UNIVERSITY OF CAMBRIDGE

MLMI14: Speech Synthesis

Mark Gales

Lent 2021

Sequence Mapping Process

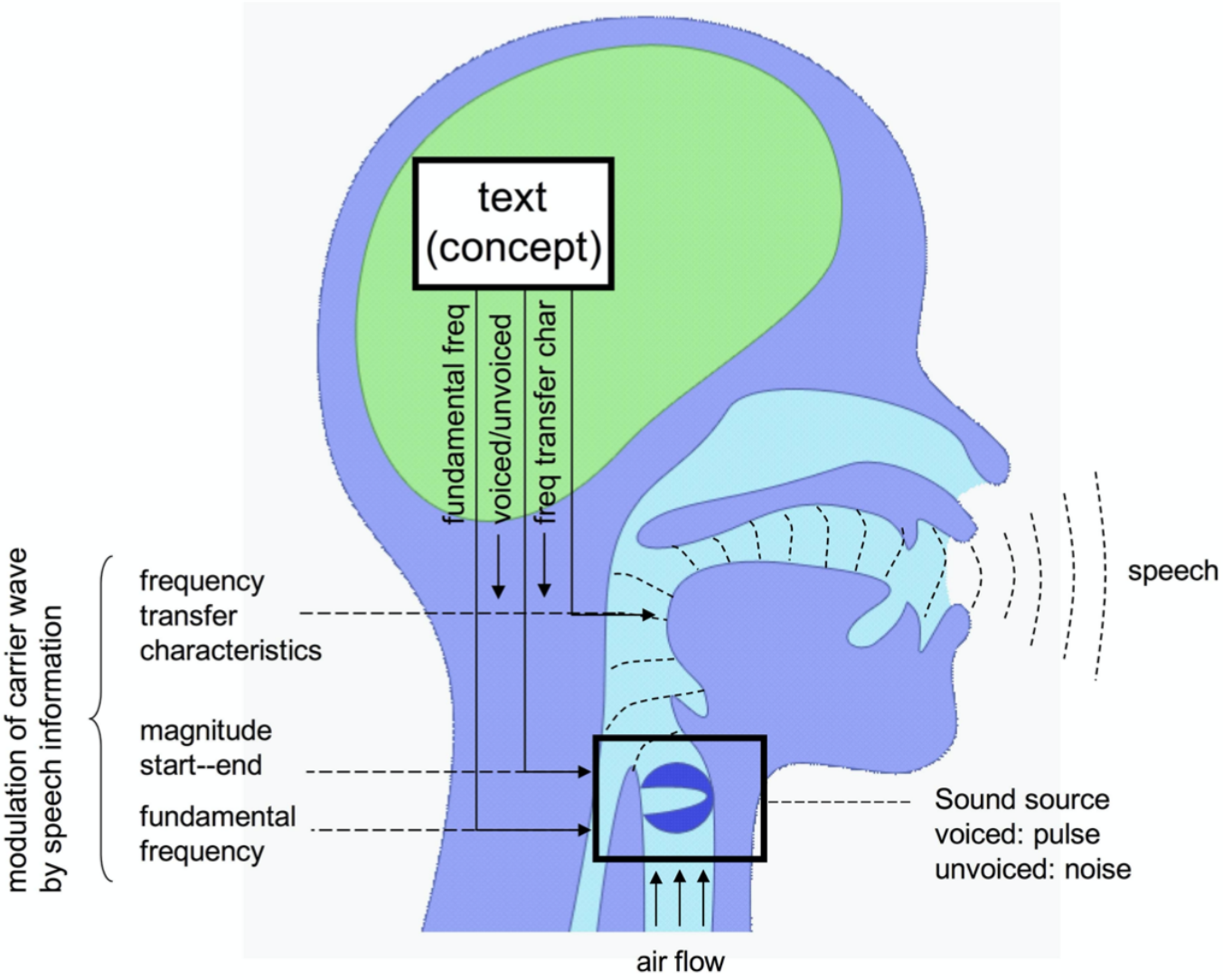
- For speech and language processing need to handle variable length sequences
  - the lengths of the source and target sequences not necessarily the same
  - number of possible sequences vast
  - often sequence-to-sequence mappings required
- Automatic Speech Recognition (ASR)

  Speech (continuous) → Text (discrete)
- Machine Translation (MT)

  Text (discrete) → Text (discrete)
- Text-to-Speech Synthesis (TTS)

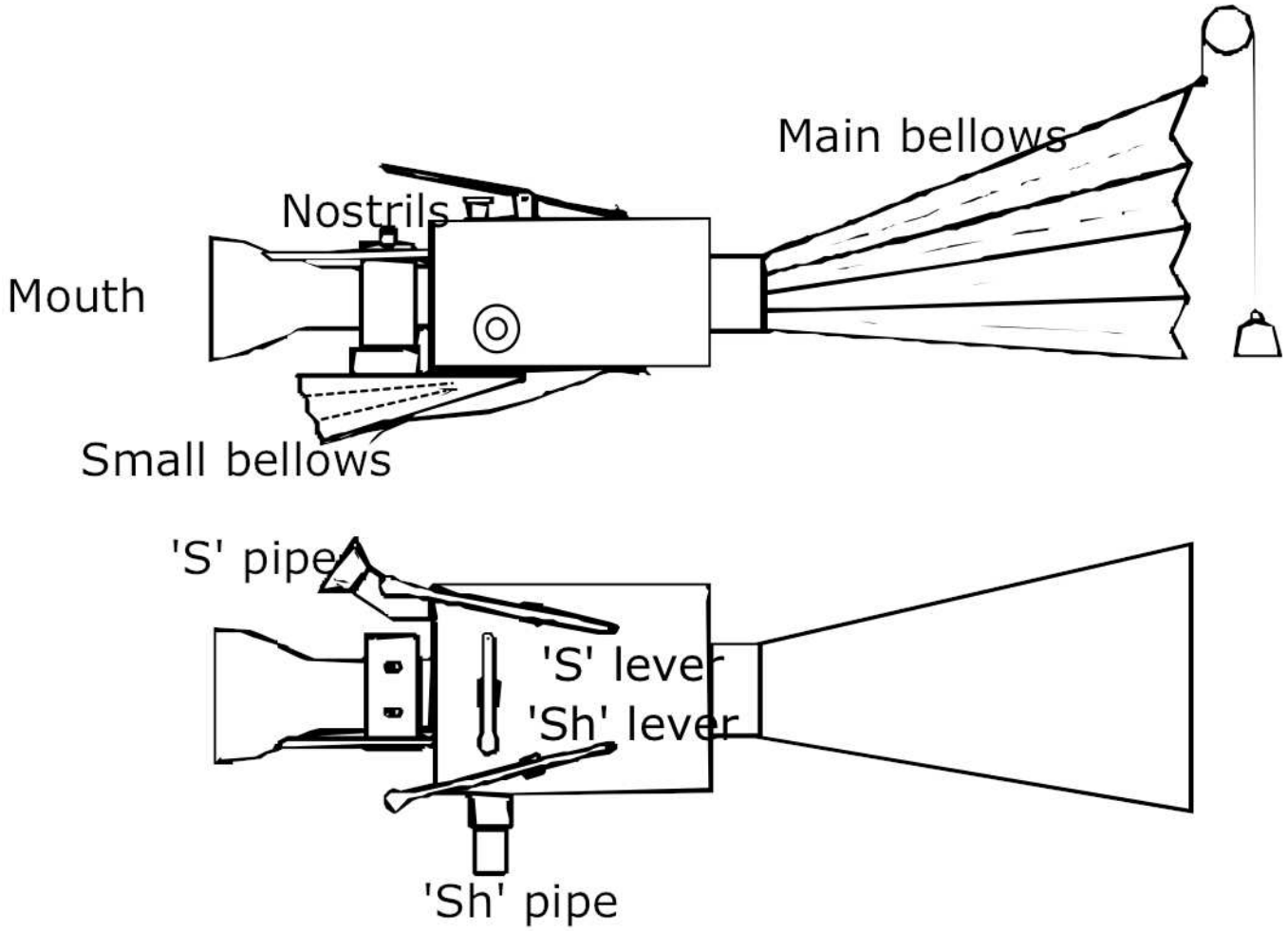
  Text (discrete) → Speech (continuous)

Speech Production

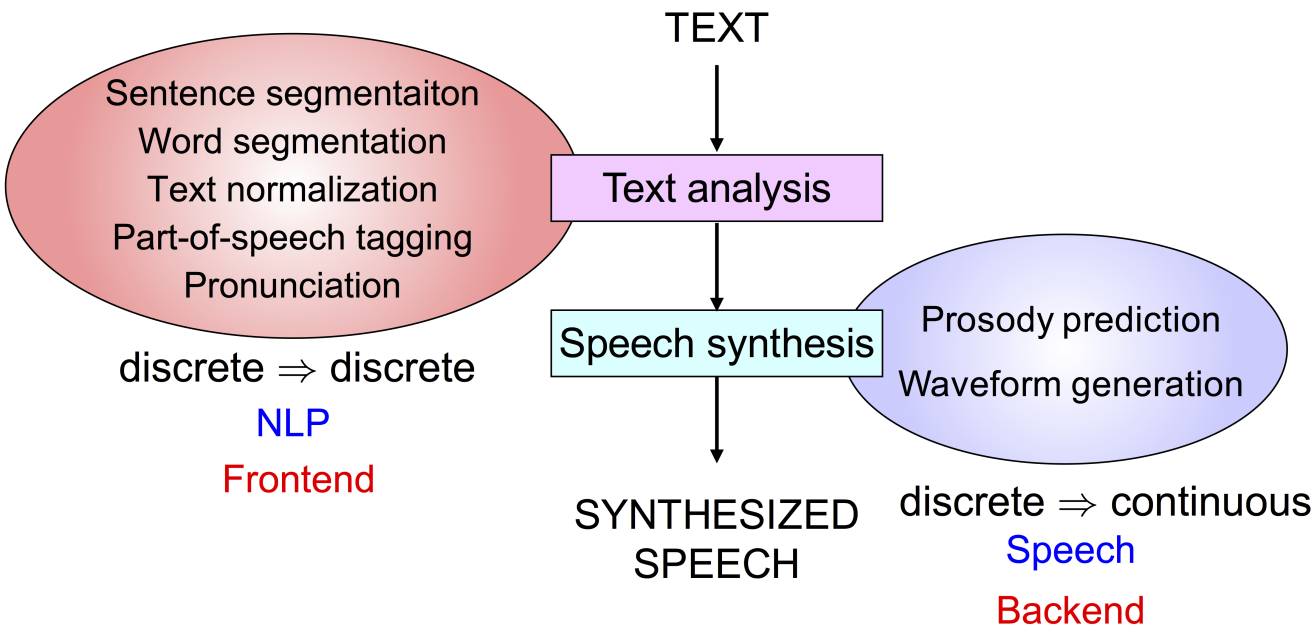


Speech Synthesis: von Kempelen (1791)

- Long history of development:



Speech Synthesis Processing Pipeline



- Typical processing pipeline: combines
  - natural language processing (NLP)
  - signal processing
  - machine learning (for statistical speech synthesis)

Text Normalisation

- Not as trivial as you might imagine ...
  - abbreviation expansion: standard abbreviations
  - numbers: dates, weights etc
  - specialist vocabulary: chemical symbols, texting ...
- Example mappings

St.	→	Street
St.	→	Saint
Mr.	→	Mister
£1984	→	one thousand, nine hundred and eighty four pounds
1984	→	nineteen eighty four
1984	→	one thousand nine hundred and eighty four
AT&T	→	A. T. and T.

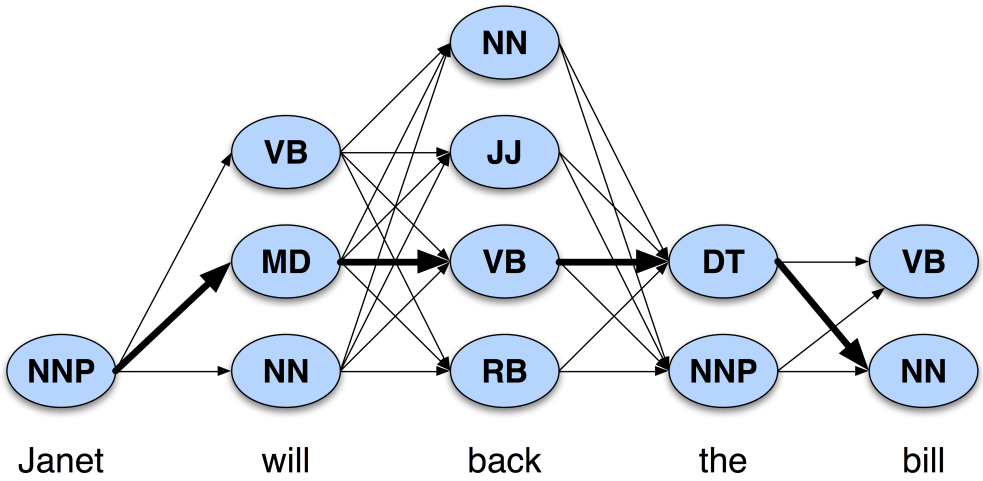
- Normalisation approach needs to be context dependent

Part of Speech (PoS) Tagger

- Given a word-sequence, $\boldsymbol{w}_{1:L}$,extract the PoS tags $\boldsymbol{t}_{1:L}$

  Janet/NNP will/MD back/VB the/DT bill/NN

  - however individual words are ambiguous (from Jurafsky & Martin)



  - pronunciation (and intonation) impacted by PoS

Penn TreeBank PoS Tags

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	*and, but, or*	SYM	symbol	+,%, &
CD	cardinal number	*one, two*	TO	"to"	*to*
DT	determiner	*a, the*	UH	interjection	*ah, oops*
EX	existential 'there'	*there*	VB	verb base form	*eat*
FW	foreign word	*mea culpa*	VBD	verb past tense	*ate*
IN	preposition/sub-conj	*of, in, by*	VBG	verb gerund	*eating*
JJ	adjective	*yellow*	VBN	verb past participle	*eaten*
JJR	adj., comparative	*bigger*	VBP	verb non-3sg pres	*eat*
JJS	adj., superlative	*wildest*	VBZ	verb 3sg pres	*eats*
LS	list item marker	*1, 2, One*	WDT	wh-determiner	*which, that*
MD	modal	*can, should*	WP	wh-pronoun	*what, who*
NN	noun, sing. or mass	*llama*	WP$	possessive wh-	*whose*
NNS	noun, plural	*llamas*	WRB	wh-adverb	*how, where*
NNP	proper noun, sing.	*IBM*	$	dollar sign	*$*
NNPS	proper noun, plural	*Carolinas*	#	pound sign	*#*
PDT	predeterminer	*all, both*	"	left quote	' or "
POS	possessive ending	*'s*	"	right quote	' or "
PRP	personal pronoun	*I, you, he*	(	left parenthesis	[, (, {, <
PRP$	possessive pronoun	*your, one's*	)	right parenthesis	],), }, >
RB	adverb	*quickly, never*	,	comma	,
RBR	adverb, comparative	*faster*	.	sentence-final punc	. ! ?
RBS	adverb, superlative	*fastest*	:	mid-sentence punc	: ; ... – -
RP	particle	*up, off*			

Pronunciation Lexicon

- Manual lexicon used for TTS, but typically more detailed
  - need to select from multiple pronunciations (use e.g. PoS)
  - only one pronunciation can be synthesised!
- Example lexicon entries (from Festival)

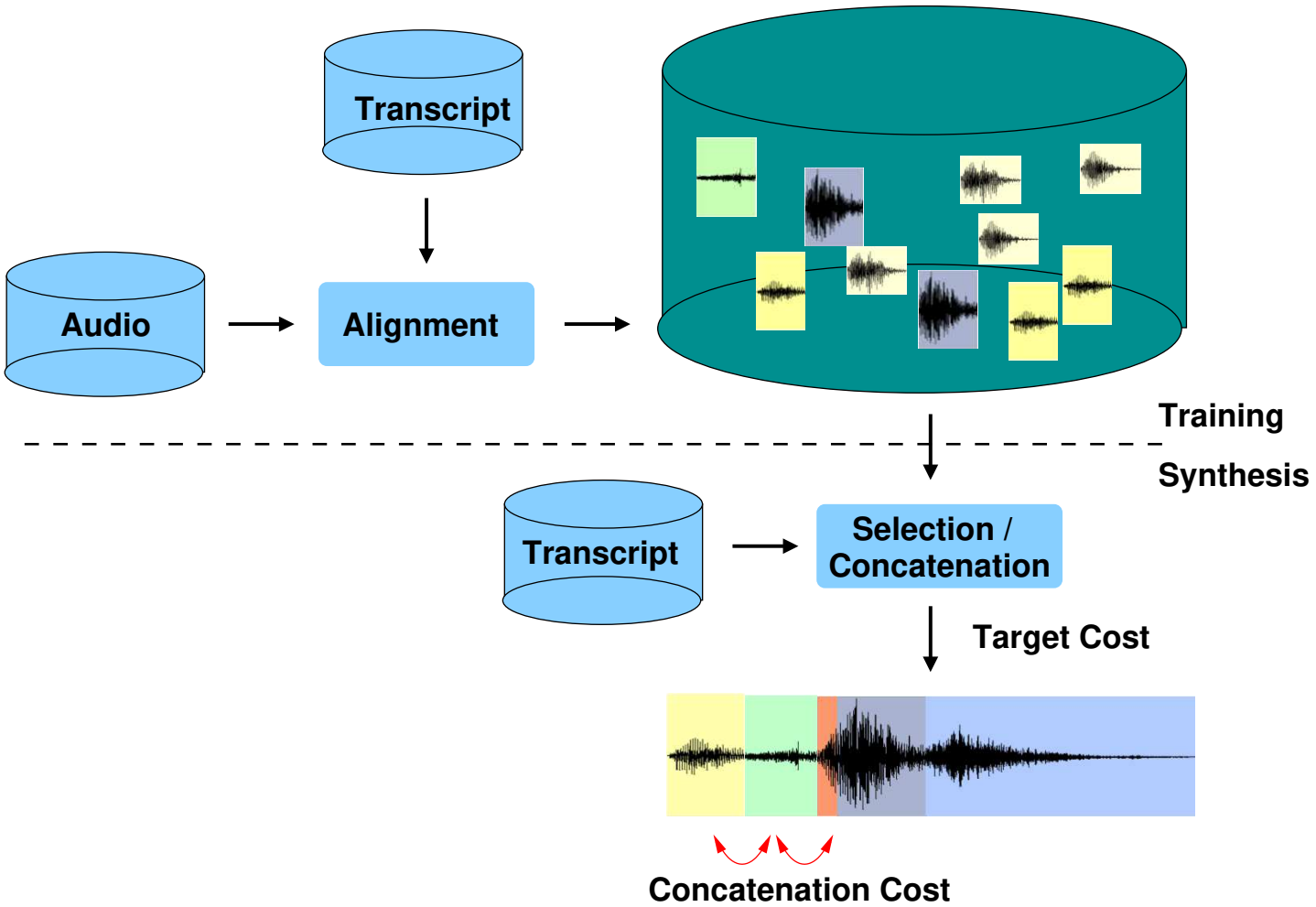
```
( "walkers" n ((( w oo ) 1) (( k @ z ) 0)) )
( "present" v ((( p r e ) 0) (( z @ n t ) 1)) )
( "lives" n ((( l ai v z ) 1)) )
( "lives" v ((( l i v z ) 1)) )
```

  - phone sequence (as in ASR)
  - syllable boundary information
  - stress-markers

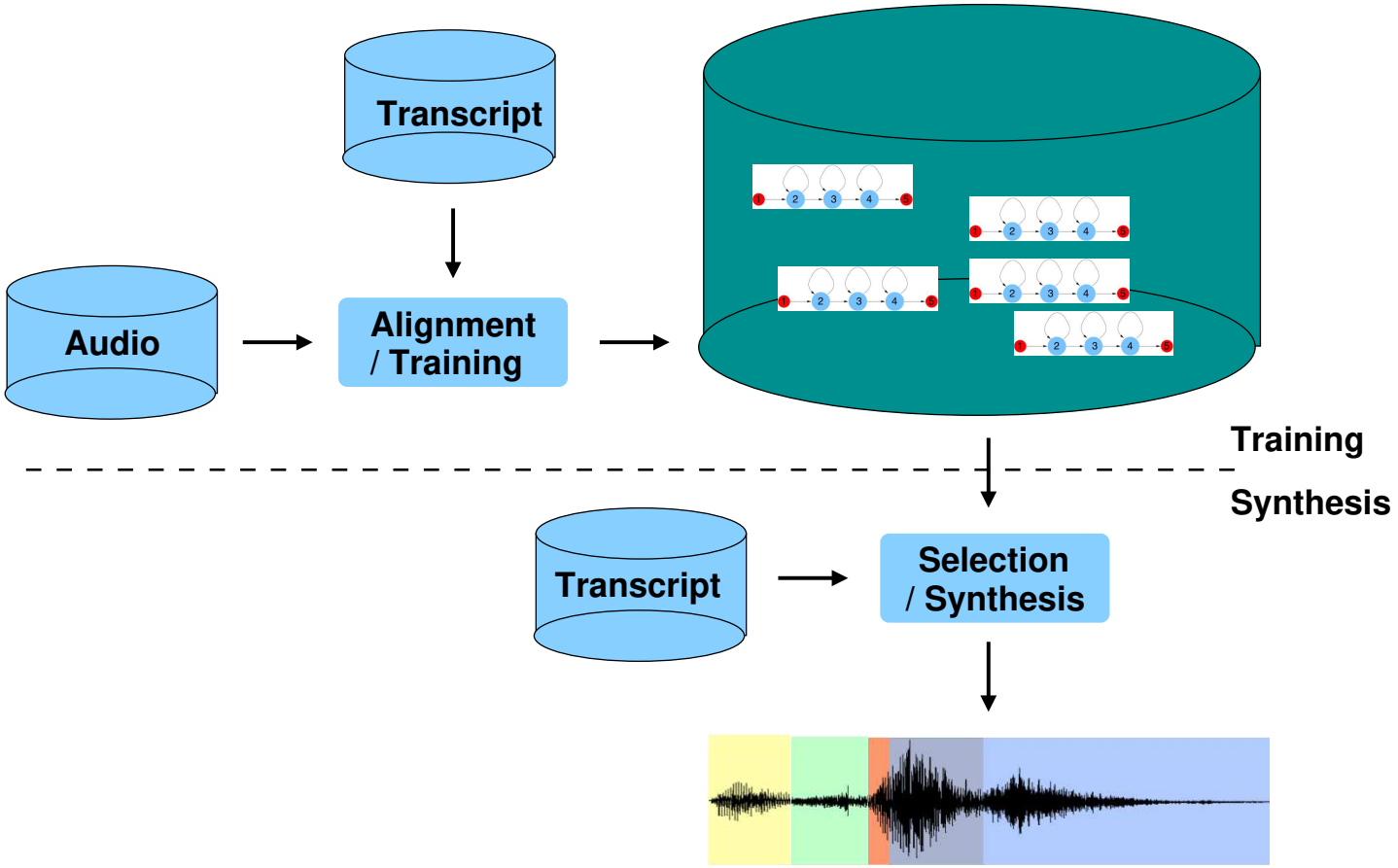
Text Analysis to Speech Synthesis

- Typical output of the text analysis stage is discrete:
  - set of phones
  - syllable positions
  - stress markers
  - word identity and part-of-speech
- These are passed to the synthesis system for:
  - aligning speech data with text
  - unit extraction/selection: concatenative synthesis
  - decision tree construction acoustic model training: statistical synthesis
- For concatenative systems additional information from text analysis
  - duration: target duration for each unit
  - F0/power: intonation pattern for target utterance

Concatenative Speech Synthesis



Statistical Speech Synthesis



Concatenative vs Statistical

- Concatenative Speech Synthesis:
  - high quality segmental accuracy
  - need large corpus of segments to cover target domain
- Statistical Speech Synthesis
  - (potentially very) small footprint
  - flexibility (able to control speaker characteristics)
  - issues with segmental naturalness
- Hybrid Approach: combining attributes of both schemes
  - use statistical approaches for unit selection
  - combine synthesis units with extracted units

Concatenative vs Statistical

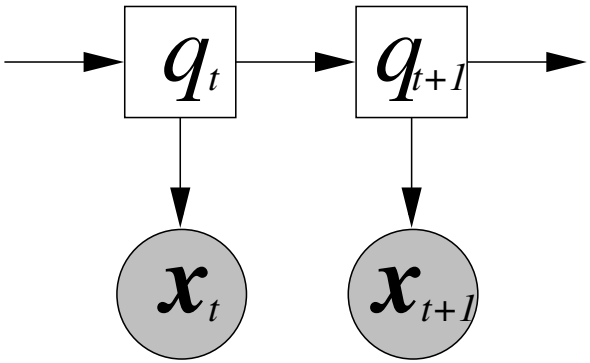
- Concatenative Speech Synthesis:
  - high quality segmental accuracy
  - need large corpus of segments to cover target domain
- Statistical Speech Synthesis
  - (potentially very) small footprint
  - flexibility (able to control speaker characteristics)
  - issues with segmental naturalness
- Hybrid Approach: combining attributes of both schemes
  - use statistical approaches for unit selection
  - combine synthesis units with extracted units

- Then shift to purely deep-learning style approaches: WaveNet
  - removed need for signal processing (and natural language processing)

HMM Trajectory Generation

Hidden Markov Models

- An important sequence model hidden Markov model (HMM)
  - an example of a dynamic Bayesian network (DBN)
  - consider a sequence of observations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$



- discrete latent variables
  - q_t describes discrete state-space
  - conditional independence assumptions

$$P(q_t|q_0, \ldots, q_{t-1}) = P(q_t|q_{t-1})$$
$$p(\boldsymbol{x}_t|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-1}, q_0, \ldots, q_t) = p(\boldsymbol{x}_t|q_t)$$

- The likelihood of the data is

$$p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T) = \sum_{\boldsymbol{q} \in \boldsymbol{Q}_T} P(q_0) \prod_{t=1}^{T} P(q_t|q_{t-1}) p(\boldsymbol{x}_t|q_t)$$

$\boldsymbol{q} = \{q_0, \ldots, q_T\}$ and $\boldsymbol{Q}_T$ is all T-length state sequences

HMM Training

- Maximum Likelihood (ML) training discussed for ASR
  - Expectation-Maximisation (EM) often used

$$\boldsymbol{\mu}_m = \frac{\sum_{t=1}^{T} \gamma_m(t)\boldsymbol{x}_t}{\sum_{t=1}^{T} \gamma_m(t)} \quad \boldsymbol{\Sigma}_m = \frac{\sum_{t=1}^{T} \gamma_m(t)(\boldsymbol{x}_t - \boldsymbol{\mu}_m)(\boldsymbol{x}_t - \boldsymbol{\mu}_m)^{\mathsf{T}}}{\sum_{t=1}^{T} \gamma_m(t)}$$

  - $\gamma_m(t) = P(q_t = m|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T, \boldsymbol{\lambda})$: $\boldsymbol{\lambda}$ "current" parameters
- Feature vector often comprises static/delta/delta-delta

$$\begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{\Delta x}_t \\ \boldsymbol{\Delta}^2 \boldsymbol{x}_t \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{x}_{t+1} - \boldsymbol{x}_{t-1} \\ \boldsymbol{\Delta x}_{t+1} - \boldsymbol{\Delta x}_{t-1} \end{bmatrix}$$

  - normalisation terms for delta and deltas ignored

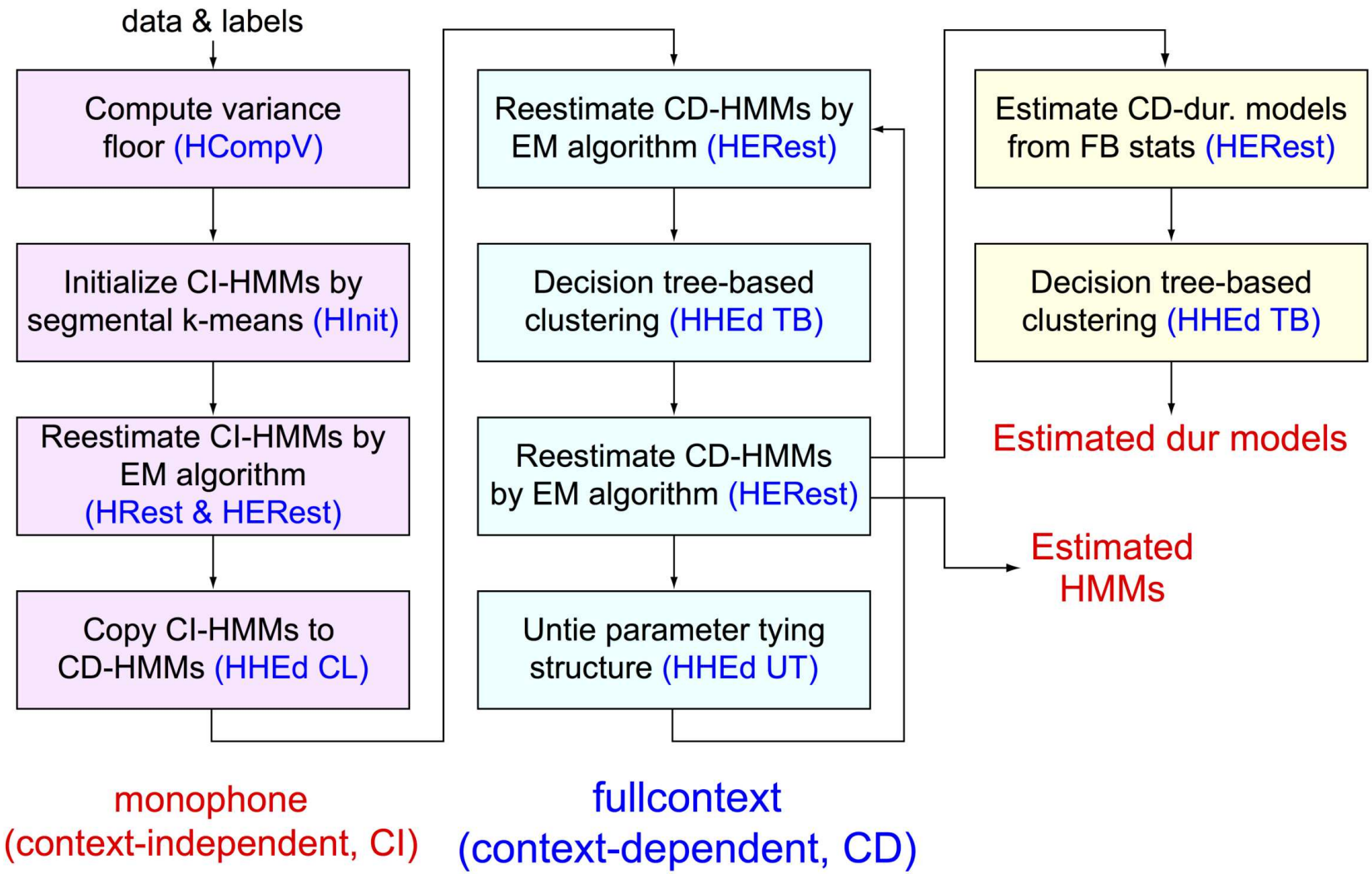
Synthesis HMMs

- Configurations for synthesis similar to ASR - major differences
  - (often) single Gaussian component per state
  - rich decision tree questions and deep trees
  - state-position roots of decision trees
  - multiple streams
  - duration model
  - multi-space distributions
- Baseline feature vector - 5ms frame-rate, 25ms window-size

$$\begin{bmatrix} \text{mel-cepstrum} - 40 - 60\,\text{dim} \\ \text{log-F0} - 1\,\text{dim} \end{bmatrix}$$

  - delta and delta-delta parameters can be added as usual

Synthesis HMM Training Process



Multiple Stream Systems

- Multiple stream states are a standard features in HTK
  - rarely used for speech recognition
  - used for statistical speech synthesis with HMMs
- Two multi-stream options - each has stream weight w_s
  - multiple component streams

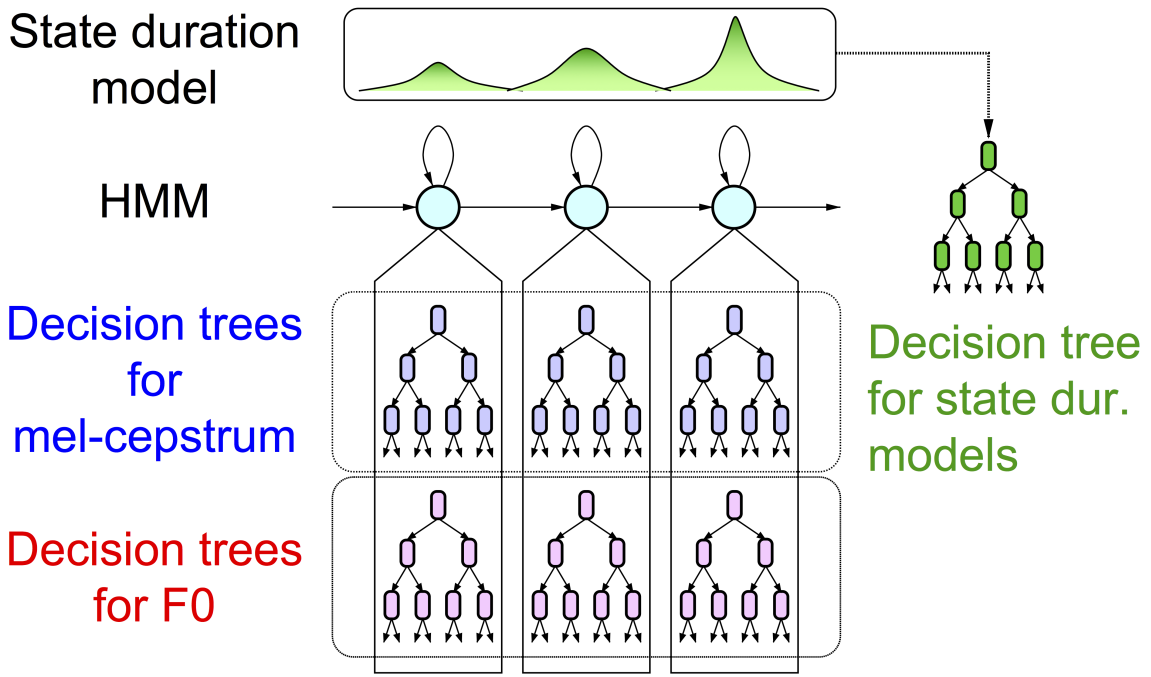
$$p\left(\left[\begin{array}{ccc} \boldsymbol{x}_t^{(1)\mathsf{T}} & \ldots & \boldsymbol{x}_t^{(S)\mathsf{T}} \end{array}\right]^{\mathsf{T}} \middle| q_t\right) \propto \prod_{s=1}^{S} \left(\sum_{m=1}^{M_s} c_{q_t}^{(sm)} \mathcal{N}(\boldsymbol{x}_t^{(s)}; \boldsymbol{\mu}_{q_t}^{(sm)}, \boldsymbol{\Sigma}_{q_t}^{(sm)}) \right)^{w_s}$$

  - multiple decision trees

$$p\left(\left[\begin{array}{ccc} \boldsymbol{x}_t^{(1)\mathsf{T}} & \ldots & \boldsymbol{x}_t^{(S)\mathsf{T}} \end{array}\right]^{\mathsf{T}} \middle| q_t^{(1)}, \ldots, q_t^{(S)}\right) \propto \prod_{s=1}^{S} \left(\mathcal{N}(\boldsymbol{x}_t^{(s)}; \boldsymbol{\mu}_{q_t^{(s)}}^{(s)}, \boldsymbol{\Sigma}_{q_t^{(s)}}^{(s)}) \right)^{w_s}$$

where $q_t^{(s)}$ indicates the context for stream s at time t

Multiple Decision Trees for Synthesis



- Root of decision tree is a state position (phone-state in ASR)
  - separate decision trees for spectral parameters/(log)F0
  - separate trees also used for delta and delta-delta (log)F0
  - separate decision tree for duration

Duration Modelling

- The standard duration (τ) model for HMMs is an exponential decay

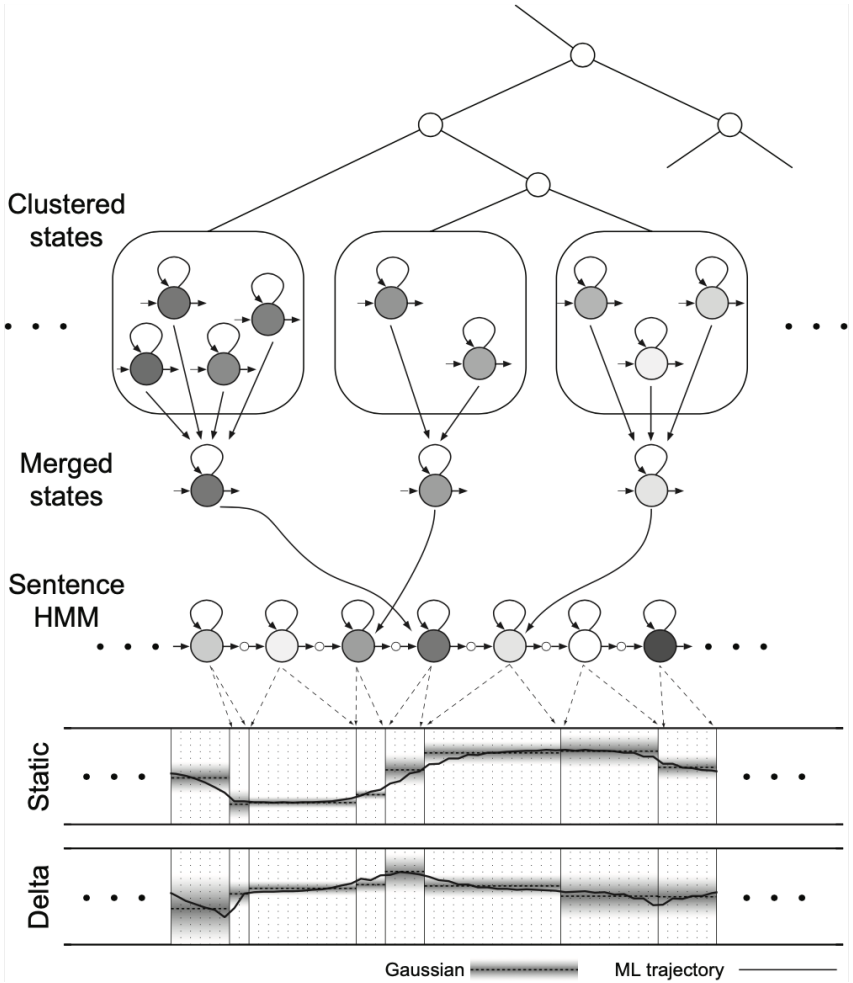
$$P(\tau|s_i) = a_{ii}^{\tau-1}(1 - a_{ii})$$

  - not a realistic model for actual durations
  - efficient as Viterbi can be directly applied
  - important for synthesis to get duration correct
- Hidden Semi Markov Models: explicit duration model
  - for synthesis single Gaussian used for duration

$$p(\tau|s_i) = \mathcal{N}(\tau; \boldsymbol{\mu}_i^{\mathtt{dur}}, \boldsymbol{\Sigma}_i^{\mathtt{dur}})$$

  - can be trained on final iteration, or integrated in training

Synthesis Trajectory Generation



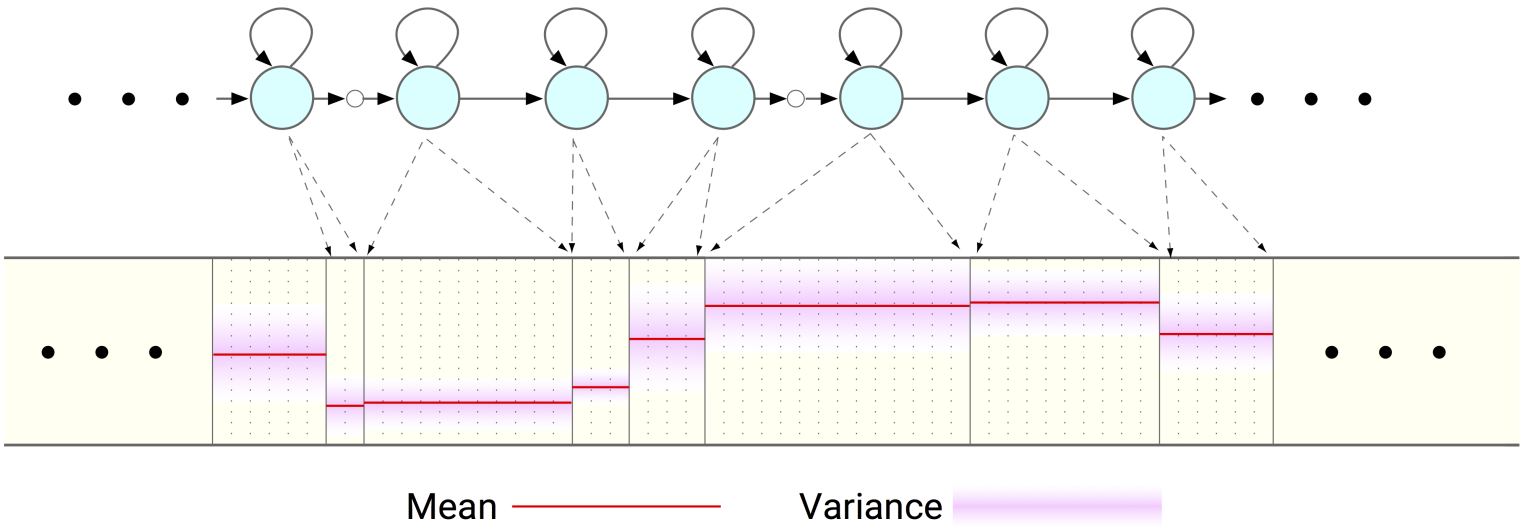
HMM Trajectory Generation

- Text analysis yield a sequence of phones and attributes
  - use decision tree to obtain sentence state sequence $\mathtt{s}_1, \ldots, \mathtt{s}_L$
  - state sequence will be different for each stream (decision tree)
- First stage is to generate the duration for each of the L states
  - simplest: take the mean of the Gaussian for state $\mathtt{s}_i$: $\boldsymbol{\mu}_i^{\mathtt{dur}}$
  - total sentence length is

$$T = \sum_{i=1}^{L} \boldsymbol{\mu}_i^{\mathtt{dur}}$$

- Yields T length vector, $\boldsymbol{q}$, indicating the state at each time
  - $\boldsymbol{q}$ can now used to generate the parameter trajectories

HMM Trajectory - Standard Model



- Given the sequence of T states $\boldsymbol{q}$ associated with a word sequence
  - need to generate distribution over the trajectory
  - assume that observation vector $\boldsymbol{x}$ is just the cepstra then

$$p(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T|\boldsymbol{q}) = \prod_{t=1}^{T} p(\boldsymbol{x}_t|q_t) = \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{\mu}_{q_t}, \boldsymbol{\Sigma}_{q_t})$$

HMM Trajectory Generation

- Given the distribution $p(\boldsymbol{x}_{1:T}|\boldsymbol{q})$ need to generate a trajectory
  - standard solution select maximum likelihood estimate

$$\hat{\boldsymbol{x}}_{1:T} = \arg\max \left\{ p(\boldsymbol{x}_{1:T}|\boldsymbol{q}) \right\}$$

- Using the standard distribution yields

$$\hat{\boldsymbol{x}}_{1:T} = \arg\max \left\{ \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{\mu}_{q_t}, \boldsymbol{\Sigma}_{q_t}) \right\} = \begin{bmatrix} \boldsymbol{\mu}_{q_1}^{\mathsf{T}} & \dots & \boldsymbol{\mu}_{q_T}^{\mathsf{T}} \end{bmatrix}^{\mathsf{T}}$$

  - most likely sequence simply the sequence of means

HMM Trajectory - Static/Delta

- Modify the "observations" to include delta parameters, $\boldsymbol{\Delta x}$,
  - possible to write for a single observation

$$\overline{\boldsymbol{x}}_t = \begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{\Delta x}_t \end{bmatrix} = \begin{bmatrix} 0 & +\mathbf{I} & 0 \\ -\mathbf{I} & 0 & +\mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_{t-1} \\ \boldsymbol{x}_t \\ \boldsymbol{x}_{t+1} \end{bmatrix}$$

$$p\left(\begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{\Delta x}_t \end{bmatrix} \middle| q_t \right) = \mathcal{N}\left(\overline{\boldsymbol{x}}_t ; \overline{\boldsymbol{\mu}}_{q_t}, \overline{\boldsymbol{\Sigma}}_{q_t} \right)$$

  - and for the complete sequence

$$\begin{bmatrix} \vdots \\ \boldsymbol{x}_t \\ \boldsymbol{\Delta x}_t \\ \boldsymbol{x}_{t+1} \\ \boldsymbol{\Delta x}_{t+1} \\ \vdots \end{bmatrix} = \begin{bmatrix} & \vdots & \vdots & \vdots & \vdots & \\ \dots & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \dots \\ \dots & -\mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \dots \\ \dots & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \dots \\ \dots & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{I} & \dots \\ & \vdots & \vdots & \vdots & \vdots & \end{bmatrix} \begin{bmatrix} \vdots \\ \boldsymbol{x}_{t-1} \\ \boldsymbol{x}_t \\ \boldsymbol{x}_{t+1} \\ \boldsymbol{x}_{t+2} \\ \vdots \end{bmatrix} = \mathbf{W}\boldsymbol{x}_{1:T}$$

  - need the distribution of $\boldsymbol{x}_{1:T}$

HMM Trajectory (cont)

- Possible to write

$$p(\mathbf{W}\boldsymbol{x}_{1:T}|\boldsymbol{q}) = \mathcal{N}\left(\mathbf{W}\boldsymbol{x}_{1:T}; \overline{\boldsymbol{\mu}}_{\boldsymbol{q}}, \overline{\boldsymbol{\Sigma}}_{\boldsymbol{q}}, \right)$$

where

$$\overline{\boldsymbol{\mu}}_{\boldsymbol{q}} = \begin{bmatrix} \vdots \\ \overline{\boldsymbol{\mu}}_{q_t} \\ \overline{\boldsymbol{\mu}}_{q_{t+1}} \\ \vdots \end{bmatrix}, \overline{\boldsymbol{\Sigma}}_{\boldsymbol{q}} = \begin{bmatrix} & \vdots & \vdots & \vdots & \\ \cdots & \mathbf{0} & \overline{\boldsymbol{\Sigma}}_{q_t} & \mathbf{0} & \cdots \\ \cdots & \mathbf{0} & \mathbf{0} & \overline{\boldsymbol{\Sigma}}_{q_{t+1}} & \cdots \\ & \vdots & \vdots & \vdots & \end{bmatrix}$$

  - but we need a distribution for $\boldsymbol{x}_{1:T}$ not $\mathbf{W}\boldsymbol{x}_{1:T}$

- Note: an example of a product of experts for details see
  - H. Zen, M. J. F. Gales, Y. Nankaku, and K. Tokuda, "Product of experts for statistical parametric speech synthesis," IEEE Trans. Audio, Speech & Language Processing, 2012

HMM Trajectory (cont)

- Expanding out the form of distribution

$$
\begin{aligned}
p(\mathbf{W}\boldsymbol{x}_{1:T}|\boldsymbol{q}) &\propto \exp\left(-\frac{1}{2}(\mathbf{W}\boldsymbol{x}_{1:T}-\overline{\boldsymbol{\mu}}_{\boldsymbol{q}})^{\mathsf{T}}\overline{\boldsymbol{\Sigma}}_{\boldsymbol{q}}^{-1}(\mathbf{W}\boldsymbol{x}_{1:T}-\overline{\boldsymbol{\mu}}_{\boldsymbol{q}})\right) \\
&\propto \exp\left(-\frac{1}{2}\left(\boldsymbol{x}_{1:T}^{\mathsf{T}}\mathbf{W}^{\mathsf{T}}\overline{\boldsymbol{\Sigma}}_{\boldsymbol{q}}^{-1}\mathbf{W}\boldsymbol{x}_{1:T}-2\overline{\boldsymbol{\mu}}_{\boldsymbol{q}}^{\mathsf{T}}\overline{\boldsymbol{\Sigma}}_{\boldsymbol{q}}^{-1}\mathbf{W}\boldsymbol{x}_{1:T}\right)\right)
\end{aligned}
$$

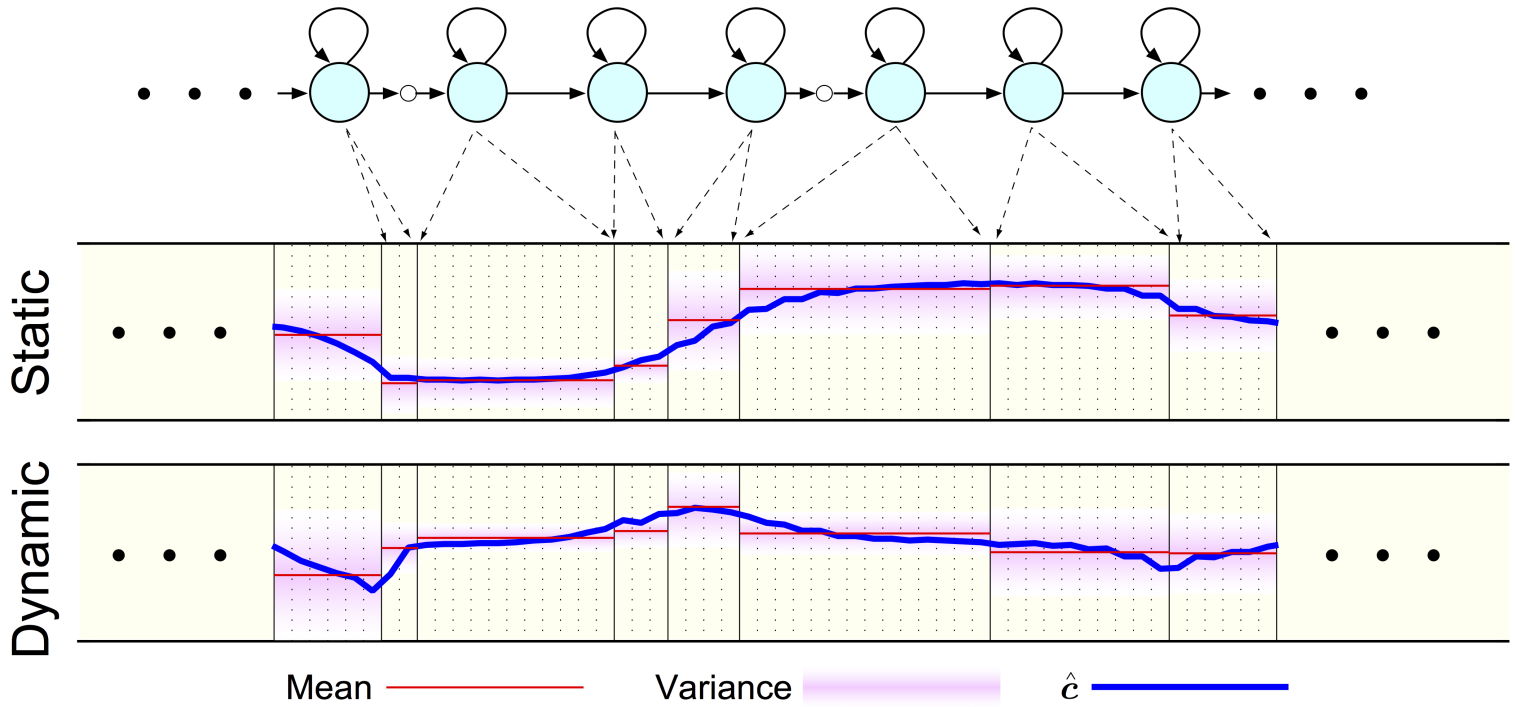
  - Equating co-efficients yields

$$
\begin{aligned}
p(\boldsymbol{x}_{1:T}|\boldsymbol{q}) &= \mathcal{N}\left(\boldsymbol{x}_{1:T}, \left(\mathbf{W}^{\mathsf{T}}\overline{\boldsymbol{\Sigma}}_{\boldsymbol{q}}^{-1}\mathbf{W}\right)^{-1}\mathbf{W}^{\mathsf{T}}\overline{\boldsymbol{\Sigma}}_{\boldsymbol{q}}^{-1}\overline{\boldsymbol{\mu}}_{\boldsymbol{q}}, \left(\mathbf{W}^{\mathsf{T}}\overline{\boldsymbol{\Sigma}}_{\boldsymbol{q}}^{-1}\mathbf{W}\right)^{-1}\right) \\
&= \mathcal{N}(\boldsymbol{x}_{1:T}; \boldsymbol{\mu}_{\boldsymbol{q}}, \boldsymbol{\Sigma}_{q})
\end{aligned}
$$

- Again estimate the ML trajectory

$$\hat{\boldsymbol{x}}_{1:T} = \boldsymbol{\mu}_{\boldsymbol{q}}$$

HMM Trajectory (cont)



- Final covariance matrix not diagonal
  - introduces correlation over time - a smooth trajectory
- Final distribution over the complete sequence is Gaussian
  - able to generate a distribution for any sequence length

Multiple Component Acoustic Models

- Previous estimation has assume a single Gaussian per state
  - possible to generalise to M components per state

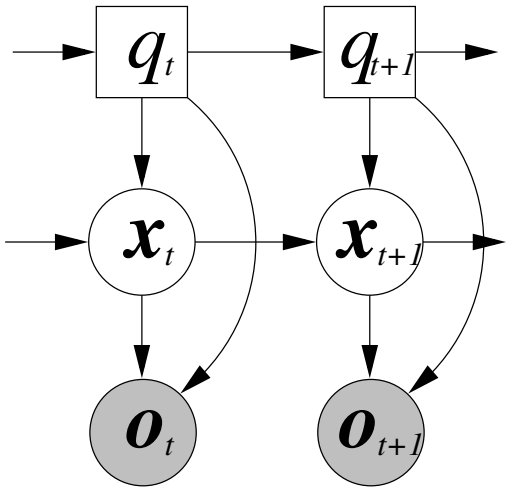
$$p(\overline{\boldsymbol{x}}_t|q_t) = \sum_{m=1}^{M} c_m \mathcal{N}(\overline{\boldsymbol{x}}_t, \overline{\boldsymbol{\mu}}_{q_t}^{(m)}, \overline{\overline{\boldsymbol{\Sigma}}}_{q_t}^{(m)})$$

- Again consider ML estimation of trajectory

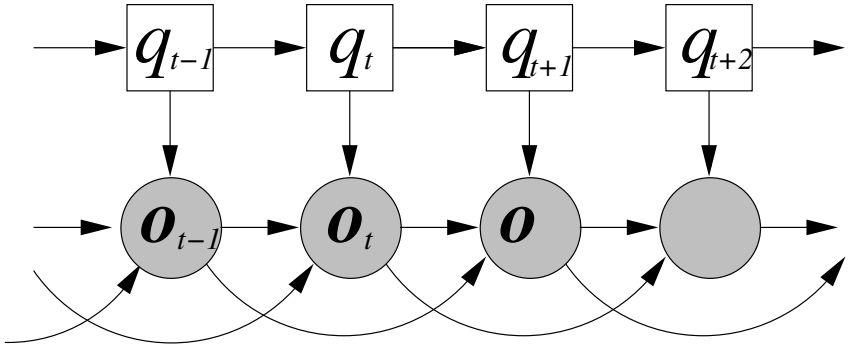
$$\hat{\boldsymbol{x}}_{1:T} = \arg\max \{p(\boldsymbol{x}_{1:T}|\boldsymbol{q}, \boldsymbol{m})\}$$

  - maximisation is over $\boldsymbol{x}_{1:T}$ and $\boldsymbol{m}$
  - $\boldsymbol{m}$ is T dimensional vector of component at each time instance
- Computationally expensive (and limited performance gains)

Alternative Acoustic Models



SLDS



AR-HMM

- Acoustic models also investigated for synthesis:
  - Switching Linear Dynamical Systems:
  - additional continuous latent variable
  - Auto-Regressive HMM:
  - additional dependencies on observations
- Both yield a smoother trajectory (without dynamics)

SLDS Speech Trajectory Modelling

Frames from phrase:
`SHOW THE GRIDLEY'S...`

Legend

- True
- HMM
- SLDS