $\Delta_{D_T}(h_t) = \mathbb{E}_{(x,y) \sim D_T}[\Delta(h_t(x), y)]$. Of course, $L_S$ can help only if $D_S$ and $D_T$ are not “too different”.

Although the feature space defined here is binary (i.e., $\mathcal{X} \subseteq \{0,1\}^p$), our analysis and technique, by no means, are restricted to discrete feature spaces. In the text applications considered in this paper, such a setting is very common; for example, the occurrences of unigrams or bigrams in the text are encoded as binary features. Also, the number of features, $p$, is typically much larger than the number of training examples.

Last, we assume that we have a decomposition of feature vector $X$ into $r + 1$ (possibly overlapping) feature vectors or groups: $X^1, X^2, \ldots, X^r \subseteq X$ with $X^0 = X$; that is, $X^i$ is a subset of features contained in $X$. Essentially, our approach considers ensembles of classifiers where each classifier is trained on one particular feature group. The nature and purpose of these groupings would become clear in section 3.

### 3 Feature Ensembles for Domain Adaptation

In this section, we propose FEAD, a Feature Ensembles approach for Domain Adaptation, which learns a classifier for the target domain in the form of an ensemble of feature group classifiers. The design of FEAD is motivated by a VC-style generalization bound analysis for ensemble classifiers in the context of domain adaptation. Compared to existing domain adaptation methods, FEAD provides an additional degree of freedom to adjust the trade-off between the generalization error and domain distribution change of individual classifiers trained on the corresponding feature groups. Although the selection of the underlying learning algorithm is not restricted, we give an interesting insight into FEAD via a concrete instantiation of our ensemble approach when using logistic regression.

#### 3.1 Algorithm

Before introducing our algorithm, we first give some notations that are useful throughout the whole section.

Let $\Delta^c : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$ be a convex loss function upper bounding $\Delta$. For a set of labeled examples $L$, $\Delta_{L}^c(f) = \frac{1}{|L|} \sum_{(x,y) \in L} \Delta^c(f(x), y)$ denotes the empirical loss of the hypothesis $f$ on $L$. Also, let $\Delta_D(f, g) = E_{(x,y) \sim D} [\Delta(f(x), g(x))]$ be the expected zero-one loss between hypotheses $f$ and $g$ on a distribution $D$. Let $H$ be the hypothesis space being considered. For example, $H$ can be the space of all linear classifiers.

For a feature group $X^i \subseteq X$, let $D_S(x,y) = D_S(X^i = x^i, y)$ be the marginal source distribution over $(x^i, y)$ where $Z_i$ is chosen to make $D_S$ a valid distribution; let $H_i$ be the space of hypotheses defined over $X^i$ (clearly $H_0 = H$ and for several popular, including linear, hypothesis spaces, $H_i \subseteq H$); and let $h_i^S = \arg\min_{h \in H_i} \Delta_{L_S}^c(h)$ be a classifier in $H_i$ minimizing the loss on $L_S$ with $\Delta_{L_S}^c(h^S) = c_i^S$.

We express the final classifier as an ensemble of classifiers given by $h(\overline{\pi}) = \sum_{i=0}^{r} \alpha_i h_i^S$, where $\overline{\pi} = (\alpha_0, \ldots, \alpha_r) \geq 0$. $\overline{\pi}$ is learned by minimizing the empirical target error. The whole learning procedure is delineated in Algorithm 1.

### Algorithm 1 Algorithm for Feature Ensembles for Domain Adaptation (FEAD)

1: **Given**: Data: $L_S$ and $L_T$; feature groups: $X_0, \ldots, X_r$; hypotheses spaces: $H_0, \ldots, H_r$; convex loss function: $\Delta^c$
2: **for** $i = 0$ to $r$ **do**
3: \hspace{1em} learn: $h_i^S \leftarrow \arg\min_{h \in H_i} \Delta_{L_S}^c(h)$
4: **end for**
5: $\overline{\pi} \leftarrow \arg\min_{\pi \succeq 0} \sum_{(x,y) \in L_T} \Delta(\sum_i \alpha_i h_i^S(x), y)$
6: **return** $\pi = \sum_{i=0}^{r} \alpha_i h_i^S$

---

1We slightly modify the notation to suit our analysis.
\textbf{Definition 1 (d_{H}-distance) [Ben-David et al., 2007].} Let \(H\) be a set of hypotheses, mapping \(X\) to \(\mathbb{R}\). The \(d_{H}\)-distance between two distributions \(Q_{1}\) and \(Q_{2}\) over \(X\) is defined as
\[
d_{H}(Q_{1}, Q_{2}) = \max_{h, h' \in H} |\Delta_{Q_{1}}(h, h') - \Delta_{Q_{2}}(h, h')|.
\]

Intuitively, \(d_{H}(Q_{1}, Q_{2})\) bounds how closely we can predict the loss between two hypotheses on \(Q_{2}\) if we know their loss on \(Q_{1}\). We first present two propositions regarding \(d_{H}\).

\textbf{Proposition 1 (Convexity).} For any distributions \(D, D_{1}\), and \(D_{2}\) with \(D_{\lambda} = \lambda D_{1} + (1 - \lambda)D_{2}, \lambda \geq 0\), we have
\[
d_{H}(D, D_{\lambda}) \leq \lambda d_{H}(D, D_{1}) + (1 - \lambda) d_{H}(D, D_{2}).
\]

\textbf{Proof.} Using \(D' = \sum_{i} \alpha_{i} D_{S,i}^{i}\) in Proposition 2 and then using Proposition 1 yields
\[
d_{H}(D_{S}, D_{T}) \leq \sum_{i} \alpha_{i} (d_{H}(D_{S}, D_{S,i}) + d_{H}(D_{S,i}^{i}, D_{T}))
\]
Substituting the above in the second step of the proof of Theorem 3 in [Ben-David et al., 2010], along with the fact that \(\Delta \leq \Delta^{*}\) and the convexity of \(\Delta^{*}\) yields the result.

\textbf{Theorem 1.} For all \(h = \sum_{i} \alpha_{i} h_{i}^{S}\) with \(\alpha_{i} \geq 0 \forall i\) and \(\sum_{i} \alpha_{i} = 1\), the following bound holds with probability at least \(1 - \delta\) for \(\delta\) for \(\forall h \in [0, 1]\)
\[
\Delta_{D_{T}}(h) \leq (1 - \beta)(e^{*} + \sum_{i} \alpha_{i} (e_{i}^{S} + d_{H}(D_{S}, D_{S,i}^{i})) + \epsilon(p_{H}, \beta, \delta))
\]
\[
+ \beta \Delta_{L_{T}}(h)
\]
where \(\Delta_{L_{T}}(h)\) is the normalized marginal source distribution over \((X^{1}, y)\), \(e^{*} = \arg\min_{h' \in H} (\Delta_{D_{T}}(h') + \Delta_{D_{S}}(h'))\), and \(\epsilon(p_{H}, \beta, \delta) = 2\sqrt{\frac{\sigma_{\epsilon}^{2}}{\eta} + \frac{(1 - \beta)\beta}{4\eta} \sqrt{\frac{2p_{H}\log(2(m + 1)) + 2\log(\frac{1}{\delta})}{m}}\}
\]
where \(m = |L_{S}| + |L_{T}|, \eta = \frac{|L_{T}|}{m}\), and \(p_{H}\) is the VC Dimension of \(H\).

\textbf{Proof.} Using \(D' = \sum_{i} \alpha_{i} D_{S,i}^{i}\) in Proposition 2 and then using Proposition 1 yields

This analysis naturally motivates an approach that is even simpler than FED — picking a single “best” feature group classifier (BFG) for the target domain, where the notion of “best” can be decided by the user. Although the ensemble approach is clearly more expressive than the BFG approach, as the former’s output space is a superset of the latter, it is valid to ask whether the additional expressiveness indeed earns FED a lower error than BFG. In general, there is no reason to expect that the expected target error or its upper bound given by Theorem 1 is minimized when only one of the classifiers is picked (in this case the result will be the same as the BFG approach.).

We give a simple artificial example which elucidates this point. Consider the following setting: We have two features \((x_{1}, x_{2}) \in \mathbb{R}^{2}\) and a label \(y \in \{-1, 1\}\), which we wish to predict. The features are conditionally independent given the label i.e. \(P(x_{1}, x_{2}, y = 1) = P(x_{1}, y)P(x_{2}, y)\) and \(P(x_{1}, x_{2}, y = -1) = P(x_{1}, y)P(x_{2}, y)\). Moreover, we are given the following description of the distribution: (1) \(P(y = 1) = P(y = 0) = \frac{1}{2}\); (2) \(D_{S}(x_{1}) = D_{S}(x_{2}) = \mathcal{N}(y, 1, 1)\); (3) \(D_{T}(x_{1}) = \mathcal{N}(y, 2, 1)\); and \(D_{T}(x_{2}) = \mathcal{N}(y, 2, 1)\), where \(D_{S}\) and \(D_{T}\) describe distributions on source and target, respectively, and \(\mathcal{N}(\mu, \sigma^{2})\) represents a Gaussian distribution with mean \(\mu\) and variance \(\sigma^{2}\).

It is easy to see that the asymptotically optimal classifiers for source domain are given by \(x_{1}\) and \(x_{1} + x_{2}\) when considering feature groups \(x_{1}\) and \((x_{1}, x_{2})\), respectively; similarly, the bayes optimal classifier for the target domain and the feature group \((x_{1}, x_{2})\) is given by \(x_{1} + \frac{1}{2}x_{2}\).
If we consider two feature groups: \((x_1, x_2)\) and \(x_1\), the BFG approach would, asymptotically, dictate picking the best of \(x_1 + x_2\) and \(x_1\) for the target domain; clearly, both of these classifiers give higher error on the target domain than the optimal classifier: \(x_1 + \frac{1}{2} x_2\). However, the WPoE approach would be able to produce the optimal classifier \(x_1 + \frac{1}{2} x_2\) as \(x_1 + \frac{1}{2} x_2 = \frac{1}{2}(x_1 + x_2) + \frac{1}{2}(x_1)\). This example illustrates how additional expressiveness of feature group ensembles can help outperform the BFG approach.

Overall, our theoretical analysis shows that FEAD is a flexible algorithm that directly tackles domain change — it allows for minimizing error on \(L_T\) while keeping the error on \(L_S\) bounded. Although the focus of this paper is not on techniques for deciding the “best” feature groups, we demonstrate in Sec. 4 that by adapting simple heuristics to determine the feature grouping, based on statistical measures or domain knowledge, FEAD can yield strong empirical results in a number of real-world domain adaptation tasks.

### 3.3 FEAD as Product of Experts

When considering a specific setting where \(H\) is the space of all linear classifiers and \(\Delta x\) is the log-loss given by \(\Delta x(h(x), y) = \log(1 + \exp(-y h(x)))\), our analysis in this section provides an interesting probabilistic perspective of FEAD. Let \(w^S_k\) be a weight vector learned for feature group \(H^i\) via logistic regression. Given a feature vector \(x\), the probability of output \(y\), as per \(w^S_k\), is given by

\[
Pr(y|x, w^S_k) = \frac{\exp(y w^S_k \cdot x)}{1 + \exp(y w^S_k \cdot x)}
\]

which implies that the odds ratio of \(y = 1\) and \(y = -1\) for this instance is given by

\[
Odds(x|w^S_k) = \frac{Pr(y = 1|x, w^S_k)}{Pr(y = -1|x, w^S_k)} = \exp(2w^S_k \cdot x)
\]

FEAD expresses the final linear classifier for the target domain, \(w_t\), as \(\sum \alpha_i w^S_i\), which is equivalent to expressing the odds for any instance as a weighted product of odds given by individual classifiers. In this sense, our approach is similar to the Product-of-Experts framework proposed by Hinton [1999], wherein classifiers based on feature groups can be thought of as experts.

### 4 Experiments

We evaluate our approach empirically on two domain adaptation tasks — sentiment analysis and email spam detection — and compare with other state-of-the-art approaches.

#### 4.1 Sentiment Classification

Sentiment analysis is the task of classifying textual reviews into positive or negative based on the expressed “sentiment” [Pang et al., 2002; Blitzer et al., 2007]. Reviews for different categories of products tend to be different in terms of vocabulary and writing style. Consequently, a sentiment classifier trained on one domain (e.g., books) suffers from considerable performance degradation when applied to another domain (e.g., electronics), thus necessitating a need for domain adaption methods.

In this set of experiments, we use the benchmark dataset released by Blitzer et al. [2007], which consists of reviews of four different product categories: books, DVDs, electronics, and kitchen appliances, collected from Amazon.com. Along with the IMDB movie reviews released by Pang et al. [2002], we generate all 20 possible source–target pairs for the empirical evaluation. For all the methods we tested, we follow the same data split and experimental procedure. Each domain contains 2000 instances and is randomly split into a training/testing set of 1600/400 examples. Furthermore, we randomly select 160 instances from the training set of each domain to serve as a validation set. For each source–target adaptation setting, we use the training set of source as \(L_S\) (labeled source training data), the validation set of target as \(L_T\) (labeled target training data), and the remaining training examples of target as \(U_T\) (unlabeled target data). Features extracted from each review are the unigrams and bigrams occurring at least twice in \(L_S \cup L_T \cup U_T\), encoded as binary features. We perform 30 such random splits for each setting and report the averaged accuracy when evaluating the model on the target testing data.

We compare four approaches – logistic regression (LR), multiview transfer learning (Multi-T), easy adapt (EA) and our feature ensemble approach (FEAD). As the baseline approach, LR is a model trained over all the features. We first find the best Gaussian smoothing prior by training the model on the source training set (\(L_S\)) and testing it on the validation set (\(L_T\)). The final model is learned on the entire labeled data (\(L_S \cup L_T\)) using the selected prior. Proposed recently by Blitzer et al. [2009], Multi-T is an enhanced version of structural correspondence learning (SCL) [Blitzer et al., 2007], which has been tested previously on the sentiment dataset. We conduct the experiments using the exact setting as reported in their paper. Another state-of-the-art method, EA [Daumé III, 2007] projects the source and target examples to a new expanded feature space first, and then trains the classifier on the new labeled examples in a way similar to the baseline LR approach. Finally, FEAD uses two groups of features. The first one is the trivial group that contains all the features — call this feature group total. For the second feature group, we aim to have features that lead to low \(d_H\)-distance and empirical error. We adapt a heuristic similar to the one used in [Blitzer et al., 2007] based on mutual information.

For each feature appearing in \(L_S\), we first compute its mutual information compared with the label. For a feature to be included in this group, its mutual information needs to be higher than at least half of the features in \(L_S\) and it needs to occur at least once in the target data (\(L_T \cup U_T\)). We call this feature group common. Notice that this by no means provides the best feature grouping for our approach. Instead, we demonstrate that even with simple groups of features selected following the principle suggested by our theoretical analysis, a significant improvement can be observed empirically.

Table 1 shows the averaged accuracy of different methods for each source–target adaptation setting. As can be seen

---

2We use the software package provided by the authors. Personal communication confirms that the results of Multi-T and SCL-MI are comparable.
clearly, FEAD yields the best results for most of the settings. Among them, FEAD performs statistically significantly better than both EA and Multi-T in 8 cases, and better than all competing approaches in 4 cases. Even when FEAD does not achieve the highest accuracy, the difference compared with the best performing method is not statistically significant, except for pairs M-E and M-K. Table 2 shows the normalized (i.e., weights for total and common feature groups sum to 1) values of the weight assigned to the common feature group. From the table, it can be inferred that the common feature group plays a significant role in the final classifier. Notice that the baseline (LR), when provided with some amount of labeled target data, is notoriously hard to defeat [Daumé III, 2007; Chang et al., 2010]; our experiments corroborate this. Overall, the experiments on the sentiment analysis empirically show that FEAD is a simple yet powerful algorithm for domain adaptation.

### 4.2 Email Spam Detection

Spam detection is a problem of classifying email messages to either spam or good. Compared with the typical binary classification setting, the main challenge of spam detection comes from its adversarial nature. As spammers often quickly react to a newly deployed filter by modifying the spam messages [Lowd and Meek, 2005b; 2005a], the sample distri-

| Setting | Algorithms | | | |
|---------|------------|------------|------------|
| Src-Tgt | LR | Multi-T | EA | FEAD |
| B-D     | 81.10 | 80.01† | 78.63† | 81.80 |
| B-E     | 78.88 | 78.82 | 79.38 | 79.34 |
| B-K     | **82.29** | 79.60† | 81.66 | 82.26 |
| B-M     | 80.23 | 77.91† | 79.25† | **80.70** |
| D-B     | 81.60† | 79.91† | 80.14† | **82.46** |
| D-E     | 81.27 | 81.09 | 80.34 | **81.54** |
| D-K     | **82.95** | 81.83† | 82.08† | 82.81 |
| D-M     | 82.53 | 79.50† | 81.53† | **82.53** |
| E-B     | 75.34 | 72.74† | **75.85** | 75.60 |
| E-D     | 75.85† | 75.86 | 74.78† | **76.76** |
| E-K     | 86.50† | 83.77† | 85.33† | **87.59** |
| E-M     | 72.60† | 70.86† | 72.63 | **73.54** |
| K-B     | 74.74† | 75.47 | 74.78 | **75.75** |
| K-D     | 75.93 | 76.97 | 75.21† | 76.88 |
| K-E     | 84.90 | 84.57 | 83.81† | **85.24** |
| K-M     | 72.38 | 71.02† | 70.45† | **72.62** |
| M-B     | 77.11† | 77.06† | 76.07† | **78.88** |
| M-D     | 77.76† | 77.94† | 76.20† | **79.52** |
| M-E     | 76.45† | **80.18** | 76.50† | 77.62† |
| M-K     | 76.72† | **79.41** | 76.48† | **77.59** |
| Avg.    | 78.86 | 78.23 | 78.06 | **79.55** |

Table 1: Results on sentiment classification of the baseline (LR), multiview transfer (Multi-T), easy adapt (EA), and our ensemble approach (FEAD). Numbers in bold-face font are $†$Numbers with $†$ are statistically significantly worse than the best performing method. Statistical significance is based on paired-t test with the $p$-value less than 0.05.

### 4.2 Email Spam Detection

Spam detection is a problem of classifying email messages to either spam or good. Compared with the typical binary classification setting, the main challenge of spam detection comes from its adversarial nature. As spammers often quickly react to a newly deployed filter by modifying the spam messages [Lowd and Meek, 2005b; 2005a], the sample distri-

**Table 2:** The weights assigned to frequent feature groups by FEAD in different sentiment classification settings.

- For book, DVD, Elec., Kitchen, Movie
- For Book, DVD, Elec., Kitchen, Movie

In order to show how different feature groups undergo different changes in their discriminative power, we compare the performance of two classifiers: one trained only on features from email-content (content features) and the other trained only on features based on sender information (sender features.) For each set of features, we used two different settings: random and time-shifted. The random setting is the control group that tests the scenario when the training and testing data come from the same distribution. In this setting, we train the classifiers on randomly chosen 765,000 instances, tune on randomly chosen 50,000, and test on the remaining 100,000. In contrast, the time-shifted setting simulates the domain adaptation scenario, where we train the classifiers on first (chronologically) 765,000 instances, tune on the next 50,000, and test on the final 100,000. Fig. 1 shows the comparisons of the content and sender based classifiers.
in these two settings, using the ROC analysis. A large performance drop in the time-shifted setting compared to the random setting can be explained by a drastic distribution or domain change. The performance degradation is more significant for content-features compared to sender-features, which empirically shows they undergo different distribution change.

In the experiments, we leverage this background knowledge to form the feature grouping used by FEAD. In addition to content and sender features, we have two other feature groups: user features and a trivial feature group that contains all the features. The content features are words in the subject and body of the email that occur at least three times in the source set. The sender features include the first 16 bits, the first 24 bits, and the entire IP address of the sender. Finally, the user features are simply the recipient ids.

We compare the performance of FEAD with two other approaches, logistic regression (LR) and easy adapt (EA)\(^3\). LR and EA do not take advantage of the background knowledge of feature grouping and are trained on all (i.e. content, sender, and user) features put together. The experimental settings are the same as described in the experiments on sentiment analysis (Sec. 4.1). To compare performance, we report the ROC curves and the normalized AUC values in a low false-positive rate (FPR) region, as the cost of losing good email is much higher than receiving spam. Normalized AUC for FPR ≤ \(\delta\) is defined as \(\frac{1}{2}\)AUC(\(\delta\)) where AUC(\(\delta\)) is the area under the curve for FPR between 0 and \(\delta\). As can be observed from Fig. 2, FEAD performs better than LR and EA consistently in this region, and has a much higher normalized AUC score. In comparison, although EA indeed performs better than the baseline LR, the improvement gain is relatively small and its ROC curve in fact crosses that of LR. This empirical result suggests that FEAD can effectively leverage the domain knowledge of the feature grouping, which is valuable in improving the model performance for domain adaptation.

5 Related Work

The problem of domain adaptation has been studied extensively from both the theoretical perspective and algorithmic sides. To understand formally the characteristics of the problem, Ben-David et al. [2007; 2010] proposed the \(d_H\)-distance framework, which was later extended to a wider variety of loss functions by Mansour et al. [2009]. Although the idea of feature grouping was never discussed previously, their work provides a solid foundation for the theoretical analysis of our ensemble approach.

A variety of algorithm-centric domain adaption methods have also been proposed. For example, EasyAdapt [Daumé III, 2007] uses feature-replication to implicitly capture the domain-invariant and domain-specific aspects of features. Blitzer et al. [2006] proposed Structural Correspondence Learning (SCL), which projects features to a latent space such that these latent features exhibit roughly the same distribution across source and target domains. Alternatively, Finkel

---

\(^3\)We attempted to include multiview transfer learning (Multi-T). Unfortunately, it does not seem to scale well to a large amount of training data, and failed to produce the final model due to memory issues, even after increasing the heap size to 20GB.
and Manning [2009] and Jiang and Zhai [2006] propose approaches based on imposing priors over individual features, where the priors can be decided directly based on the generalizability of the features. FEAD is related to these works as it massages the weights of different features in terms of their contribution to the final prediction function. However, unlike these approaches, the feature weights in FEAD are tweaked after creating the classifiers.

The use of feature grouping in FEAD is analogous to adaptation from multiple sources [Mansour et al., 2008; Ben-David et al., 2010], where feature groups can be thought of as different sources created artificially. However, adaptation from multiple sources typically assumes that the instances from different sources are independent, which is clearly not the case in our setting. Finally, we share the idea of using ensemble-like learning for domain adaptation with Dai et al. [2007], who use boosting for the same purpose. However, our approach considers ensembles over different feature groups, whereas they create ensembles by re-weighting instances.

6 Conclusion

In this paper, we introduced a technique that leverages the expressiveness of ensembles to adapt to distribution change between domains. Building on this notion, we presented a simple and easy-to-implement method based on re-weighting classifiers learned on different feature groups. Our approach provides the flexibility to re-adjust the influence of different feature groups on the target domain classifier based on their cross-domain distributional stability. This design is motivated by our theoretical analysis, and its empirical effectiveness is demonstrated through experiments on the benchmark sentiment analysis task and spam detection, where our approach outperforms several state-of-the-art methods.

Although in this work we showed that by using simple statistical measures or domain knowledge to generate the feature grouping, our approach can lead to significant improvement, it is nevertheless interesting to find automatically the best feature grouping for the task. We leave that as future work.

Acknowledgment We would like to thank John Blitzer for providing the Multiview-T software, Michael Gamon for preparing sentiment analysis data, Sumit Basu and Chris Meek for their comments and the anonymous reviewers for their useful suggestions. The work was partly supported by Army Research Laboratory (ARL) under agreement W911NF-09-2-0053 and by an ONR award on “Guiding Learning and Decision Making in the Presence of Multiple Forms of Information.”

References


