Writer Adaptation using Bottleneck Features and Discriminative Linear Regression for Online Handwritten Chinese Character Recognition

Jun Du, Jin-Shui Hu, Bo Zhu, Si Wei, **Li-Rong Dai**
University of Science and Technology of China
iFlytek Research

ICFHR 2014, Crete, Greece, September 1-4, 2014

Background

- Chinese handwriting recognition is popular
  - Especially on portable devices in mobile internet era
- User experience largely depends on the writing style
  - Mismatch even with more and more diversified training data
- Solution: writer adaptation
  - Can really improve the user experience for a specific writer
  - Supervised mode with automatically labeled data by users

Related Work for Writer Adaptation

- For handwriting recognition of western languages
  - Adaptable output layer of a time delay neural network (1993)
  - Adding a radial basis function to neural networks (1997)
  - MLLR and MAP for HMM based system (2001)
  - Biased regularization for SVM (2006)

- For Chinese handwriting recognition
  - STM: Style Transfer Mapping (2011)

Core Innovations

- Bottleneck features (BNF) for feature extraction
  - A highly nonlinear and discriminative transformation
  - Superior to linear transformation based on LDA

- Discriminative linear regression (DLR) for writer adaptation

- Incorporate BNF and DLR with prototype-based classifier
  - Significantly outperform STM

Multi-prototype based Classifier

- Classification with discriminant functions

$$r(\mathbf{x};\boldsymbol{\Lambda}) = \arg\max_i g_i(\mathbf{x};\lambda_i)$$

$$g_i(\mathbf{x};\lambda_i) = -\min_k \parallel \mathbf{x} - \mathbf{m}_{ik} \parallel^2$$

- Minimum Classification Error (MCE) criterion

$$l(\mathcal{X};\boldsymbol{\Lambda}) = \frac{1}{R}\sum_{r=1}^{R} \frac{1}{1+\exp[-\alpha d(\mathbf{x}_r;\boldsymbol{\Lambda})+\beta]}$$

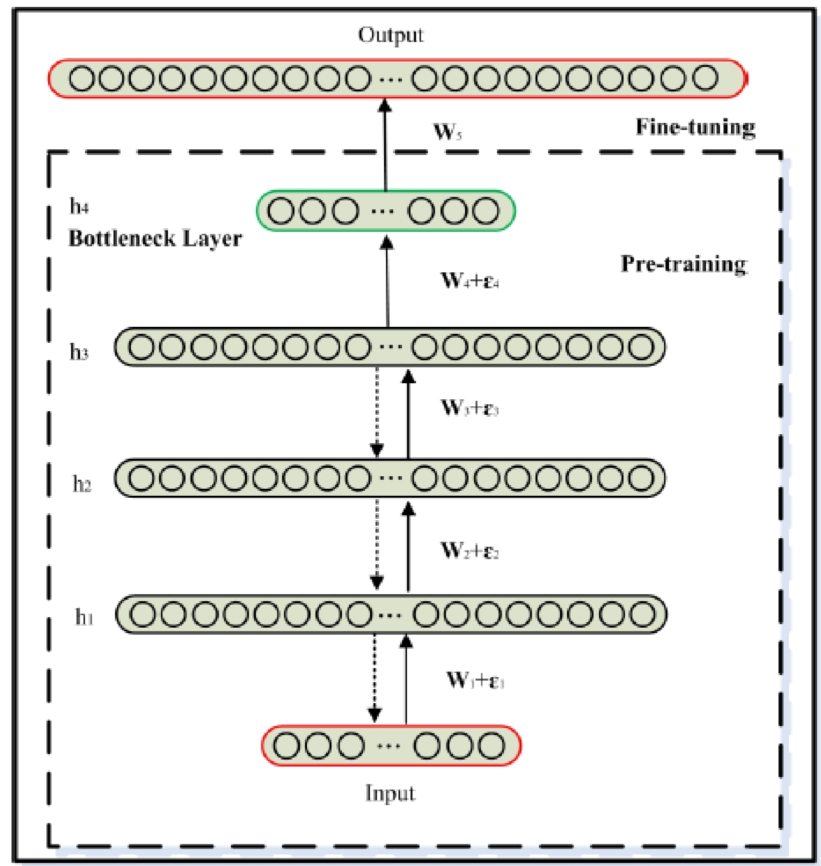
- Misclassification measure
  - Sample Separation Margin (SSM)

$$d(\mathbf{x}_r;\boldsymbol{\Lambda}) = \frac{-g_p(\mathbf{x}_r;\lambda_p)+g_q(\mathbf{x}_r;\lambda_q)}{2\parallel \mathbf{m}_{p\hat{k}} - \mathbf{m}_{q\bar{k}} \parallel}$$

Bottleneck feature extractor

- Extracting from a bottleneck layer of DNN
  - DNN input: LDA transformed feature vector
  - DNN output: the posterior probability of character classes

- Hinton's training recipe
  - Layer-by-layer RBM pre-training
  - Cross-entropy fine-tuning



Writer Adaptation via Linear Regression

- Feature transformation

$$\mathbf{x}_r = \mathcal{F}(\mathbf{y}_r; \boldsymbol{\Theta}) = \mathbf{A}\mathbf{y}_r + \mathbf{b}$$

- Style transfer mapping

$$\min_{\mathbf{A}} \sum_{r=1}^{R'} f_r \|\mathbf{A}\mathbf{s}_r - \mathbf{t}_r\|_2^2 + \beta_1 \|\mathbf{A} - \mathbf{I}\|_2^2$$

- Discriminative linear regression (SSM-MCE)

$$l(\mathcal{Y}; \boldsymbol{\Lambda}, \boldsymbol{\Theta}) = \frac{1}{R'} \sum_{r=1}^{R'} \frac{1}{1 + \exp[-\alpha d(\mathbf{y}_r; \boldsymbol{\Lambda}, \boldsymbol{\Theta}) + \beta]}$$

$$d(\mathbf{y}_r; \boldsymbol{\Lambda}, \boldsymbol{\Theta}) = \frac{-g_p(\mathbf{x}_r; \lambda_p) + g_q(\mathbf{x}_r; \lambda_q)}{2 \parallel \mathbf{m}_{p\hat{k}} - \mathbf{m}_{q\bar{k}} \parallel}$$

Experimental Setup

- Database
  - Training: 15167 character classes, totally 14846606 samples
  - Data from 105 real users written in several months
    - 5000-30000 character samples for each user
    - Random half for adaptation and testing
- Feature extraction
  - 392-dimensional raw feature: 8-directional features
  - LDA transformation: 392 -> 96
- DNN architecture for BNF: 96-1024-1024-1024-96-15167

No Adaptation: BNF vs. LDA

- BNF significantly outperforms LDA with LBG initialization
- The gap between BNF and LDA is smaller after SSM-MCE

Table 1. Performance (character error rate in %) comparison of systems using prototype-based classifiers with different features and different training criteria on the testing set of all 105 writers.

	#prototype	LBG	SSM-MCE
LDA	1	33.97	22.16
	2	30.63	20.20
	4	27.08	19.14
BNF	1	26.06	19.66
	2	23.56	19.12
	4	22.01	18.79

Writer Adaptation using Different Approaches

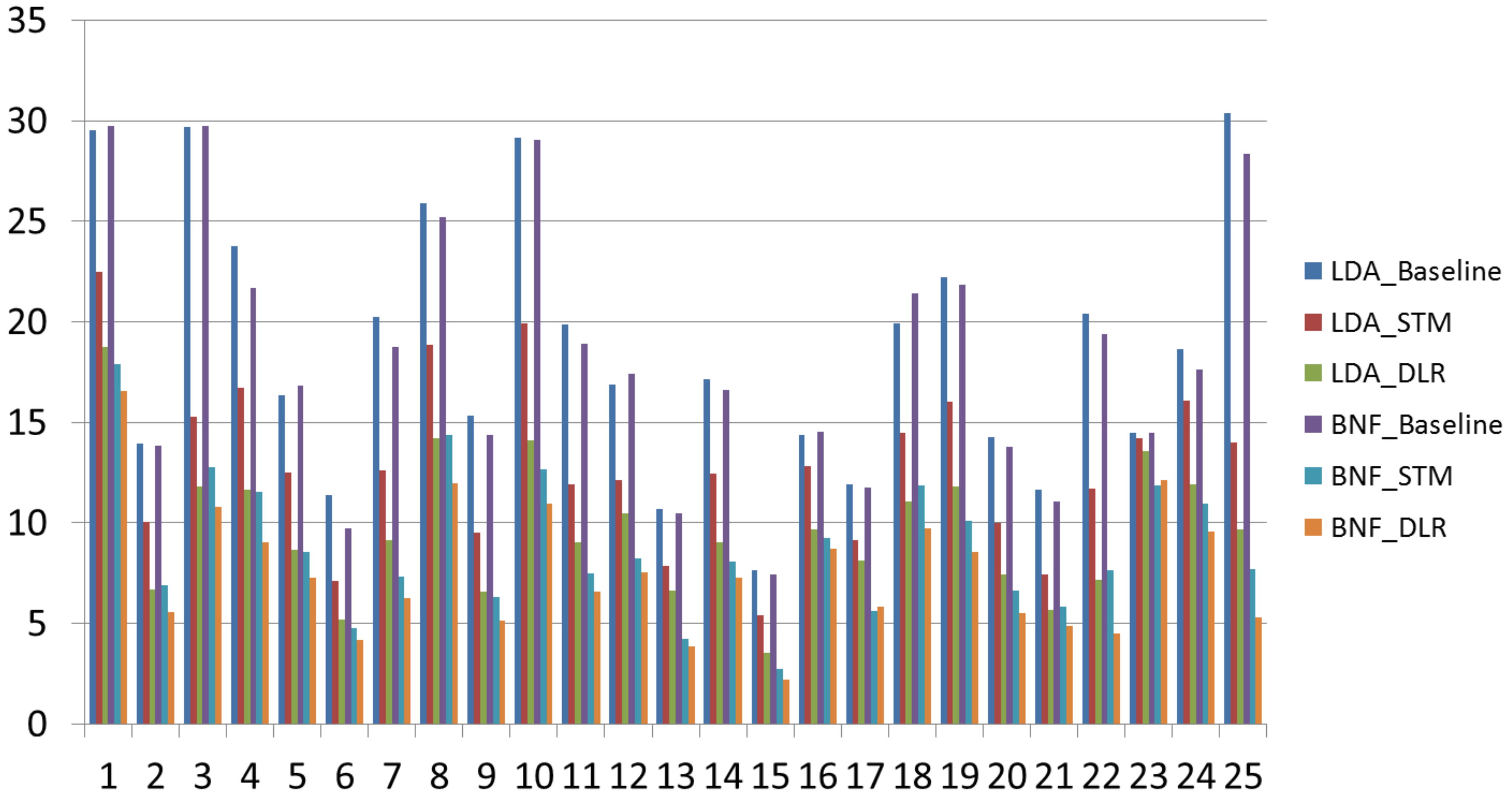
- Both BNF and DLR bring significant improvements
- BNF and DLR are complementary (40% ERR over LDA+STM)
- More adaptation data is useful for DLR rather than STM

Table 2. Performance (character error rate in %) comparison of systems using different adaptation strategies averaged across each testing set of all 105 writers.

	LDA		BNF	
	STM	DLR	STM	DLR
Baseline	19.14		18.79	
WA(1000)	13.49	11.96	9.79	9.42
WA(3000)	13.29	10.51	9.31	8.53
WA(5000)	13.24	10.11	9.24	8.14

Comparison for 25 selected writers

- In most cases, BNF+DLR achieves the best performance



Summary and Future Work

- BNF+DLR achieves promising results
  - Writer adaptation is easier in highly nonlinear feature space
  - Discriminatively trained linear regression is more powerful

- Future work
  - Unsupervised, semi-supervised adaptation
  - Extend the linear regression to nonlinear for writer adaptation
  - Writer adaptation on deep learning based classifiers