Introduction to Discriminative Training in Speech Recognition

Ralf Schlüter, Georg Heigold

Lehrstuhl für Informatik 6
Human Language Technology and Pattern Recognition
Computer Science Department, RWTH Aachen University
D-52056 Aachen, Germany

January 14, 2010

Contents

Outline

Motivation

Aim of discriminative methods: improve class separation

Motivation

Aim of discriminative methods: improve class separation

- standard maximum likelihood (ML) training: maximize reference class conditional $p_\theta(x|c)$

Motivation

Aim of discriminative methods: improve class separation

- standard maximum likelihood (ML) training: maximize reference class conditional $p_\theta(x|c)$
- maximum mutual information (MMI) training: maximize reference class posterior $p_\theta(c|x) = \dfrac{p(c) \cdot p_\theta(x|c)}{\sum\limits_{c'} p(c') \cdot p_\theta(x|c')}$

Motivation

Aim of discriminative methods: improve class separation

- standard maximum likelihood (ML) training: maximize reference class conditional $p_\theta(x|c)$
- maximum mutual information (MMI) training: maximize reference class posterior $p_\theta(c|x) = \dfrac{p(c) \cdot p_\theta(x|c)}{\sum\limits_{c'} p(c') \cdot p_\theta(x|c')}$

Where's the difference?

Motivation

Aim of discriminative methods: improve class separation

- standard maximum likelihood (ML) training: maximize reference class conditional $p_\theta(x|c)$
- maximum mutual information (MMI) training: maximize reference class posterior $p_\theta(c|x) = \dfrac{p(c) \cdot p_\theta(x|c)}{\sum\limits_{c'} p(c') \cdot p_\theta(x|c')}$

Where's the difference?

- Ideally: (almost) no difference! In case of infinite training data and correct model assumptions, the true probabilities are obtained in both cases. They lead to equal decisions, provided the class prior $p(c)$ is known. (Proof: model free optimization.)

Motivation

Aim of discriminative methods: improve class separation

- standard maximum likelihood (ML) training: maximize reference class conditional $p_\theta(x|c)$
- maximum mutual information (MMI) training: maximize reference class posterior $p_\theta(c|x) = \dfrac{p(c) \cdot p_\theta(x|c)}{\sum_{c'} p(c') \cdot p_\theta(x|c')}$

Where's the difference?

- Ideally: (almost) no difference! In case of infinite training data and correct model assumptions, the true probabilities are obtained in both cases. They lead to equal decisions, provided the class prior $p(c)$ is known. (Proof: model free optimization.)
- ML training: classes are handled independently, therefore decision boundaries are not considered explicitly in training.

Motivation

Aim of discriminative methods: improve class separation

- standard maximum likelihood (ML) training: maximize reference class conditional $p_\theta(x|c)$
- maximum mutual information (MMI) training: maximize reference class posterior $p_\theta(c|x) = \dfrac{p(c) \cdot p_\theta(x|c)}{\sum\limits_{c'} p(c') \cdot p_\theta(x|c')}$

Where's the difference?

- Ideally: (almost) no difference! In case of infinite training data and correct model assumptions, the true probabilities are obtained in both cases. They lead to equal decisions, provided the class prior $p(c)$ is known. (Proof: model free optimization.)
- ML training: classes are handled independently, therefore decision boundaries are not considered explicitly in training.
- in MMI training and generally in discriminative training, the reference class directly competes against all other classes, decision boundaries become relevant in training.

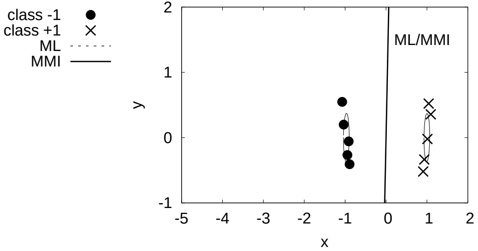
Motivation

- In practice, model assumptions are incorrect, and training data is limited. Here discriminative training can be beneficial.

Motivation

- In practice, model assumptions are incorrect, and training data is limited. Here discriminative training can be beneficial.

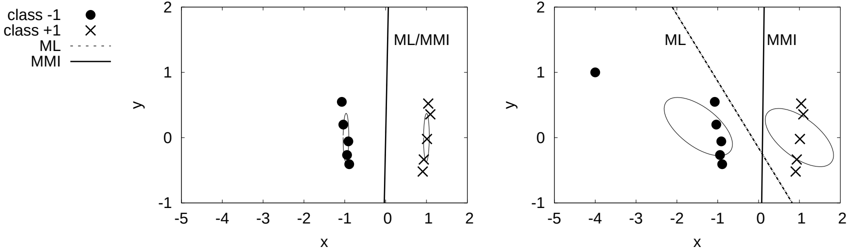
Example: a two class problem (with pooled covariance matrix)



Motivation

- In practice, model assumptions are incorrect, and training data is limited. Here discriminative training can be beneficial.

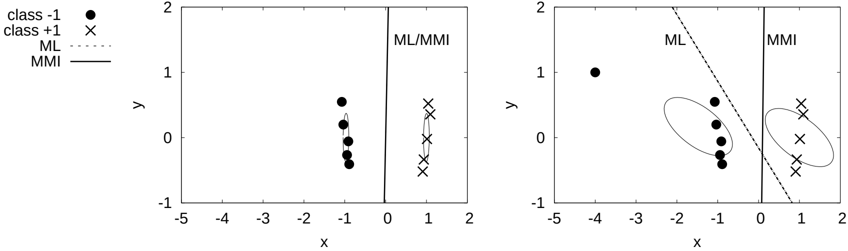
Example: a two class problem (with pooled covariance matrix)



Motivation

- In practice, model assumptions are incorrect, and training data is limited. Here discriminative training can be beneficial.

Example: a two class problem (with pooled covariance matrix)

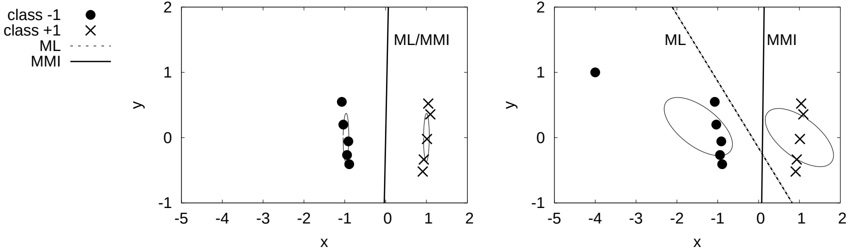


- Clearly, in case of ML training, the outlier deteriorates the decision boundary, whereas MMI training registers the minor importance of the outlier.

Motivation

- In practice, model assumptions are incorrect, and training data is limited. Here discriminative training can be beneficial.

Example: a two class problem (with pooled covariance matrix)



- Clearly, in case of ML training, the outlier deteriorates the decision boundary, whereas MMI training registers the minor importance of the outlier.
- MMI captures decision boundary, although model assumption does not fit in second case (pooled covariance).

Outline

Overview

Questions:

Overview

Questions:

- Which discriminative criterion to take?

Overview

Questions:

- Which discriminative criterion to take?
- Relation to decision rule and evaluation measure?

Overview

Questions:

- Which discriminative criterion to take?
- Relation to decision rule and evaluation measure?
- How to optimize criterion?

Overview

Questions:

- Which discriminative criterion to take?
- Relation to decision rule and evaluation measure?
- How to optimize criterion?
- Efficiency?

Overview

Questions:

- Which discriminative criterion to take?
- Relation to decision rule and evaluation measure?
- How to optimize criterion?
- Efficiency?
- Influence of modeling?

Overview

Questions:

- Which discriminative criterion to take?
- Relation to decision rule and evaluation measure?
- How to optimize criterion?
- Efficiency?
- Influence of modeling?
- Uniqueness of solution?

Overview

Questions:

- Which discriminative criterion to take?
- Relation to decision rule and evaluation measure?
- How to optimize criterion?
- Efficiency?
- Influence of modeling?
- Uniqueness of solution?
- Generalization?

Overview

Questions:

- Which discriminative criterion to take?
- Relation to decision rule and evaluation measure?
- How to optimize criterion?
- Efficiency?
- Influence of modeling?
- Uniqueness of solution?
- Generalization?

Bottomline:

Overview

Questions:

- Which discriminative criterion to take?
- Relation to decision rule and evaluation measure?
- How to optimize criterion?
- Efficiency?
- Influence of modeling?
- Uniqueness of solution?
- Generalization?

Bottomline:

- How to utilize available training material
  to obtain optimum recognition performance?

Outline

Notation

X_r	sequence $x_{r,1}, x_{r,2}, ..., x_{r,T_r}$ acoustic observation vectors
W_r	spoken word sequence $w_{r,1}, w_{r,2}, ..., w_{r,N_r}$ in training utterance r
W	any word sequence
$p(W)$	language model probability, supposed to be given
$p_\theta(X_r\|W)$	acoustic emission probability/acoustic model
θ	set of all parameters of the acoustic model
$\mathcal{M}_r$	set of competing word sequences to be considered
f	smoothing function

Outline

General Approach

Training

- input: training data and stochastic model $p_\theta(X, W)$
  with free model parameters θ
- output: "optimal" model parameters $\hat{\theta}$
- optimality defined via training criterion

$$\hat{\theta} \;\; := \;\; \arg\max_{\theta} \{F(\theta)\}$$

General Approach

Training

- input: training data and stochastic model $p_\theta(X, W)$
  with free model parameters θ
- output: "optimal" model parameters $\hat{\theta}$
- optimality defined via training criterion

$$\hat{\theta} \quad := \quad \arg\max_{\theta} \{F(\theta)\}$$

Unified training criterion [Macherey+ 2005]

$$F(\theta) \quad = \quad \sum_{r=1}^{R} f\left(\log\left(\frac{\sum_W p(W) p_\theta(X_r|W) \cdot A(W, W_r)}{\sum_{W \in \mathcal{M}_r} p(W) p_\theta(X_r|W)}\right)\right)$$

General Approach

Training

- input: training data and stochastic model $p_\theta(X, W)$
  with free model parameters θ
- output: "optimal" model parameters $\hat{\theta}$
- optimality defined via training criterion

$$\hat{\theta} \;:=\; \arg\max_{\theta} \{F(\theta)\}$$

Unified training criterion [Macherey$^+$ 2005]

$$F(\theta) \;=\; \sum_{r=1}^{R} f\left(\log\left(\frac{\sum_W p(W) p_\theta(X_r|W) \cdot A(W, W_r)}{\sum_{W \in \mathcal{M}_r} p(W) p_\theta(X_r|W)}\right)\right)$$

- covers maximum mutual information (MMI), minimum classification error (MCE), minimum phone/word error (MPE/MWE)

General Approach

Training

- input: training data and stochastic model $p_\theta(X, W)$
  with free model parameters θ
- output: "optimal" model parameters $\hat{\theta}$
- optimality defined via training criterion

$$\hat{\theta} \quad := \quad \arg\max_{\theta} \{F(\theta)\}$$

Unified training criterion [Macherey$^+$ 2005]

$$F(\theta) \quad = \quad \sum_{r=1}^{R} f\left(\log\left(\frac{\sum_W p(W) p_\theta(X_r|W) \cdot A(W, W_r)}{\sum_{W \in \mathcal{M}_r} p(W) p_\theta(X_r|W)}\right)\right)$$

- covers maximum mutual information (MMI), minimum classification error (MCE), minimum phone/word error (MPE/MWE)
- control set $\mathcal{M}_r$ of competing hypotheses, cost function, smoothing function, scaling of models (not shown)

Outline

Probabilistic Training Criteria

Objective

- find good estimate of probability distribution
- optimality regarding error via Bayes' decoding
  (asymptotic w.r.t. amount of training data)

Maximum Likelihood (ML)

- optimization of joint probability

$$\arg\max_{\theta} \sum_{r} \log\left(p(W_r)p_{\theta}(X|W_r)\right) = \arg\max_{\theta} \sum_{r} \log p_{\theta}(X_r|W_r)$$

- Tutorial on HMM [Rabiner 1989].
- Maximization of probability of reference word sequences (classes).
- Model correctness important.
- HMM: maximization for each class separately.
- Neglects competing classes.
- Expectation-maximiation: local convergence guaranteed.
- Estimation efficient, easily parallelizable.

Maximum Mutual Information (MMI)

- optimization of conditional probability

$$\arg\max_{\theta} \sum_r \log p_\theta(W_r|X_r) = \arg\max_{\theta} \sum_r \log \frac{p(W_r)p_\theta(X_r|W_r)}{\sum_V p(V)p_\theta(X_r|V)}$$

- Considers competing classes and therefore decision boundaries
- Necessitates set of competing classes on training data.
- Optimization for standard modeling (HMMs, mixture distributions): only gradient descent or similar.
- Optimization using log-linear modeling: convex problem
- First application of MMI for ASR using discrete HMMs [Bahl$^+$ 1986]:
  - 2000 isolated words, 18% rel. improvement in word error rate.
- MMI for discrete and continuous probability densities [Brown 1987]:
  - isolated E-set letters, 18% rel. improvement in recognition rate.
- MMI for discrete and continuous probabilty densities [Normandin 1991]:
  - digit strings, up to 50% rel. improvement in string error rate.

Outline

Error-Based Training Criteria

Objective: Optimize some error measure directly, e.g.:

- Empirical recognition error on training data
  - Advantage: direct relation to decision rule
  - Problem: non-differentiable training criterion, use of differentiable approximations in practice
  - Problem: ASR classes (words/word sequences) difficult to handle
- Model-based expected error on training data
  - Advantage: word or phoneme error easy to handle
  - Usually, approximated word/phoneme error, but correct edit distance also is viable [Heigold^{+} 2005]
  - Relation to decision rule less straight-forward.
  - Over-training and generalization becomes an issue ($\rightarrow$ regularization, margin)

Minimum classification error (MCE)

- For ASR: minimization of smoothed empirical sentence error [Juang & Katagiri 1992, Chou+ 1992].

$$\arg\min_{\theta} \frac{1}{R} \sum_{r=1}^{R} \frac{1}{1 + \left[\frac{p_\theta^\alpha(X_r|W_r) \cdot p^\alpha(W_r)}{\sum\limits_{W \neq W_r} p_\theta^\alpha(X_r|W) \cdot p^\alpha(W)} \right]^{2\varrho}}$$

- Smoothing parameters α and ϱ.
- Upper bound to Bayes' error rate for any acoustic model [Schlüter+ 2001]
- Lesser effect of incorrect model assumptions.

Minimum word/phone error (MWE/MPE)

- minimization of model-based expected word/phone error on training data [Povey & Woodland 2002]

$$\arg\max_{\theta} \sum_{r=1}^{R} \frac{\sum_W A(W, W_r) p(W) p_\theta(X_r|W)}{\sum_W p(W) p_\theta(X_r|W)}$$

- Criterion: *maximum* expected *accuracy* $A(W, W_r)$.
- Accuracy usually approximate, but exact case based on edit (*Levenshtein*) distance also possible [Heigold$^+$ 2005].
- Regularization (e.g. I-smoothing [Povey & Woodland 2002]) necessary due to overtraining.
- Usually better than MMI and MCE.

Outline

Practical Issues

- Importance of language model in training of acoustic model.
- Relative and absolute scaling of language and acoustic model in training.
- Necessity for recognition of training data.
- Efficient calculation of discriminative training statistics using word lattices.

Language Models for Discriminative Training

Potential Importance of Language Model Choice:

- language model for recognition of alternative word sequences
- language model dependence of discriminative training criterion itself
- interaction of language model of acoustic model parameters

Correlation hypothesis:

only those acoustic models need optimization, which even together with a language model do not sufficiently discriminate.
$\longrightarrow$ language model choice would correlate for training and recognition

Masking hypothesis:

language model usually largely improves recognition accuracy and might mask deficiencies of the acoustic models.
$\longrightarrow$ suboptimal language models for training would give better performance

Language Model for Discriminative Training

- Discriminative training includes language model.
- In training, unigram language model usually leads to the best word error rates [Schlüter^{+} 1999] (WSJ 5k):

language models		criterion	word error rates[%]			
recog	train		dev	eval	dev& eval	
bi	–	ML	6.91	6.78	6.86	
	zero	MMI	6.71	6.03	6.41	
	uni		6.59	6.00	6.33	-8%
	bi		6.71	6.20	6.48	
	tri		6.87	6.54	6.72	
tri	–	ML	4.82	4.11	4.51	
	zero	MMI	4.63	4.05	4.38	
	uni		4.30	3.64	4.01	-11%
	bi		4.48	3.94	4.24	
	tri		4.58	4.00	4.33	

Scaling of likelihoods

- recognition: absolute scaling of likelihoods irrelevant (language model scale vs. acoustic model scale)
- absolute scaling does have impact on word posterior calculation [Wessel$^+$ 1998, Woodland & Povey 2000]
- use language model scale β also in training:

$$p(X, W) = p(W)^{\beta} p_{\theta}(X|W)$$

- replace $p(X, W)$ with:

$$p(X, W)^{\gamma} = p(W)^{\beta\gamma} p_{\theta}(X|W)^{\gamma} \qquad \text{for} \quad \gamma \in [0, 1]$$

- optimum approx. for $\gamma = \frac{1}{\beta}$., i.e. use

$$p(X, W)^{\frac{1}{\beta}} = p(W) p_{\theta}(X|W)^{\frac{1}{\beta}}$$

- For simplicity here usually omitted in equations.

Competing Word Sequences

- Problem: Exponential number of competing word sequences.
- Competing word sequences need to be estimated:
  - Hypothesis-generation on training data using recognizer.
  - Initial lattice generation using recognizer sufficient.
  - Later acoustic model rescoring constrained to lattice.
- Representation and processing of competing word sequences.
  - Efficient algorithms to process word lattices.
  - Generic implementation: weighted finite state transducers.

Competing Word Sequences

History:

- best recognized word sequence for MMI (Corrective Training) [Normandin 1991]:
  - considers incorrectly recognized training sentences only
- best *incorrectly* recognized word sequence for MCE [Juang & Katagiri 1992]:
  - interpretation of smoothed sentence error still valid
- N-best recognized word sequences for MMI [Chow 1990]:
  - continuous speech recognition, 1000 words
  - only minor improvements in word error rate
- word graphs from recognition for MMI training [Valtchev^{+} 1997]:
  - large vocabulary, 64k words
  - efficient implementation
  - 5-10% relative improvement in word error rate

Outline

Comparative Experimental Results

Crit.	SieTill Test	WSJ 5k Dev	WSJ 5k Evl	EPPS English Dev06	EPPS English Evl06	EPPS English Evl07	Mandarin BN/BC Dev07	Mandarin BN/BC Evl06
	WER [%]							
ML	1.81	4.55	3.74	14.4	10.8	12.0	15.1	21.9
MMI	1.79	4.07	3.53	13.8	11.0	12.0	14.4	20.8
MCE	1.69	4.02	3.47	13.8	11.0	11.9		
MWE		3.98	3.44					
MPE		4.17	3.62	13.4	10.2	11.5	14.2	20.6

- SieTill [Schlüter 2000]
- WSJ 5k [Macherey 2010]
- EPPS/broadcasts [Heigold 2010]

Outline

Motivation

Goal: optimization method for discriminative training criteria $F(\theta)$ w.r.t. set of parameters θ which provides *reasonable* convergence.

- Various approaches, e.g.:
  - extended *Baum-Welch* (EBW) [Normandin 1991]
  - gradient descent, study: e.g. [Valtchev 1995]
  - MMI with log-linear models: generalized iterative scaling (GIS)
  - generalization of GIS to log-linear models with hidden variables and further criteria like MPE and MCE [Heigold$^+$ 2008a]
- Problems:
  - robust setting of step sizes/iteration constants (EBW and gradient descent),
  - convergence speed (especially GIS).

Extended Baum-Welch

- Motivated by a growth transformation [Gopalakrishnan+ 1991]
- Widely used for discriminative training of *Gaussian* mixture HMMs, e.g. [Normandin 1991, Valtchev+ 1997, Schlüter 2000, Woodland & Povey 2002]
- Highly optimized heuristics for finding right order of magnitude for iteration constants.
- Training of *Gaussian* mixture HMMs: require positive variances to obtain estimate for iteration constants.

Outline

Gradient descent

Follow gradient to optimize parameter:

$$\hat{\theta} = \theta + \gamma \nabla_\theta \mathcal{F}_\theta$$

Step sizes:

- heuristic, e.g. for MCE [Chou$^+$ 1992])
- by comparison to EBW [Schlüter 2000]

Convergence:

- local optimum
- better convergence: general purpose approaches, e.g. Qprop, Rprop, or L-BFGS, for experimental comparisons see [McDermott & Katagiri 2005, McDermott$^+$ 2007, Gunawardana$^+$ 2005, Mahajan$^+$ 2006]

Outline

Rprop [Riedmiller & Braun 1993]

General purpose gradient based optimization:

- assume iteration n
- parameter update:

$$\theta_i^{(n+1)} = \theta_i^{(n)} + \gamma_i^{(n)} \mathrm{sign}\left(\frac{\partial F(\theta^{(n)})}{\partial \theta_i}\right)$$

- update of step sizes $\gamma_i^{(n)}$:

$$\gamma_i^{(n+1)} = \begin{cases} \min\{\gamma_i^{(n)} \cdot \eta^+, \gamma_{max}\} & \mathit{if}\ \frac{\partial F(\theta^{(n)})}{\partial \theta_i} \cdot \frac{\partial F(\theta^{(n-1)})}{\partial \theta_i} > 0 \\ \max\{\gamma_i^{(n)} \cdot \eta^-, \gamma_{min}\} & \mathit{if}\ \frac{\partial F(\theta^{(n)})}{\partial \theta_i} \cdot \frac{\partial F(\theta^{(n-1)})}{\partial \theta_i} < 0 \\ \gamma_i^{(n)} & \mathit{otherwise} \end{cases}$$

- $\eta + \in (1, \infty)$, $\eta^- \in (0, 1)$

Outline

Formal gradient of MMI

- notation:
  - W: word sequence $w_1, \ldots, w_N$
  - r: index of training segment/utterance given by (X_r, W_r)
  - X_r: acoustic observation vector sequence $x_{r1}, \ldots, x_{rT}$
  - W_r: reference/spoken word sequence $w_{r1}, \ldots, w_{rN}$
  - s_1^T: HMM state sequence $s_1, \ldots, s_T$
- MMI training criterion:

$$F_{\text{MMI}}(\theta) = \sum_r \log \left(\frac{p(W_r) p_\theta(X_r|W_r)}{\sum_W p(W) p_\theta(X_r|W)} \right)$$

$$= \sum_r \left(\log p(W_r) p_\theta(X_r|W_r) - \log \sum_W p(W) p_\theta(X_r|W) \right)$$

- acoustic model (HMM):

$$p_\theta(X_r, W) = \sum_{s_1^{T_r}} \prod_{t=1}^{T_r} p(s_t|s_{t-1}) p_\theta(x_{rt}|s_t)$$

Derivative of MMI w.r.t. Parameter

Gradient of MMI criterion:

$$\nabla_\theta F_{\text{MMI}}(\theta) = \sum_r \left(\nabla_\theta \log p_\theta(X_r|W_r) - \frac{\sum_W p(W) p_\theta(X_r|W) \nabla_\theta \log p_\theta(X_r|W)}{\sum_{W'} p(W') p_\theta(X_r|W')} \right)$$

For efficient evaluation, consider derivative of acoustic model, $\nabla_\theta \log p_\theta(X_r|W)$.

Derivative of MMI w.r.t. Parameter

Derivative of acoustic model:

$$\nabla_\theta \log p_\theta(X_r, W) = \nabla_\theta \log \sum_{s_1^{T_r}:W} \prod_{t=1}^{T_r} p_\theta(x_{rt}|s_t) p(s_t|s_{t-1})$$

$$= \sum_{t=1}^{T_r} \frac{\sum_{s_1^{T_r}:W} \left(\nabla_\theta \log p_\theta(x_{rt}|s_t)\right) \cdot \prod_{t'=1}^{T_r} p_\theta(x_{rt'}|s_{t'}) p(s_{t'}|s_{t'-1})}{\sum_{\sigma_1^{T_r}:W} \prod_{\tau=1}^{T_r} p_\theta(x_{r\tau}|s_\tau) p(s_\tau|s_{\tau-1})}$$

$$= \sum_{t=1}^{T_r} \sum_s \left(\nabla_\theta \log p_\theta(x_{rt}|s)\right) \cdot \frac{\sum_{s_1^{T_r}:s_t=s} p_\theta(X_r, s_1^{T_r}|W)}{p_\theta(X_r|W)}$$

$$= \sum_{t=1}^{T_r} \sum_s \gamma_{rt}(s|W) \cdot \nabla_\theta \log p_\theta(x_{rt}|s)$$

with the word sequence conditioned state posterior (occupancy):

$$\gamma_{rt}(s|W) = \frac{\sum_{s_1^{T_r}:s_t=s} p_\theta(X_r, s_1^{T_r}|W)}{p_\theta(X_r|W)} = p_{\theta,t}(s|X_r, W)$$

Derivative of MMI w.r.t. Parameter

resubstitute derivative of acoustic model into derivative of MMI criterion:

$$\nabla_\theta F_{\text{MMI}}(\theta) = \sum_r \sum_{t=1}^{T_r} \sum_s \big(\nabla_\theta \log p_\theta(x_{rt}|s)\big) \cdot \left(\gamma_{rt}(s|W_r) - \frac{\sum_W p(W) p_\theta(X_r|W) \gamma_{rt}(s|W)}{\sum_{W'} p(W') p_\theta(X_r|W')}\right)$$

$$= \sum_r \sum_{t=1}^{T_r} \sum_s \big(\nabla_\theta \log p_\theta(x_{rt}|s)\big) \cdot \big(\gamma_{rt}(s|W_r) - \gamma_{rt}(s)\big)$$

with the general state posterior (occupancy):

$$\gamma_{rt}(s) = \frac{\sum_W p(W) p_\theta(X_r|W) \gamma_{rt}(s|W)}{\sum_{W'} p(W') p_\theta(X_r|W')} = p_{\theta,t}(s|X_r)$$

Efficient Calculation of State Occupancies

In general:

- efficient calculation of spoken word sequence conditional state occupancy $\gamma_{rt}(s|W_r)$: forward-backward state probabilities on trellis of word sequence
- efficient calculation of general state occupancy $\gamma_{rt}(s)$: forward-backward probabilities on trellis of word lattice

Viterbi approximation:

- $\gamma_{rt}(s|W) = \delta_{s,s_{rt}(W)}$ with forced alignment $S_r(W) = s_{r1}(W), \ldots, s_{rT_r}(W)$ of spoken word sequence
- assume a (word) lattice $\mathcal{M}_r$ for utterance r, with edges ω representing a word $w(\omega)$ (in context) with start time $t_s(\omega)$ and end time $t_e(\omega)$, and a corresponding forced alignment $s_{t_s}^{t_e}(\omega)$. An edge sequence $\mathcal{W} \in \mathcal{M}_r$ then corresponds to the word sequence $W(\mathcal{W})$. Consequently, the language model and acoustic model can also be defined for an edge sequence, which then might specify word boundaries, phonetic and language model context.

Word Posterior Probabilities

For the general state occupancy in Viterbi approximation we obtain:

$$\gamma_{rt}(s) = \frac{\sum_{W} p(W) p_\theta(X_r|W) \delta_{s,s_{rt}(W)}}{p_\theta(X_r)}$$

$$= \sum_{\omega} \delta_{s,s_{rt}(\omega)} \frac{\sum_{\mathcal{W}:\omega\in\mathcal{W}} p(\mathcal{W}) p_\theta(X_r|\mathcal{W})}{p_\theta(X_r)}$$

$$= \sum_{\omega} \delta_{s,s_{rt}(\omega)} p(\omega|X_r)$$

with the edge (or word in context) posterior

$$p(\omega|X_r) = \sum_{\mathcal{W}:\omega\in\mathcal{W}} \frac{p(\mathcal{W}) p_\theta(X_r|\mathcal{W})}{p_\theta(X_r)}$$

A forward-backward algorithm is used to efficiently compute edge (word in context) posterior probabilities using word lattices.

Outline

Formal gradient of MPE

- $A(W, W_r)$: accuracy (negated error) between string W and W_r
- example (MPE): approximate phone accuracy [Povey & Woodland 2002]
- expectation of accuracy:

$$E_\theta[A(\cdot, W_r)] := \sum_W A(W, W_r) \cdot \frac{p(W)p_\theta(X_r|W)}{\sum_{W'} p(W')p_\theta(X_r|W')}$$

- MPE training criterion:

$$F_{\mathrm{MPE}}(\theta) = \sum_r E_\theta[A(\cdot, W_r)]$$

Derivative of MPE w.r.t. Parameter

Derivative of MPE criterion:

$$\nabla_\theta p_\theta(X_r|W) = p_\theta(X_r|W) \cdot \big(\nabla_\theta \log p_\theta(X_r|W)\big)$$

$$\nabla_\theta F_{\text{MPE}}(\theta) = \sum_r \sum_W \big(A(W, W_r) - E_\theta[A(\cdot, W_r)]\big) \cdot \big(\nabla_\theta \log p_\theta(X_r|W)\big) \cdot \frac{p(W)p_\theta(X_r|W)}{\sum\limits_{W'} p(W')p_\theta(X_r|W')}$$

For efficient evaluation, consider derivative of acoustic model:

$$\nabla_\theta \log p_\theta(X_r|W) = \sum_{t=1}^{T_r} \sum_s \big(\nabla_\theta \log p_\theta(x_{rt}|s)\big) \cdot \frac{\sum_{s_1^{T_r}:s_t=s} p_\theta(X_r, s_1^{T_r}|W)}{p_\theta(X_r|W)}$$

Derivative of MPE w.r.t. Parameter

resubstitute derivative of acoustic model into derivative of MPE criterion:

$$\nabla_\theta F_{\text{MPE}}(\theta) = \sum_r \sum_{t=1}^{T_r} \sum_s \left(\nabla_\theta \log p_\theta(x_{rt}|s)\right) \cdot \tilde{\gamma}_{rt}(s)$$

with the general state accuracy:

$$\tilde{\gamma}_{rt}(s) = \sum_W \left(A(W, W_r) - E_\theta[A(\cdot, W_r)]\right) \cdot \frac{\sum\limits_{s_1^{T_r}: s_t = s} p(W) p_\theta(X_r, s_1^{T_r}|W)}{\sum\limits_{W'} p(W') p_\theta(X_r|W')}$$

which can be computed efficiently, similar to the case of general state occupancies.

Efficient Calculation of State Accuracy

In general:

- assumption: $A(W, W_r) = \sum_{t=1}^{T_r} A(s_{rt}(W), s_{rt}(W_r))$
- example: approximate phone accuracy [Povey & Woodland 2002]
- efficient calculation of general state accuracy $\tilde{\gamma}_{rt}(s)$: forward-backward accuracies on trellis of word lattice [Povey & Woodland 2002]

Word Posterior Accuracies

For the general state accuracy we in Viterbi approximation obtain:

$$\tilde{\gamma}_{rt}(s) = \frac{\sum_{W} \left(A(W, W_r - E_\theta[A(\cdot, W_r)]]\right) \cdot p(W) p_\theta(X_r|W) \delta_{s, s_{rt}(W)}}{p_\theta(X_r)}$$

$$= \sum_{\omega} \delta_{s, s_{rt}(\omega)} \frac{\sum_{W:\omega \in W} \left(A(W, W_r - E_\theta[A(\cdot, W_r)]]\right) \cdot p(W) p_\theta(X_r|W)}{p_\theta(X_r)}$$

$$= \sum_{\omega} \delta_{s, s_{rt}(\omega)} \tilde{p}(\omega|X_r)$$

with the edge (or word in context) posterior accuracies

$$\tilde{p}(\omega|X_r) = \sum_{W:\omega \in W} \frac{\left(A(W, W_r - E_\theta[A(\cdot, W_r)]]\right) \cdot p(W) p_\theta(X_r|W)}{p_\theta(X_r)}$$

Later, an efficient way of computing edge (word in context) posterior accuracies using word lattices will be presented.

Outline

Forward/Backward Probabilities on Word Lattices

- Let $\omega_s(\mathcal{W})$ and $\omega_e(\mathcal{W})$ be the first and last edge of a continuous edge sequence $\mathcal{W}$ on a word lattice.
- Assume that the lattice fully encodes the language model context:

$$p(W(\mathcal{W})) = p(\mathcal{W} = \omega_1^N) = \prod_{n=1}^{N} p(\omega_n | \omega_{n-1})$$

Let ω_{ri} and ω_{rf} be the initial and final edges of a word lattice for utterance r. Then define the following forward (Φ) and backward (Ψ) probabilities on initial and final partial edge sequences on the word lattice respectively:

$$\Phi(\omega) = \sum_{\substack{\mathcal{W}:\omega_s(\mathcal{W})=\omega_{ri} \\ \omega_e(\mathcal{W})=\omega}} p(\mathcal{W}) p_\theta({x_r}_1^{t_e(\mathcal{W})} | \mathcal{W})$$

$$\Psi(\omega) = \sum_{\substack{\mathcal{W}:\omega_s(\mathcal{W})=\omega \\ \omega_e(\mathcal{W})=\omega_{rf}}} p(\mathcal{W}) p_\theta({x_r}_{t_s(\mathcal{W})}^{T_r} | \mathcal{W})$$

Forward/Backward Probabilities on Word Lattices

For the forward probability a recursion formulae can be derived by separating the last edge from the edge sequence in the summation and $\prec$ denoting direct predecessor edges:

$$
\begin{aligned}
\Phi(\omega) &= \sum_{\substack{\mathcal{W}:\omega_s(\mathcal{W})=\omega_{ri}\\ \omega_e(\mathcal{W})=\omega}} p(\mathcal{W}) p_\theta(x_{r}{}_1^{t_e(\mathcal{W})}|\mathcal{W}) \\
&= \sum_{\omega'\prec\omega} \sum_{\substack{\mathcal{W}':\omega_s(\mathcal{W}')=\omega_{ri}\\ \omega_e(\mathcal{W}')=\omega'}} p(\mathcal{W}') p(\omega|\omega') p_\theta(x_{r}{}_1^{t_e(\mathcal{W}')}|\mathcal{W}') p_\theta(x_{r}{}_{t_s(\omega)}^{t_e(\omega)}|\omega) \\
&= \sum_{\omega'\prec\omega} \Phi(\omega') p(\omega|\omega') p_\theta(x_{r}{}_{t_s(\omega)}^{t_e(\omega)}|\omega).
\end{aligned}
$$

Using this recursion formula, the forward probabilities can be calculated efficiently on word lattices.

Forward/Backward Probabilities on Word Lattices

Similar to the forward probabilities, a recursion formula can be derived for efficient calculation of the backward probabilities and $\succ$ denoting direct successor edges:

$$
\begin{aligned}
\Psi(\omega) &= \sum_{\substack{\mathcal{W}:\omega_s(\mathcal{W})\succ\omega \\ \omega_e(\mathcal{W})=\omega_{rf}}} p(\mathcal{W}) p_\theta({x_r}_{t_s(\omega)}^{T_r}|\omega\mathcal{W}) \\
&= \sum_{\omega'\succ\omega} \sum_{\substack{\mathcal{W}':\omega_s(\mathcal{W}')\succ\omega' \\ \omega_e(\mathcal{W}')=\omega_{rf}}} p(\omega'|\omega) p(\mathcal{W}') p_\theta({x_r}_{t_s(\omega)}^{t_e(\omega)}|\omega) p_\theta({x_r}_{t_s(\omega')}^{T_r}|\omega'\mathcal{W}') \\
&= \sum_{\omega'\succ\omega} p_\theta({x_r}_{t_s(\omega)}^{t_e(\omega)}|\omega) p(\omega'|\omega) \Psi(\omega')
\end{aligned}
$$

Forward/Backward Probabilities on Word Lattices

Using the forward and backward probabilities, the edge/word posterior on a word lattice can be written as

$$p(\omega|X_r) = \frac{\Phi(\omega) \sum_{\omega' \succ \omega} p(\omega'|\omega)\Psi(\omega')}{\Phi(\omega_{rf})}$$

with $p_\theta(X_r) = \Phi(\omega_{rf}) = \Psi(\omega_{ri})$.

Word posterior probabilities follow naturally from MPE and similar discriminative training criteria. They also are the basis for confidence measures, which are used for unsupervised training, adaptation, or dialog systems. They are also part of approximate approaches to *Bayes*' decision rule with word error cost, like confusion networks [Mangu$^+$ 1999], or minimum frame word error [Wessel$^+$ 2001a, Hoffmeister$^+$ 2006].

Outline

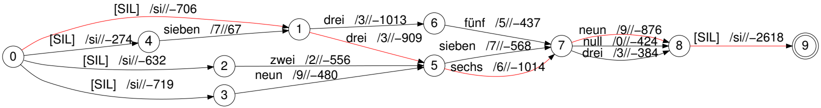
FB Probabilities: Generalization to WFSTs

Replace word lattice with WFST

- edge label: word with pronunciation
- weight of edge ω: $p \leftarrow p(\omega|\omega') \cdot p_\theta(x_{r\,t_s(\omega)}^{\;t_e(\omega)}|\omega)$
- semiring: substitute arithmetic operations (multiplication, addition, inversion) with operations of probability semiring

Semiring	$\mathbb{K}$	$p \oplus p'$	$p \otimes p'$	$\bar{0}$	$\bar{1}$	$\text{inv}(p)$
probability	$\mathbb{R}^+$	$p + p'$	$p \cdot p'$	0	1	$\frac{1}{p}$

Example WFST from SieTill, $\mathcal{W}_r =$"drei sechs neun" (in red)



FB Probabilities: Generalization to WFSTs

Forward probabilities ($pre(\omega) \in \mathcal{W}$ such that $pre(\omega) \prec \omega$)

$$\Phi(\omega) := \bigoplus_{\substack{\mathcal{W}:\omega_s(\mathcal{W})=\omega_{ri} \\ \omega_e(\mathcal{W})=\omega}} \bigotimes_{\omega \in \mathcal{W}} p(\omega|pre(\omega)) \otimes p_\theta(x_r{}_{t_s(\omega)}^{t_e(\omega)}|\omega)$$

$$= \bigoplus_{\omega' \prec \omega} \Phi(\omega') \otimes p(\omega|\omega') \otimes p_\theta(x_r{}_{t_s(\omega)}^{t_e(\omega)}|\omega)$$

Backward probabilities: similar

Using the forward and backward probabilities, the edge posterior on a WFST $\mathcal{X}_r$ can be written as

$$p(\omega|\mathcal{X}_r) = \Phi(\omega) \otimes \left(\bigoplus_{\omega' \succ \omega} p(\omega'|\omega) \otimes \Psi(\omega') \right) \otimes \mathrm{inv}(\Phi(\omega_{rf}))$$

Expectation Semiring

vector weight (p, v) of edge ω with

- $p \leftarrow p(\omega|\omega') \cdot p_\theta(x_r{}_{t_s(\omega)}^{t_e(\omega)}|\omega)$
- $v \leftarrow A(\omega) \cdot p$
  - accuracy of edge ω such that $\bigotimes_{\omega \in \mathcal{W}} A(\omega) = A(\mathcal{W}, \mathcal{W}_r)$
  - approximate phone accuracy [Povey & Woodland 2002] can be decomposed in this way
  - such a decomposition not possible in general

expectation semiring [Eisner 2001]:
vector semiring whose first component is a probability semiring

Semiring	$\mathbb{K}$	$(p,v) \oplus (p',v')$	$(p,v) \otimes (p',v')$	$\bar{0}$	$\bar{1}$	$\mathrm{inv}(p,v)$
expectation	$\mathbb{R}^+ \times \mathbb{R}$	$(p+p', v+v')$	$(p \cdot p', p \cdot v' + p' \cdot v)$	$(0,0)$	$(1,0)$	$\left(\frac{1}{p}, -\frac{v}{p^2}\right)$

Edge Posteriors & Expectation Semiring

probability semiring

- word posterior probabilities (see MMI derivative) identical to edge posteriors using probability semiring

$$p(\omega|X_r) = p_{\text{probability}}(\omega|\mathcal{X}_r)$$

- intuitive and classical result [Rabiner 1989]

expectation semiring

- word posterior accuracies (see MPE derivative) identical to v-component of edge posteriors using expectation semiring [Heigold$^+$ 2008b]

$$\tilde{p}(\omega|X_r) = p_{\text{expectation},v}(\omega|\mathcal{X}_r)$$

- also use this identity to efficiently calculate
  - derivative of unified training criterion
  - covariance between two random additive variables (related to MPE derivative)

Outline

Definition of Models

assume feature vector $x \in \mathbb{R}^D$ and class $c \in \{1, \dots, C\}$

Gaussian model $\mathcal{N}(x|\mu_c, \Sigma_c)$ with

- means $\mu c \in \mathbb{R}^D$
- positive-definite covariance matrices $\Sigma_c \in \mathbb{R}^{D \times D}$

induces posterior $p_\theta(c|x)$

$$\frac{p(c)\mathcal{N}(x|\mu_c, \Sigma_c)}{\sum_{c'} p(c')\mathcal{N}(x|\mu_{c'}, \Sigma_{c'})}$$

- include priors $p(c) \in \mathbb{R}^+$

Log-linear model with unconstrained parameters

- $\lambda_{c0} \in \mathbb{R}$
- $\lambda_{c1} \in \mathbb{R}^D$
- $\lambda_{c2} \in \mathbb{R}^{D \times D}$

$$\frac{\exp\left(x^\top \lambda_{c2} x + \lambda_{c1}^\top x + \lambda_{c0}\right)}{\sum_{c'} \exp\left(x^\top \lambda_{c'2} x + \lambda_{c'1}^\top x + \lambda_{c'0}\right)}$$

Transformation: Gaussian into Log-Linear Model

Comparison of terms quadratic, linear, and constant in observations x leads to the transformation rules [Saul & Lee 2002, Gunawardana$^+$ 2005]:

1.	λ_{c2}	$=$	$-\frac{1}{2}\Sigma_c^{-1}$
2.	λ_{c1}	$=$	$\Sigma_c^{-1}\mu_c$
3.	λ_{c0}	$=$	$-\frac{1}{2}\left(\mu_c^\top \Sigma_c^{-1}\mu_c + \log\lvert 2\pi\Sigma_c\rvert\right) + \log p(c)$

Outline

Transformation: Log-Linear into Gaussian Model

- invert transformation from Gaussian to log-linear model

1.	Σ_c	$=$	$-\frac{1}{2}\lambda_{c2}^{-1}$
2.	μ_c	$=$	$\Sigma_c^{-1}\lambda_{c1}$
3.	$p(c)$	$=$	$\exp\left(\lambda_{c0} + \frac{1}{2}\left(\mu_c^\top \Sigma_c^{-1}\mu_c + \log\|2\pi\Sigma_c\|\right)\right)$

- problem: parameter constraints not satisfied in general
  - covariance matrices Σ_c must be positive-definite
  - priors $p(c)$ must be normalized
- solution: model parameters for posterior are ambiguous
  e.g. for $\Delta\lambda_2 \in \mathbb{R}^{D\times D}, \Delta\lambda_0 \in \mathbb{R}$

$$\frac{\exp\left(x^\top(\lambda_{c2}+\Delta\lambda_2)x + \lambda_{c1}^\top x + (\lambda_{c0}+\Delta\lambda_0)\right)}{\sum\limits_{c'}\exp\left(x^\top(\lambda_{c'2}+\Delta\lambda_2)x + \lambda_{c'1}^\top x + (\lambda_{c'0}+\Delta\lambda_0)\right)}$$

$$= \frac{\exp\left(x^\top\lambda_{c2}x + \lambda_{c1}^\top x + \lambda_{c0}\right)}{\sum\limits_{c'}\exp\left(x^\top\lambda_{c'2}x + \lambda_{c'1}^\top x + \lambda_{c'0}\right)}$$

Transformation: Log-Linear into Gaussian Model

invert transformation rules for transformed log-linear model

1.	Σ_c	$=$	$-\frac{1}{2}(\lambda_{c2} + \Delta\lambda_2)^{-1}$
2.	μ_c	$=$	$\Sigma_c^{-1}\lambda_{c1}$
3.	$p(c)$	$=$	$\exp\left((\lambda_{c0} + \Delta\lambda_0) + \frac{1}{2}\left(\mu_c^\top \Sigma_c^{-1}\mu_c + \log\lvert 2\pi\Sigma_c\rvert\right)\right)$

use additional degrees of freedom to impose parameter constraints

- choose $\Delta\lambda_2 \in \mathbb{R}^{D\times D}$ such that $\lambda_{c2} + \Delta\lambda_2$ are negative-definite
- choose $\Delta\lambda_0$ such that $p(c)$ is normalized, i.e.,

$$\Delta\lambda_0 \quad := \quad -\log\sum_c \exp\left(\lambda_{c0} + \frac{1}{2}\left(\mu_c^\top \Sigma_c^{-1}\mu_c + \log\lvert 2\pi\Sigma_c\rvert\right)\right)$$

Outline

Motivation

Conventional approach:

- depends on initialization and choice of optimization algorithm
- spurious local optima (non-convex training criterion)
- many heuristics required
- i.e., involves much engineering work

"Fool-proof" approach:

- unique optimum (independent of initialization)
- accessibility of global optimum (convex training criterion)
- joint optimization of all model parameters, no parameters to be tuned

Outline

Assumptions

Assumptions to cast HCRF into CRF

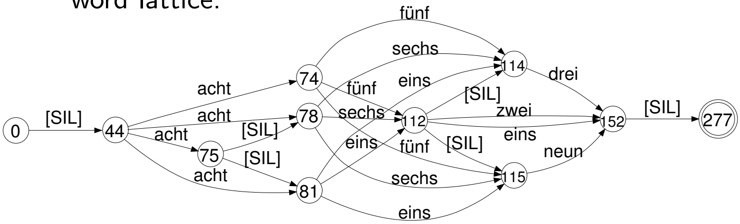
- log-linear parameterization, e.g.
  $p(x|s) = \exp\left(x^\top \lambda_{s2} x + \lambda_{s1}^\top x + \lambda_{s0}\right)$ and $p(s|s') = \exp(\alpha_{s's})$
- MMI-like training criterion
- alignment represents spoken sequence
- alignment of spoken sequence known and kept fixed
- use single densities with augmented features instead of mixtures
- exact normalization constant

Lattice-Based MMI

$$F_{lattice}(\lambda) = \sum_r \frac{\sum\limits_{s_1^{T_r} \in \mathcal{N}_r} p_\lambda(x_1^{T_r}, s_1^{T_r})}{\sum\limits_{s_1^{T_r} \in \mathcal{D}_r} p_\lambda(x_1^{T_r}, s_1^{T_r})}$$

- numerator word lattice $\mathcal{N}_r$: state sequences s_1^T representing correct hypothesis
- denominator word lattice $\mathcal{D}_r$: correct and competing state sequences, use word pair approximation and pruning
- non-convex

word lattice:

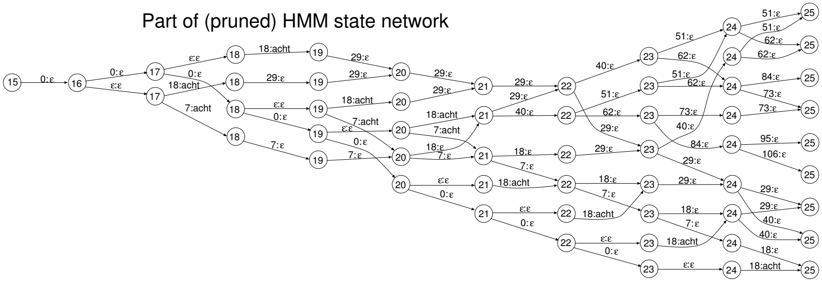


Fool-Proof MMI

$$F_{fool}(\lambda) = \sum_r \frac{p_\lambda(x_1^{T_r}, \hat{s}_1^{T_r})}{\sum\limits_{s_1^{T_r} \in \mathcal{S}_r} p_\lambda(x_1^{T_r}, s_1^{T_r})}$$

- consider only best state sequence $\hat{s}_1^T$ in numerator, kept fixed
- sum over full state sequence network in denominator
- convex

Part of (pruned) HMM state network



Frame-Based MMI

$$F_{frame}(\lambda) = \sum_r \sum_{t=1}^{T_r} \frac{p_\lambda(x_t, \hat{s}_t)}{\sum_{s=1}^{S} p_\lambda(x_t, s)}$$

- frame discrimination, cf. hybrid approach
- assume alignment for numerator s_1^T, kept fixed
- summation over all HMM states $s \in \{1, \ldots, S\}$ in denominator
- convex

Refinements to MMI

These refinements do not break convexity:

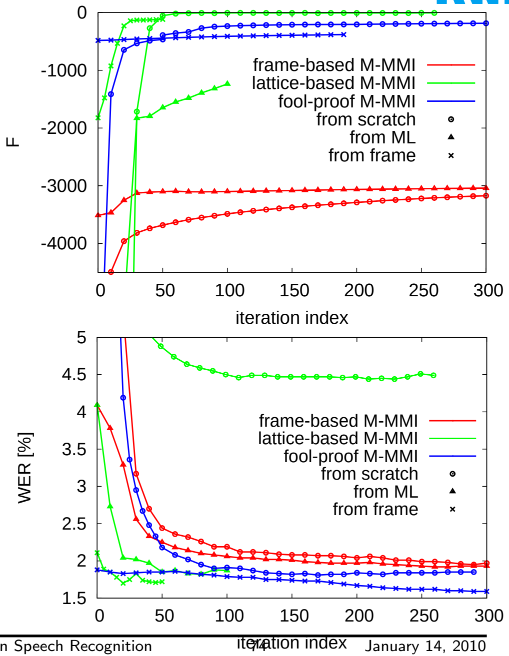
- ℓ_2-regularization
- margin term

Outline

Initialization

Analyze effect of initial parameters on training.

- vary initialization for different training criteria
- experiments: digit strings (SieTill, German, telephone)



Read Speech (WSJ)

- 5k-vocabulary, trigram language model
- phone-based HMMs, 1,500 CART-tied triphones
- audio data: 15h (training), 0.4h (test)

- log-linear model with kernel-like features $f(x)$
  - first ($f_d(x) = x_d$) and second ($f_{dd'}(x) = x_d \cdot x_{d'}$) order features
  - cluster features: assume GMM of marginal distribution, $p(x) = \sum_l p(x, l)$

$$f_l(x) = \begin{cases} p(l|x) & \text{if } p(l|x) \geq \textit{threshold} \\ 0 & \text{otherwise} \end{cases}$$

- starting from scratch (model) and linear segmentation
- frame-based MMI, with re-alignments
- details: [Wiesler$^+$ 2009]

Read Speech (WSJ)

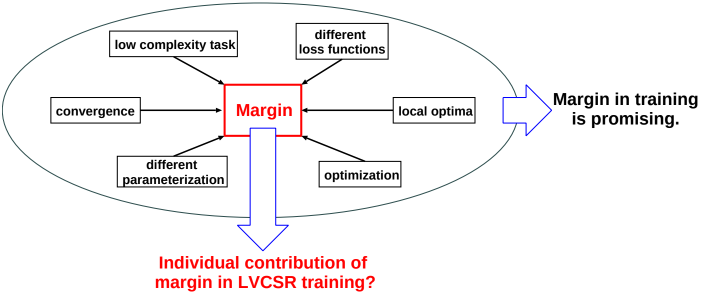
Feature setup	WER [%]
First order features, monophones	22.7
+second order features	10.3
$+2^{10}$ cluster features + temporal context of size 9	6.2
+1,500 CART-tied HMM states (triphones)	3.9
+realignment	3.6
GHMM (ML)	3.6
(MMI)	3.0

Outline

Motivation

Goal: incorporation of margin term into conventional training criteria

- replace likelihoods $p(W)p(X|W)$ with margin-likelihoods $p(W)p(X|W)\exp(-\rho A(W, W_r))$
- $A(W, W_r)$: accuracy between hypothesis W and reference W_r
- interpretation (boosting):
  emphasize incorrect hypotheses by up-weighting
- interpretation (large margin): next slides



Outline

Support Vector Machines (Hinge Loss)

Optimization problem for SVMs

$$SVM(\lambda) \;=\; -\frac{C}{2}\|\lambda\|^2 - \sum_{r=1}^{R} l(W_r, d_r; \rho)$$

- feature functions $f(X, W)$, model parameters λ
- distance $d_{rW} := \lambda^\top(f(X_r, W_r) - f(X_r, W))$
- hinge loss function $l^{\,(hinge)}(W_r, d_r; \rho) :=$ $\max_{W \neq W_r} \{\max\{-d_{rW} + \rho(A(W_r, W_r) - A(W, W_r)), 0\}\}$
- ℓ_2-regularization with constant $C > 0$
- [Altun+ 2003, Taskar+ 2003]

Outline

Smooth Approximation to SVM: Margin-MMI

Margin-based/modified MMI (M-MMI)

$$F_{\text{M-MMI},\gamma}(\lambda) = -\frac{C}{2}\|\lambda\|^2 + \sum_{r=1}^{R} \frac{1}{\gamma} \log\left(\frac{\exp(\gamma(\lambda^\top f(X_r, W_r) - \rho A(W_r, W_r)))}{\sum_W \exp(\gamma(\lambda^\top f(X_r, W) - \rho A(W, W_r)))}\right)$$

Lemma: $F_{\text{M-MMI},\gamma} \overset{\gamma\to\infty}{\to} \text{SVM}^{hinge}$ (pointwise convergence).

- [Heigold$^+$ 2008b]

Proof

$$\Delta A(W, W_r) := A(W_r, W_r) - A(W, W_r)$$

$$
\begin{aligned}
& -\frac{1}{\gamma} \log \left(\frac{\exp(\gamma(\lambda^\top f(X_r, W_r) - \rho A(W_r, W_r)))}{\sum_W \exp(\gamma(\lambda^\top f(X_r, W) - \rho A(W, W_r)))} \right) \\
= \quad & \frac{1}{\gamma} \log \left(1 + \sum_{W \neq W_r} \exp(\gamma(-d_{rW} + \rho \Delta A(W, W_r))) \right) \\
\xrightarrow{\gamma \to \infty} & \begin{cases} \max\limits_{W \neq W_r} \{-d_{rW} + \rho \Delta A(W, W_r)\} & \text{if } \exists W \neq W_r : d_{rW} < \rho \Delta A(W, W_r) \\ 0 & \text{otherwise} \end{cases} \\
= \quad & \max_{W \neq W_r} \{\max\{-d_{rW} + \rho \Delta A(W, W_r), 0\}\} \\
=: \quad & l^{(hinge)}(W_r, d_r; \rho).
\end{aligned}
$$

Outline

Support Vector Machines (Margin Error)

Optimization problem for SVMs

$$SVM(\lambda) \quad = \quad -\frac{C}{2}\|\lambda\|^2 - \sum_{r=1}^{R} l(W_r, d_r; \rho)$$

- feature functions $f(X, W)$, model parameters λ
- distance $d_{rW} := \lambda^\top(f(X_r, W_r) - f(X_r, W))$
- margin error loss function
  $l^{(error)}(W_r, d_r; \rho) := E\big(A(\arg\min_W [d_{rW} + \rho A(W, W_r)], W_r)\big)$
- ℓ_2-regularization with constant $C > 0$
- [Heigold$^+$ 2008b]

Outline

Smooth Approximation to SVM: Margin-MPE

Margin-based/modified MPE (M-MPE)

$$F_{\text{M-MPE},\gamma}(\lambda) = -\frac{C}{2}\|\lambda\|^2 + \sum_{r=1}^{R}\sum_{W} E(W, W_r)\left(\frac{\exp(\gamma(\lambda^\top f(X_r, W_r) - \rho A(W_r, W_r)))}{\sum_V \exp(\gamma(\lambda^\top f(X_r, V) - \rho A(V, W_r)))}\right)$$

Lemma: $F_{\text{M-MPE},\gamma} \overset{\gamma\to\infty}{\to} \text{SVM}^{error}$.

- [Heigold$^+$ 2008b]

Outline

Experimental Evaluation of Margin

Digit strings (SieTill, German, telephone)

dns/state	feature orders	# param,	criterion	WER [%]
1	first	11k	ML	3.8
			MMI	2.9
			M-MMI	2.7
64	first	690k	ML	1.8
			MMI	1.8
			M-MMI	1.6
1	first, second, and third	1,409k	Frame	1.8
			MMI	1.7
			M-MMI	1.5

Experimental Evaluation of Margin

European parliament plenary sessions in English (EPPS) and Mandarin broadcasts

	WER [%]		
	EPPS En	Mandarin BN/BC	
Criterion	90h	230h	1500h
ML	12.0	21.9	17.9
MMI		20.8	
M-MMI		20.6	
MPE	11.5	20.6	16.5
M-MPE	11.3	20.3	16.3

Experimental Evaluation of Margin

Handwriting Recognition (IFN/ENIT)

- isolated town names, handwritten
- choose slice features to use 1D HMM
- details: see [Dreuw^{+} 2009]

Criterion	WER [%] abc-d	abd-c	acd-b	bcd-a	abcd-e
ML	7.8	8.7	7.8	8.7	16.8
MMI	7.4	8.2	7.6	8.4	16.4
M-MMI	6.1	6.8	6.1	7.0	15.4

Outline

Effective Discriminative Training

- Discriminative Criteria
  - fit decision rule: minimize training error
  - limit overfitting: include regularization and margin to exploit remaining degrees of freedom of the parameters

Effective Discriminative Training

- Discriminative Criteria
  - fit decision rule: minimize training error
  - limit overfitting: include regularization and margin to exploit remaining degrees of freedom of the parameters
- Optimization Methods
  - general purpose methods give robust estimates
  - in convex case gradient descent still faster than growth transform (GIS)

Effective Discriminative Training

- Discriminative Criteria
  - fit decision rule: minimize training error
  - limit overfitting: include regularization and margin to exploit remaining degrees of freedom of the parameters
- Optimization Methods
  - general purpose methods give robust estimates
  - in convex case gradient descent still faster than growth transform (GIS)
- Log-Linear Modeling
  - convex (w/o hidden variables)
  - covers *Gaussians* completely, w/o constraints on e.g. variance
  - opens modeling up to arbitrary features
  - initialization: from scratch or from *Gaussians*

Effective Discriminative Training

- Discriminative Criteria
  - fit decision rule: minimize training error
  - limit overfitting: include regularization and margin to exploit remaining degrees of freedom of the parameters
- Optimization Methods
  - general purpose methods give robust estimates
  - in convex case gradient descent still faster than growth transform (GIS)
- Log-Linear Modeling
  - convex (w/o hidden variables)
  - covers *Gaussians* completely, w/o constraints on e.g. variance
  - opens modeling up to arbitrary features
  - initialization: from scratch or from *Gaussians*
- Estimation of Statistics
  - efficiency: use word lattice to represent competing word sequences
  - implementation: generic approach using WFSTs, covers class of criteria

Thanks for your attention!

Outline

References

Y. Altun, I. Tsochantaridis, and T. Hofmann, "Hidden Markov support vector machines," in International Conference on Machine Learning (ICML) 2003, Washington, DC, USA, 2003.

L. R. Bahl, P. F. Brown, P. V. de Souza, R. L. Mercer. "Maximum Mutual Information Estimation of Hidden Markov Model Parameters for Speech Recognition," Proc. *1986 Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 1, pp. 49-52, Tokyo, Japan, May 1986.

P. F. Brown. *The Acoustic-Modeling Problem in Automatic Speech Recognition*, Ph.D. thesis, Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA, May 1987.

W. Chou, B.-H. Juang, and C.-H. Lee, "Segmental GPD training of HMM based speech recognizer," in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 1992, San Francisco, CA, USA, March 1992, pp. 473-476.

Y.-L. Chow: "Maximum Mutual Information Estimation of HMM Parameters for Continuous Speech Recognition using the N-best Algorithm," in*Proc. Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 701–704, Albuquerque, NM, April 1990.

P. Dreuw, G. Heigold, and H. Ney, "Confidence-based discriminative training for model adaptation in offline Arabic handwriting recognition," in International Conference on Document Analysis and Recognition (ICDAR), Barcelona, Spain, July 2009.

J. Eisner, "Expectation semirings: Flexible EM for finite-state transducers," in International Workshop on Finite-State Methods and Natural Language Processing (FSMNLP), Helsinki, Finland, August 2001.

P. S. Gopalakrishnan, D. Kanevsky, A. Nádas, D. Nahamoo. "An Inequality for Rational Functions with Applications to Some Statistical Estimation Problems," IEEE Transactions on Information Theory, Vol. 37, Nr. 1, pp. 107-113, January 1991.

A. Gunawardana, M. Mahajan, A. Acero, and J. Platt, “Hidden conditional random fields for phone classification,” in Interspeech, pp. 117 120, Lisbon, Portugal, Sept. 2005.

G. Heigold, W. Macherey, R. Schlüter, and H. Ney: "Minimum Exact Word Error Training," in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 186–190, San Juan, Puerto Rico, November 2005.

G. Heigold, T. Deselaers, R. Schlüter, H. Ney: “GIS-like Estimation of Log-Linear Models with Hidden Variables,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4045–4048, Las Vegas, NV, USA, April 2008.

G. Heigold, T. Deselaers, R. Schlüter, H. Ney, “Modified MMI/MPE: A direct evaluation of the margin in speech recognition,” in International Conference on Machine Learning (ICML), pp. 384-391, Helsinki, Finland, July 2008.

G. Heigold: *A Log-Linear Modeling Framework for Speech Recognition*, Doctoral Thesis to be submitted, RWTH Aachen University, Aachen, Germany, 2010.

B. Hoffmeister, T. Klein, R. Schlüter, and H. Ney: “Frame Based System Combination and a Comparison with Weighted ROVER and CNC,” in *Proc. Interspeech*, pages 537–540, Pittsburgh, PA, USA, September 2006.

B.-H. Juang and S. Katagiri, “Discriminative learning for minimum error classification,” IEEE Transactions on Signal Processing, vol. 40, no. 12, pp. 3043-3054, 1992.

D. Kanevsky: “Extended Baum Welch transformations for general functions,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 821-824, Montreal, Quebec, Canada, May 2004.

W. Macherey, L. Haferkamp, R. Schlüter, H. Ney: “Investigations on Error Minimizing Training Criteria for Discriminative Training in Automatic Speech Recognition,” in *Proc. European Conference on Speech Communication and Technology (Interspeech)*, Lisbon, September 2005.

W. Macherey, "Discriminative training and acoustic modeling for automatic speech recognition," Ph.D. thesis to be submitted, RWTH Aachen University, 2010.

M. Mahajan, A. Gunawardana, A. Acero: "Training algorithms for hidden conditional random fields," in *Proc IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Toulouse, France, May 2006.

L. Mangu, E. Brill, A. Stolcke: "Finding Consensus Among Words: Lattice-Based Word Error Minimization," *Proc. European Conference on Speech Communication and Technology (EUROSPEECH)*, pp. 495–498, Budapest, Hungary, Sept. 1999.

E. McDermott, S. Katagiri: "Minimum Classification Error for Large Scale Speech Recognition Tasks using Weighted Finite State Transducers," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Philadelphia, PA, USA, April 2005.

E. McDermott, T. Hazen, J.L. Roux, A. Nakamura, S. Katagiri: "Discriminative training for large vocabulary speech recognition using Minimum Classification Error," in *Proc. IEEE Transactions on Audio, Speech and Language Processing (ICASSP)*, Vol. 15, No. 1, pp. 203–223, April 2007.

Y. Normandin, "Hidden Markov Models, Maximum Mutual Information, and the Speech Recognition Problem," Ph.D. thesis, McGill University, Montreal, Canada, 1991.

D. Povey and P. C. Woodland, "Minimum phone error and I- smoothing for improved discriminative training," in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2002, Orlando, FL, May 2002, vol. 1, pp. 105–108.

L.R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in Proc. of the IEEE, vol. 77, no. 2, pp. 257–286, 1989.

M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The Rprop algorithm," in IEEE International Conference on Neural Networks (ICNN) 1993, San Francisco, CA, USA, 1993, pp. 586 591.

L. Saul and D. Lee, “Multiplicative updates for classification by mixture models,” in T.G. Dietterich, S. Becker, and Z. Ghahramani, editor, Advances in Neural Information Processing Systems (NIPS). MIT Press, 2002.

R. Schlüter, B. Müller, F. Wessel, and H. Ney: “Interdependence of Language Models and Discriminative Training,” in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, Vol. 1, pages 119–122, Keystone, CO, December 1999.

R. Schlüter: *Investigations on Discriminative Trainings Criteria*, Doctoral Thesis, RWTH Aachen University, Aachen, Germany, Sept. 2000.

R. Schlüter, H. Ney: "Model-based MCE Bound to the True Bayes' Error," *IEEE Signal Processing Letters*, Vol. 8, No. 5, pages 131–133, May 2001.

B. Taskar, C. Guestrin, and D. Koller, “Max-margin Markov networks,” in Advances in Neural Information Processing Systems (NIPS) 2003, 2003.

V. Valtchev: *Discriminative Methods in HMM-based Speech Recognition,* Ph.D. thesis, St. John's College, University of Cambridge, Cambridge, March 1995.

V. Valtchev, J. J. Odell, P. C. Woodland, S. J. Young. “MMIE Training of Large Vocabulary Recognition Systems,” Speech Communication, Vol. 22, No. 4, pp. 303-314, September 1997.

F. Wessel, K. Macherey, and R. Schlüter, “Using word probabilities as confidence measures,” in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 1998, Seattle, WA, USA, May 1998, pp. 225-228.

F. Wessel, R. Schlüter, and H. Ney: “Explicit Word Error Minimization using Word Hypothesis Posterior Probabilities,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 33–36, Salt Lake City, Utah, May 2001.

F. Wessel, R. Schlüter, H. Ney: "Explicit Word Error Minimization using Word Hypothesis Posterior Probabilities," Proc. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 33–36, Salt Lake City, Utah, May 2001.

S. Wiesler, et al., "Investigations on features for log-linear acoustic models in continuous speech recognition," in IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Merano, Italy, Dec. 2009.

P. C. Woodland and D. Povey, "Large scale discriminative training for speech recognition," in Automatic Speech Recognition (ASR) 2000, Paris, France, September 2000, pp. 7–16.

P.C. Woodland, D. Povey: "Large Scale Discriminative Training of Hidden Markov Models for Speech Recognition." *Computer Speech and Language*, Vol. 16, No. 1, pp. 2548, 2002.

Outline

Speech Tasks: Corpus Statistics & Setups

Identifier	Description	Train/Test [h]
SieTill	German digit strings	11/11 (Test)
EPPS En	English European Parliament plenary speech	92/2.9 (Evl07)
BNBC Cn 230h	Mandarin broadcasts	230/2.2 (Evl06)
BNBC Cn 1500h	Mandarin broadcasts	1,500/2.2 (Evl06)

Identifier	#States/#Dns	Features
SieTill	430/27k	25 LDA(MFCC)
EPPS En	4,500/830k	45 LDA(MFCC+voicing) +VTLN+SAT/CMLLR
BNBC Cn 230h	4,500/1,100k	45 LDA(MFCC)+3 tones +VTLN+SAT/CMLLR
BNBC Cn 1500h	4,500/1,200k	45 SAT/CMLLR(PLP+voicing +3 tones+32 NN)+VTLN

Outline

Handwriting Recognition Tasks

IFN/ENIT:

- isolated Tunisian town names
- 4 training folds + 1 additional fold for testing
- simple appearance-based image slice features
- each fold comprises approximately 500,000 frames

Corpus		#Observations [k] Towns	Frames
IFN/ENIT	a	6.5	452
	b	6.7	459
	c	6.5	452
	d	6.7	451
	e	6.0	404