Unsupervised Transfer Learning

Irene Li
18, July, 2018
Why Transfer Learning?

Figure 2. Examples of domain pairs used in the experiments. See Section 4.1 for details.

Train on MNIST: .9891  
Test on MNIST-M: .5749

DROP

Overview: related areas

<table>
<thead>
<tr>
<th>Target Data</th>
<th>Source Data (not directly related to the task)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>labelled</td>
</tr>
<tr>
<td></td>
<td>labelled</td>
</tr>
<tr>
<td></td>
<td>Fine-tuning</td>
</tr>
<tr>
<td></td>
<td>Multitask Learning</td>
</tr>
<tr>
<td></td>
<td>Domain-adversarial training</td>
</tr>
<tr>
<td></td>
<td>Zero-shot learning</td>
</tr>
<tr>
<td></td>
<td>Domain Adaptation</td>
</tr>
<tr>
<td></td>
<td>unlabeled</td>
</tr>
<tr>
<td></td>
<td>Self-taught learning</td>
</tr>
</tbody>
</table>
|             | Rajat Raina, Alexis Battle, Honglak Lee, Benj...
|             | Self-taught learning: transfer learning from...
|             | Wenyuan Dai, Qiang Yang, Gui-Rong Xue, Yong ...
Transfer Learning In NLP?

- Unsupervised TL setting
  - Source domain: $D_S = \{(x_S, y_S)\}$
  - Target domain: $D_T = \{(x_T)\}$

- Problems in NLP
  - Frequency bias: $P(x_S) \neq P(x_T)$
    - Different frequencies: same word in different domains
  - Context feature bias: $P(y_S|x_S) \neq P(y_T|x_T)$
    - “monitor” in Wall Street Journal and Amazon reviews
General Methods

**Feature-based method (popular!):**
- Transfer the features into a same feature space!
- Multi-layer feature learning (representation learning)

**Model-based method:**
- Parameter init + fine-tune (a lot!)
- Parameter sharing

**Instance-based method:**
- Re-weighting: make source inputs similar with target inputs
- Pseudo samples for target domain
Feature-based method: Intuition

Target Domain

Source Domain

New Feature Space
Feature-based method: Deep Adaptation Network

Source domain with labels
Target domain without labels

**Figure 1.** The DAN architecture for learning transferable features. Since deep features eventually transition from general to specific along the network, (1) the features extracted by convolutional layers $conv1$–$conv3$ are general, hence these layers are frozen, (2) the features extracted by layers $conv4$–$conv5$ are slightly less transferable, hence these layers are learned via fine-tuning, and (3) fully connected layers $fc6$–$fc8$ are tailored to fit specific tasks, hence they are not transferable and should be adapted with MK-MMD.

Loss function: discriminativeness and domain invariance

$$\min_{\Theta} \frac{1}{n_a} \sum_{i=1}^{n_a} J(\theta(x_i^a), y_i^a) + \lambda \sum_{\ell=l_1}^{l_2} d^2_k(D_s^\ell, D_t^\ell)$$

Source error (CNN loss) + domain discrepancy (MK-MMD)

Multi-kernel Maximum Mean Discrepancy
Maximum Mean Discrepancy (MMD)

Two-sample problem (unknown p and q):

\[ X := \{x_1, x_2, \ldots, x_m\} \sim p \text{ and } Y := \{y_1, y_2, \ldots, y_n\} \sim q, \text{ test whether } p = q \]

Maximum Mean Discrepancy (Muller, 1997):

Map the layers into a Reproducing Kernel Hilbert Space H with kernel function k:

\[
MMD^2(p, q) = \|p - q\|^2_H = \frac{1}{m^2} \sum_{i,j=1}^m k(x_i, x_j) - \frac{2}{mn} \sum_{i,j=1}^{m,n} k(x_i, y_j) + \frac{1}{n^2} \sum_{i,j=1}^n k(y_i, y_j)
\]

\[O(n^2)\]
MK-MMD: Optimization

Unbiased estimation in $O(n)$:

$$MMD_u^2[F, X, Y] = \frac{1}{m(m-1)} \sum_{i \neq j}^m h(z_i, z_j)$$

$$h(z_i, z_j) := k(x_i, x_j) + k(y_i, y_j) - k(x_i, y_j) - k(x_j, y_i)$$

Kernel:

Gaussian Kernel (RBF), bandwidth sigma could be estimated.

$$K(x, y) = \exp\left(-\frac{||x - y||^2}{2\sigma^2}\right)$$

Multi-kernel:

$$\mathcal{K} \triangleq \left\{ k = \sum_{u=1}^m \beta_u k_u : \sum_{u=1}^m \beta_u = 1, \beta_u \geq 0, \forall u \right\}$$
Unsupervised domain adaptation method: aligns the second-order statistics of the source and target distributions with a linear transformation

\[ l = l_{\text{CLASS}} + \sum_{i=1}^{t} \lambda_i l_{\text{CORAL}} \]

\[ l_{\text{CORAL}} = \frac{\| C_S - C_T \|_F^2}{4d^2} \]

**Correlation Matrix**

\[ C_S = \frac{1}{n_S-1} (D_S^T D_S - \frac{(V^T D_S)^T (V^T D_S)}{n_S}) \]

\[ C_T = \frac{1}{n_T-1} (D_T^T D_T - \frac{(V^T D_T)^T (V^T D_T)}{n_T}) \]

---

**Fig. 1.** Sample Deep CORAL architecture based on a CNN with a classifier layer. For generalization and simplicity, here we apply the CORAL loss to the fc8 layer of AlexNet [20]. Integrating it to other layers or network architectures should be straightforward.

Model-based Method (1): share word embeddings

Task: sentiment classification (pos or neg)
Method: word embeddings -> sentence encoding -> Logistic Regressor

Domain Adapted (DA) embeddings
- Generic embeddings + Domain Specific (DS) embeddings via CCA/KCCA.

Word embeddings to sentence encoding:
- i.e. a weighted combination of their constituent word embeddings.

Use a Logistic Regressor to do classification (pos or neg).
- Canonical Correlation Analysis (CCA)
- Kernel CCA (nonlinear CCA)

Domain Adapted Word Embeddings for Improved Sentiment Classification ACL 2018 (short)
Combine two embedding feature space

\[ \bar{w}_{i,DS} = w_{i,DS} \phi_{DS} \]
\[ \bar{w}_{i,G} = w_{i,G} \phi_{G}. \]

CCA maximizes the correlation between \( \bar{w}_{i,DS} \)
and \( \bar{w}_{i,G} \) to obtain \( \phi_{DS} \) and \( \phi_{G} \) such that

\[ \rho(\phi_{DS}, \phi_{G}) = \max_{\phi_{DS}, \phi_{G}} \frac{E[(\bar{w}_{i,DS}, \bar{w}_{i,G})]}{\sqrt{E[\bar{w}_{i,DS}^2]E[\bar{w}_{i,G}^2]}} \] (2)

Domain Adapted Word Embeddings for Improved Sentiment Classification ACL 2018 (short)
Canonical Correlation Analysis (CCA)

Canonical Correlation Analysis (CCA): $X = (X_1, ..., X_n)$ and $Y = (Y_1, ..., Y_m)$ of random variables, and there are correlations among the variables, then canonical-correlation analysis will find linear combinations of $X$ and $Y$ which have maximum correlation with each other.

$$\bar{w}_{i,DS} = \underbrace{\bar{w}_{i,DS}}_{\text{LSA Embedding}} \phi_{DS}$$

$$\bar{w}_{i,G} = \underbrace{\bar{w}_{i,G}}_{\text{GloVe Embedding}} \phi_G.$$

$$\min_{\alpha, \beta} \| \bar{w}_{i,DS} - (\alpha \bar{w}_{i,DS} + \beta \bar{w}_{i,G}) \|_2^2 + \| \bar{w}_{i,G} - (\alpha \bar{w}_{i,DS} + \beta \bar{w}_{i,G}) \|_2^2.$$

$$\hat{w}_{i,DA} = \frac{1}{2} \bar{w}_{i,DS} + \frac{1}{2} \bar{w}_{i,G}$$

Final Embedding

Domain Adapted Word Embeddings for Improved Sentiment Classification ACL 2018 (short)
Model-based Method(1): share word embeddings

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Embedding</th>
<th>Avg Precision</th>
<th>Avg F-score</th>
<th>Avg AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yelp</td>
<td>(\text{KCCA(\text{Glv, LSA})})</td>
<td>85.36± 2.8</td>
<td>81.89±2.8</td>
<td>82.57±1.3</td>
</tr>
<tr>
<td></td>
<td>(\text{CCA(\text{Glv, LSA})})</td>
<td>83.69± 4.7</td>
<td>79.48±2.4</td>
<td>80.33±2.9</td>
</tr>
<tr>
<td></td>
<td>(\text{KCCA(\text{w2v, LSA})})</td>
<td>87.45± 1.2</td>
<td>83.36±1.2</td>
<td>84.10±0.9</td>
</tr>
<tr>
<td></td>
<td>(\text{CCA(\text{w2v, LSA})})</td>
<td>84.52± 2.3</td>
<td>80.02±2.6</td>
<td>81.04±2.1</td>
</tr>
<tr>
<td></td>
<td>(\text{KCCA(\text{GlvCC, LSA})})</td>
<td><strong>88.11±3.0</strong></td>
<td><strong>85.35±2.7</strong></td>
<td><strong>85.80±2.4</strong></td>
</tr>
<tr>
<td></td>
<td>(\text{CCA(\text{GlvCC, LSA})})</td>
<td>83.69± 3.5</td>
<td>78.99±4.2</td>
<td>80.03±3.7</td>
</tr>
<tr>
<td></td>
<td>(\text{KCCA(\text{w2v, DSw2v})})</td>
<td>78.09± 1.7</td>
<td>76.04±1.7</td>
<td>76.66±1.5</td>
</tr>
<tr>
<td></td>
<td>(\text{CCA(\text{w2v, DSw2v})})</td>
<td>86.22± 3.5</td>
<td>84.35±2.4</td>
<td>84.65±2.2</td>
</tr>
<tr>
<td></td>
<td>(\text{concSVD(\text{Glv, LSA})})</td>
<td>80.14± 2.6</td>
<td>78.50±3.0</td>
<td>78.92±2.7</td>
</tr>
<tr>
<td></td>
<td>(\text{concSVD(\text{w2v, LSA})})</td>
<td>85.11± 2.3</td>
<td>83.51±2.2</td>
<td>83.80±2.0</td>
</tr>
<tr>
<td></td>
<td>(\text{concSVD(\text{GlvCC, LSA})})</td>
<td>84.20± 3.7</td>
<td>80.39±3.7</td>
<td>80.83±3.9</td>
</tr>
<tr>
<td></td>
<td>(\text{GloVe})</td>
<td>77.13± 4.2</td>
<td>72.32±7.9</td>
<td>74.17±5.0</td>
</tr>
<tr>
<td></td>
<td>(\text{GloVe-CC})</td>
<td>82.10± 3.5</td>
<td>76.74±3.4</td>
<td>78.17±2.7</td>
</tr>
<tr>
<td></td>
<td>(\text{word2vec})</td>
<td>82.80± 3.5</td>
<td>78.28±3.5</td>
<td>79.35±3.1</td>
</tr>
<tr>
<td></td>
<td>(\text{LSA})</td>
<td>75.36± 5.4</td>
<td>71.17±4.3</td>
<td>72.57±4.3</td>
</tr>
<tr>
<td></td>
<td>(\text{word2vec})</td>
<td>73.08± 2.2</td>
<td>70.97±2.4</td>
<td>71.76±2.1</td>
</tr>
</tbody>
</table>

Result on Yelp Dataset

Domain Adapted Word Embeddings for Improved Sentiment Classification ACL 2018 (short)
## Model-based Method (2): fine-tune

### Datasets:
- **(Source) MovieQA**
- **(Target 1) TOEFL listening comprehension**
- **(Target2) MCTest**

### Task: QA
Read an article + a question, find out a correct answer from 4 or 5 choices.

### Models: MemN2N, QACNN

<table>
<thead>
<tr>
<th>S</th>
<th>MovieQA</th>
</tr>
</thead>
<tbody>
<tr>
<td>After entering the boathouse, the trio witness Voldemort telling Snape that the elder Wand cannot serve Voldemort until Snape dies ... Before dying, Snape tells Harry to take his memories to the Pensieve ...</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Q</th>
<th>What does Snape tell Harry before he dies?</th>
</tr>
</thead>
</table>

| C<sub>1</sub> | To bury him in the forest |
| C<sub>2</sub> | That he always respected him |
| C<sub>3</sub> | To remember to him for the good deeds |
| C<sub>4</sub> | To take his memories to the Pensieve |
| C<sub>5</sub> | To write down his memories in a book |

---

_Supervised and Unsupervised Transfer Learning for Question Answering. Naccl, 2018_
Model-based Method(2): fine-tune

**Algorithm 1** Unsupervised QA Transfer Learning

**Input:** Source dataset with correct answer to each question; Target dataset without any answer; Number of training epochs.

**Output:** Optimal QA model $M^*$

1. Pre-train QA model $M$ on the source dataset.
2. **repeat**
3. For each question in the target dataset, use $M$ to predict its answer.
4. For each question, assign the predicted answer to the question as the correct one.
5. Fine-tune $M$ on the target dataset as usual.
6. **until** Reach the number of training epochs.

<table>
<thead>
<tr>
<th>Model</th>
<th>Training</th>
<th>TOEFL</th>
<th>MCTest</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>manual</td>
<td>ASR</td>
</tr>
<tr>
<td>QACNN</td>
<td>(a) Target Only</td>
<td>48.9</td>
<td>47.5</td>
</tr>
<tr>
<td></td>
<td>(b) Source Only</td>
<td>51.2</td>
<td>49.2</td>
</tr>
<tr>
<td></td>
<td>(c) Source + Target</td>
<td>52.5</td>
<td>49.7</td>
</tr>
<tr>
<td></td>
<td>(d) Fine-tuned (1)</td>
<td>53.4 (4.5)</td>
<td>51.5 (4.0)</td>
</tr>
<tr>
<td></td>
<td>(e) Fine-tuned (2)</td>
<td><strong>56.1</strong> (7.2)</td>
<td><strong>55.3</strong> (7.8)</td>
</tr>
<tr>
<td></td>
<td>(f) Fine-tuned (all)</td>
<td>56.0 (7.1)</td>
<td>55.1 (7.6)</td>
</tr>
<tr>
<td>MemN2N</td>
<td>(g) Target Only</td>
<td>45.2</td>
<td>44.4</td>
</tr>
<tr>
<td></td>
<td>(h) Source Only</td>
<td>43.7</td>
<td>41.9</td>
</tr>
<tr>
<td></td>
<td>(i) Source + Target</td>
<td>46.8</td>
<td>45.7</td>
</tr>
<tr>
<td></td>
<td>(j) Fine-tuned</td>
<td>48.6 (3.4)</td>
<td>46.6 (2.2)</td>
</tr>
<tr>
<td>Fang et al. (2016)</td>
<td>49.1</td>
<td>48.8</td>
<td>-</td>
</tr>
<tr>
<td>Trischler et al. (2016)</td>
<td>-</td>
<td>-</td>
<td>74.6</td>
</tr>
<tr>
<td>Wang et al. (2015)</td>
<td>-</td>
<td>-</td>
<td>75.3</td>
</tr>
</tbody>
</table>

*Results*
References


http://alex.smola.org/icml2008/
https://github.com/jindongwang/transferlearning
http://cs231n.github.io/transfer-learning/
THANKS

Q & A