VisDA Workshop

Heterogeneous Domain Adaptation: Learning Visual Classifiers from Textual Description

Mohamed Elhoseiny, Babak Saleh, Ahmed Elgammal
Rutgers University
New Results for VisDA In addition to our ICCV13 paper

Two more formulations that was not studied in ICCV13.

A) F1’: Reg Con formulation

We also tried a model that predicts hyper-plane parameters using regression, while utilizing that existing images are negative examples.

\[
\hat{c}(t_*) = \arg\min_{c,\zeta_i} [c^Tc - \beta c^T \Sigma^{-\frac{1}{2}} \hat{c}(t_*)] \\
+ \gamma \sum \zeta_i \\
\text{s.t.} : -(c^Tx_i) \geq \zeta_i, \quad i = 1 \cdots N \\
\zeta_i \geq 0 \\
\gamma, \beta: \text{hyperparameter}
\]
B) F3’: DA-Reg 2 formulation

We also modified this model to penalize deviation of $t_k^T W$ from $c_k$ seen classifier. $W$ is learnt as follows then plugged into our quadratic solver.

$$
\min_{W} r(W) + \lambda_1 \sum_{i} c_i (TWX^T) + \lambda_2 \sum_{i} (ct_i - t_i W)^T (ct_i - t_i W))
$$
## Additional Results - (Not in our ICCV13 paper)

### Results of Different formulations-Flower

<table>
<thead>
<tr>
<th>Formulation</th>
<th>ROC-AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1: Regression</td>
<td>0.580</td>
</tr>
<tr>
<td>F2: DA</td>
<td>0.620</td>
</tr>
<tr>
<td>F1’: Regression Con</td>
<td>0.629</td>
</tr>
<tr>
<td>F3: DA-Reg Con</td>
<td>0.680</td>
</tr>
</tbody>
</table>
### Results of Different formulations - Birds

<table>
<thead>
<tr>
<th>Formulation</th>
<th>ROC-AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1: Regression</td>
<td>0.61</td>
</tr>
<tr>
<td>F2: DA</td>
<td>0.59</td>
</tr>
<tr>
<td>F1’:Regression Con</td>
<td>0.56</td>
</tr>
<tr>
<td>F3: DA-Reg Con</td>
<td>0.62</td>
</tr>
<tr>
<td>F3’ DA-Reg Con2</td>
<td>0.64</td>
</tr>
</tbody>
</table>

F3’ is a newer formulation that beats our ICCV13 formulation by 2% as illustrated in the Table.
In contrast to our ICCV13 and our VisDA13 formulation, we are now able to predict unseen classifier parameters in the form of the representer theorem (the beta vector)

\[ g_k(x^*) = \sum_{i=1}^{N} \beta_k^i k_X(x^*, x_i) + b = \beta_k^T K_{XX}^x \]
Thank You

Project Website: https://sites.google.com/site/mhelhoseiny/projects/computer-vision-projects/Write_a_Classifier includes the data. The code will be available shortly on it.

References: