Recent work on Discriminative Training

Dan Povey & Phil Woodland

Apr 24th 2003

Cambridge University Engineering Department

Discriminative Training

- Discriminative training is training HMM parameters not using ML ...
- ... but maximising some other criterion (e.g. MMI) which reflects goodness-of-recognition of train-data
- Recent work at Cambridge on discriminative training includes:
  - Work on implementing MMI for LVCSR (using lattices)
  - Minimum Phone Error (MPE)
  - Also (won't cover today but)
    - Adaptation, e.g. gender adaptation with discriminative training (MPE-MAP)
    - SAT for discriminative training (relates to MLLR)

Overview

- MPE objective function
- Typical results for MPE vs MMI vs ML
- Overview of implementation issues

Minimum Phone Error (MPE)

- Maximise the following function:

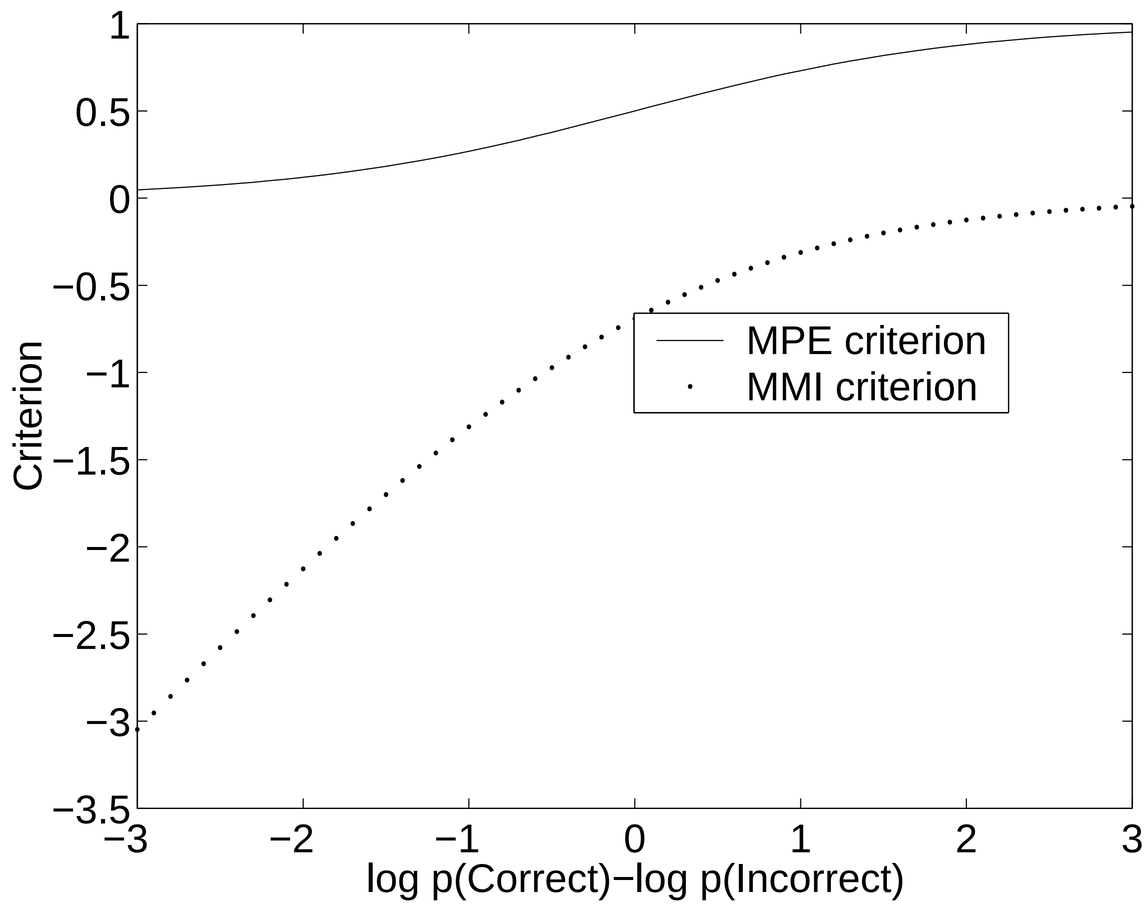
$$\mathcal{F}_{\mathrm{MPE}}(\lambda) = \sum_{r}^{R} \sum_{s} P_{\lambda}(s|\mathcal{O}_r)\mathrm{RawPhoneAccuracy}(s, s_r)$$

- i.e. an average of phone accuracy, weighted by sentence likelihood
- where $\mathrm{RawPhoneAccuracy}(s, s_r)$ is #phones in reference, minus #phone errors
- When maximising criterion, we try to increase likelihood of sentences which are more accurate than average

Maximum Mutual Information (MMI)

- $\mathcal{F}_{\mathrm{MMIE}}(\lambda) = \sum_{r=1}^{R} \log \frac{p_\lambda(\mathcal{O}_r|s_r)^\kappa P(s_r)^\kappa}{\sum_s p_\lambda(\mathcal{O}_r|s)^\kappa P(s)^\kappa}$
- Equals posterior probability of correct sentence given data & HMM

Comparison of objective functions (for 2 sentences)



Prior Information for robust parameter estimates

- Discriminative objective functions make it difficult to get robustly estimated model parameters (overtraining)

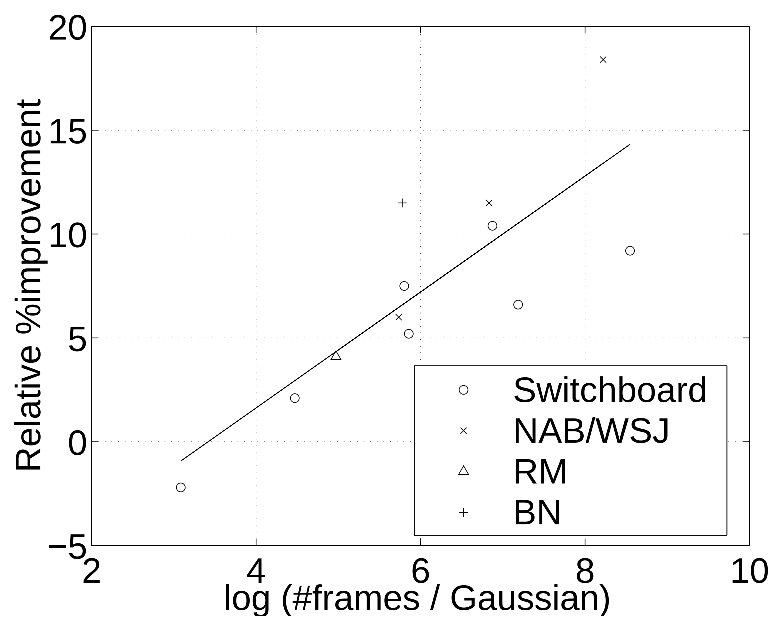
- This is especially true of MPE

- We use a technique we call I-smoothing, to back off parameters to the ML values where there is not enough training data for a Gaussian
  - Mathematically, I-smoothing is like MAP
  - We use a prior over the parameter values, center of prior is at ML estimate
  - In I-smoothing, evidence is discriminative objective function (in MAP, evidence is speaker-dependent ML objective function)

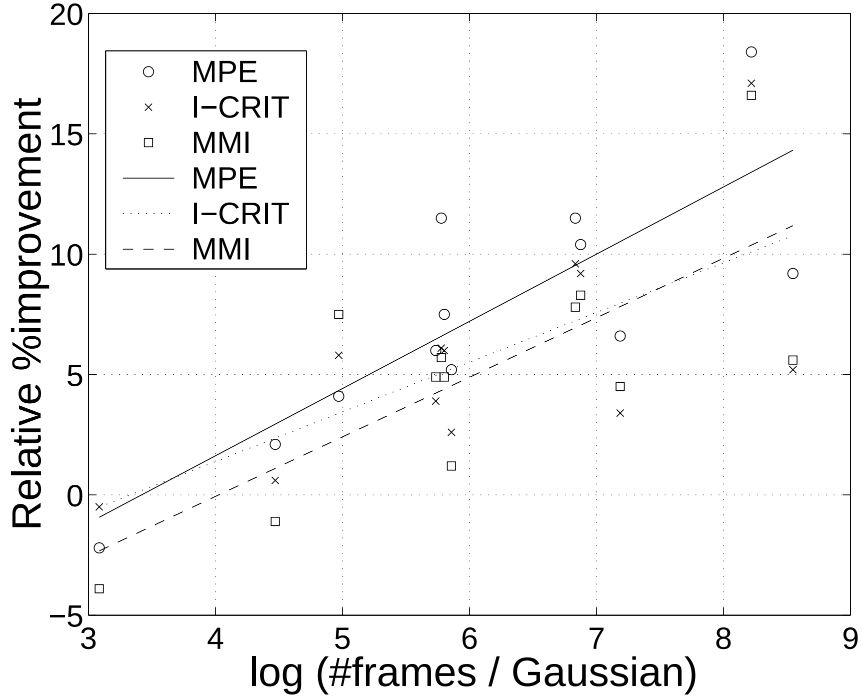
- Without I-smoothing, MPE is worse than MMI and gives only small improvement over ML

Improvement vs. ML

- Relative improvement of MPE vs ML, on various corpora (no MLLR)
- (with varying HMM set sizes and amounts of training data)
- Shows how improvement varies with ratio of train-data to # Gaussians in HMM set



Comparison of MPE with MMI, I-smoothed MMI



E.g. of MPE for an evaluation Switchboard system

- From 2002 NIST evaluation, tested on subset of 2001 development data
- Our system was the best

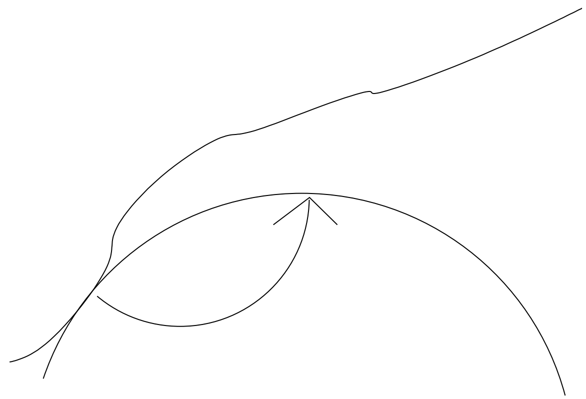
(%WERs)	ML	MPE	% Rel impr
No MLLR	33.3%	30.1%	9.6%
MLLR	30.7%	28.5%	5.3%

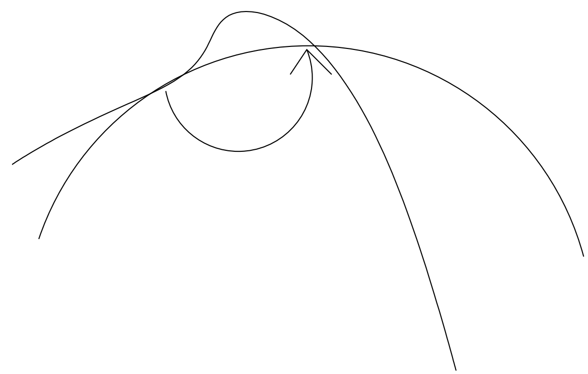
- This year we improved our system further, but were slightly beaten
- (although the winning result was a combination of results from two other sites)

Optimisation of MPE

- Optimised in a number of iterations; on each iteration, optimise an auxiliary function (as in ML)
- Uses a "weak-sense" auxiliary function (see next slide)
- To construct the auxiliary function, need differential of objective function w.r.t. data-likelihood of each phone in the lattice
- Need to find this differential without enumerating each possible sentence in the lattice
- This can be calculated efficiently using an algorithm similar to the forward-backward algorithm

Auxiliary functions





(a) (b)

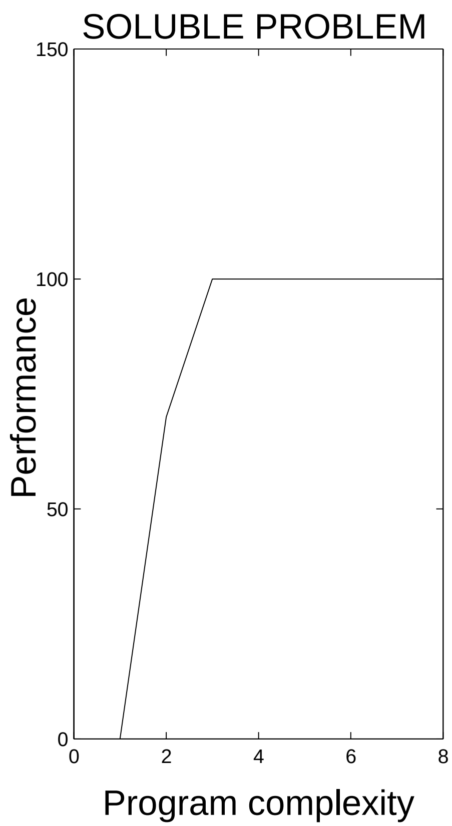
Use of (a) strong-sense and (b) weak-sense auxiliary functions for function optimisation

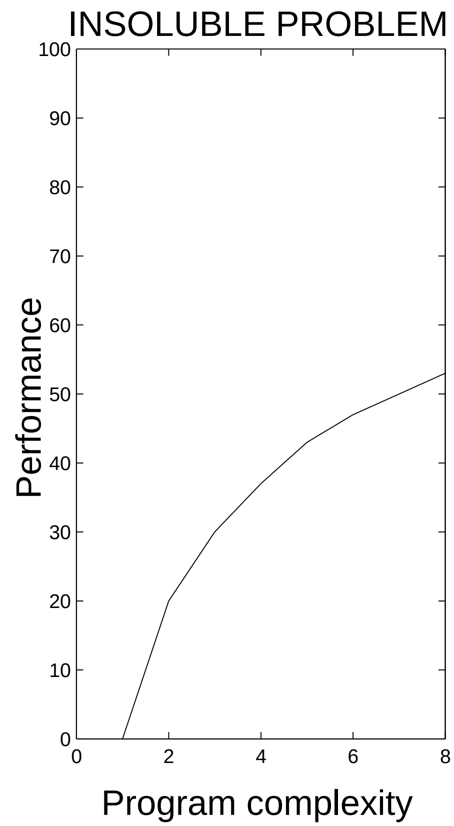
- Strong-sense auxiliary function: has the same value as real objective function at a local point $\lambda = \lambda'$, but $\leq$ objf everywhere else
- Weak-sense auxf has same differential around local point $\lambda = \lambda'$

On another topic...

- Interesting question:
- Should we be looking for complex or simple solutions to the speech recognition problem?
- Clearly simple is better if we have the choice, but ...
- Is there a “simple” solution?
- Traditional science expects simple solutions (e.g. physics)
- What if a problem has no simple solution?

“Soluble” vs “Insoluble” problems.





- Goodness of solution for the best solution with a particular description length, as f(description length)

“Soluble” vs “Insoluble” problems cont’d

- If this is right, we can’t find a “solution” that can be written in a few pages
- ... so what can we do (other than give up)?
- Some ideas:
  - Find convienient ways of creating and transmitting complex solutions to the problem
  - Find new representations of the solution (e.g. weird new programming language)
  - Swap code (and write programs so this is possible)
  - Use evolution (not maths research) as a model for how to solve the problem