The Development of the Cambridge University RT-04 Diarisation System

S.E. Tranter, M.J.F. Gales, R. Sinha, S. Umesh, P.C. Woodland

9th November 2004

Cambridge University

Overview

- The Diarisation Task and Data
- The CU Diarisation System
- Development Results
- Results on RT-04f Evaluation Data
- Summary and Future Plans

The Diarisation Task

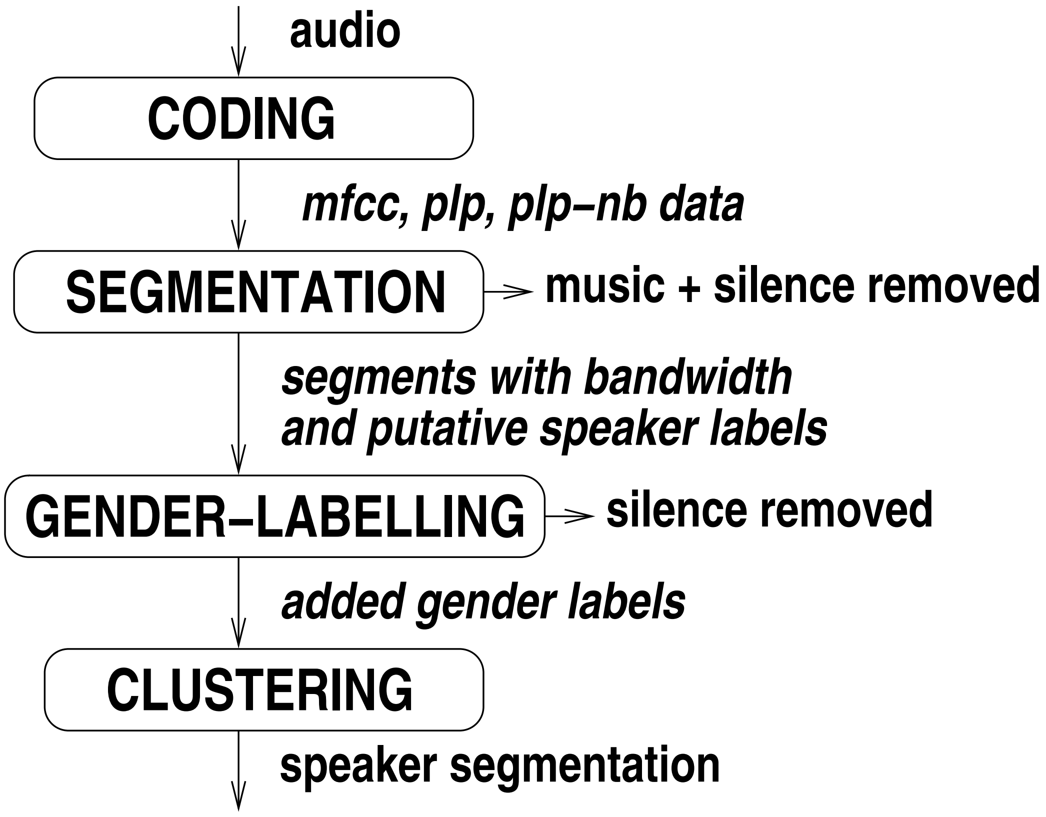
- Task
  - Label 'who spoke when' from audio data.
  - Essentially a speaker segmentation and clustering task.
- Data
  - Development sets : each consisted of 6 shows of approx. 30 minutes

didev03	:	RT-03s dev data, epoch Oct-Dec 2000.
sttdev04	:	manually marked at CU, epoch Jan 2001.
eval03	:	RT-04 dev data, epoch Feb 2001.
dev04f2	:	RT-04 dev data, epoch Nov/Dec 2003.
devall	:	represents sum of all dev sets.

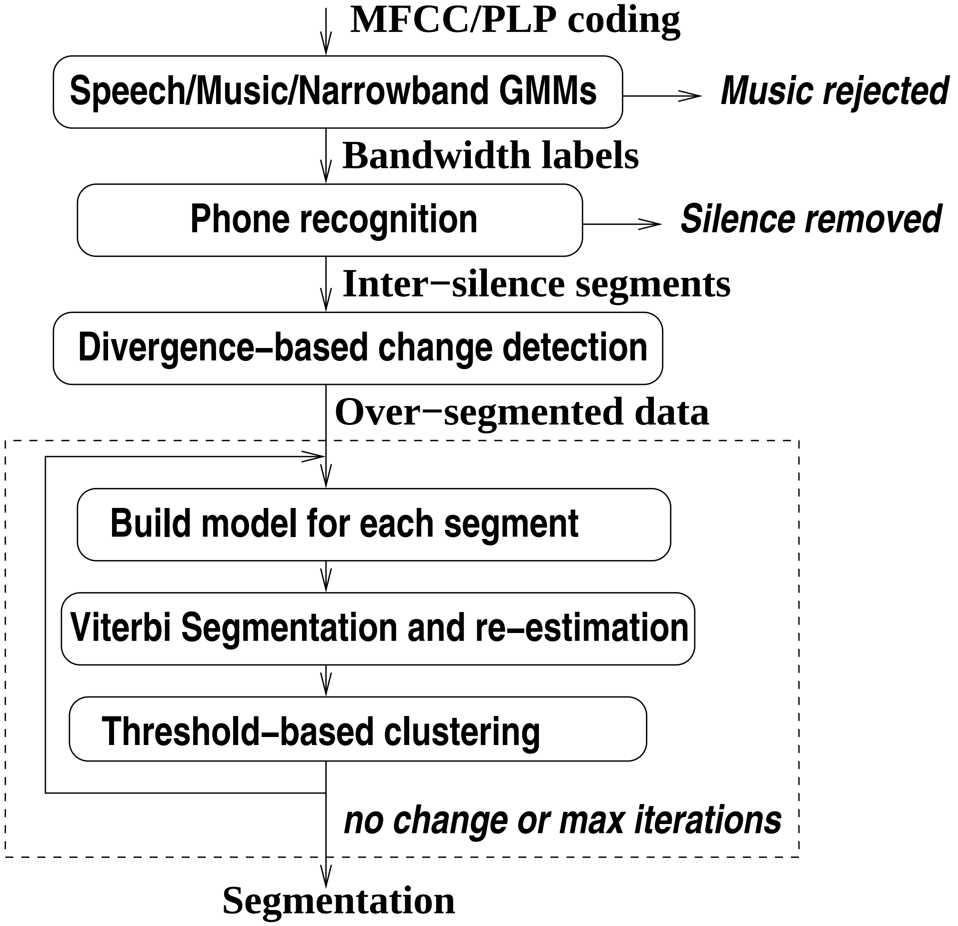
  - Evaluation set : eval04f consisted of 12 shows, epoch Dec 2003.

The CU Diarisation System - Overview

3 stage process : Segmentation →Gender Labelling →Clustering



The CU Diarisation System - Segmenter


MFCC/PLP coding
Speech/Music/Narrowband GMMs
Music rejected
Bandwidth labels
Phone recognition
Silence removed
Inter−silence segments
Divergence−based change detection
Over−segmented data
Build model for each segment
Viterbi Segmentation and re−estimation
Threshold−based clustering
no change or max iterations
Segmentation

- Over-segmented data is combined using LIMSI-style iterative scheme.

The CU Diarisation System - Segmenter (2)

- Single Gaussian model is built for each segment.
- Segments having loss likelihood less than a threshold if merged are combined.
- Viterbi decoding using new models then resegments the data.
- First few iterations used diagonal covariance to model segments as there were many short segments.
- In subsequent iterations a full covariance model is used.
- RT-04 segmenter also produces speaker labels unlike RT-03s segmenter.

The CU Diarisation System - Gender Labelling

- The first-pass of CU BN STT system is run to transcribe the segmenter output.

- Segments with no transcription are discarded from segmenter output.

- A forced alignment with GD models then determines the most likely gender of each segment.

The CU Diarisation System - Clusterer

- Clustering is done bandwidth and gender dependently.
- Clusterer uses *only* the start/end times of segments and *ignores* the segmenter speaker labels.
- Segments are sorted by mid-time based segment cluster-id before clustering.
- Clustering is done *top-down* using AHS distance metric and BIC-based stopping criterion.
- Single full *correlation* matrix of static PLP features is used to model segments.

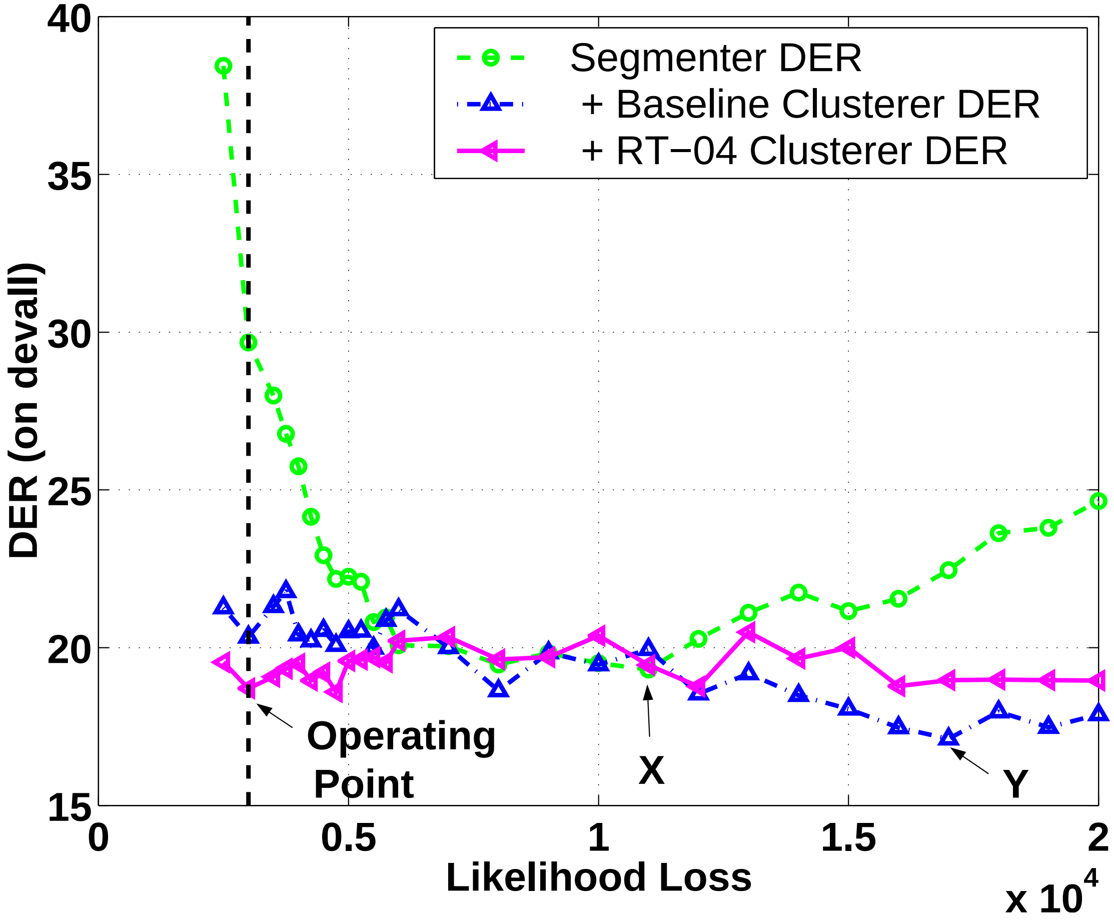
Development Results - Segmenter Improvement

Segmentation	Dataset	Segment-Impurity †	Seg	+Clust
		MS/FA/SPE/SI @ NumSeg	DER	DER
RT-03s	didev03	0.1/3.0/1.9/5.07 @ 875	-	18.8
	eval03	0.3/1.9/1.7/3.92 @ 869	-	19.8
	sttdev04	1.0/0.9/2.1/4.01 @ 913	-	22.9
	dev04f2	1.3/4.1/1.0/6.33 @ 1077	-	32.7
	devall	**0.69/2.34/1.70/4.74 @ 3734**	-	**23.2**
RT-04	didev03	0.6/1.6/1.0/3.16 @ 790	27.9	18.0
	eval03	0.6/0.7/0.9/2.17 @ 706	31.2	15.9
	sttdev04	2.2/0.3/0.9/3.36 @ 786	30.1	21.2
	dev04f2	1.5/1.8/0.6/3.93 @ 632	39.9	26.9
	devall	**1.26/1.03/0.85/3.14 @ 2914**	**29.7**	**20.3**

† sometimes called oracle clustering

- 34% relative drop in SI along with 22% reduction in # of segments on devall.
- DER using RT-03s clusterer improved by 12% relative.

Development Results - Segmenter Tuning



- Best Segmenter DER = 19.3% (**X**), Best Clusterer DER = 17.1% (**Y**)

Development Results - Silence Removal

- Silence stripping after phone recogniser stage in segmenter

Silence Threshold	Segment Impurity MS/FA/SPE/SI	@ NumSeg	Segmenter DER (on `devall`)
0.5s	3.62/0.39/0.85/4.86	@ 5190	36.1
1.0s	1.22/1.08/0.85/**3.15**	@ 3005	**32.0**
2.0s	0.77/2.12/0.94/3.83	@ 3045	32.9

Segment impurity as well as DER lowest for 1s value.

- Empty segments removal in gender-labelling stage

Stage	Segment Impurity MS/FA/SPE/SI	@ NumSeg
before P1	1.22/1.08/0.85/3.15	@ 3005
after P1	1.26/1.03/0.85/3.14	@ 2914

Number of segments dropped by 3% with no to loss segment purity.

Development Results - Clusterer Initialisation

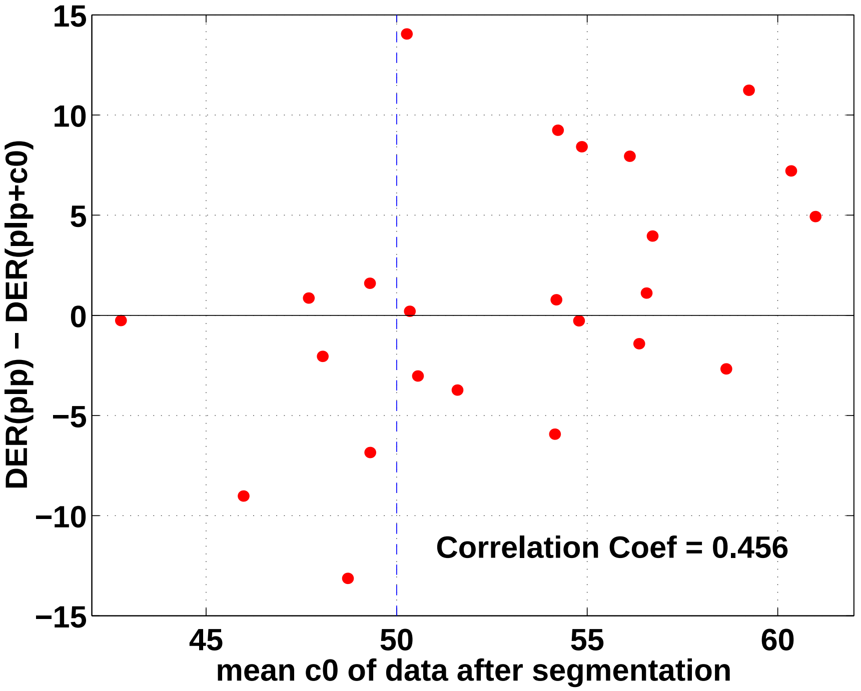
Aim : sorting the segments before clustering to help initialisation

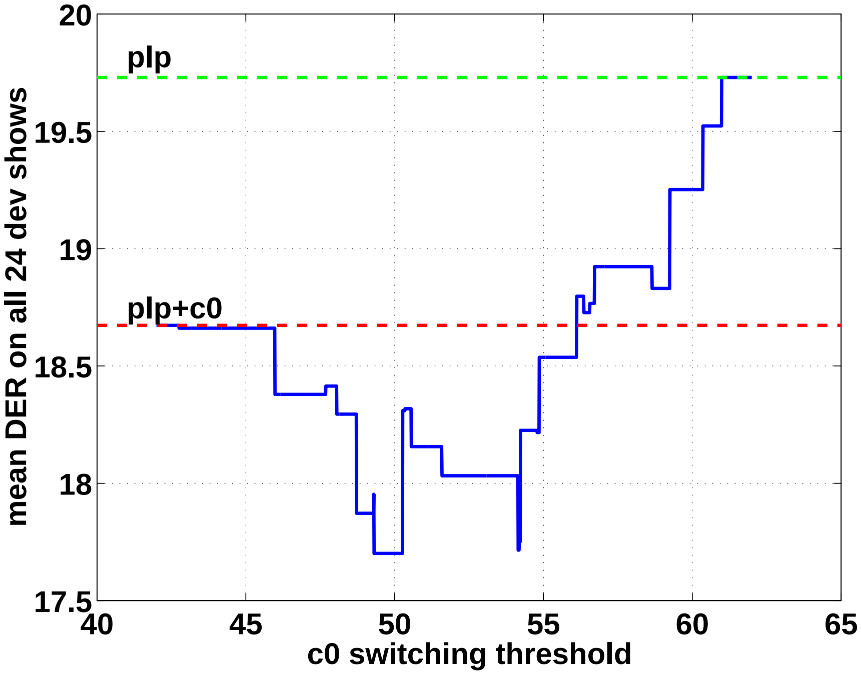
sorting	didev03	eval03	sttdev04	dev04f2	devall
none	18.0	15.9	21.2	26.9	20.3
time	17.5	16.7	21.5	25.7	20.2
spkr-start	17.5	17.9	22.6	17.5	19.0
spkr-mid	14.0	15.2	22.2	23.5	**18.7**

- Relative improvement of 8% in DER on `devall`.
- Clusterer highly sensitive to initialisation.

Development Results - Changing Feature in Clustering

Motivation : DER dropped by 5% absolute on `dev04f2` by using PLP *without* c0





- Clustering uses c0 if and only if mean c0 value is above a threshold.

- c0-switching reduced DER by 1% absolute on `devall`.

Results on RT-04f Evaluation Data - Progress

Coding	Segmentation	Clustering	DER main	DER c0switch
RT-03s	**RT-03s**	**RT-03s**	**36.33**	-
RT-03s	RT-03s	RT-04f	27.90	24.45
RT-03s	**RT-04f**	**RT-04f**	**22.48**	**22.35**
†RT-04f	RT-04f	RT-04f	23.86	24.12

† Official evaluation submission

- New segmenter and clusterer resulted in 14% absolute drop in primary DER.

- Slight degradation in evaluation submission performance due to compiler switch affecting only *coding*.

Results on RT-04f Eval Data - Impact of Different Strategies

Different strategies to pick segmenter and clusterer on dev data:
(a) use segmenter with best segmenter DER on `devall`
(b) use clusterer with best DER on `devall`
(c) use clusterer with best DER on `dev04f2`
(e) RT-04 evaluation system

Likelihood Threshold	Segmenter DER	RT-03s Clusterer DER	RT-04 Clusterer DER
3000	35.15	22.03	22.48(e)
11000	18.72(a)	22.90	21.02
16000	21.17	20.50(c)	22.18
17000	22.05	22.06(b)	21.44

- NB using segmenter output only, `eval04f` DER could be reduced to 18.7%.
- using static only coefficients in final segmentation stage reduced this to 18.1%.

Summary and Future Plans

- On RT-04f evaluation data the final system gave DER of 23.9%.
- Modifications since RT-03s resulted in 34% relative improvement in DER.
- DER of 18.1% is possible using segmenter output directly.
- Clustering stage rather sensitive to segmentation.
- Possible future work includes :
  - exploiting speaker labels from segmenter in clusterer.
  - cluster voting of segmenter and clusterer outputs.
  - investigating the use of proxy speaker models (a la MIT).