Cross Language Text Classification via Multi-View Subspace Learning

Yuhong Guo and **Min Xiao**

Dept. of Computer and Information Sciences
Temple University

Problem

- Documents in different languages may share the same set of categories
  - E.g., newsgroup dataset in English and French can cover the same set of categories
- Standard monolingual classification methods
  - Require sufficient number of labels in each language to train a monolingual classifier
  - Expensive document annotation in each language
- How about using labeled data in one language to help the classification in the other language via cross-language text classification?

Cross Language Text Classification

- Idea of cross-language text classification (CLTC):
  - Exploit labeled data existing in language A to classify documents in language B
  - Reduce expensive re-labeling process in language B
- Existing simple CLTC methods rely on machine translation:
  - First translate documents from the source language A to the target language B, or vice versa
  - Then apply standard monolingual classification

Cross Language Text Classification

- Two problems:
  - Feature distribution divergence between the original documents and the translated documents in each language
    - can be addressed by domain adaptation methods
  - Information loss and translation errors in machine translation process
    - can be alleviated by multi-view learning methods that exploit original data in both languages
- How about addressing these two types of problems simultaneously to gain more advantages?

Proposed Approach

- A Subspace Co-regularized Multi-view Learning Method
  - Translate the documents in each language into the other language using machine translation to form two parallel data matrices (two views): $\mathbf{X}_1, \quad \mathbf{X}_2$
  - Exploit data from both views (languages) to alleviate the translation loss
  - Learn discriminative subspace representations of multi-view documents
    - capture the intrinsic structure of the data, bridge domain divergence

Prediction Function

- In each view, we project the data into low-dimensional subspace, and then use a linear prediction function. In the *i*th view, it is:

$$f_i(X_i^\ell) = X_i^\ell \Theta_i \mathbf{w}_i + b_i$$

where $\Theta_i \in I\!R^{d_i \times m}, \Theta_i^\top \Theta_i = I$ projects data into low-dimensional subspace

Formulation

- Two-view learning formulation:

loss function for predictors on labeled data

regularizer

$$\min_{\{\Theta_i, \mathbf{w}_i, b_i\}} \quad \sum_{i=1}^{2} \|X_i^{\ell}\Theta_i \mathbf{w}_i + b_i - \mathbf{y}\|^2 + \alpha_i \|\mathbf{w}_i\|^2$$
$$+\gamma \|X_1\Theta_1 - X_2\Theta_2\|_F^2$$
$$\text{s. t.} \quad \Theta_1^\top \Theta_1 = I, \quad \Theta_2^\top \Theta_2 = I.$$

Subspace co-regularization:
distance of the two views on projected low-dimensional space

Formulation

- After solving the minimization over $\{\mathbf{w}_i, b_i\}$ for closed-form solutions, we obtain the following orthogonal constrained problem

$$\min_{\Theta_1, \Theta_2} L(\Theta_1, \Theta_2) \quad \text{s. t. } \Theta_1^\top \Theta_1 = I, \ \Theta_2^\top \Theta_2 = I.$$

$$\begin{aligned} L(\Theta_1, \Theta_2) &= \gamma \| X_1 \Theta_1 - X_2 \Theta_2 \|_F^2 + 2\mathbf{y}^\top H \mathbf{y} \\ &- \sum_{i=1}^{2} \mathbf{z}_i^\top \Theta_i (\Theta_i^\top M_i \Theta_i + \alpha_i I)^{-1} \Theta_i^\top \mathbf{z}_i \end{aligned}$$

Optimization Algorithm

- Gradient descent with curvilinear search:
  - ✓ requires no local projections
  - ✓ always stays in feasible orthogonal region
  - ✓ converges to local optimal solution

Optimization Algorithm

- Given gradients:

$$G_1 = \nabla_{\Theta_1} L(\Theta_1, \Theta_2), \quad G_2 = \nabla_{\Theta_2} L(\Theta_1, \Theta_2).$$

- Compute skew symmetric matrices:

$$F_1 = G_1 \Theta_1^\top - \Theta_1 G_1^\top, \quad F_2 = G_2 \Theta_2^\top - \Theta_2 G_2^\top$$

Such that

- Curvilinear local search:

$$\begin{aligned} Q_1(\tau) &= \Big(I + \frac{\tau}{2} F_1\Big)^{-1} \Big(I - \frac{\tau}{2} F_1\Big) \Theta_1 \\ Q_2(\tau) &= \Big(I + \frac{\tau}{2} F_2\Big)^{-1} \Big(I - \frac{\tau}{2} F_2\Big) \Theta_2 \end{aligned}$$

note: $Q_1(\tau)^\top Q_1(\tau) = I$ and $Q_2(\tau)^\top Q_2(\tau) = I$

Local descent path at $\tau \geq 0$

Experiments

➢ Dataset

- A multilingual dataset: Reuters RCV1/RCV2
- 5 languages
  - English (**E**), French (**F**), German (**G**), Italian (**I**), Spanish (**S**)

➢ Comparison Approaches

- Baselines methods: TB, TSB
- Domain adaptation methods: EA++
- Multi-view co-classification method: MVMV, MVCC
- Proposed Method: SCMV

Results

Table 1. Average classification accuracy results over 10 runs for 20 CLTC tasks.

TASKS	TB	TSB	EA++	MVMV	MVCC	SCMV
E2F	78.60±0.80	79.24±0.51	79.52±0.47	81.13±0.46	83.20±0.38	**86.10**±0.42
E2G	75.65±0.67	75.01±0.51	75.25±0.46	80.37±0.76	81.62±0.54	**83.51**±0.74
E2I	79.80±0.69	76.39±0.98	76.48±1.02	80.01±0.69	83.75±0.64	**84.87**±0.51
E2S	84.54±1.52	85.24±1.01	85.43±1.03	86.30±0.69	89.98±0.42	**92.26**±0.34
F2E	77.04±0.92	80.32±0.47	80.60±0.48	81.15±0.44	82.51±0.36	**83.86**±0.35
F2G	76.41±0.92	76.32±0.62	76.68±0.49	79.66±0.91	81.84±0.76	**83.16**±0.70
F2I	78.32±0.82	77.02±0.78	78.87±0.75	79.53±0.63	82.98±0.47	**83.25**±0.43
F2S	84.77±1.05	86.24±0.71	86.90±0.69	87.53±0.68	90.96±0.44	**92.81**±0.25
G2E	77.04±0.88	78.57±0.37	78.42±0.36	78.68±0.68	80.52±0.50	**82.52**±0.47
G2F	75.93±0.70	77.08±0.51	77.22±0.42	77.99±0.61	80.57±0.48	**83.55**±0.36
G2I	79.88±0.77	78.54±1.05	78.61±0.99	78.07±0.78	81.85±0.54	**84.20**±0.51
G2S	85.82±0.91	86.22±0.55	86.61±0.57	84.73±0.62	89.24±0.37	**90.67**±0.61
I2E	76.98±0.74	76.76±0.42	77.80±0.40	78.86±0.61	80.45±0.47	**81.34**±0.48
I2F	76.88±0.94	78.10±0.35	78.61±0.47	78.11±0.65	80.58±0.60	**81.73**±0.42
I2G	76.79±0.57	76.56±0.55	77.66±0.48	79.69±0.61	80.50±0.53	**84.76**±0.35
I2S	85.36±1.42	87.68±0.50	88.63±0.51	89.42±0.56	90.66±0.33	**94.15**±0.44
S2E	74.35±0.94	74.73±0.63	74.83±0.69	77.89±0.54	79.45±0.58	**80.50**±0.44
S2F	75.89±1.10	77.48±0.58	77.62±0.57	77.93±0.62	82.82±0.22	**84.86**±0.33
S2G	75.88±0.44	74.28±0.40	74.31±0.34	77.91±0.56	80.90±0.44	**81.12**±0.53
S2I	79.36±0.84	79.72±0.69	80.54±0.75	82.46±0.65	87.18±0.46	**88.59**±0.47

Thanks!