Language Recognition via Sparse Coding

Overview

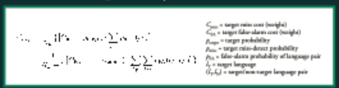
- Language Recognition (LRE)
  - Discriminative task to identify spoken language in speech utterance
  - Acoustic signal processing techniques and learning algorithms crucial in modeling distinguishing characteristics of language
- Approach
  - Extract low-level spectral features of speech as input to sparse coding
  - Improve discriminative quality of sparse-coded speech features via maximum a posteriori (MAP) adaptation for sparse coding dictionary
- Result
  - Outperforms Lincoln i-vector pipeline developed for NIST LRE 2015 on subset comprising Arabic and Chinese clusters

Experiments

- Task: NIST Language Recognition Evaluation (LRE) 2015
  - Aims to examine average performance of language recognition system that can classify target language correctly within six predefined language clusters for a given speech sample
  - LRE 2015 covers 20 different languages in Arabic, Chinese, English, French, Slavic, and Iberian clusters
  - Our evaluation focuses on Arabic and Chinese clusters only

Cluster	Target languages
Arabic	Egyptian, Iraqi, Levantine, Maghrebi, Modern Standard
Chinese	Cantonese, Mandarin, Min, Wu

- Evaluation metric: NIST average cost performance



Sparse Coding Background

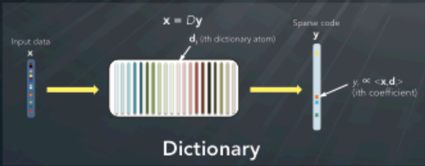
- Unsupervised method to learn efficient representation of data using small number of basis vectors (or atoms) from dictionary D
- While solving for sparse representation of input, sparse coding also learns D
- Two forms
  - L_1-regularized LASSO (least absolute shrinkage and selection operator) and LARS (least angle regression)

$$\min_{D,y} \|x - Dy\|_2^2 + \lambda \|y\|_1 \quad \text{s.t. } \|d_i\|_2 \le 1, \forall i$$

  - L_0-regularized matching pursuit (OMP)

$$\min_{D,y} \|x - Dy\|_2^2 \quad \text{s.t. } \|y\|_0 \le S.$$

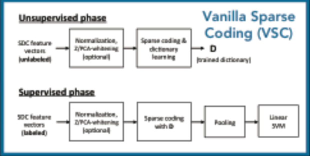
- Computing sparse representation of data = discovering higher-level features present in data
- Computing sparse representation of data = discovering higher-level features present in data



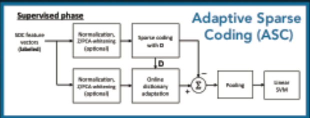
Approach

- Low-level feature extraction
  - Compute shifted delta cepstra (SDC) by windowing speech waveforms, passing through mel-scale and RASTA filterbanks, and normalizing
- Vanilla sparse coding (VSC)
  - Classical semi-supervised approach
  - Unsupervised high-level feature learning by sparse coding and dictionary learning
  - Sparse-coded features are pooled and applied to supervised training of classifiers (e.g., linear SVM)
- Adaptive sparse coding (ASC)
  - Improves VSC by adapting VSC dictionary D to utterance-specific dictionary Da in supervised procedure
  - Using both D and Da, compute two sparse codes
  - Arithmetic difference vector of the two sparse codes are used to train classifiers

Vanilla Sparse Coding (VSC)

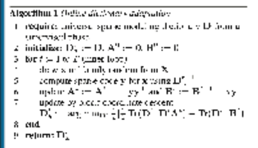


Adaptive Sparse Coding (ASC)



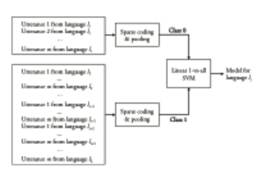
Dictionary Adaptation Algorithm

Key idea: using the dictionary trained on data examples from all classes as initial point, adjust dictionary atoms to fit class-specific examples based on maximum a posteriori (MAP) criterion



Language Classification via SVM

We train 1-vs-all linear SVM classifier for each language



Postprocessing of Classification Results

- Historical NIST LRE systems have benefitted from fusing classification results of multiple LRE pipelines
- We use simple linear Z-score fusion based for postprocessing

$$\mathrm{llr}^{l_T}_{\text{fusion}} = \rho \cdot \frac{\mathrm{llr}_1^{l_T} - \mu_1^{l_T}}{\sigma_1^{l_T}} + (1 - \rho) \cdot \frac{\mathrm{llr}_2^{l_T} - \mu_2^{l_T}}{\sigma_2^{l_T}}$$

- Final classification result is computed using mixing ratio ρ to combine log-likelihood ratio scores (llr1 and llr2) from two different classification pipelines

Summary

- Sparse coding can also be applied to discriminative speech applications (e.g., language and speaker recognition)
- Sparse modeling can improve discriminative quality of spectral speech features
- In addition to vanilla sparse coding (VSC), we propose adaptive sparse coding (ASC), an improvement over VSC via MAP adaptation
- We experimentally validate improved performance of sparse coding over iVector-SDC on Arabic and Chinese subset from NIST LRE 2015
- Future work
  - Evaluation on full NIST LRE dataset
  - Explore different speech feature inputs such as DNN bottleneck features

Results: Individual Pipelines

(lower is better)

Classification pipeline	Arabic	Chinese
iVector-SDC baseline	0.2566	0.2054
Vanilla OMP-1024 sparse coding on SDC	0.2486	0.2120
Vanilla LARS-1024 sparse coding on SDC	0.2393	0.2043
Adaptive OMP-1024 sparse coding on SDC	0.2015	0.1983
Adaptive LARS-1024 sparse coding on SDC	**0.1874**	**0.1634**

- We evaluated both forms of sparse coding LARS and OMP that train 1024 dictionary atoms
- Adaptive sparse coding with LARS-1024 is our best scheme
  - Significantly outperforms iVector-SDC baseline

Fusion scheme	Arabic	Chinese
iVector-SDC baseline + Vanilla LARS-1024 sparse coding on SDC	0.1988	0.1857
iVector-SDC baseline + Adaptive LARS-1024 sparse coding on SDC	0.1652	0.1226

- We calibrated mixing ratio ρ between 0.1 to 0.9
- Fusion enables us to achieve the new best cost performance for both Arabic and Chinese

Youngjune L. Gwon, William M. Campbell, Douglas Sturim, MIT Lincoln Laboratory and H. T. Kung, Harvard University

This material is based upon work supported by the Department of Defense under Air Force Contract No. (FA8721-05-C-0002 and/or FA8702-15-D-0001). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of Defense.

LINCOLN LABORATORY
Massachusetts Institute of Technology