Complementary System Combination and Generation for ASR

Mark Gales

20 June 2006

Cambridge University Engineering Department

Outline

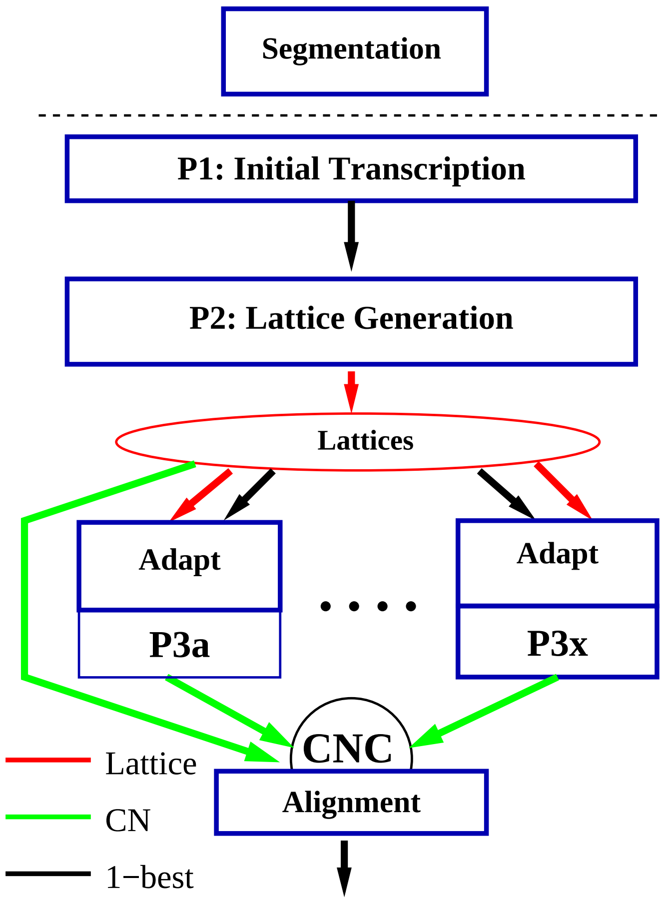
- LVCSR framework - minimum Bayes' risk training/decoding
- System Combination
  - "Implicit" System Combination
    Cross-system adaptation/ N-best or lattice rescoring
  - "Explicit" System Combination
    likelihood/hypothesis combination
- Complementary Systems
  - "random" selection
  - complementary system training
- Example LVCSR Systems
- Relationship to MT system combination

LVCSR Systems

- Most LVCSR systems have the same general framework
- Front-end:
  - Mel-warped PLP/MFCC feature vectors plus linear transformation/projection
  - Cepstral mean normalisation (possibly VTLN/variance-normalisation)
- Acoustic model:
  - hidden Markov model (HMM)-based
  - decision-tree state-clustered tri-phones
  - Gaussian mixture model state-output distributions
  - discriminative/minimum Bayes' risk training
- Language model:
  - tri-gram/4-gram word/class-based language model
- Acoustic model adaptation:
  - maximum likelihood linear regression (MLLR) [1]/constrained MLLR [2]

CU-HTK Multi-Pass/Combination Framework

- Multi-pass/combination framework used at CU for BN/CTS decoding [3, 4]



- P1 used to generate initial hypothesis
- P1 hypothesis used for rapid adaptation
  – LSLR, diagonal variance transforms
- P2: lattices generated for rescoring
  – apply complex LMs to trigram lattices
- P3 Adaptation/rescoring
  – unsupervised adaptation
  – lattice rescoring
- CN Decoding/Combination

Minimum Bayes Risk Training/Decoding

- Discriminative [5, 6]/MBR [7] training is commonly used in LVCSR systems
- MBR training may be expressed in terms of the expected loss [8, 7]

$$\hat{\mathcal{M}} = \arg\min_{\mathcal{M}} \left\{ \sum_{\mathcal{H}} P(\mathcal{H}|\boldsymbol{O}_{\tt trn}; \mathcal{M}) \mathcal{L}(\mathcal{H}, \mathcal{H}_{\tt ref}) \right\}$$

  where $\mathcal{L}(\mathcal{H}, \tilde{\mathcal{H}})$ is the the loss function of $\mathcal{H}$ against the reference $\mathcal{H}_{\tt ref}$

- MBR decoding framework may also be used [9]

$$\hat{\mathcal{H}} = \arg\min_{\tilde{\mathcal{H}}} \left\{ \sum_{\mathcal{H}} P(\mathcal{H}|\boldsymbol{O}; \mathcal{M}) \mathcal{L}(\mathcal{H}, \tilde{\mathcal{H}}) \right\}$$

  – $\tilde{\mathcal{H}}$ and $\mathcal{H}$ are normally selected from N-best list

Forms of ASR Loss Function

- MMI-like training/Viterbi decoding equate to a "sentence" level cost function:

$$\mathcal{L}(\mathcal{H},\tilde{\mathcal{H}}) = \begin{cases} 0, & \mathcal{H} = \tilde{\mathcal{H}} \\ 1, & \mathcal{H} \neq \tilde{\mathcal{H}} \end{cases}$$

- A number of loss functions have been examined
  - sentence level (1/0 loss function): MMI-like training [10]
  - word level: MWE suited to WER cost function [9]
  - phone level: MPE better generalisation than MWE training [11]
- Training schemes based on these have been implemented [11, 8]
- Word-level MBR decoding can be directly implemented using N-best lists [9]
  - limits possible results to one of the N-best
  - lattice-based word-level MBR decoding commonly used [12, 13]

Calculating the Loss Function

- Some loss functions (e.g. MWE) require aligning the two hypotheses

$$\mathcal{L}(\mathcal{H},\tilde{\mathcal{H}}) = \sum_{\mathbf{A}} \mathcal{L}(\mathcal{H},\tilde{\mathcal{H}}|\boldsymbol{a})P(\boldsymbol{a})$$

where $\boldsymbol{a}$ is a possible word-alignment between $\mathcal{H}$ and $\tilde{\mathcal{H}}$

- Given an alignment the loss calculation is trivial

REF:	BUT DIDN'T ELABORATE FURTHER
HYP:	IN IT DIDN'T ELABORATE
Loss:	3

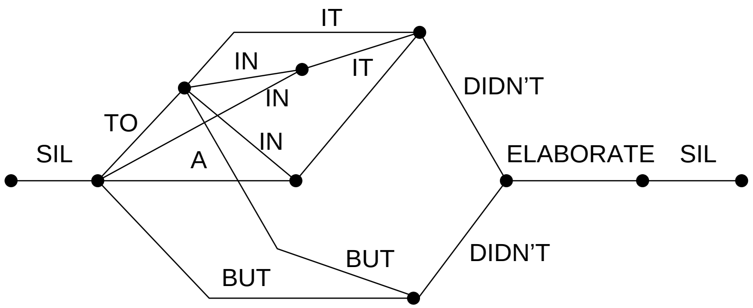
***	BUT	DIDN'T	ELABORATE	FURTHER
IN	IT	DIDN'T	ELABORATE	***
I	S			D

- Most techniques select a single alignment that minimises the loss, $\boldsymbol{a}_{\tt min}$

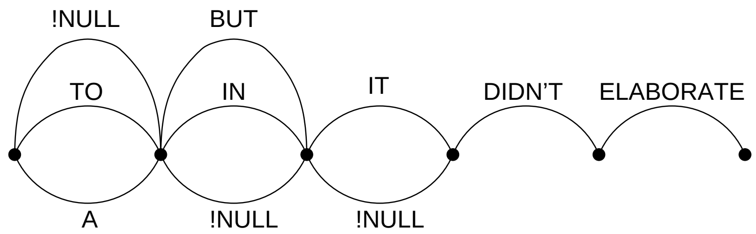
$$\mathcal{L}(\mathcal{H},\tilde{\mathcal{H}}|\boldsymbol{a}_{\tt min}) \leq \mathcal{L}(\mathcal{H},\tilde{\mathcal{H}})$$

Confusion Network Decoding

- Aligning N-best lists is simple, but limits possible hypotheses and gains
  - implicit word posteriors from hypothesis posteriors and N-best list
- Confusion networks (CNs)[12] use lattices and word-level confidences
  - use standard HMM decoder to generate word lattice;
  - iteratively align/merge links to form CN and obtain word posteriors



Word lattice



Confusion network

$$\hat{\mathcal{W}}^{(i)} = \arg\max_{\mathcal{W}^{(i)}} \left\{ P(\mathcal{W}^{(i)} | \boldsymbol{O}; \mathcal{M}) \right\}$$

  - allows hypothesis not in the original lattice (good and bad!)

System Combination

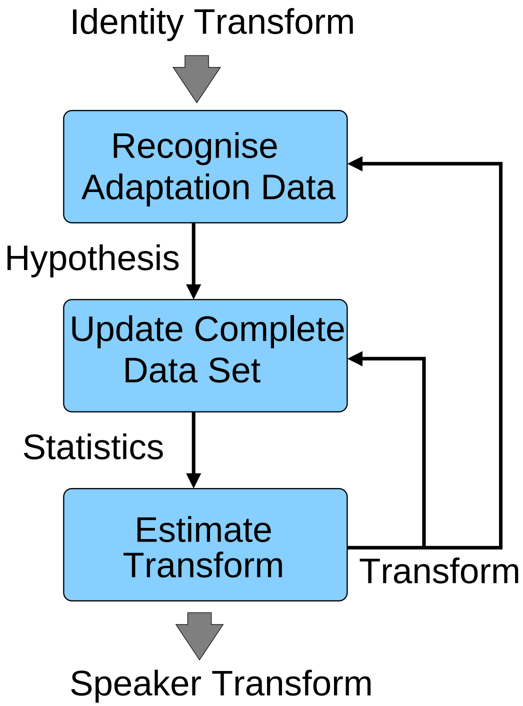
- For many tasks a single system cannot correctly classify all data
- System combination allows multiple systems to be used
  - rely on systems making different errors
- Two forms of system combination used
  - "implicit" combination - indirectly combine systems
  - "explicit" combination - directly combine scores from systems

"Implicit" System Combination

- Propagate information from one system to another
  - perform decoding/adaptation given the propagated information
- Two common forms of information propagation:
  - N-best/lattices for rescoring
  - 1-best hypothesis (and confidence scores) for adaptation
- N-best/lattice rescoring:
  - restricts search space - restricting possible errors from rescoring system
  - often done by "accident" ...
- Cross-adaptation used in many LVCSR systems
  - based on unsupervised adaptation

Unsupervised Adaptation

- An essential part of any LVCSR system is speaker/environment adaptation
  - for tasks like CTS and BN transcription unsupervised adaptation is required
- Approach to estimate MLLR [1] and CMLLR [2]

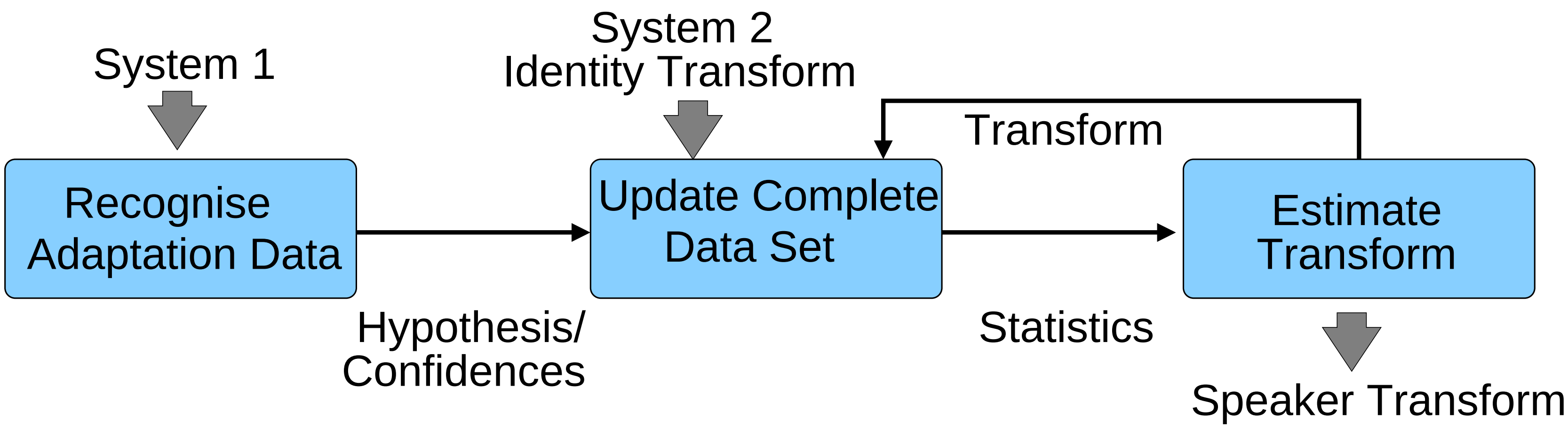


- Two iterative loops for estimation:
  1. estimate hypothesis given transform
  2. update complete-dataset given transform and hypothesis

  referred to as Iterative MLLR[14]
- For supervised training hypothesis is known
- Can also vary complexity of transform with iteration

Cross-System Adaptation

- Use hypothesis (and confidences) from a different system for adaptation
  - complexity of transform balances level of information propagated



- Generated speaker transform used in standard decoding framework
  - may be used in MBR decoding as well

"Explicit" System Combination

- MBR decoding for multiple systems can be expressed as

$$\hat{\mathcal{H}} = \arg\min_{\tilde{\mathcal{H}}} \left\{ \sum_{\mathcal{H}} P(\mathcal{H}|\boldsymbol{O}; \mathcal{M}^{(1)}, \ldots, \mathcal{M}^{(S)}) \mathcal{L}(\mathcal{H}, \tilde{\mathcal{H}}) \right\}$$

- Fundamental issue for system combination

Need to obtain an estimate of $P(\mathcal{H}|\boldsymbol{O}; \mathcal{M}^{(1)}, \ldots, \mathcal{M}^{(S)})$

  – since generative models used, Bayes' allows posterior to be obtained from

$$P(\mathcal{H}|\boldsymbol{O}; \mathcal{M}^{(1)}, \ldots, \mathcal{M}^{(S)}) = \frac{p(\boldsymbol{O}|\mathcal{H}; \mathcal{M}^{(1)}, \ldots, \mathcal{M}^{(S)}) P(\mathcal{H}; \mathcal{M})}{\sum_{\tilde{\mathcal{H}}} p(\boldsymbol{O}|\tilde{\mathcal{H}}; \mathcal{M}^{(1)}, \ldots, \mathcal{M}^{(S)}) P(\tilde{\mathcal{H}}; \mathcal{M})}$$

  – so either direct or likelihood combination posteriors may be used

Hypothesis/Score Combination

- Possible information available from the individual models:
  - posterior score: $P(\mathcal{H}|\boldsymbol{O};\mathcal{M}^{(s)})$
  - acoustic likelihood score: $p(\boldsymbol{O}|\mathcal{H};\mathcal{M}^{(s)})$
  - language model score: $P(\mathcal{H};\mathcal{M}^{(s)})$
  - classification result: $\mathcal{D}_s^{(\mathcal{H})}(\boldsymbol{O})$ (classifies sequence $\boldsymbol{O}$ as $\mathcal{H}$)

What "scores" should be combined?

How should the "scores" be combined?

- Two standard forms of score that are combined:
  - likelihood (or distribution parameter)/hypothesis posterior combination
- Two standard forms of combination approach:
  - (weighted) linear/log-linear combination

Likelihood Combination Schemes

- Mixture of Experts combination; standard approach

$$p(\boldsymbol{O}|\mathcal{H};\mathcal{M}^{(1)},\ldots,\mathcal{M}^{(S)}) \approx \sum_{s=1}^{S}\alpha_s p(\boldsymbol{O}|\mathcal{H};\mathcal{M}^{(s)})$$

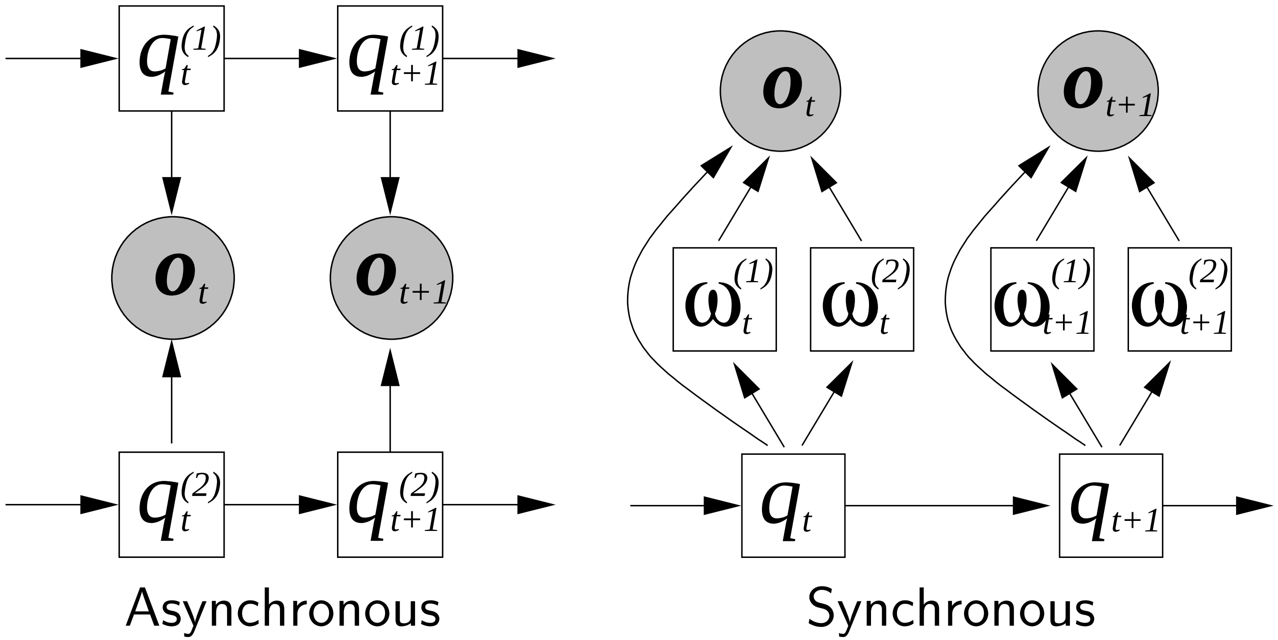
  – α_s are the component priors

- Product of Experts framework [15] may be expressed as

$$\begin{aligned} p(\boldsymbol{O}|\mathcal{H};\mathcal{M}^{(1)},\ldots,\mathcal{M}^{(S)}) &\approx \frac{1}{Z}\exp\left(\sum_{s=1}^{S}\alpha_s \log\left(p(\boldsymbol{O}|\mathcal{H};\mathcal{M}^{(s)})\right)\right) \\ &= \frac{1}{Z}\prod_{s=1}^{S} p(\boldsymbol{O}|\mathcal{H};\mathcal{M}^{(s)})^{\alpha_s} \end{aligned}$$

  – used for discriminative model combination [16, 17]
  – and (not very successfully) to products of Gaussians [18]

Synchronous vs Asynchronous Likelihood Combination



- In likelihood combination can either be synchronous or asynchronous:
  - asynchronous: systems have independent state processes:
    factorial HMMs [19], loosely coupled models [20], system combination [16]
  - synchronous: likelihoods combined at the state level
    single latent variable state-space, e.g. GMMs, PoGs [18]

Hypothesis Combination

- Linear hypothesis combination

$$P(\mathcal{H}|\boldsymbol{O};\mathcal{M}^{(1)},\dots,\mathcal{M}^{(S)}) \approx \sum_{s=1}^{S} \alpha_s P(\mathcal{H}|\boldsymbol{O};\mathcal{M}^{(s)})$$

  α_s is the "confidence" and satisfies the probability constraints.

  – used in CN combination

- Log-Linear hypothesis combination

$$P(\mathcal{H}|\boldsymbol{O};\mathcal{M}^{(1)},\dots,\mathcal{M}^{(S)}) \approx \frac{1}{Z}\exp\left(\sum_{s=1}^{S} \alpha_s \log\left(P(\mathcal{H}|\boldsymbol{O};\mathcal{M}^{(s)})\right)\right)$$

Likelihoods to Posteriors

- Many of the combination approaches require the hypothesis posterior (or a Confidence Measure), usually generative models used
  - directly applying Bayes' rule yields

$$P(\mathcal{H}|\boldsymbol{O};\mathcal{M}) = \frac{p(\boldsymbol{O}|\mathcal{H};\mathcal{M})P(\mathcal{H};\mathcal{M})}{\sum_{\tilde{\mathcal{H}}} p(\boldsymbol{O}|\tilde{\mathcal{H}};\mathcal{M})P(\tilde{\mathcal{H}};\mathcal{M})}$$

  - assumes models "correct" - tend to have "exaggerated" dynamic range
- Posterior estimates use acoustic deweighting

$$P(\mathcal{H}|\boldsymbol{O};\mathcal{M}) \approx \frac{p(\boldsymbol{O}|\mathcal{H};\mathcal{M})^{\lambda}P(\mathcal{H};\mathcal{M})}{\sum_{\tilde{\mathcal{H}}} p(\boldsymbol{O}|\tilde{\mathcal{H}};\mathcal{M})^{\lambda}P(\tilde{\mathcal{H}};\mathcal{M})}$$

  - λ set to around $1/$`grammar scale factor`.

Posterior Estimate Mapping

- Posteriors tend to be over-estimated, partly due to the lattice sizes
- Simple approaches to handle this are:
  - Decision tree mapping: using a held-out data set generate piecewise linear transformation from "posterior probabilities" to confidence scores [21]
  - Rank-based mapping: only the rank order is believed:

$$P(\mathcal{H}|\boldsymbol{O};\mathcal{M}) \approx \frac{1}{Z}\exp\left(-\alpha\mathtt{rank}(\mathcal{H}|\boldsymbol{O};\mathcal{M})\right)$$

    doesn't need scores - just rank ordering of hypotheses

- For some systems hard to get consistent posterior scores
  - just use 1-best output
  - systems use weighted voted, global system weights used

Consensus Decoding

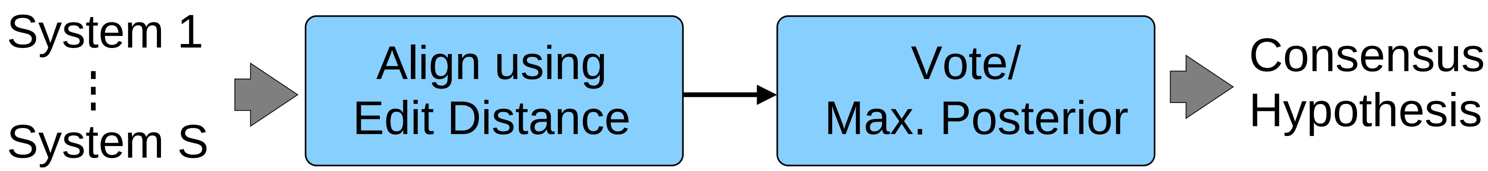
- Using the standard MBR decoding criterion for multiple systems

$$\hat{\mathcal{H}} = \arg\min_{\tilde{\mathcal{H}}} \left\{ \sum_{\mathcal{H}} P(\mathcal{H}|\boldsymbol{O};\mathcal{M}^{(1)},\ldots,\mathcal{M}^{(S)})\mathcal{L}(\mathcal{H},\tilde{\mathcal{H}}) \right\}$$

How to select the set of $\tilde{\mathcal{H}}$

  – using N-best list may be too restrictive

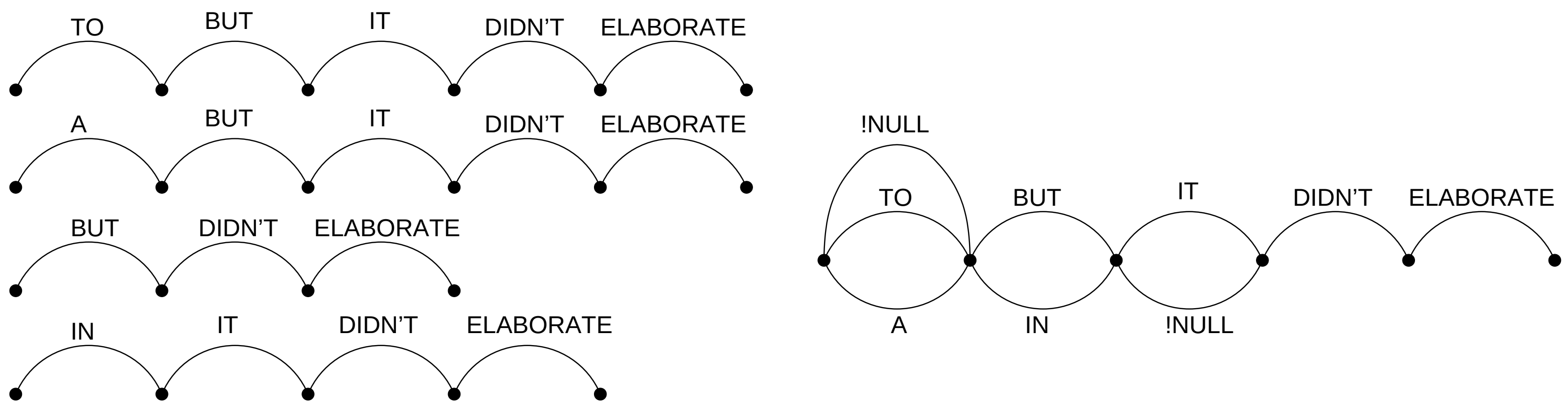
- Consensus decoding reduces this problem by using an alignment stage



- Two standard approaches to word-level hypothesis combination: ROVER[22], CN Combination[21]

ROVER

- ROVER takes the 1-best output from multiple recognition then:
  - convert outputs, $\mathcal{D}_s^{(\mathcal{H})}(\boldsymbol{O})$, into Word Transition Networks (WTNs)
  - align using edit distance and combine (WTNs) in a pre-specified order
  - use weighted voting to decide between aligned WTNs



Multiple System WTNs　　Aligned/Combined

- Output doesn't have to be in the original hypotheses:
  - `BUT IT DIDN'T ELABORATE`

ROVER Scores

- ASR systems commonly output a "confidence" score for each word
  - normally generated from recognition lattices
    other features may be used [23], e.g. LM score, N-best homogeneity
  - acoustic de-weighting again important
- The score for rover combination is usually of combination of frequence and confidence

$$P(\mathcal{W}_i|\boldsymbol{O};\mathcal{M}^{(1)},\ldots,\mathcal{M}^{(S)}) \approx \sum_{s=1}^{S}\left((1-\alpha)P(\mathcal{W}_i|\boldsymbol{O};\mathcal{M}^{(s)}) + \alpha\mathcal{D}_s^{(\mathcal{W}_i)}(\boldsymbol{O})/S\right)$$

There are two parameters to set

- α: the weighting between the frequency and confidence scores
- $P(!\texttt{NULL})$: confidence score associated with a NULL transition

Confusion Network Combination

- In contrast to ROVER, align and combine CN
  - use multiple hypothesis rather than 1-best
  - combined "posterior" found by

$$P(\mathcal{W}_i|\boldsymbol{O};\mathcal{M}^{(1)},\ldots,\mathcal{M}^{(S)}) \approx \sum_{s=1}^{S} \alpha_s P(\mathcal{W}_i|\boldsymbol{O};\mathcal{M}^{(s)})$$

  α_s can be used to represent the global confidence in system s

- CNC generally works slightly better than ROVER
  - multiple system word posteriors, rather than 1-best
  - but alignment more complex - not normally used with different segmentations

Complementary System Selection/Training

- When combining systems together would like systems that:
  - make different errors to each other
  - (normally) have approximately same error rate
- Approaches applied in ASR are:
  - “random” selection
  - complementary system training

Complementary System Selection ("Random")

- Variability to systems can be obtained by varying for example:
  - segmentation and clustering [3]
  - acoustic model decision tree [24]
  - acoustic model context (tri/quin-phone) [4]
  - speaker/environment adaptation (MLLR/CMLLR/lattice-based) [4]
  - dictionary/phone-set [4, 3, 25]
  - "bugs" etc. etc.
- Simple process (but computationally expensive!)
  - build set of systems using range of configurations
  - using development data see which systems combine best
- Used in the vast majority of ASR combination systems
  - cross-site combination best - combines many of the above

Complementary System Training

- Rather than "random" selection, how to build systems that are designed to be complementary?
- For likelihood combination schemes standard schemes available
  - mixture of experts - Expectation Maximisation (EM) [26]
  - product of experts - Generalised EM [18], or Contrastive Divergence [27]

  normally maximise likelihood, but can be applied to MBR training.
- For posterior/hypothesis combination Minimum Bayes Risk Leveraging tells us how to build complementary systems [28]

$$\hat{\mathcal{M}}^{(s+1)} = \arg\min_{\mathcal{M}} \left\{ \sum_{\mathcal{H}} P(\mathcal{H}|\boldsymbol{O}_{\texttt{trn}}; \mathcal{M}^{(1)}, \ldots, \mathcal{M}^{(s)}, \mathcal{M}) \mathcal{L}(\mathcal{H}, \mathcal{H}_{\texttt{ref}}) \right\}$$

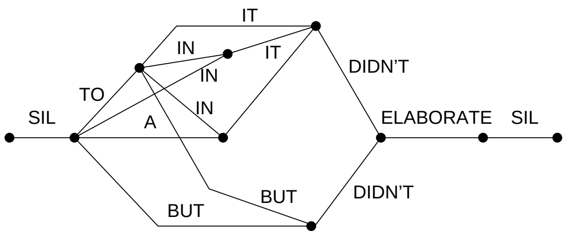
where the loss function $\mathcal{L}(\mathcal{H}, \mathcal{H}_{\texttt{ref}})$ is associated with scoring

Boosting

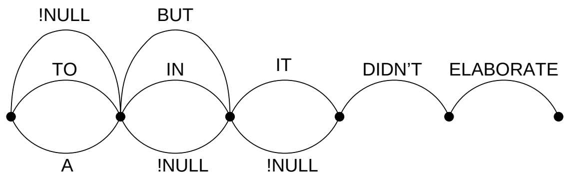
- Boosting [29] is a standard (and successful) machine-learning approach
  - build initial classifier
  - weight data depending on classification
  - train classifier using weighted data (and iterate)
  - classifiers combined using weighted voting
- Normally applied to static data
- For ASR need to determine at what level to perform boosting
  - frame-level [30]: simplest approach
  - phone-level [31]: requires alignments at phone-level
  - hypothesis-level [32]: approximations for decoding
- Combine with consensus decoding allows combination/training at various levels
  - loss-based alignment, e.g. at word level

Code-breaking Framework

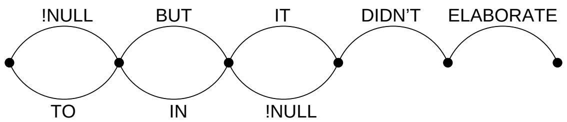
- Boosting normally applied to the same form of acoustic model
  - interesting to combine very different classifiers
- Build classifiers that resolve specific confusions given an initial system
  - the Code-Breaking framework [33]
  - version based on CNs described here [34]



Word lattice



Confusion Network



Pruned confusion network

- use standard HMM decoder to generate word lattice;
- generate confusion networks (CN) from word lattice and prune

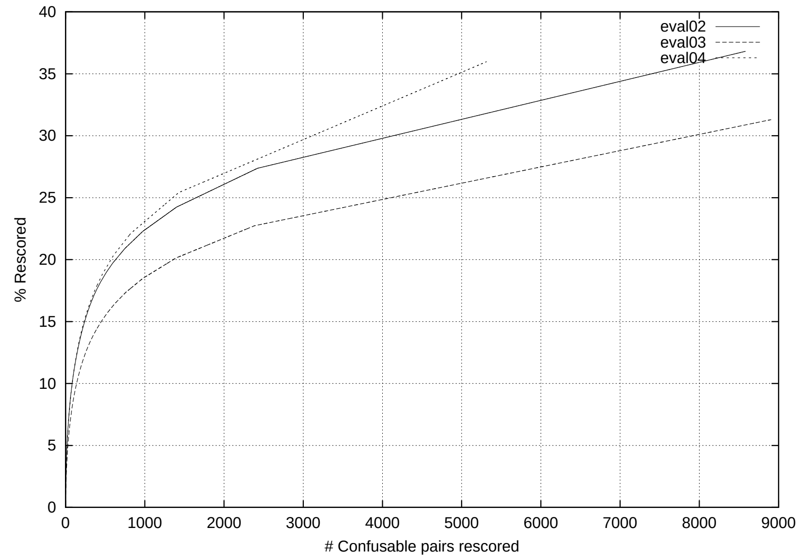
- Train classifiers to resolve specific binary confusions

Binary Classification using Support Vector Machines

- Wide range of discriminative classifiers for binary tasks
- Support Vector Machines (SVMs) [35] are a powerful classifier
  - dynamic kernels used to handle variable length speech data
  - generative kernels attractive form
    distance from decision boundary is a posterior ratio [34]
- Log-linear combination of CN posteriors and SVM posterior ratios

# SVMs	#corrected /#pairs	% corrected
10 SVMs	56/1250	4.5%

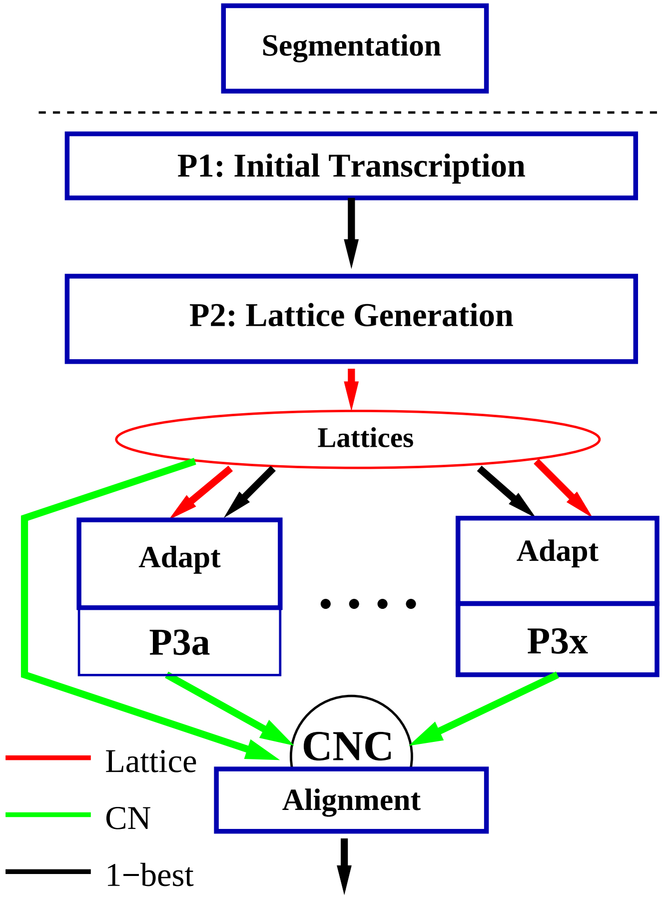
- performance on `eval03` CTS task
- only 1.6% of 76157 words rescored
- more SVMs required!



English BN/CTS Systems

- For full description of systems see[3, 4]
- Acoustic model training data:
  - BN - 1350 hours of data, 1200 hours closed caption transcriptions
  - CTS - 2300 hours of data, 2000 hours quick transcriptions
- Language model training data:
  - BN- 928MWords of text split into 5 language models and interpolated
  - CTS- 1,000MWords of text split into 6 language models and interpolated
- P3 Branch models:
  selected from a range of possible configurations
  - GD multiple pronunciation dictionary model (`P3b GD-MPron`)
  - GD single pronunciation dictionary model[36] (`P3c GD-SPron`)
  - quinphone SAT single pron. dictionary model (`P3e SAT-SPron-Quin`)

Acoustic Model Diversity



- P1 used to generate initial hypothesis
- P1 hypothesis used for rapid adaptation
  - LSLR, diagonal variance transforms
- P2: lattices generated for rescoring
  - apply complex LMs to trigram lattices
- P3 Adaptation
  - 1-best CMLLR
  - Lattice-based MLLR
  - Lattice-based full variance
- CN Decoding/Combination

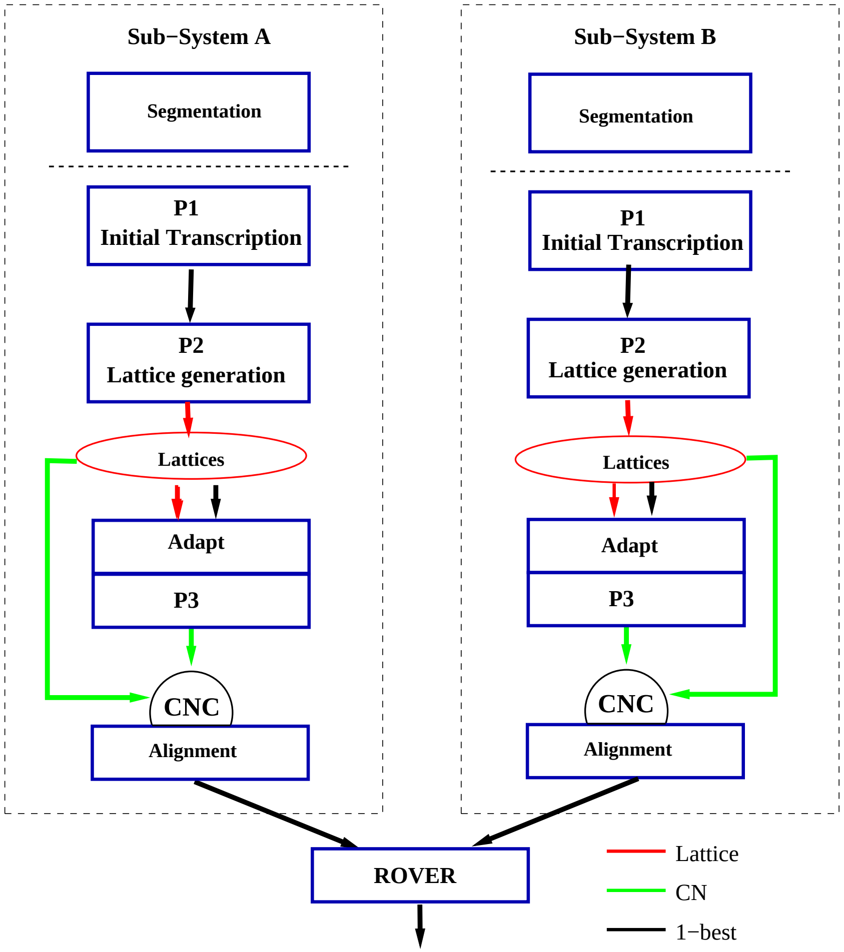
- Segmentation/P1-P2 branches runs in < 5xRT, full configuration < 10xRT.

Acoustic Model Diversity - CTS

System		WER(%) eval04
P2-cn	GD-MPron	19.1
P3b-cn	GD-MPron	18.1
P3e-cn	SAT-SPron-Quin	18.3
P3b+P3e	CNC	16.9

- System combination works well - very different models being combined
  - quinphone SAT single pronunciation and
  - a triphone GD multiple pronunciation system

Segmentation Diversity



- Different segmentations/clusterings
- Each subsystem
  - P1/P2 branches
  - P3c GD-SPron models
- P3 Adaptation
  - 1-best CMLLR
  - Lattice-based MLLR
  - Lattice-based full variance
- CN Decoding
- P2+P3c Combination within branch
- ROVER combination cross branch

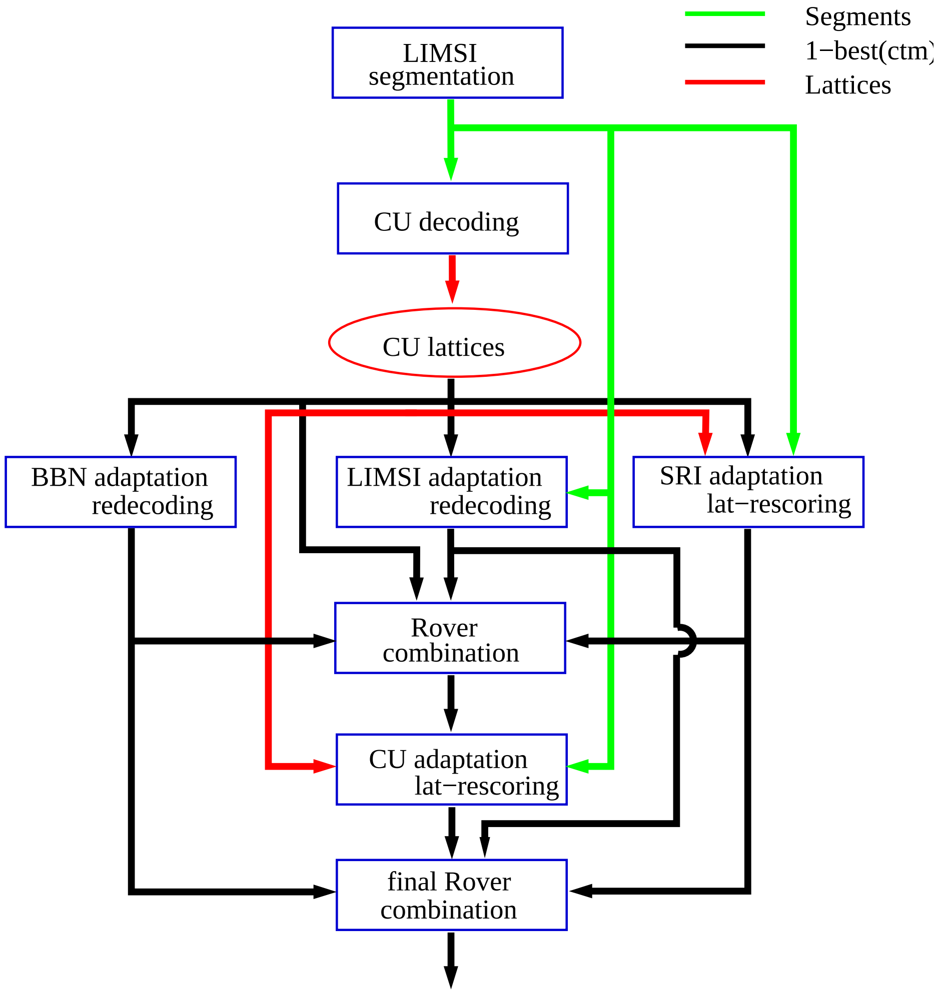
- Each branch runs in < 5xRT, full configuration < 10xRT.

Segmentation Diversity - BN

System	Segment/ Clustering		WER(%) eval04
L0+P3c	LIMSI	CNC	12.8
B0+P3c	BBN	CNC	13.0
C0+P3c	CU	CNC	13.3
L0+P3c $\oplus$ C0+P3c		ROVER	12.6
L0+P3c $\oplus$ B0+P3c		ROVER	12.4

- Three segmentations and clusterings: CU, BBN and LIMSI (thanks to BBN and LIMSI)
  - all segmentations/clusterings very different (CU deliberately very different)
- Diversity in segmentation gives gains in combination
  - combining BBN and LIMSI 0.5% better than using general framework
- Framework used for the RT04f BN-English EARS evaluation

Cross-Site Diversity - "SuperEARS"



- Initial pass using CU P1/P2 system
- BBN P3 branch (P3B)
  - use 1-best output for adaptation
  - decode using BBN segmentation
- LIMSI P3 branch (P3L)
  - P3B except LIMSI segmentation
- SRI P3 branch (P3S)
  - use 1-best output for adaptation
  - rescore CU lattices
- CU P4 branch (P4)
  - P2$\oplus$P3B$\oplus$P3L$\oplus$P3S adaptation
  - rescore CU lattices

Cross-Site Diversity - BN

System			WER(%) eval04
P2-cn	CU	MPron	13.6
P3B	BBN	decode	12.8
P3L	LIMSI	decode	14.0
P3S	SRI	rescore	14.6
P2⊕P3B⊕P3L⊕P3S		ROVER	12.2
P4	CU	SPron	12.8
P3B⊕P3L⊕P3S⊕P4		ROVER	11.6

- Further system description in[37], ran in < 10xRT.
- Complementary systems - built at different sites (BBN,LIMSI,SRI,CU)
  - 0.8% absolute better than using models from CU
  - works well - generally not that practical!

Relevance to Machine Translation

- Techniques based on MBR decoding/training applied to MT
  - MBR decoding using N-best lists applied to SMT [38]
  - minimum error rate training [39]
  - bilingual text alignment [40]
- Similar problems to ASR for system combination:
  - need systems that make different errors
  - consistent posterior scores for all systems useful
- Implicit combination using N-best lists straightforward
  - equivalent of cross-system adaptation??
- Diversity in systems
  - my impression is that no single (completely) dominating statistical model
  - "random" selection should work well!

Loss Functions/Consensus Decoding

- There are a number of evaluation criteria that have been used for MT
  - WER: alignment integral to scoring, efficient to compute using DP
  - PER: independent of alignment.
  - BLEU: alignment not part of scoring
  - TER: alignment integral to scoring.

  Note: for `BLEU` an alignment will minimise the loss function

- Consensus decoding has been applied to MT systems [41, 42]
- Alignment in MT systems is significantly more complex than in ASR systems
  - phrase/word re-ordering complicates the whole business
  - Could use edit distance [41], but doesn't allow re-ordering ...
  - use statistical alignment [42], but not tuned to loss function
- Not clear importance of tuning alignment precisely to evaluation loss-function

Conclusions

- System combination an important part of LVCSR systems
  - likelihood, posterior and decision combination all possible
- System combination using consensus decoding is used in most systems

alignment is central for system combination

- "Random" selection over space of models used to select systems to combine
- Complementary system training an on-going research area
  - alignment also important for complementary system training
- For text/audio translation many similar issues to ASR:
  - obtaining meaningful scores
  - how to get diversity into systems to combine (without going cross-site)
  - aligning hypotheses for combination/training

References

[1] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density HMMs," *Computer Speech and Language*, vol. 9, pp. 171–186, 1995.

[2] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[3] M. J. F. Gales, D. Kim, P. C. Woodland, H. Chan, D. Mrva, R. Sinha, and S. Tranter, "Progress in the CU-HTK Broadcast News transcription system," *IEEE Transactions Audio, Speech and Language Processing*, 2006, to appear.

[4] G. Evermann, H. Chan, M. J. F. Gales, B. Jia, X. Liu, D. Mrva, K. Sim, L. Wang, P. C. Woodland, and K. Yu, "Development of the 2004 CU-HTK English CTS systems using more than two thousand hours of data," in *Proc. Fall 2004 Rich Transcription Workshop (RT-04f)*, 2004.

[5] P. Gopalakrishnan, D. Kanevsky, A. Nádas, and D. Nahamoo, "An inequality for rational functions with applications to some statistical estimation problems," *IEEE Trans. Information Theory*, 1991.

[6] P. C. Woodland and D. Povey, "Large scale discriminative training of hidden Markov models for speech recognition," *Computer Speech & Language*, vol. 16, pp. 25–47, 2002.

[7] W. Byrne, "Minimum Bayes risk estimation and decoding in large vocabulary continuous speech recognition," *IEICE Special Issue on Statistical Modelling for Speech Recognition*, 2006.

[8] J. Kaiser, B. Horvat, and Z. Kacic, "A novel loss function for the overall risk criterion based discriminative training of HMM models," in *Proc. ICSLP*, 2000.

[9] A. Stolcke, E. Brill, and M. Weintraub, "Explicit word error minimization in N-Best list rescoring," in *Proc. Eur. Conf. Speech Commun. Technol.*, 1997.

[10] K. Na, B. Jeon, D. Chang, S. Chae, and S. Ann, "Discriminative training of hidden Markov models using overall risk criterion and reduced gradient method," in *Proceedings Eurospeech*, 1995, pp. 97–100.

[11] D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Orlando, FL, May 2002.

[12] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus among words: Lattice-based word error minimization," in *Proc. Eur. Conf. Speech Commun. Technol.*, 1999.

[13] V. Goel and W. Byrne, "Task dependent loss functions in speech recognition: A* search over recognition lattices," in *Proc. Eur. Conf. Speech Commun. Technol.*, 1999.

[14] P. C. Woodland, D. Pye, and M. J. F. Gales, "Iterative unsupervised adaptation using maximum likelihood linear regression," in *Proc. Int. Conf. Spoken Lang. Process.*, Philadelphia, 1996, pp. 1133–1136.

[15] G. Hinton, "Products of experts," in *Proceeding of ICANN*, 1999.

[16] P. Beyerlein, "Discriminative model combination," in *Proc. ASRU*, 1997.

[17] P. Beyerlein, X. L. Aubert, R. Haeb-Umbach, M. Harris, D. Klakow, A. Wendemuth, S. Loau, M. Pitz, and S. A., "The Phillips/RWTH systems for transcription of broadcast news," in *Proc. DARPA Broadcast News and Transcription Workshop, Herndon, Virginia,*, 1999.

[18] M. J. F. Gales and S. S. Airey, "Product of Gaussians for speech recognition," *Computer Speech and Language*, 2006.

[19] Z. Ghahramani and M. I. Jordan, "Factorial hidden Markov models," *Machine Learning*, vol. 29, pp. 245–275, 1997.

[20] H. Nock, "Techniques for modelling phonological processes in automatic speech recognition," Ph.D. dissertation, Cambridge University, 2001.

[21] G. Evermann and P. C. Woodland, "Posterior probability decoding, confidence estimation and system combination," in *Proc. Speech Transcription Workshop*, College Park, MD, May 2000.

[22] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recogniser Output Voting Error Reduction (ROVER)," in *Proc. IEEE ASRU Workshop*, 1997.

[23] H. Jiang, "Confidence measures for speech recognition: A survey," *Speech Communication*, vol. 45, pp. 455–470, 2006.

[24] O. Siohan, B. Ramabhadran, and B. Kingsbury, "Constructing ensembles of ASR systems using randomized decision trees," in *Proceedings ICASSP 2005*, 2005.

[25] R. Sinha, M. J. F. Gales, D. Kim, X. Liu, K. Sim, and P. C. Woodland, "The CU-HTK Mandarin Broadcast News transcription system," in *Proceedings ICASSP*, 2006.

[26] L. Baum and J. Eagon, "An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology," *Bull. Amer. Math. Soc.*, vol. 73, pp. 360–363, 1967.

[27] G. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, pp. 1771–1800, 2002.

[28] C. Breslin and M. Gales, "Generating complementary systems for speech recognition," in *To appear in Proceedings ICSLP*, 2006.

[29] R. Schapire, "The strength of weak lerners," *Machine Learning*, pp. 197–227, 1990.

[30] G. Zweig and M. Padmanabhan, "Boosting Gaussian mixtures in an LVCSR system," in *Proceedings ICASSP*, 2000.

[31] D. Dimitrakakis and S. Bengio, "Boosting HMMs with an application to speech recognition," in *Proceedings ICASSP*, 2004.

[32] C. Meyer and H. Schramm, "Boosting HMM acoustic models in large vocabulary speech recognition," *Speech Communication*, pp. 532–548, 2006.

[33] V. Venkataramani, S. Chakrabartty, and W. Byrne, "Support vector machines for segmental minimum Bayes risk decoding of continuous speech," in *Proceedings ASRU*, 2001.

[34] M. J. F. Gales and M. I. Layton, "Training augmented models using svms," *IEICE Special Issue on Statistical Modelling for Speech Recognition*, 2006.

[35] V. Vapnik, *Statistical learning theory*. John Wiley & Sons, 1998.

[36] T. Hain, "Implicit pronunciation modelling in ASR," in *ISCA ITRW PMLA*, 2002.

[37] P. C. Woodland, H. Y. Chan, G. Evermann, M. J. F. Gales, D. Y. Kim, X. A. Liu, D. Mrva, K. C. Sim, L. Wang, K. Yu, J. Makhoul, R. Schwartz, L. Nguyen, S. Matsoukas, B. Xiang, M. Afify, S. Abdou, J.-L. Gauvain, L. Lamel, H. Schwenk, G. Adda, F. Lefevre, D. Vergyri, W. Wang, J. Zheng, A. Venkataraman, R. R. Gadde, and A. Stolcke, "SuperEARS: Multi-site broadcast news system," in *Proc. Fall 2004 Rich Transcription Workshop (RT-04)*, Palisades, NY, November 2004.

[38] S. Kumar and W. Byrne, "Minimum Bayes-risk decoding for stastical machine translation," in *Proc. HLT-NAACL*, 2004.

[39] F. Och, "Minimum error training in statistical machine translation," in *Proceeding ACL*, 2002.

[40] S. Kumar and W. Byrne, "Minimum Bayes-risk alignment of bilingual texts," in *Proc. of the Conference on Empirical Methods in Natural Language Processing*, 2002.

[41] S. Bangalore, G. Bordel, and G. Riccardi, "Computing consensus translation from multiple machine translation systems," in *Proceedings ASRU*, 2001.

[42] E. Matusov, N. Ueffing, and H. Ney, "Computing consensus translation from multiple machine translation systems using enhanced hypotheses alignment," in *Proceedings EACL*, 2006.