CU-HTK March 2001 Hub5 system

Phil Woodland, Thomas Hain, Gunnar Evermann & Dan Povey

May 3rd 2001

Cambridge University Engineering Department

Overview

- CU-HTK 2000 system
- Cellular data
- MMIE training
- Lattice MLLR
- 2001 system & results
- Conclusions
- HTK3

CU-HTK 2000 System: Basic Features

- Front-end
  - Reduced bandwidth 125–3800 Hz
  - 12 MF-PLP cepstral parameters + C0 and 1st/2nd derivatives
  - Side-based cepstral mean and variance normalisation
  - Vocal tract length normalisation in training and test
- Decision tree state clustered, context dependent triphone & quinphone models: MMIE and MLE versions
- Generate lattices with MLLR-adapted models
- Rescore using iterative MLLR + Full-Variance transform adaptation
- Posterior probability decoding via confusion networks
- System combination

Acoustic Training/Test Data

h5train00 248 hours Switchboard (Swbd1), 17 hours CallHome English (CHE)

h5train00sub 60 hours Swbd1, 8 hours CHE

Development test sets

dev01 40 sides Swbd2 (eval98), 40 sides Swbd1 (eval00), 38 sides Swbd2 cellular (dev01-cell)

dev01sub half of the **dev01** selected to give similar WER to full set

eval98 40 sides Swbd2 (eval98-swbd2), 40 sides of CHE (eval98-che)

eval97sub 20 side subset of eval97 evaluation set (Swbd2 + CHE)

Earlier development used eval98/eval97sub and later work on dev01sub.

CU-HTK 2000/1 Systems: MLE acoustic models

- MLE triphone models
  - Initial models trained on h5train00sub using VTLN data (6168 states / 12 mix)
  - Extended training using 265 hour set h5train00 (16 mix comps)
  - Soft-tying of closest states for each phone
  - Create gender dependent (GD) versions

- MLE quinphone models
  - ± 2 phone context $+$ word boundary clustering on h5train00 VTLN data
  - Trained up to 16 mix comps (9642 states)
  - Soft-tied gender dependent versions

CU-HTK 2000 System: MMIE acoustic models

- Starting point: MLE models (triphone/quinphone, GI, VTLN)
- Trained using extended Baum-Welch algorithm with lattice-based MMIE
- Lattices on training data using MLE models and a bigram language model
  - Numerator/denominator word level lattices with model alignment and Hub5 unigram LM probabilities.
  - Scaling of acoustic likelihoods (instead of LM)
- Best performance after 2 iterations

CU-HTK 2000/1 Systems: Vocabulary & Language Modelling

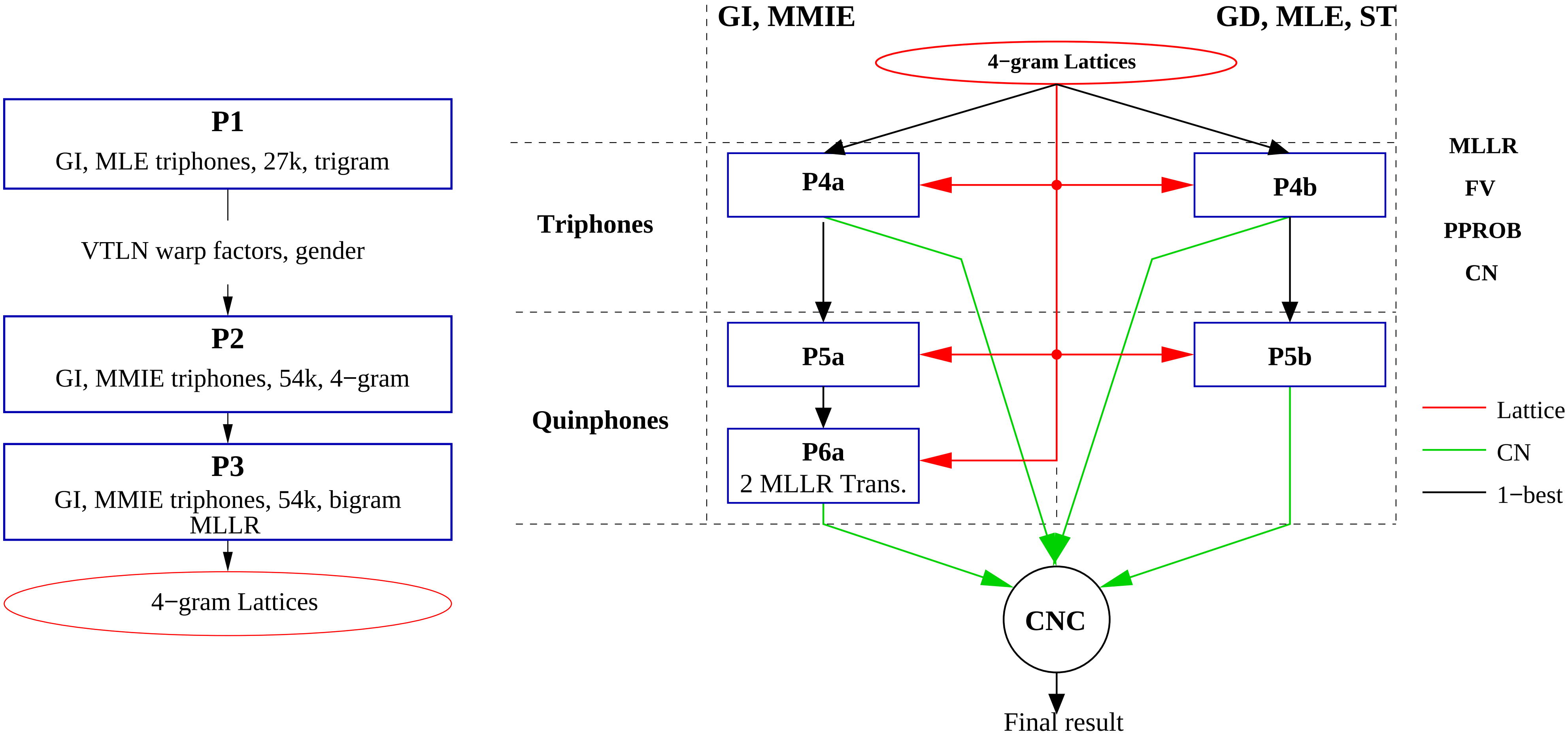
- Vocabulary
  - 54537 words: Hub5 vocabulary plus top 50k words of Broadcast News data (0.30% OOV rate on eval00)
  - Multiple pronunciation dictionary (based on LIMSI'93 + TTS)
  - Pronunciation probabilities estimated from forced alignment

- Language models
  - Training data
    - 204MW Broadcast News
    - 3MW 1998 Hub5 data
    - 3MW Hub5 data (CHE + Jan 2000 MSU transcriptions)
  - 3-fold interpolated/merged bigram, trigram, and 4-gram word LMs
  - Class based trigram model (400 classes) to smooth word LM

CU-HTK 2000/1 System: Decoding

- Lattice generation/rescoring with time-synchronous Viterbi decoder
- Post-process lattices to yield confusion networks
- Find 1-best min word error rate hypothesis from confusion network
- Combine networks from different stages using Confusion Network Combination
- Confidence scores estimated using confusion networks
- Piecewise linear mapping of word posteriors to confidence scores via decision tree.

CU-HTK 2000 System Stages



P1

GI, MLE triphones, 27k, trigram

VTLN warp factors, gender

P2

GI, MMIE triphones, 54k, 4−gram

P3

GI, MMIE triphones, 54k, bigram MLLR

4−gram Lattices

GI, MMIE

GD, MLE, ST

4−gram Lattices

Triphones

P4a

P4b

MLLR

FV

PPROB

CN

Quinphones

P5a

P5b

P6a

2 MLLR Trans.

Lattice

CN

1−best

CNC

Final result

CU-HTK 2000 performance on eval00

	Models	Ctxt	CN	MLLR	FV	Swbd2	CHE	Total
P1	1998 P1					31.7	45.4	38.6
P2	MMIE	tri	n	-	n	25.5	38.1	31.8
P3	MMIE	tri	n	1	n	22.9	35.7	29.3
P4a	MMIE	tri	y	1	y	20.9	33.5	27.2
P4b	MLE/ST/GD	tri	y	1	y	21.9	33.7	27.8
P5a	MMIE	quin	y	1	y	20.3	32.7	26.6
P5b	MLE/ST/GD	quin	y	1	y	21.0	32.8	26.9
P6a	MMIE	quin	y	2	y	20.3	32.6	26.5
CNC	P4a+P6a+P4b+P5b					19.3	31.4	25.4

%WER of CUHTK 2000 system on eval00

CU-HTK 2000 performance on dev01

dev01	Swbd1	Swbd2	Swdb2 cell	Total
P1	31.7	46.9	48.1	42.1
P2	25.5	40.1	41.7	35.7
P3	22.9	37.5	38.3	32.8
P4a	20.9	34.5	34.9	30.0
P4b	21.9	35.6	35.9	31.0
P5a	20.3	33.9	34.5	29.5
P5b	21.0	34.5	35.1	30.1
P6a	20.3	33.6	34.3	29.4
CNC (cuhtk1)	19.3	32.5	33.2	28.3

%WER of CUHTK 2000 system on dev01

Dealing with Cellular Data-I

- No cellular training data that is really appropriate (all available data sets have problems)
- Investigated simulating the GSM channel with the “toast” simulator
- GSM coding/decoding of the eval98-swbd2 data

	eval98-swdb2		dev01-cellular
	original	GSM-simulated	
MMI	40.0	43.6	41.7
MMI+MLLR	37.5	41.4	38.3

%WER in cellular data using MMI triphones trained on h5train00

Dealing with Cellular Data-II

- Try GSM coding/decoding the h5train00sub training data and re-training
- Used MLE models trained on 68 hours of data and single-pass retraining

	eval98-gsm			dev01-cellular
	Swbd2	CHE	Total	
baseline MLE	46.4	52.7	49.6	44.3
GSM simul. training	45.8	51.8	48.8	44.6

%WER for single-pass decoding with tg LM, h5train00sub models trained on original or simulated GSM data

- Training on simulated GSM coded data helps when test uses same process but not real cellular data!
- Decided to stick with baseline system for cellular data

Review of CU-HTK MMIE Training

- Maximum mutual information estimation (MMIE) maximises the sentence level posterior : in log form

$$\mathcal{F}_\lambda = \sum_{r=1}^{R} \log \frac{P_\lambda\left(\mathcal{O}_r|\mathcal{M}_{w_r}\right) P\left(w_r\right)}{\sum_w P_\lambda\left(\mathcal{O}_r|\mathcal{M}_w\right) P\left(w\right)}$$

  - *Numerator* is likelihood of data given correct transcription
  - *Denominator* is total likelihood, calculated as sum over *all* word sequences
  - Need to optimise rational function: use extended Baum-Welch algorithm
  - Compute using word lattices for numerator and denominator

Extended Baum-Welch Algorithm

EBW re-estimation formulae are of the form:

$$\hat{\mu}_{j,m} = \frac{\left\{\theta_{j,m}(\mathcal{O}) - \theta^{\text{den}}_{j,m}(\mathcal{O})\right\} + D\mu_{j,m}}{\left\{\gamma_{j,m} - \gamma^{\text{den}}_{j,m}\right\} + D}$$

- Due to high computational load need to ensure fast convergence
  - Gaussian-specific D setting with flooring

- Improve MMIE generalisation
  - Acoustic scaling by inverse of normal language model scale factor
  - Weakened language model (unigram) helps focus on acoustic distinctions

Lattice Based MMIE

- MLE triphone models used to generate word lattices
- Model-marked lattices for triphone/quinphone
- Run EBW with model-boundaries with margin for pruning
- Best performance after two iterations
- Applied to gender-independent non-soft-tied triphones and quinphones
- WER reductions of 2-3% absolute with 265 hours of training
- Also works well with MLLR, confusion networks etc.
- Investigate MMIE (& variants) for different styles of models

Fixed Boundary Estimation (“Exact Match”)

- Exact match scheme scheme relies on boundaries in model-marked lattices
- Comparisons show equal or better performance to using boundaries with margin
  - More efficient—runs about twice as fast
  - Allows more exact acoustic scaling
  - Fixed boundary estimation used for all experiments here
  - Used larger D flooring values to reduce overtraining after two iterations

Iteration	h5train00sub		h5train00	
Number	eval97sub	eval98	eval97sub	eval98
0 (MLE)	46.0	46.6	44.4	45.6
1	44.4	45.4	42.6	44.0
2	43.7	44.7	41.9	42.9
3	43.9	44.4	41.6	42.7
4	43.9	44.3	41.4	42.2

%WER GI models rescoring 1998 trigram lattices

Interpolated Objective Functions

- Maximising MMIE criterion tends to over-train, especially for smaller data sets
- Alternative: use modified objective function that combines MLE and MMIE objective function: a type of "H-criterion"
- Function of the form $\alpha\mathcal{F}_{\mathrm{MMIE}} + (1-\alpha)\mathcal{F}_{\mathrm{MLE}}$. Typically α in range 0.6-0.9
- Unlike pure MMIE, test data WER minimised as objective function maximised

	α = fraction MMIE				
	1.0	0.9	0.8	0.7	0.5
eval97sub	44.2	44.0	44.1	43.6	43.9
eval98	44.3	44.0	44.0	44.2	44.9

%WER of h5train00sub models with H-criterion training

MMIE Training for Gender Dependent Models

- Std method of training GD models starts from GI models
  - splits training data for male/female
  - update gender dependent mean and mix weights only
- Used H-criterion GD models with $\alpha = 0.75$
- On eval97sub with h5train00sub training WER reduced from 43.7% after 3 iterations to 43.1% after 2 iterations of GD updating
- With h5train00 training only 0.1% reduction in WER from GD modelling
- MMIE GD models not used in 2001 eval system

MMIE Training for Soft-Tied Models

- MLE models in 2000 evaluation system used soft-tying, but MMIE did not
- Initial investigations using h5train00sub triphones:
  - similar gains from soft-tying for MMIE as for MLE models
  - 1% reduction in WER on eval97sub and 0.5% on eval98
- Attempted to extend this to h5train00 using realigned lattices
  - improvements appeared to be very small on eval98
  - insufficient time to fully investigate
  - not used for MMIE models in 2001 eval system

MMIE-based Model Alignments

- Regenerate model-marked lattices from MMIE-trained triphones/quinphones

- Continue MMIE training for several iterations

	triphones		quinphones
Training type	eval97sub	eval98	eval97sub
MLE	44.4	45.6	42.0
2000 MMIE	41.9	42.7	39.8
fixed bound	41.4	42.2	39.2
realigned	40.9	41.5	38.6

%WER rescoring 1998 tg lattices, h5train00 non-adapted models, quinphones also use pronunciation probabilities

- More than 1% total reduction in WER due to new model training

- MMIE models now 3.4% to 4.1% better than MLE

Summary of MMIE Developments

- Lattice MMIE using fixed boundaries
- Use slower convergence—typically used 4 iterations
- Can use interpolated objective function to aid generalisation: more important with smaller training sets
- Worthwhile benefits from gender dependent models and soft-tying on 68 hour training appears not to scale to 265 hours: hence still using vanilla GI MMIE models
- Realigning lattices helps: overall MMIE triphones are 1-1.2% lower WER and quinphones 1.2% lower WER than 2000 evaluation MMIE models.

Lattice MLLR

- Unsupervised MLLR requires a transcription from a recogniser
  - transcription assumed correct
  - used to derive a single model sequence for MLLR forward-backward pass
- Gains from adaptation reduced due to supervision errors
  - Can estimate fewer transform parameters
  - Effect is most important when adaptation needed most!
- Lattice MLLR (Uebel & Woodland, ICASSP 2001) uses a recognition lattice for forward-backward pass rather than single model sequence
  - Take into account all model sequences found in lattice weighted by posterior probability
  - Use acoustic scaling to broaden posterior as for MMIE
  - Similar principles to Padmanabhan, Saon & Zweig, ASR 2000.

Lattice MLLR: Triphone Results

- Triphone MMIE models on eval98 set using 2000 system and lattices
- Generate phone-marked lattices once and iteratively update MLLR transforms in sequence 1,2,4,8 MLLR transforms
- Iteratively update FV transform: true interleaved updates of MLLR and FV transforms

	#MLLR(+FV)	%WER (Viterbi)	%WER (Conf-Net)
std MLLR	1	38.7	37.1
lattice MLLR	1	38.5	36.9
lattice MLLR	2	38.2	36.7
lattice MLLR	4	38.0	36.6
lattice MLLR	8	37.7	36.7

%WER on eval98: iterative lattice MLLR vs std MLLR

Lattice MLLR: Triphone Dev Results

#MLLR	FV iterative	Swbd1	Swbd2	Swbd2 cell	Total
1	Y	18.8	34.3	34.7	29.2
4	Y	18.7	34.2	34.4	29.0
8	Y	18.8	33.9	34.3	28.9
4	N	18.7	34.0	34.3	28.9
8	N	18.6	34.0	34.5	28.9

%WER (confnet) on dev01sub: single vs iterative FV estimation

- Iterative estimation of FV transform not required
- 2001 system used up to 4 MLLR transforms with FV transform estimated once

Resegmentation of Audio Data

- Concerned that MSU-style silence “bracketing” (0.5s) would effect normalisation factors relative to training
  - silence at segment boundaries reduced to 200ms
- Initially investigated VTLN: implementation very stable wrt amount of silence
- Cepstral Mean/Variance normalisation is more of an issue
  - re-segment test data using same rules as training
  - use aligned P1 output for segmentation points
  - re-compute mean/variance normalisation factors
- Investigated effect of re-segmentation on adapted and non-adapted models

Resegmentation of Audio Data: Results

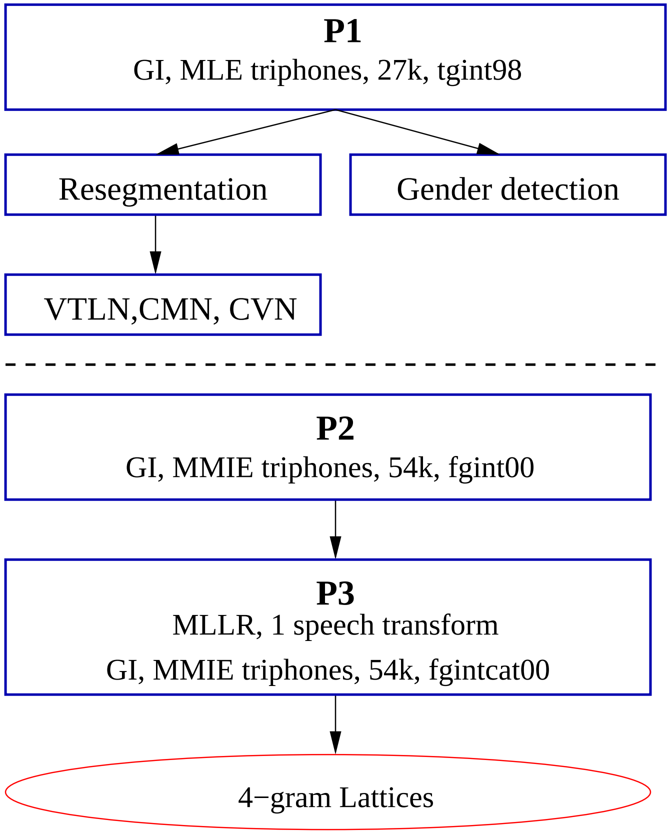
	MLLR	dev01-cellular	eval00-sw1
original seg	N	42.4	26.2
new seg for normalisation	N	41.2	25.1
original seg	Y	39.2	—
new seg for normalisation	Y	38.6	—

%WER 2000 MMIE/VTLN models models, tg LM

- More than 1% abs reduction in WER without adaptation for mismatched training/test segmentation
  - variance normalisation is rather sensitive to segmentation!
- Considerably smaller improvement when using adaptation
- Negligible impact on eval98

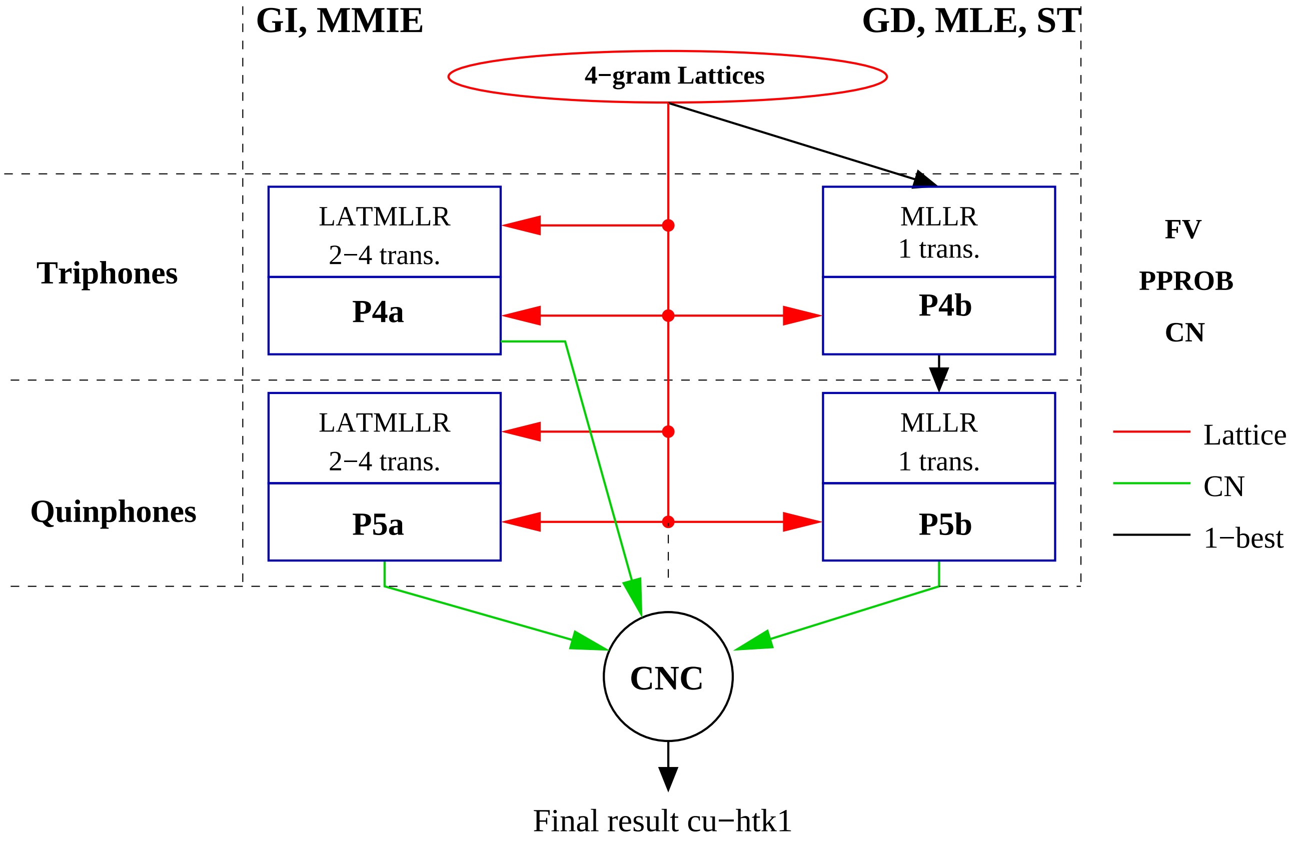
2001 system – Lattice Generation

- Stages similar to last year
- New MMIE models
- Normalisation after resegmentation based on P1 output



2001 system – Rescoring & Combination

- MLLR in rescoring replaced by iterative Lattice MLLR (4 transforms)



Results on dev01 set

		Swbd1	Swbd2	Swbd2 cell	Total
P1	VTLN/gender det	31.7	46.9	48.1	42.1
P2	initial trans.	23.5	38.6	39.2	33.7
P3	lat gen	21.1	36.0	36.7	31.2
P4a	MMIE tri	20.0	33.5	34.0	29.1
P4b	MLE tri	21.3	35.0	35.4	30.5
P5a	MMIE quin	19.8	33.2	33.4	28.7
P5b	MLE quin	20.2	34.0	34.2	29.4
CNC	P5a+P4a+P5b	18.3	31.9	32.1	27.3
Rover	vote	18.9	32.5	32.6	27.9
Rover	conf	18.6	32.3	32.4	27.6

%WER on dev01 for all stages of 2001 system

- final confidence scores have NCE 0.254

Results on eval01 set

		Swbd1	Swbd2	Swbd2 cell	Total
P1	VTLN/gender det	31.9	39.2	45.5	39.1
P2	initial trans.	24.3	29.9	36.6	30.4
P3	lat gen	22.5	27.8	33.9	28.2
P4a	MMIE tri	21.3	26.3	31.6	26.5
P4b	MLE tri	22.6	27.8	32.9	27.9
P5a	MMIE quin	21.5	26.1	30.8	26.2
P5b	MLE quin	21.3	26.7	32.0	26.8
CNC	P5a+P4a+P5b	19.8	24.5	29.2	24.6
Rover	vote	20.1	25.2	30.2	25.3
Rover	conf	19.9	24.6	29.8	24.9

%WER on eval01 for all stages of 2001 system

- final confidence scores have NCE 0.294

Lattice MLLR & System combination

- Effect of lattice MLLR for quinphone models
  - Compare with std iterative MLLR based on triphone output.
  - Lattice MLLR is much more independent of previous stages
  - No cross-system adaptation effects
  - Possible explanation for relatively little gain from quinphones
- Include the effects of confusion networks and system combination.
- Results on dev01sub
  - cross-system adaptation important for quinphones
  - system combination means that single best quinphone system less important!

Quinphone Lattice MLLR & System Combinations

		#MLLR(+FV)	%WER (Vit)	%WER (CN)
Q1	lattice MLLR (P5a-1)	1	30.2	29.1
Q2	std MLLR (tri adapt)	1	29.9	28.6
Q3	lattice MLLR	4	29.9	28.8
Q4	std MLLR (quin adapt)	2	29.5	28.5

%WER on dev01sub for MMI quinphone models

	Swbd1	Swbd2	Swbd2 cell	Total
Q1 + P4a + P5b	16.9	32.5	32.8	27.3
Q2 + P4a + P5b	17.0	32.4	32.7	27.3
Q3 + P4a + P5b	17.1	32.5	32.6	27.3
Q4 + P4a + P5b	17.0	32.5	32.6	27.3

%WER on dev01sub for various adapted quinphone combinations

Computation

Pass	Speed ($\times$RT)	Memory (MB)
P1	12	357
P2	13	280
P3	39	335
P4a	30	280
P4b	34	299
P5a	27	380
P5b	36	391

Times based on Pentium III 1GHz

- MLLR/Full-Variance 4xRT
- Time marked lattices for LatMLLR (tri+quin) 48xRT+43xRT
- Lattice MLLR/Full-Variance 10xRT
- Overall 298xRT

Faster Contrast

- Contrast cuhtk2 is first part of the full system
- Confusion network output of P3 lattices (only MMIE models)
- No rescoring with multiple models, LatMLLR or sys combination
- Runs in a fraction of the time of cuhtk1 (65xRT vs. 298xRT)

	Swbd1	Swbd2	Swbd2 cell	Total	NCE
cuhtk1	19.8	24.5	29.2	24.6	0.294
cuhtk2	21.6	27.0	32.6	27.2	0.308

%WER on eval01 for primary and contrast systems

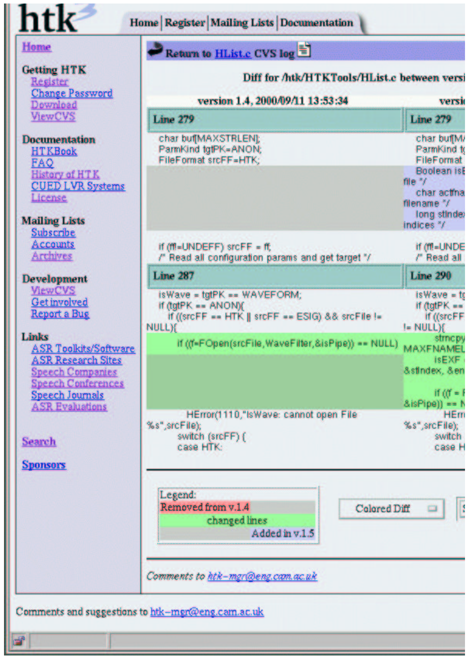
Conclusions

- Improved MMIE training 1% abs lower WER
- Improved adaptation using Lattice MLLR
  - Allows use of more transforms - improvements of 1-best decoding 1% abs
  - Doesn't exploit cross-adaptation effects and overall probably little win
- Re-segmentation for improved normalisation 0.1-0.7% better with adaptation
- Overall system improvement
  - 1% improvement over 2000 system: rather less than sum of parts!
  - Still lowest overall WER
- Faster contrast system
  - no rescoring passes and sys combination still yields competitive system
  - improved 1.5% abs over corresponding 2000 system

HTK3

- Available free of charge since September 2000
- Includes full C source & 300 page HTK book
- Aims to lower entry barrier for ASR research
- Web site got hits from 25k unique IP addresses
- 5000 registered users
- User/developer mailing lists (100 posts/month)
- Meeting: 10th May 7pm Hilton Salt Lake City

`http://htk.eng.cam.ac.uk`



Return to HLlist.c CVS log

Diff for /htk/HTKTools/HList.c between versi

version 1.4, 2000/09/11 13:53:34 | versi

Line 279 | Line 279

```
   char buf[MAXSTRLEN];                               char buf[MA
   ParmKind tgtPK=ANON;                               ParmKind tg
   FileFormat srcFF=HTK;                              FileFormat
                                                      Boolean isE
                                                   file */
                                                      char actfna
                                                   filename */
                                                      long stInde
                                                   indices */

   if (ff!=UNDEFF) srcFF = ff;                        if (ff!=UNDE
   /* Read all configuration params and get target */ /* Read all
```

Line 287 | Line 290

```
   isWave = tgtPK == WAVEFORM;                        isWave = tg
   if (tgtPK == ANON){                                if (tgtPK ==
      if ((srcFF == HTK || srcFF == ESIG) && srcFile !=   if ((srcFF
NULL){                                             != NULL){
         if ((f=FOpen(srcFile,WaveFilter,&isPipe)) == NULL)   strncpy
                                                   MAXFNAMEL
                                                         isEXF
                                                   &stIndex, &en

                                                         if ((f = F
                                                   &isPipe)) == N
            HError(1110,"IsWave: cannot open File        HErr
%s",srcFile);                                      %s",srcFile);
         switch (srcFF) {                                 switch
         case HTK:                                        case H
```

Legend:
Removed from v.1.4
changed lines
Added in v.1.5

Comments to htk-mgr@eng.cam.ac.uk

Comments and suggestions to htk-mgr@eng.cam.ac.uk

HTK3 features

- Discrete and (semi-)continuous HMMs
- Decision-tree state clustering of cross-word triphones
- Baum-Welch training, Viterbi recognition & alignment
- Bigram language models and finite state grammars
- Lattice generation & rescoring
- MLLR and MAP adaptation
- New version supports PLP, VTLN as used in eval
- Future plans: lattice tools, LM toolkit, MMIE, LVCSR decoder