The ICSI RT-09 Speaker Diarization System

David Sun

Papers

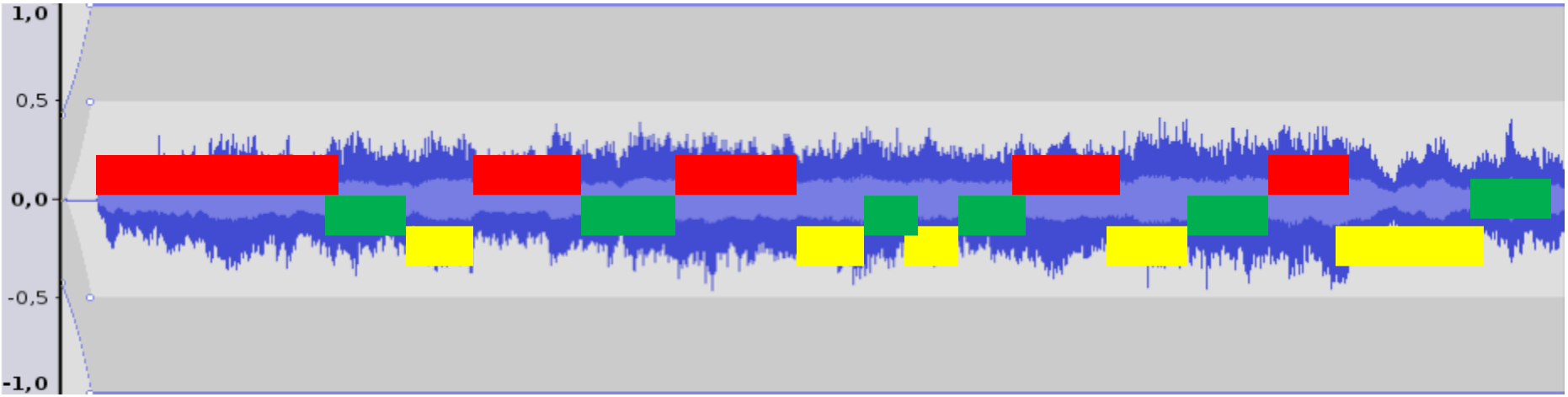
- ***The ICSI RT-09 Speaker Diarization System***, Gerald Friedland, Adam Janin, David Imseng, Xavier Anguera, Luke Gottlieb, Marijn Huijbregts, Mary Tai Knox and Oriol Vinyals, Transactions on Audio, Speech and Language Processing (TASLP, July 2011
- *Speaker Diarization: a review of recent research*, Xavier Anguera, Simon Bozonnet, Nicholas Evans, Corinne Fredouille, Gerald Friedland, and Oriol Vinyals, Transactions on Audio, Speech and Language Processing (TASLP), July 2011
- *Robust speech/non-speech classification in heterogeneous multimedia content,* Huijbregts, Marijn and de Jong, Franciska, Speech Commun, Feb 2011

Topics

- Speech diarization overview
- ICSI RT-09
- Signal pre-processing
- Speech activity detection
- Speaker segmentation clustering
- Joint audio video diarization

Speech Diarization

- Question “who spoke when?”



- Unsurpervised segmentation into speaker-homogenous regions.
- Challenge:
  - Number of speakers unknown
  - Amount of speech unknown

Applications

- Annotating broadcast news , TV, radio .
- Meeting content indexing, linking, summarization, navigation
  - Multiple audio, video and textual streams.
- Behavior analysis
  - Find dominant speakers
  - Engagement
  - Emotions
- Speech-to-text
  - Speaker model adaptation

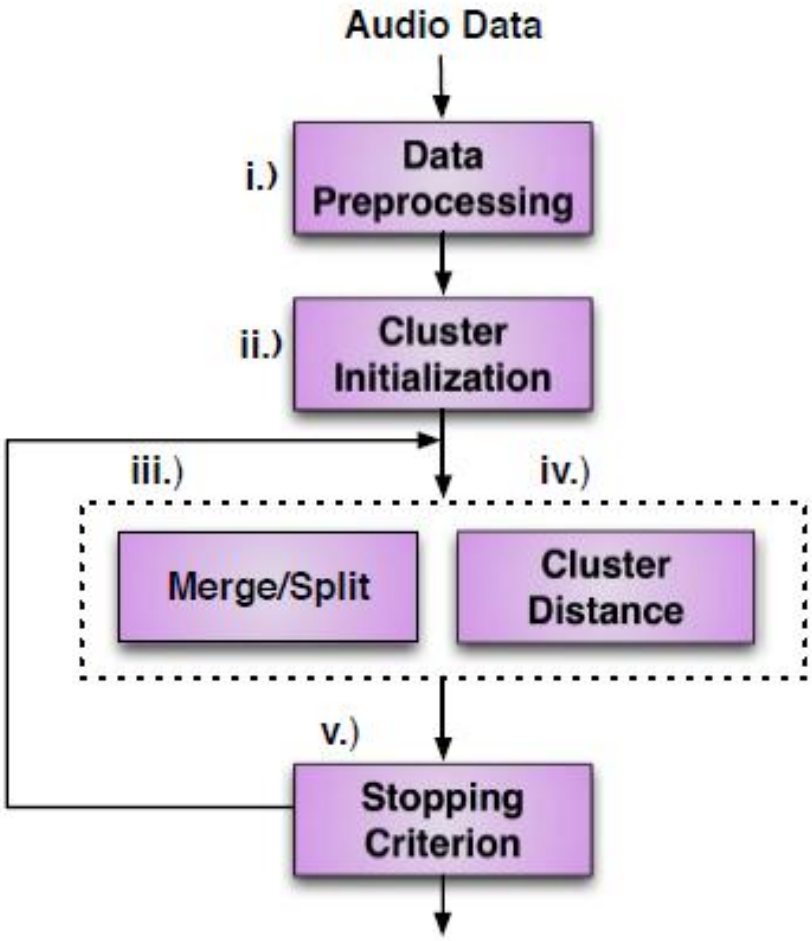
Major Projects

- Euro
  - European Union (EU) Multimodal Meeting Manager (M4) project,
  - The Swiss Interactive Multimodal Information Management (IM2) project
  - The EU Augmented Multi-party Interaction (AMI) project/ EU Augmented Multi-party Interaction with Distant Access (AMIDA) project
  - EU Computers in the Human Interaction Loop (CHIL) project
- USA?

Evaluation

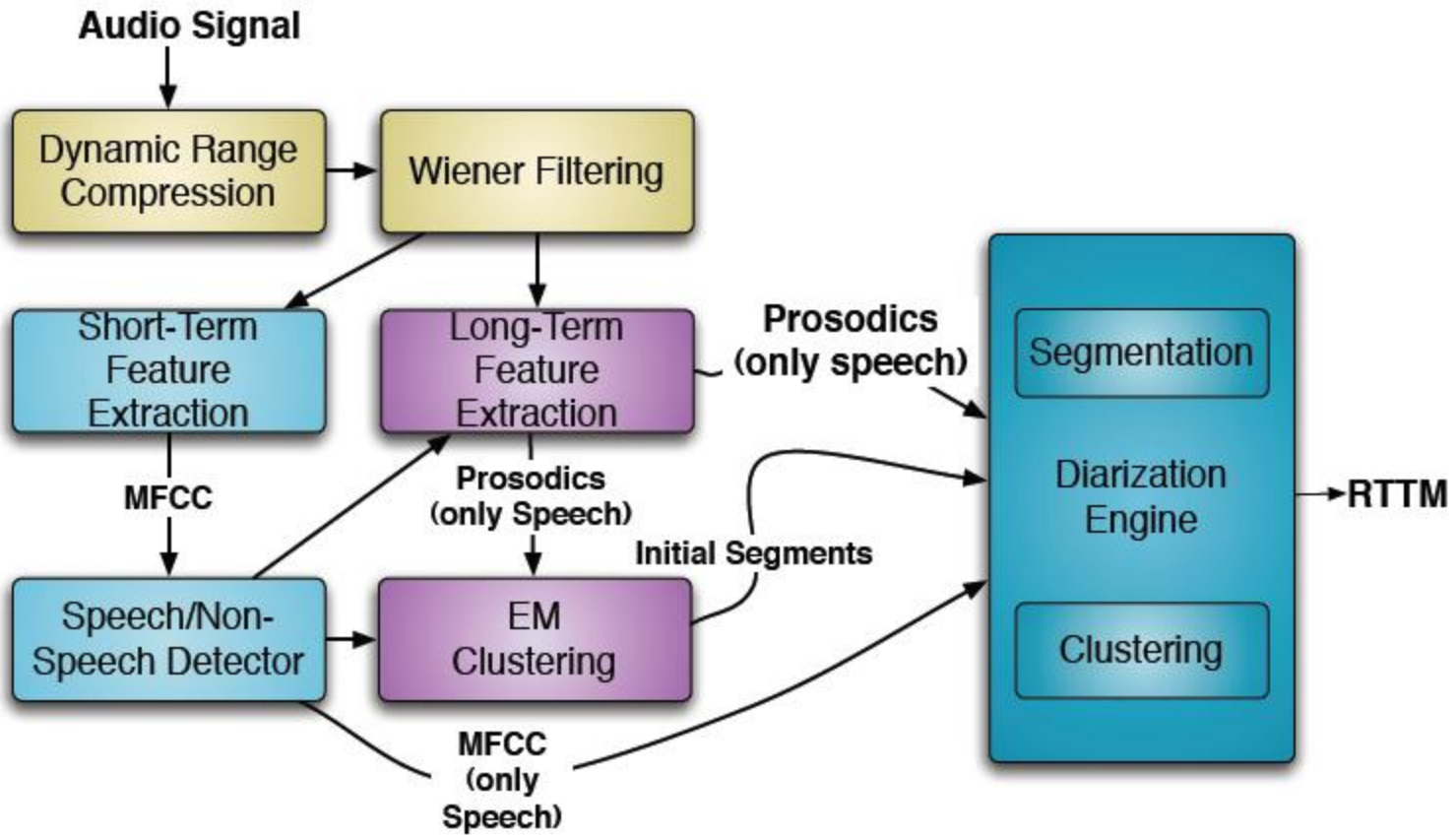
- US National Institute for Standards and Technology (NIST) official speaker diarization (Rich-Transcription) evaluation
  - Standard protocol and database
  - broadcast news (BN)
    - Recorded in studio (high S/N ratio)
    - Structured speech
  - meeting data
    - Recorded using far field mics (high variability, room artifacts)
    - More spontaneous and overlapping
  - lecture meetings
  - coffee breaks

General Architecture

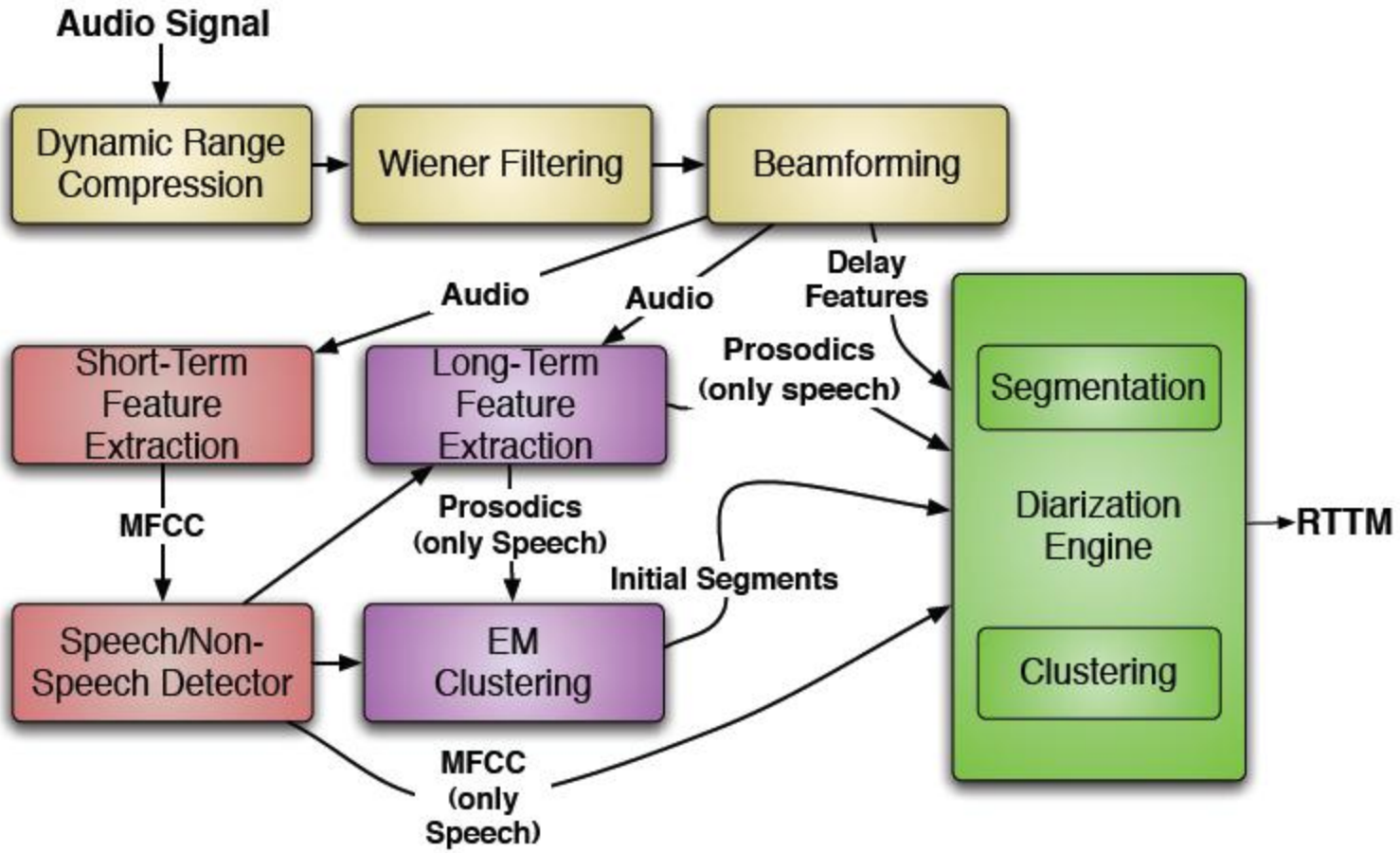


Noise reduction
Multichannel processing*
Feature extraction
Voice activity Detection

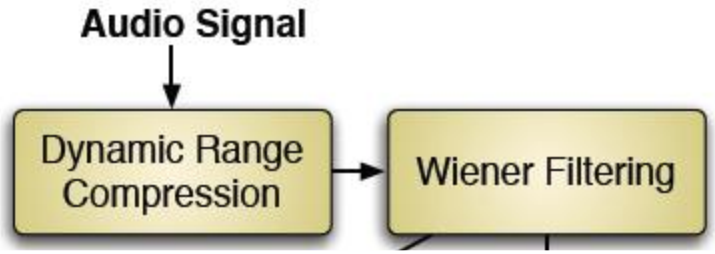
ICSI Speaker Diarization System (Single Distant Microphone)



ICSI Speaker Diarization System (Multiple Distant Microphone)

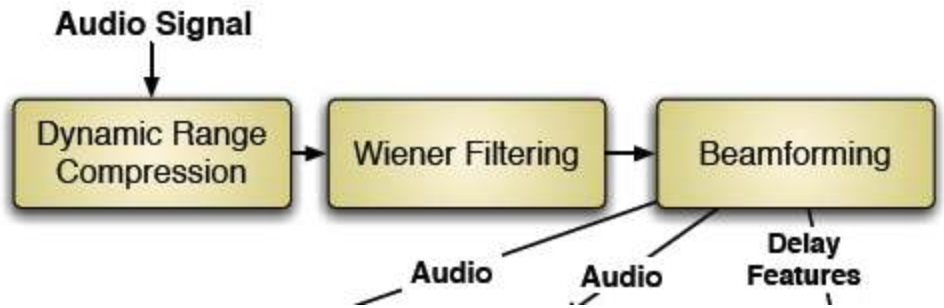


Signal Preprocessing



- Dynamic Range Compression
  - Convert to linear 16-bit PCM (truncate high order bits)
  - Downsample to 16KHz (little impact on performance)

Multichannel Processing



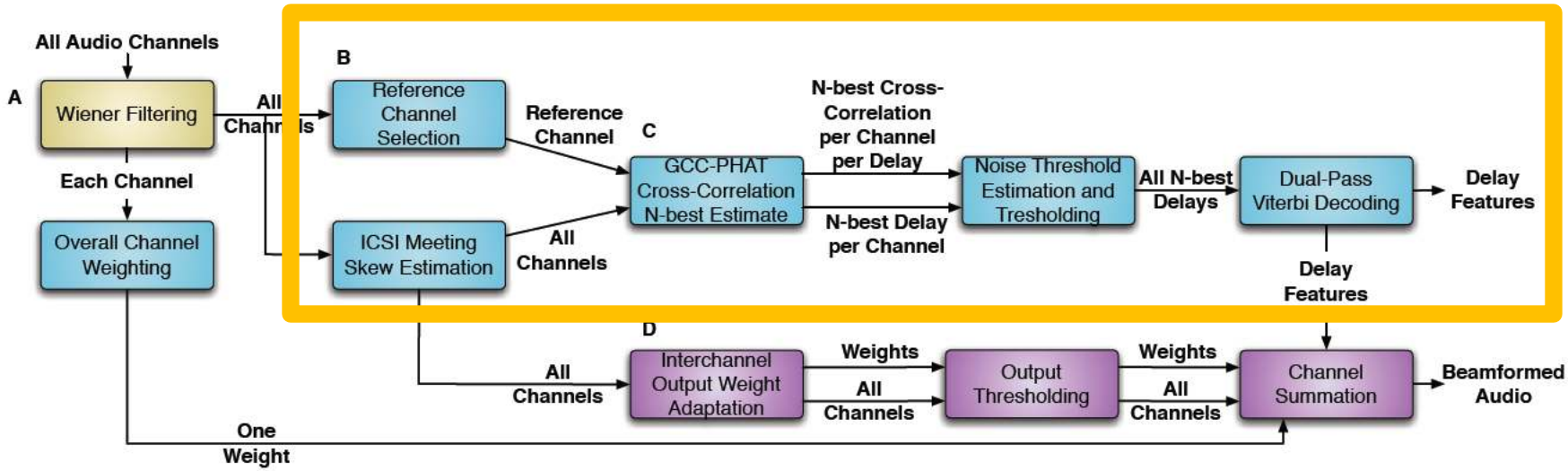
- Beamforming
  - S/N enhancement technique
  - Combine recording from multiple microphones into a single enhanced audio source.

Beamforming

- Microphones are spatially located, each captures random noise.
- Adding multiple channels together:
  - Desired signal enhanced
  - Noise cancels out or suppressed
- Delay-(filter)-sum
  - Output is a weighted sum of delayed inputs

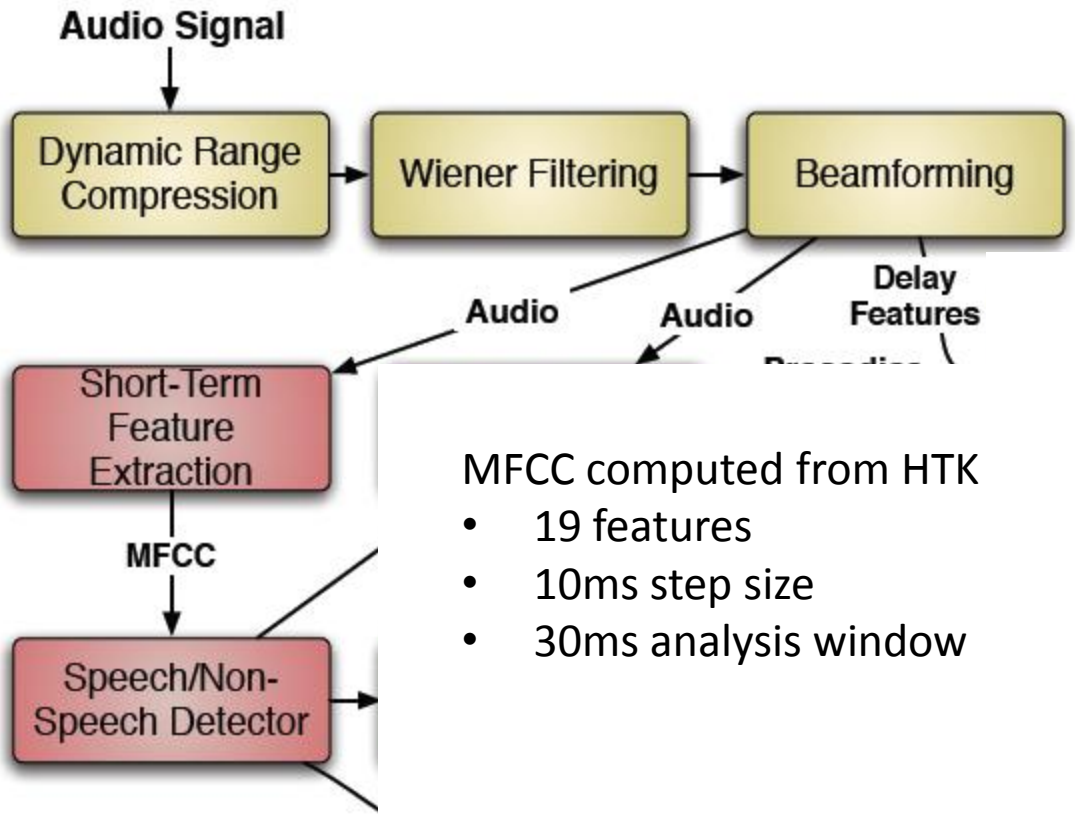
$$y[n] = \sum_{m=1}^{M} W_m[n] x_m[n - \mathrm{TDOA}^{(m,\mathrm{ref})}[n]]$$

Robust Beamformer (BeamformIt)



$$y[n] = \sum_{m=1}^{M} W_m[n] x_m[n - \mathrm{TDOA}^{(m,\mathrm{ref}}[n]]$$

Feature Extraction



MFCC computed from HTK

- 19 features
- 10ms step size
- 30ms analysis window

Speech Activity Detection (SAD)

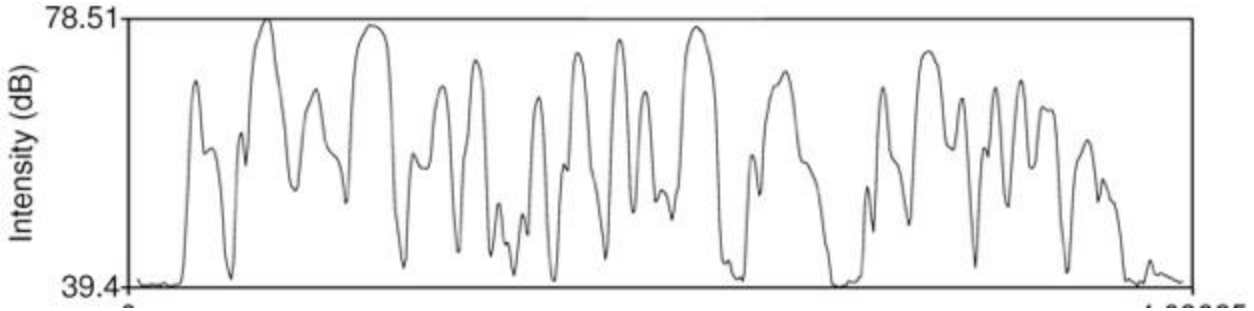
- Task: detecting the fragments in an audio recording that contain speech
- Simultaneous classification and classification
- Classification task
  - Given a fragment of audio distinguishing *speech* from *non-speech*
  - *Non-speech* can be *silence* or *audible non-speech.*
- Segmentation task
  - Determine the start time and end time of each fragment

Why?

- More practical to process small speech segments instead of an entire recording
- Performance of the ASR system can be enhanced
  - ASR will always produce hypothesis on input audio, even for audible non-speech => more insertion errors
  - cluster the speech segments on a speaker for automatic ASR tuning.
- All non-speech presented to a speaker clustering system will contaminate the speaker models and this will decrease the clustering quality

Silence Based Detection

- Assume audio only contains speech and silence.
- Algorithm:
  - Calculate the energy of short (often overlapping) windows.
  - The local minima of this energy series are considered silence.



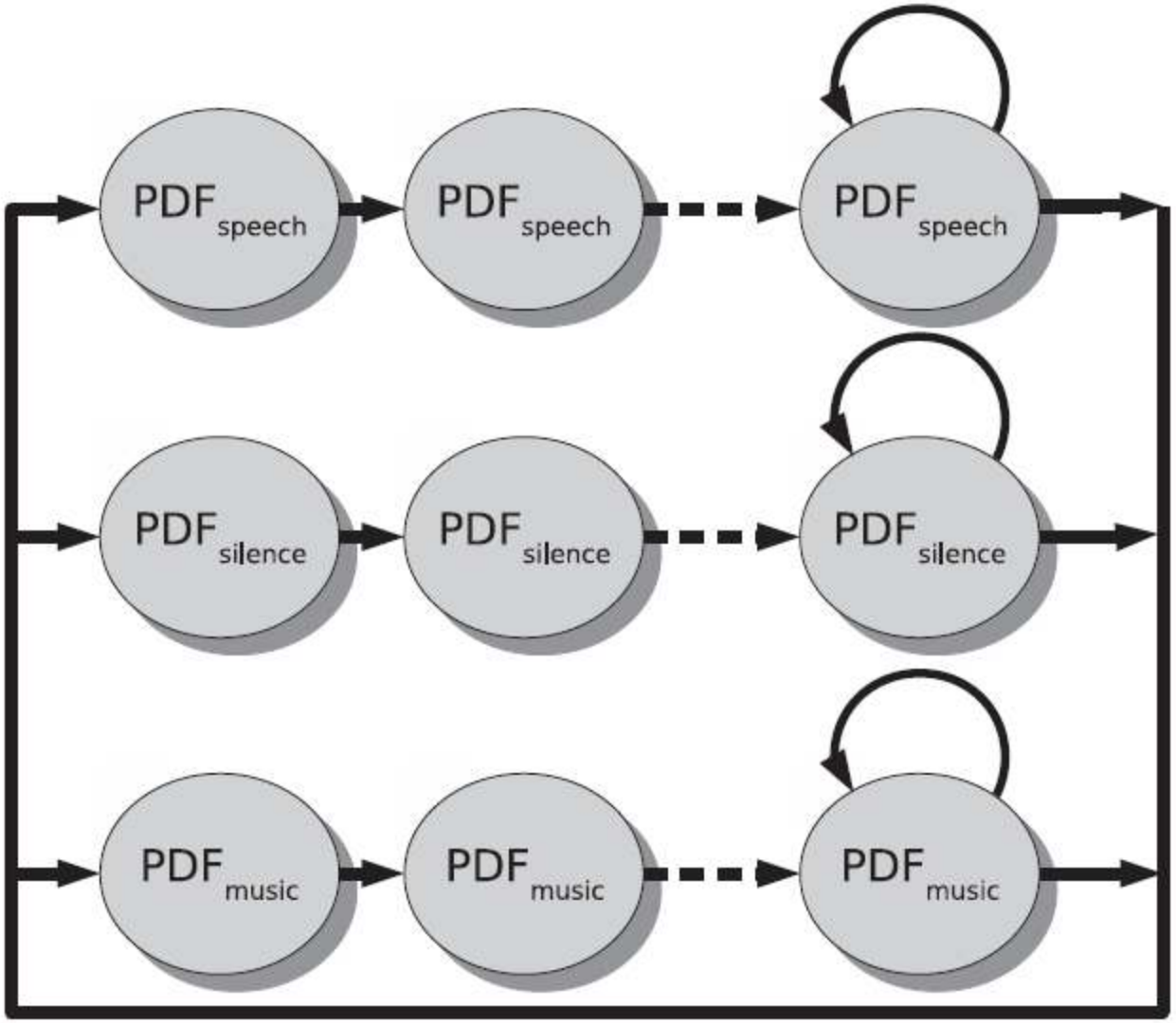
Silence Based Detection

- Broadcast News (BN) recordings
  - major part of the recording consists of speech and small pauses between utterances or topics
- Meetings
  - more spontaneous speech

Model Based Detection

- Train a GMM for each class.
- HMMs with one state for each class tend to produce short segments
  - even with low transition probabilities
- To force minimum time constraints on segments
  - HMMs are created with a string of states per class that each share the same GMM.
  - Control minimum time of each segment with the number of strings in each string

Model Based Detection



Model Based Detection

- Performing a Viterbi decoding run using HMM results in the segmentation and classification of an audio file.
- Advantage:
  - Very easy to add classes.
- Disadvantage
  - The GMMs need to be trained on some training set
  - Acoustic mismatch between training and testing data sets

Model Based Detection

- How to do it without a training set?
- Can one use the data itself during classification?

SAD

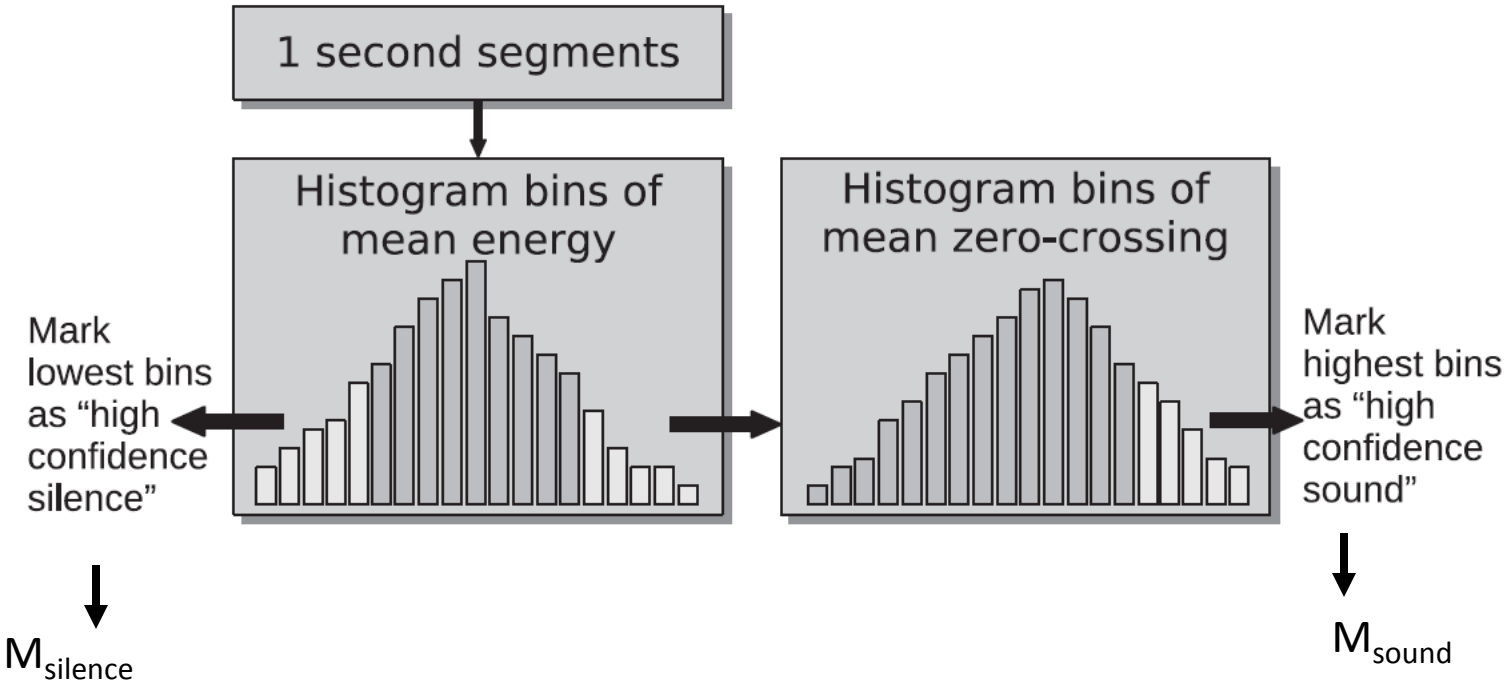
Step1. Bootstrapping Speech/Silence

- Perform initial segmentation using some standard model-based algorithm
  - Two parallel HMM: initial models $M_{non\text{-}speech}$ and M_{speech}
  - Diagonal covariance
  - Minimum of 30 states for silence and 75 states for speech
- Feature extraction
  - 12 MFCCs
  - Zero-crossing rates
  - 1^{st} and 2^{nd} derivatives
  - 39-D feature vector every 32 ms window/10ms overlap
- Data segmented into sets of speech / non-speech regions.

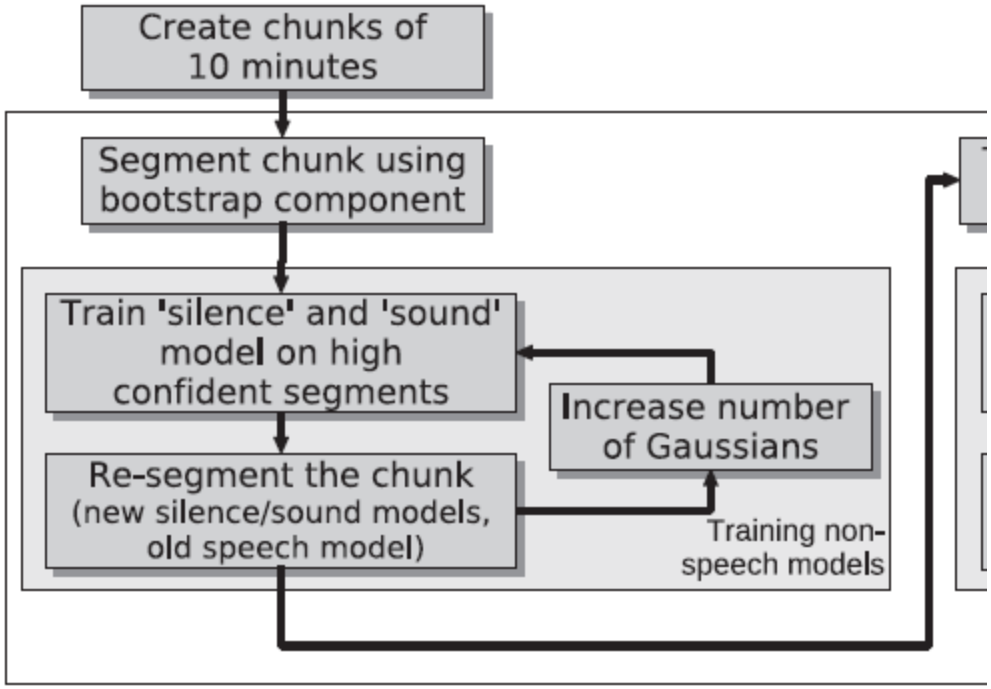
SAD

Step2. Training Models for Non-speech

- Non-speech (silence + sound) model trained from data
  - Evaluate confidence score on segments classified as non-speech.
  - Normalize all segments longer than 1sec to 1sec intervals



SAD


Create chunks of 10 minutes
Segment chunk using bootstrap component
Train 'silence' and 'sound' model on high confident segments
Increase number of Gaussians
Re-segment the chunk (new silence/sound models, old speech model)
Training non-speech models

SAD

Step2. Training Models for Non-speech

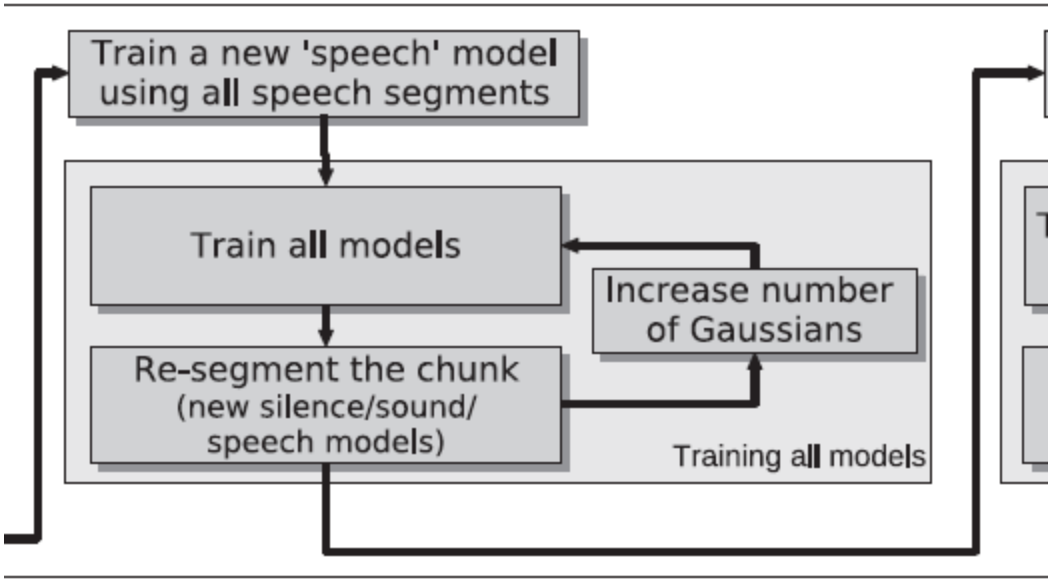
- Use $M_{silence}$ and M_{sound} and M_{speech} to reclassify data.
  - Data assigned to sound and silence models are merged
- $M_{silence}$ and M_{sound} trained on input data, but M_{speech} trained on outside data.
  - $M_{silence}$ and M_{sound} will likely give higher likehood to *all* data compared to M_{speech}
  - samples pulled from speech model are dropped
- We are more confident about the models
  - Evaluate confidence score on remaining data via lower threshold.
  - Additional Gaussians can be used to train GMM.
- Iterate the above process three times.
  - No data from Speech model has been moved to Sound model

SAD

Step3: Training all three models

- $M_{silence}$ and M_{sound} are well trained
  - All non-speech segments are likely be correctly classified.
- M_{speech} can be trained with all remaining data
- Retrain $M_{silence}$, M_{sound}, M_{speech} together by increasing the number of Gaussians at each step until hit threshold.

SAD


Train a new 'speech' model using all speech segments
Train all models
Increase number of Gaussians
Re-segment the chunk (new silence/sound/ speech models)
Training all models

SAD

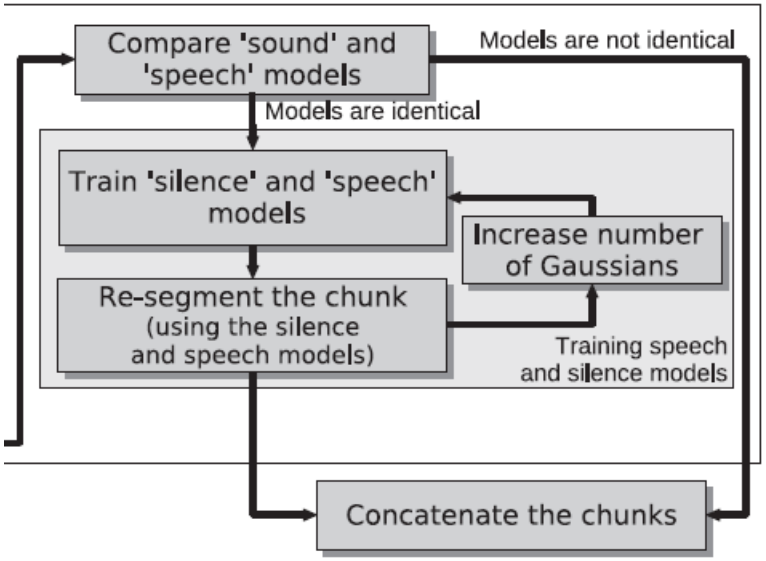
Step4: Training Speech and Silence Models

- Algorithm not well suited for data that contains only *speech* and *silence* with no *non-speech sound.*
  - M_{sound} will be trained on misclassfied speech
  - After a couple of iterations, M_{sound} and M_{speech} becomes competing models
  - Use Bayesian Information Criterion to check for model similarity

$$S(i,j) = \mathcal{L}(x_{i\cup j}|\Theta_{i\cup j}) - \mathcal{L}(x_i|\Theta_i) - \mathcal{L}(x_j|\Theta_j)$$

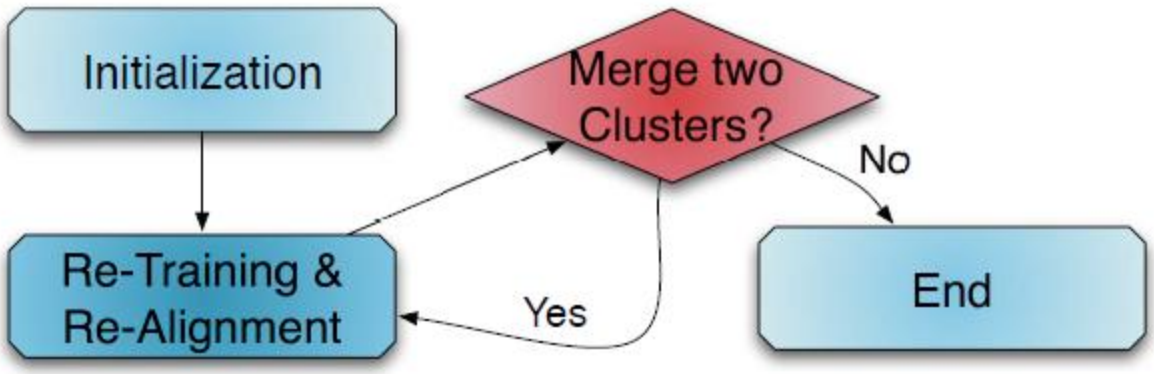
  - If *S(i, j) > 0,* replace with a single speech mode.

SAD


Compare 'sound' and 'speech' models
Models are not identical
Models are identical
Train 'silence' and 'speech' models
Increase number of Gaussians
Re-segment the chunk (using the silence and speech models)
Training speech and silence models
Concatenate the chunks

Speaker Segmentation and Clustering

- Agglomerative hierarchical clustering
  - Start with large number of clusters
  - Iterative procedure for cluster merging, model re-training, realignment.



Initialization

- We want to
  - estimate *k* the number of clusters
  - estimate *g* the number of Gaussians per GMM

Step 1 Pre-clustering

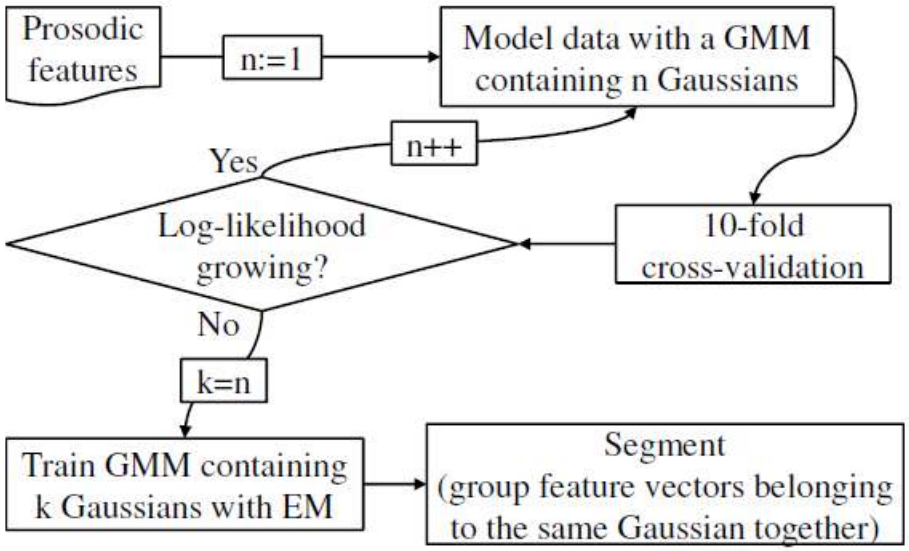
- Extract long-term acoustic features with good speaker discrimination.
  - 100 pitch values and 80 formants /second
  - Hamming window size of 1000 ms
  - Speech/non-speech segment less than 2000ms untouched, larger than 2000ms split to 1000ms segments.
  - Trade-off between accurate estimation of features or a large number of feature vectors.

Category	Short description
pitch	median of the pitch
pitch	minimum of the pitch
pitch	mean of the pitch tier
formants	standard deviation of the 4th formant
formants	minimum of the 4th formant
formants	mean of the 4th formant
formants	standard deviation of the 5th formant
formants	minimum of the 5th formant
formants	mean of the 5th formant
harmonics	mean of the harmonics-to-noise ratio
formant	mean of the formant dispersion
pitch	mean of the pointprocess of the periodicity contour

Intialization

Step 1 Pre-clustering

- Feature vectors clustered with diagonal covariances
  - Over-estimate the number of initial clusters
  - Merging will only reduce clusters



Initialization

- Adaptive seconds per Gaussian:
  - Number of seconds of data available per Gaussian for training
  - *k* should be chosen in relation to the number of different speakers
  - *g* is related to the total amount of speech
  - Use linear regression to estimate *g*

$$secpergauss = 0.01 \cdot \text{speech in seconds} + 2.6$$

$$g = \frac{\text{speech in seconds}}{secpergauss \cdot k}$$

Core Algorithm

- Model
  - HMM to capture temporal structure of acoustic observations
  - GMM as emission probabilities. State represent different speakers.
- Agglomerative Hierarchical Clustering
  - Iterative algorithm
  - Compares clusters via metric, merge ones that are similar

Core Algorithm

Step 1: Model retraining and re-segmentation

- Given speech, goal is to generate speaker models and segment data without prior information.
- Iterative procedure (like EM)
  - Training based on current segmentation
    - Each frame is assigned to an single state
    - Using all the segments belonging to state *k*, update GMM using standard EM.
  - Recompute segmentation based on updated model.
    - Using Viterbi algorithm
- HMM need to remain in the same state for at least 2.5 seconds (min duration of speech of 250 samples).
  - To ensure the clusters are not modeling small units such as phones.
  - Each speaker takes the floor for at least that amount of time.

Core Algorithm

Step2: Model merging:

- Each cluster will correspond to one speaker but a speaker will have many clusters.
- Metric to determine if two clusters should be merged.
- Model selection problem:
  - Given two clusters, are the two separate models better than a joint model.
  - Measure the change in BIC score.

$$S(i,j) = \mathcal{L}(x_{i\cup j}|\Theta_{i\cup j}) - \mathcal{L}(x_i|\Theta_i) - \mathcal{L}(x_j|\Theta_j)$$

  - Decision rule: *S(i, j) > 0 then merge, otherwise not merge*

Core Algorithm

Step2: Model merging:

- At each iteration, merge the pair with largest *S(i, j).*
- How to merge:
  - The sum of the two merged GMM
  - Initialize each mixture with the same mean and variance as the original
  - Mixture weights re-scaled to sum to one.

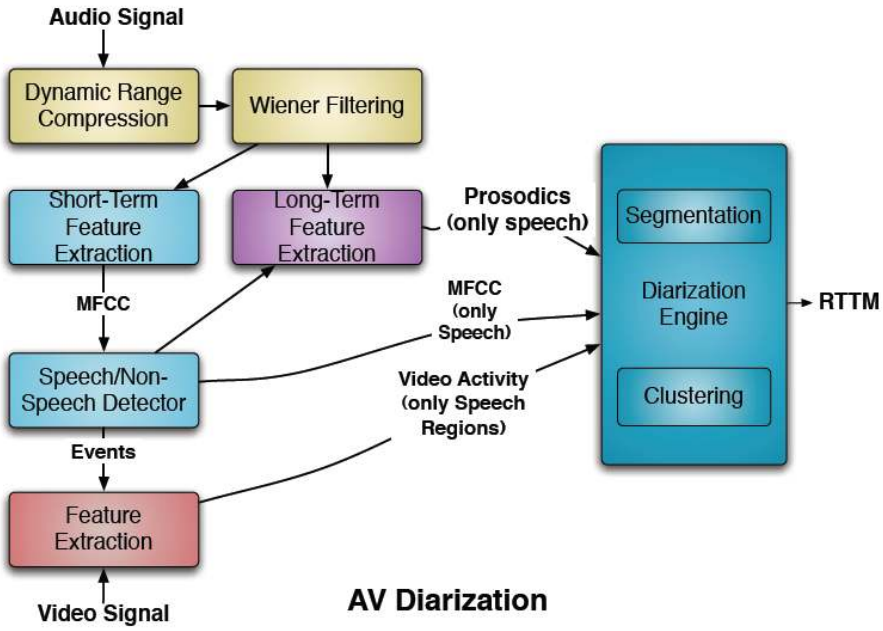
Step 4 Stopping criteria:

- No more merging required when all delta-BIC are negative.
- Final segmentation based on current cluster models.

Audiovisual Diarization

- One distant mic and close up camera.
  - MFCC (same as before)
  - Prosodic: 10 features
  - Video: motion vector magnitudes over estimated skin blocks.
  - Alpha = 0.75, beta = 0.1

$$\hat{\mathcal{L}}(x[i]|\Theta_k) \doteq \alpha \cdot \mathcal{L}(x_{MFCC}[i]|\Theta_{MFCC,k}) + \beta \cdot \mathcal{L}(x_{pros}[i]|\Theta_{pros,k}) + (1-\alpha-\beta) \cdot \mathcal{L}(x_{vid}[i]|\Theta_{vid,k})$$



Results

System	Condition	Speech Non-Speech Error Rate	Diarization Error Rate
Batch Audio	adm	6.43	28.52
	mm3a	6.29	28.32
	mdm	4.92	17.24
	sdm	5.92	31.30
Online Audio	mdm	7.94	39.27
	sdm	15.03	44.61
Audiovisual	sdm	6.89	32.56