Domain Adaptation with Ensemble of Feature Groups

Rajhans Samdani\(^1\) and Scott Wen-Tau Yih\(^2\)

\(^1\)University of Illinois at Urbana-Champaign \(^2\)Microsoft Research

Simple and Effective Algorithm for Domain Adaptation

---

**Domain Adaptation (DA)**

- Supervision for one domain (Source S): e.g., movie review sentiment detection
- Little/no supervision for different but related domain (Target T): book reviews
- Use data on S to improve accuracy on T
- Pervasive: labeled data is scarce!!!
- We show experiments on:
  - DA for Sentiment Analysis
  - Email spam detection: spammers adapt by hacking new computers and changing spam text – need for DA for detection

---

**Our Approach: FEAD**

- Feature Ensemble for Domain Adaptation
- Outputs weighted ensemble of classifiers based on different groups of features
  1. Classifiers are largely trained on source data (plus some labeled target data)
  2. Weights are tuned on small amount of labeled target data (not used in 1)
- Fast and very easy to implement
- Allows for incorporating knowledge about features via feature groups
- Beats state-of-the-art algorithms experimentally

---

**Feature Groups: Concept Drift**

- Moving from source to target, features undergo a change in distribution
  - Features \(x_i\) change little – reliable
  - Features \(x_i\) change a lot – unreliable
- E.g., Email Spam Detection: time changes – domain changes
  - Email text features: easy to change; unreliable across time
  - Sender-ip features: hard to hack new computers; reliable
- Use “feature groups” as learned units with similar cross-domain behavior and learn ensemble

---

**Our Algorithm: FEAD**

1. Given: Data: \(L_S\) and \(L_T\); feature groups: \(X_0, \ldots, X_r\); convex loss function: \(\Delta^r\)
   - // Local classifiers learned on source
   - // Ensemble weights tuned on target
   - // Local classifiers “re-learned” on source-target

2. for \(i = 0\) to \(r\) do
   3. learn \(h_S^i = \arg\min_h \Delta^r_S(h)\)
   4. end for

5. \(\alpha_0, \ldots, \alpha_r \leftarrow \arg\min_{\alpha_0, \ldots, \alpha_r \geq 0} \sum_{(x,y) \in L_T} \Delta^r \left( \sum_i \alpha_i h_S^i (x), y \right)\)
   - // Local classifiers “re-learned” on source-target

6. for \(i = 0\) to \(r\) do
   7. re-learn \(h_S^i = \arg\min_h \Delta^r_{S,T} (h)\)
   8. end for

9. return \(w_k = \sum_i \alpha_i h_S^i\) // final weighted ensemble

---

**Discover Feature Groups by**

- Domain knowledge: different feature generating functions
  - E.g. Email spam detection: email, sender-ip features, user-id features
  - Simple ad-hoc measures: mutual information, frequency, etc.
  - E.g. Sentiment detection: features based on mutual information and frequency

**Generalization Bound for Ensemble \(h\)**

\[
\Delta_{D_T^r}(h) \leq (1 - \beta) \left( \epsilon^* + \sum_i \alpha_i \Delta_{D_S^r}(h_i) + d_H(D_S, D_T) \right) + \epsilon (\rho_M, \beta, \delta) + \beta \Delta_{D_{S,T}}(h) \]

- \(d_H\) distance: a measure of distance between distributions (Ben-David et al., 2006)
- Ensemble coefficients balance empirical target error, empirical source error(s), and distribution distances

---

**Experiments: Sentiment Analysis**

- 2000 labeled reviews for Movies, Books, Kitchen Appliances, DVDs and Electronics (Blitzer et al. 2006, Pang et al. 2002)
- Baselines: Logistic Regression on all data, EasyAdapt, and Multiview-Transfer
- Feature groups:
  - Total: All features
  - Frequent: features which have high M.I. with output labels in source and occur frequently in target
- FEAD stat. significantly better than
  - Multiview-Transfer and EA in 8 cases
  - Better than LR in 5 cases

**Experiments: Spam Detection**

- Hotmail Data: using historical data hence need DA as spam evolves
- Total 915,000 messages: first 765,000 as source data, next 30,000 as target tuning data, last 120,000 as target test data
- Feature Groups: Email-features, Sender-features, Use-features
- Evaluation: ROC curve at low FPR values

**Experiments: Spam Detection**

- Hotmail Data: using historical data hence need DA as spam evolves
- Total 915,000 messages: first 765,000 as source data, next 30,000 as target tuning data, last 120,000 as target test data
- Feature Groups: Email-features, Sender-features, Use-features
- Evaluation: ROC curve at low FPR values

---

**Table: Setting Algorithms**

<table>
<thead>
<tr>
<th>Setting</th>
<th>Algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Src-Tgt</td>
<td>LR Multi-T EA FEAD</td>
</tr>
<tr>
<td>B-E</td>
<td>78.88 78.82 79.38 79.34</td>
</tr>
<tr>
<td>B-D</td>
<td>81.10 80.01 78.63 81.80</td>
</tr>
<tr>
<td>B-K</td>
<td>81.29 79.60 81.66 82.26</td>
</tr>
<tr>
<td>B-M</td>
<td>80.23 77.91 79.25 80.70</td>
</tr>
<tr>
<td>E-B</td>
<td>75.34 72.74 75.85 75.60</td>
</tr>
<tr>
<td>E-D</td>
<td>75.85 75.86 74.78 76.76</td>
</tr>
<tr>
<td>E-K</td>
<td>86.50 83.77 85.33 87.59</td>
</tr>
<tr>
<td>E-M</td>
<td>72.60 70.86 72.63 73.54</td>
</tr>
<tr>
<td>D-B</td>
<td>81.60 79.91 80.14 82.46</td>
</tr>
<tr>
<td>D-E</td>
<td>81.27 81.09 80.34 81.54</td>
</tr>
<tr>
<td>D-K</td>
<td>82.95 81.83 82.08 82.81</td>
</tr>
<tr>
<td>D-M</td>
<td>82.53 79.50 81.53 82.53</td>
</tr>
<tr>
<td>K-B</td>
<td>74.74 75.47 74.78 75.75</td>
</tr>
<tr>
<td>K-E</td>
<td>84.90 84.57 83.81 85.24</td>
</tr>
<tr>
<td>K-D</td>
<td>75.93 76.97 75.21 76.88</td>
</tr>
<tr>
<td>K-M</td>
<td>72.38 71.02 70.45 72.62</td>
</tr>
<tr>
<td>M-B</td>
<td>77.11 77.06 76.07 78.98</td>
</tr>
<tr>
<td>M-E</td>
<td>76.45 80.18 76.50 77.62</td>
</tr>
<tr>
<td>M-D</td>
<td>77.76 77.94 76.20 79.52</td>
</tr>
<tr>
<td>M-K</td>
<td>76.72 79.41 76.48 77.59</td>
</tr>
<tr>
<td>Avg.</td>
<td>78.86 78.23 78.06 79.55</td>
</tr>
</tbody>
</table>