Deep Network for Speech Emotion Recognition

Master Thesis

Zhuowei Han

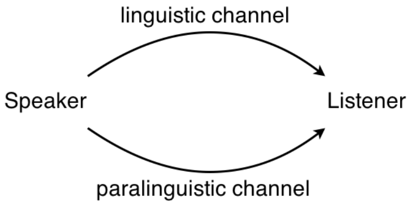
Institut für Signalverarbeitung
und Systemtheorie

Universität Stuttgart

16/04/2015

Speech Emotion Recognition

- More natural human-machine interaction requires paralinguistic information such as age, gender, emotion.
- Emotion is high-dimensional complex data with non-linear time-variant hidden features
- Traditional feature learning is labor expensive



Deep Learning

- New research area of machine learning
- Deep architecture for building high-level representations via unsupervised feature learning
- Learning both temporal and non-temporal features
- Application in vision/audition processing, e.g. handwriting recognition (Graves, Alex, et al. 2009), traffic sign classification (Schmidhuber, et al. 2011), text translation (Google, 2014)

Table of Contents

Table of Contents

Framework of Emotion Recognition

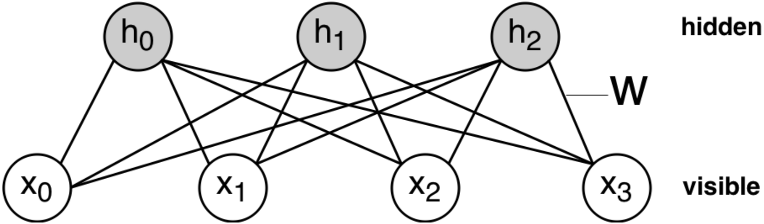
- Extract spectrum features: Mel Frequency Cepstral Coefficients
- Aggregate MFCCs to build high-level representations via unsupervised learning
- Classification based on high-level features via supervised learning

Table of Contents

Conditional Restricted Boltzmann Machine

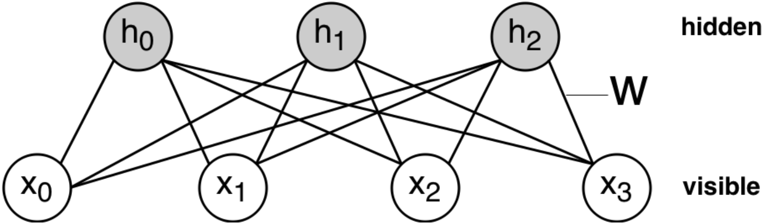
- Energy-based undirected graphical model
- Contains hidden variables (hidden units), increases the modeling capacity.
- Unsupervised feature learning
  - build high-level features from low-level features
  - learned features used for prediction or classification
- Successfully applied in motion capture (Graham W. Taylor, Geoffrey E. Hinton, 2006)

Structure



visible/input layer	$\mathbf{x} \in \{0, 1\}$
hidden layer	$\mathbf{h} \in \{0, 1\}$
weight	$\mathbf{W}$
visible bias	$\mathbf{b}$
hidden bias	$\mathbf{c}$
parameter set	$\boldsymbol{\theta} = \{\mathbf{W}, \mathbf{b}, \mathbf{c}\}$

Structure



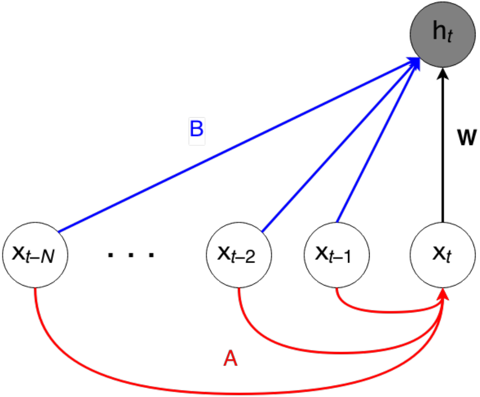
Energy Function: $E_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{h}) = -\mathbf{x}^{\mathbf{T}}\mathbf{W}\mathbf{h} - \mathbf{b}^{\mathbf{T}}\mathbf{x} - \mathbf{c}^{\mathbf{T}}\mathbf{h}$

Joint Distribution: $P^{RBM}(\mathbf{x},\mathbf{h}) = \frac{1}{Z} e^{-E_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{h})}$

Partition Function: $Z = \sum_{\mathbf{x},\mathbf{h}} e^{-E_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{h})}$

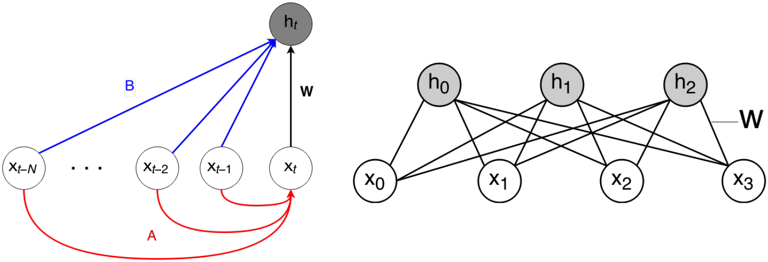
Free Energy: $\mathcal{F}(\mathbf{x}) = -\log \sum_{h} e^{-E(\mathbf{x},\mathbf{h})}$

Conditional RBM



- Consider visible units from previous time step as additional bias for current visible and hidden layer
- Visible layer consists of linear units with independent Gaussian noise to model real-valued data, e.g. spectral features

Conditional RBM



Energy Function: $E_{\boldsymbol{\theta}}^{CRBM}(\mathbf{x}, \mathbf{h}) = \left\| \frac{\mathbf{x} - \tilde{\mathbf{b}}}{2} \right\|^2 - \tilde{\mathbf{c}}^T \mathbf{h} - \mathbf{x}^T \mathbf{W} \mathbf{h}$

$$\tilde{\mathbf{b}} = \mathbf{b} + \mathbf{A} \cdot \mathbf{x}_{<t}$$

$$\tilde{\mathbf{c}} = \mathbf{c} + \mathbf{B} \cdot \mathbf{x}_{<t}$$

$$\boldsymbol{\theta} = \{\mathbf{W}, \mathbf{A}, \mathbf{B}, \mathbf{b}, \mathbf{c}\}$$

Free Energy: $\mathcal{F}(\mathbf{x}) = -\log \sum_h e^{-E(\mathbf{x},\mathbf{h})}$

Inference

$$P(\mathbf{x}) = \sum_{\mathbf{h}} P(\mathbf{x}, \mathbf{h})$$

$$P(\mathbf{h}) = \sum_{\mathbf{x}} P(\mathbf{x}, \mathbf{h})$$

$$P(\mathbf{h}|\mathbf{x}) = \frac{P(\mathbf{x}, \mathbf{h})}{P(\mathbf{x})}$$

$$P(\mathbf{x}|\mathbf{h}) = \frac{P(\mathbf{x}, \mathbf{h})}{P(\mathbf{h})}$$

$$P(h_j = 1 \mid \mathbf{x}) = sigmoid(\sum_i x_i W_{ij} + c_j)$$

$$P(x_i = 1 \mid \mathbf{h}) = sigmoid(\sum_j W_{ij} h_j + b_i)$$

Training of Energy-based Model

Maximum Likelihood Estimation $P^{\boldsymbol{\theta}}(\mathbf{x})$

Training of Energy-based Model

Maximum Likelihood Estimation $P^{\boldsymbol{\theta}}(\mathbf{x})$

Kullback-Leibler Divergence:

$$\begin{aligned} Q(\mathbf{x}) \| P^{\boldsymbol{\theta}}(\mathbf{x}) &= \int_{-\infty}^{\infty} Q(\mathbf{x}) \cdot \log \frac{Q(\mathbf{x})}{P^{\boldsymbol{\theta}}(\mathbf{x})} \mathrm{d}\mathbf{x} \\ &= \int_{-\infty}^{\infty} Q(\mathbf{x}) \cdot \log Q(\mathbf{x}) \mathrm{d}\mathbf{x} - \int_{-\infty}^{\infty} Q(\mathbf{x}) \cdot \log P^{\boldsymbol{\theta}}(\mathbf{x}) \mathrm{d}\mathbf{x} \\ &= \langle \log Q(\mathbf{x}) \rangle_{Q(\mathbf{x})} - \Big\langle \log P^{\boldsymbol{\theta}}(\mathbf{x}) \Big\rangle_{Q(\mathbf{x})} \end{aligned}$$

$Q(\mathbf{x})$, true data distribution

$P^{\boldsymbol{\theta}}(\mathbf{x})$, parameterized distribution, to be estimated

$\langle \cdot \rangle_{Q(\mathbf{x})}$, expectation w.r.t. $Q(\mathbf{x})$

Training of Energy-based Model

Maximum Likelihood Estimation $P^{\boldsymbol{\theta}}(\mathbf{x})$

Kullback-Leibler Divergence:

$$
\begin{aligned}
Q(\mathbf{x}) \| P^{\boldsymbol{\theta}}(\mathbf{x}) &= \int_{-\infty}^{\infty} Q(\mathbf{x}) \cdot \log \frac{Q(\mathbf{x})}{P^{\boldsymbol{\theta}}(\mathbf{x})} \mathrm{d}\mathbf{x} \\
&= \int_{-\infty}^{\infty} Q(\mathbf{x}) \cdot \log Q(\mathbf{x}) \mathrm{d}\mathbf{x} - \int_{-\infty}^{\infty} Q(\mathbf{x}) \cdot \log P^{\boldsymbol{\theta}}(\mathbf{x}) \mathrm{d}\mathbf{x} \\
&= \langle \log Q(\mathbf{x}) \rangle_{Q(\mathbf{x})} - \left\langle \log P^{\boldsymbol{\theta}}(\mathbf{x}) \right\rangle_{Q(\mathbf{x})}
\end{aligned}
$$

$Q(\mathbf{x})$, true data distribution

$P^{\boldsymbol{\theta}}(\mathbf{x})$, parameterized distribution, to be estimated

$\langle \cdot \rangle_{Q(\mathbf{x})}$, expectation w.r.t. $Q(\mathbf{x})$

Training of Energy-based Model

$$-\log P^{\boldsymbol{\theta}}(\mathbf{x}) = \mathcal{F}(\mathbf{x}) + \log \sum_{\mathbf{x}} \sum_{\mathbf{h}} e^{-E_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{h})}$$

$$-\frac{\partial \log P^{\boldsymbol{\theta}}(\mathbf{x})}{\partial \boldsymbol{\theta}} = \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} - \sum_{\tilde{\mathbf{x}}} P(\tilde{\mathbf{x}}) \frac{\partial \mathcal{F}(\tilde{\mathbf{x}})}{\partial \boldsymbol{\theta}}$$

$\mathbf{x}$, input (visible) data space
$\tilde{\mathbf{x}}$, all possible vectors in the data space, generated by model.

Training of Energy-based Model

$$-\log P^{\boldsymbol{\theta}}(\mathbf{x}) = \mathcal{F}(\mathbf{x}) + \log \sum_{\mathbf{x}} \sum_{\mathbf{h}} e^{-E_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{h})}$$

$$-\frac{\partial \log P^{\boldsymbol{\theta}}(\mathbf{x})}{\partial \boldsymbol{\theta}} = \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} - \sum_{\tilde{\mathbf{x}}} P(\tilde{\mathbf{x}}) \frac{\partial \mathcal{F}(\tilde{\mathbf{x}})}{\partial \boldsymbol{\theta}}$$

$\mathbf{x}$, input (visible) data space
$\tilde{\mathbf{x}}$, all possible vectors in the data space, generated by model.

objective function by averaging log-likelihood over data:

$$-\left\langle \frac{\partial \log P^{\boldsymbol{\theta}}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{Q(\mathbf{x})} = \left\langle \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{Q(\mathbf{x})} - \left\langle \frac{\partial \mathcal{F}(\tilde{\mathbf{x}})}{\partial \boldsymbol{\theta}} \right\rangle_{P(\tilde{\mathbf{x}})}$$

Training of Energy-based Model

$$-\log P^{\boldsymbol{\theta}}(\mathbf{x}) = \mathcal{F}(\mathbf{x}) + \log \sum_{\mathbf{x}} \sum_{\mathbf{h}} e^{-E_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{h})}$$

$$-\frac{\partial \log P^{\boldsymbol{\theta}}(\mathbf{x})}{\partial \boldsymbol{\theta}} = \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} - \sum_{\tilde{\mathbf{x}}} P(\tilde{\mathbf{x}}) \frac{\partial \mathcal{F}(\tilde{\mathbf{x}})}{\partial \boldsymbol{\theta}}$$

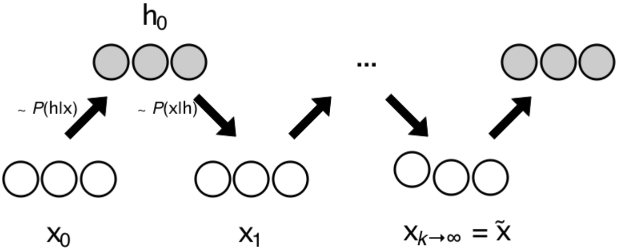
$\mathbf{x}$, input (visible) data space
$\tilde{\mathbf{x}}$, all possible vectors in the data space, generated by model.

objective function by averaging log-likelihood over data:

$$-\left\langle \frac{\partial \log P^{\boldsymbol{\theta}}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{Q(\mathbf{x})} = \left\langle \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{Q(\mathbf{x})} - \left\langle \frac{\partial \mathcal{F}(\tilde{\mathbf{x}})}{\partial \boldsymbol{\theta}} \right\rangle_{P(\tilde{\mathbf{x}})}$$

Contrastive Divergence (Hinton)

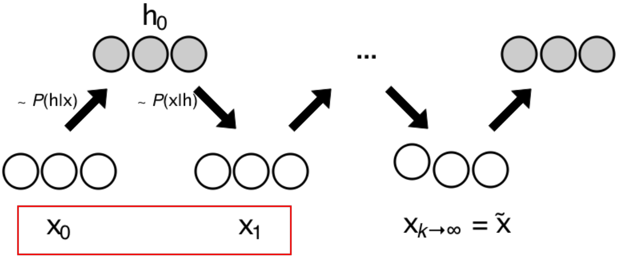
- Obtain $P(\tilde{\mathbf{x}})$ by Gibbs sampling
- k=0, $P_0(\mathbf{x})(= Q(\mathbf{x}))$ is true data distribution, independent of parameter $\boldsymbol{\theta}$
- $P^{\boldsymbol{\theta}}_{\infty}(\mathbf{x}) \to P(\tilde{\mathbf{x}})$



Contrastive Divergence (Hinton)

- In practise perform 1-Gibbs step will work well:

$$-\left\langle \frac{\partial \log P^{\boldsymbol{\theta}}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{P_0(\mathbf{x})} = \left\langle \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{P_0(\mathbf{x})} - \left\langle \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{P_1^{\boldsymbol{\theta}}(\mathbf{x})}$$



Contrastive Divergence (Hinton)

- In practise perform 1-Gibbs step will work well:

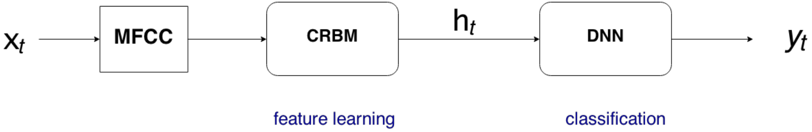
$$-\left\langle \frac{\partial \log P^{\boldsymbol{\theta}}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{P_0(\mathbf{x})} = \left\langle \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{P_0(\mathbf{x})} - \left\langle \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{P_1^{\boldsymbol{\theta}}(\mathbf{x})}$$

$$\Delta\boldsymbol{\theta} \sim \left\langle \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{P_0} - \left\langle \frac{\partial \mathcal{F}(\mathbf{x})}{\partial \boldsymbol{\theta}} \right\rangle_{P_1^{\boldsymbol{\theta}}}$$

Table of Contents

DNN for Classification

- Using high-level features to perform classification
- DNN Structure
  - Feedforward network
  - Recurrent network



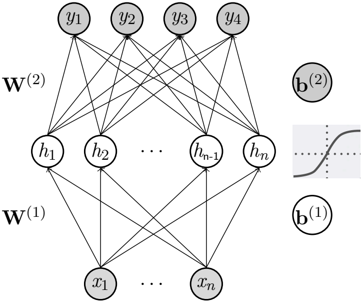
Feedforward Structure

Hidden layer pre-activation:

$$\mathbf{a}(\mathbf{x}) = \mathbf{W}^{(1)}\mathbf{x} + \mathbf{b}^{(1)}$$

Hidden layer activation:

$$\mathbf{h} = f(\mathbf{a})$$



Output layer activation of single hidden layer:

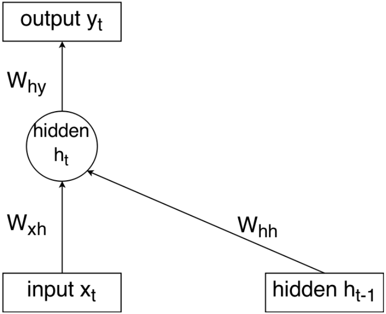
$$\hat{y}(\mathbf{x}) = o(\mathbf{W}^{(2)}\mathbf{h}^{(1)} + \mathbf{b}^{(2)})$$

Output layer activation of N hidden layers:

$$\hat{y}(\mathbf{x}) = o(\mathbf{W}^{(N+1)}\mathbf{h}^{(N)} + \mathbf{b}^{(N+1)})$$

Recurrent Structure

$$h_t = \mathcal{H}(W_{xh}x_t + W_{hh}h_{t-1} + b_h)$$
$$y_t = W_{hy}h_t + b_y$$



Empirical Risk Minimization

- Objective

$$\arg\min_{\boldsymbol{\theta}} \frac{1}{M} \sum_{m} l(\hat{y}(\mathbf{x}^{(m)};\boldsymbol{\theta}), y^{(m)}) + \lambda \Omega(\boldsymbol{\theta})$$

- Loss function $l(\hat{y}(\mathbf{x}^{(m)};\boldsymbol{\theta}), y^{(m)})$, $\boldsymbol{\theta} = \{\mathbf{W}, \mathbf{b}\}$
- Regularizer $\lambda \Omega(\boldsymbol{\theta})$, L1 & L2 regularization

Optimization

- Stochastic gradient descent
- Layerwise pre-training & Backpropagation (BP)

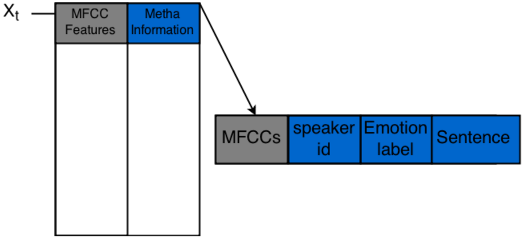
Table of Contents

Experiment Setup

EmoDB Database

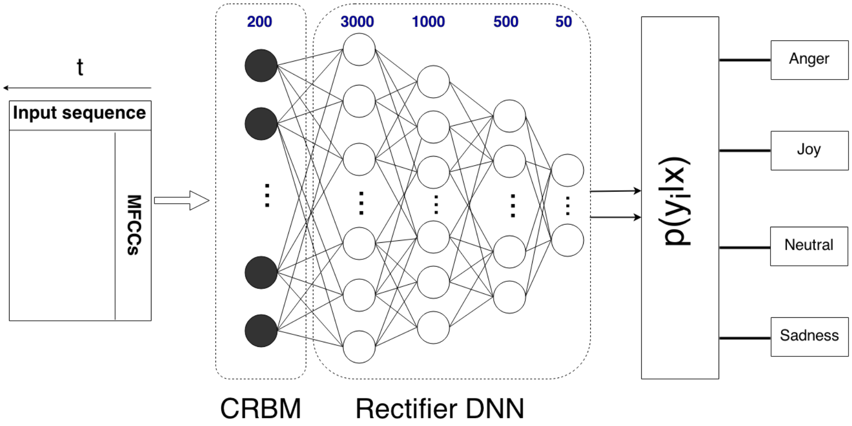
	Joy	Neutral	Sadness	Anger	Total
No. of sentences	71	79	62	127	339
Percent (%)	21	23.2	18.3	37.5	100

Data Structure



Experiment Result

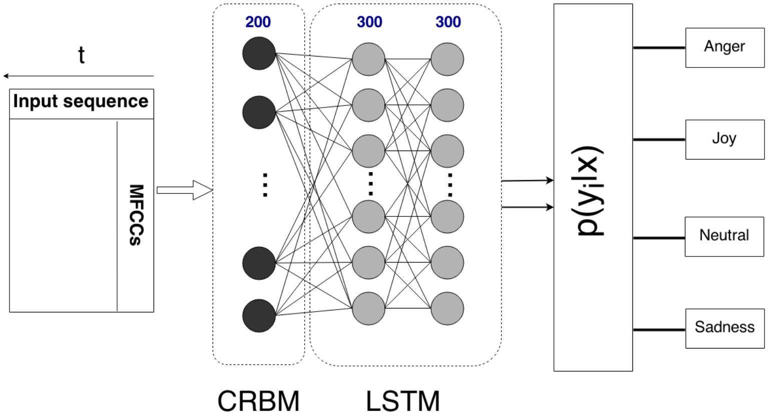
- CRBM-DNN



Confusion matrix of CRBM-DNN result.

		Classfied			
		Joy	Neutral	Sadness	Anger
	Joy	57.7%	1.4%	0.0%	40.8%
True	Neutral	17.7%	54.4%	25.3%	2.5%
	Sadness	1.6%	27.9%	70.5%	0.0%
	Anger	39.4%	1.6%	0.0%	59.1%
		recognition rate:59.76%			

- CRBM-LSTM


t
Input sequence
MFCCs
200
300
300
CRBM
LSTM
p(y_i|x)
Anger
Joy
Neutral
Sadness

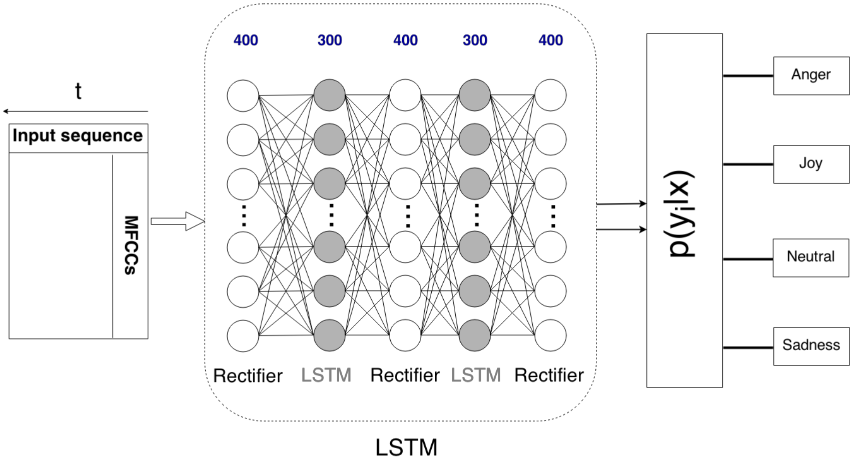
Experiment Result

Confusion matrix of CRBM-LSTM result.

		Classfied			
		Joy	Neutral	Sadness	Anger
	Joy	11.3%	9.9%	2.8%	76.1%
True	Neutral	0.0%	72.2%	17.7%	10.1%
	Sadness	0.0%	4.8%	88.7%	6.5%
	Anger	0.8%	1.6%	0.0%	97.6%
		recognition rate: 71.98%			

Experiment Result

- LSTM with rectifier units



Experiment Result

Confusion matrix of LSTM-Rectifier result.

		Classfied			
		Joy	Neutral	Sadness	Anger
	Joy	57.7%	7.0%	0.0%	35.2%
True	Neutral	6.3%	86.1%	6.3%	1.3%
	Sadness	0.0%	6.6%	93.4%	0.0%
	Anger	8.7%	0.0%	0.0%	91.3%
recognition rate: 83.43%					

Table of Contents

Conclusion

- Capturing long-term dependencies is necessary for speech emotion recognition
- CRBM-DNN is inappropriate for speech emotion recognition (ER: 40.24%)
- CRBM can capture non-temporal and temporal dependencies, but only short term

Conclusion

- Capturing long-term dependencies is necessary for speech emotion recognition
- CRBM-DNN is inappropriate for speech emotion recognition (ER: 40.24%)
- CRBM can capture non-temporal and temporal dependencies, but only short term

Model	Temporal Dependency	Memory	Generaltive
DNN	-	-	-
RBM	-	-	✓
CRBM	✓	2-5	✓
AE	-	-	-
RNN	✓	1-100	-
LSTM	✓	1-1000	-

Conclusion

- Capturing long-term dependencies is necessary for speech emotion recognition
- CRBM-DNN is inappropriate for speech emotion recognition (ER: 40.24%)
- CRBM can capture non-temporal and temporal dependencies, but only short term
- Frame-based classification can also reach good result
  - CRBM-LSTM 71.98%
  - LSTM with rectifier layers 83.43%
  - Sentence-based model SVM 84.26% (Tobias Gruber 2014)

Outlook

- Stacking CRBM to form deep belief network
- Second order optimization to speed up learning process, e.g. Newton methods
- Bi-directional LSTM

Thank You!

Deep Network for Speech Emotion Recognition

Master Thesis

Zhuowei Han

Institut für Signalverarbeitung
und Systemtheorie

Universität Stuttgart

16/04/2015