Speech Synthesis

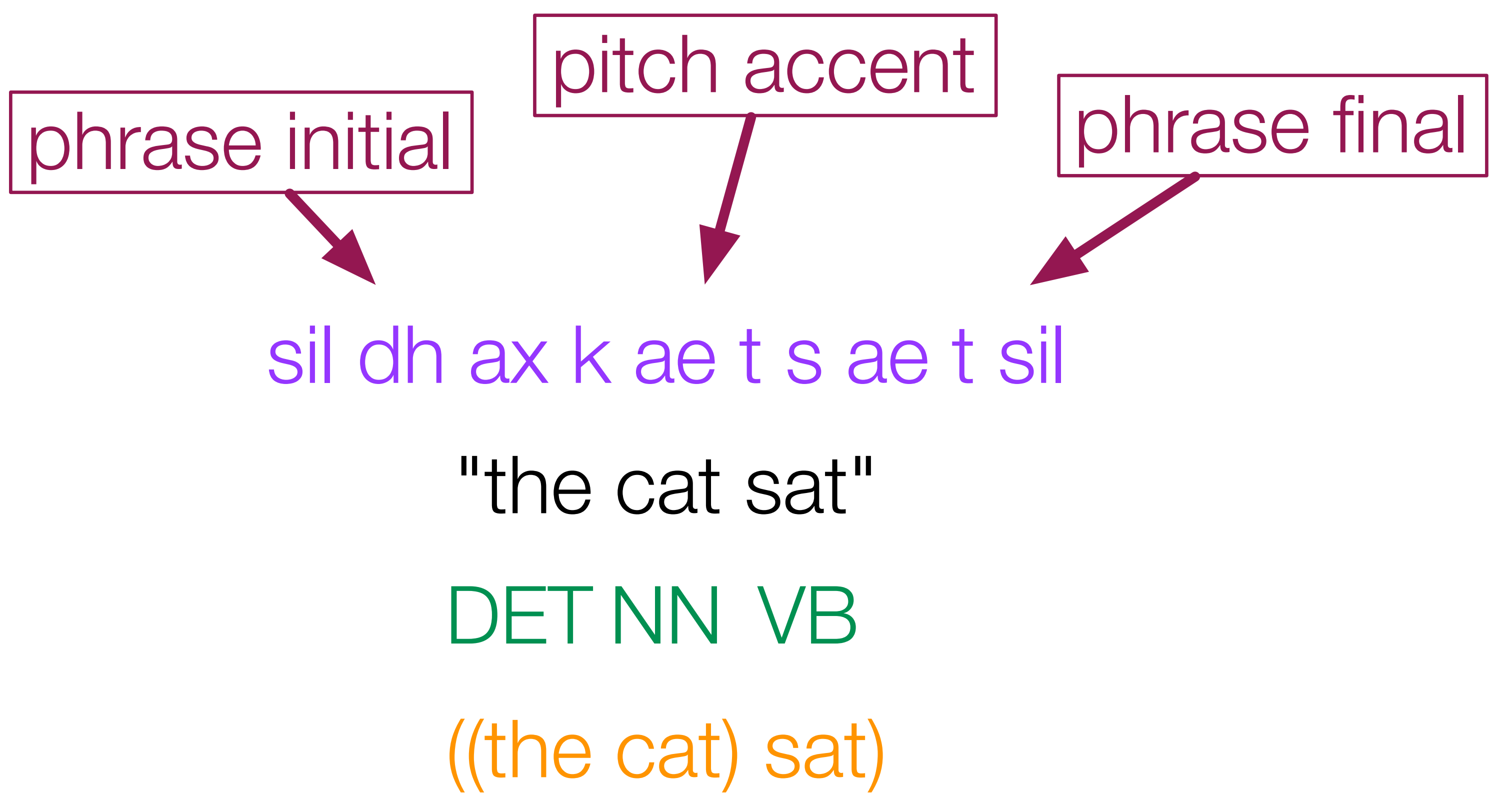
Text-to-speech (TTS)

- **Definition: a text-to-speech system must be**
  - Able to read any text
  - Intelligible
  - Natural sounding

- The first of these puts a constraint on the method we can choose:
  - *playback of whole words or phrases in not a solution*
- The second is actually closer to being a '*solved problem*' than the third

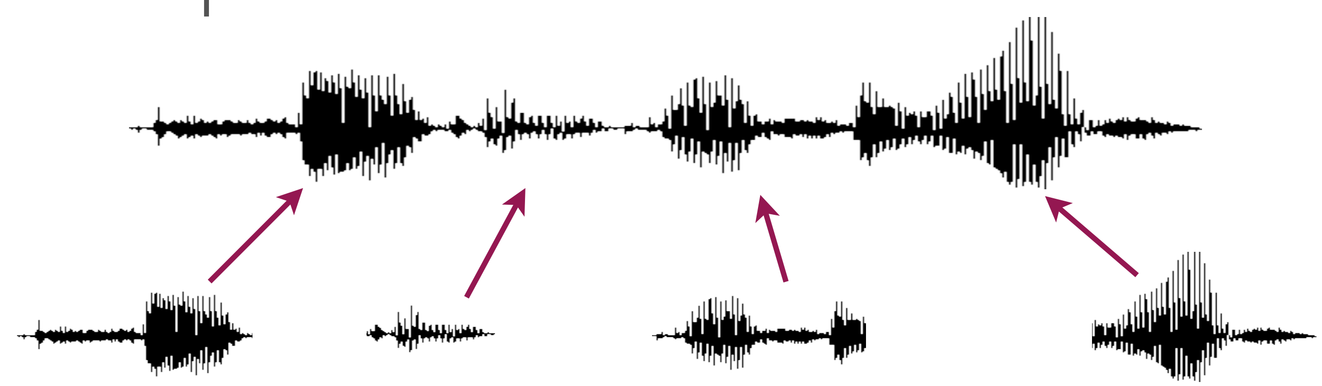
- **A generation task**
  - although not completely clear what objective function we are optimising

From text to linguistic specification

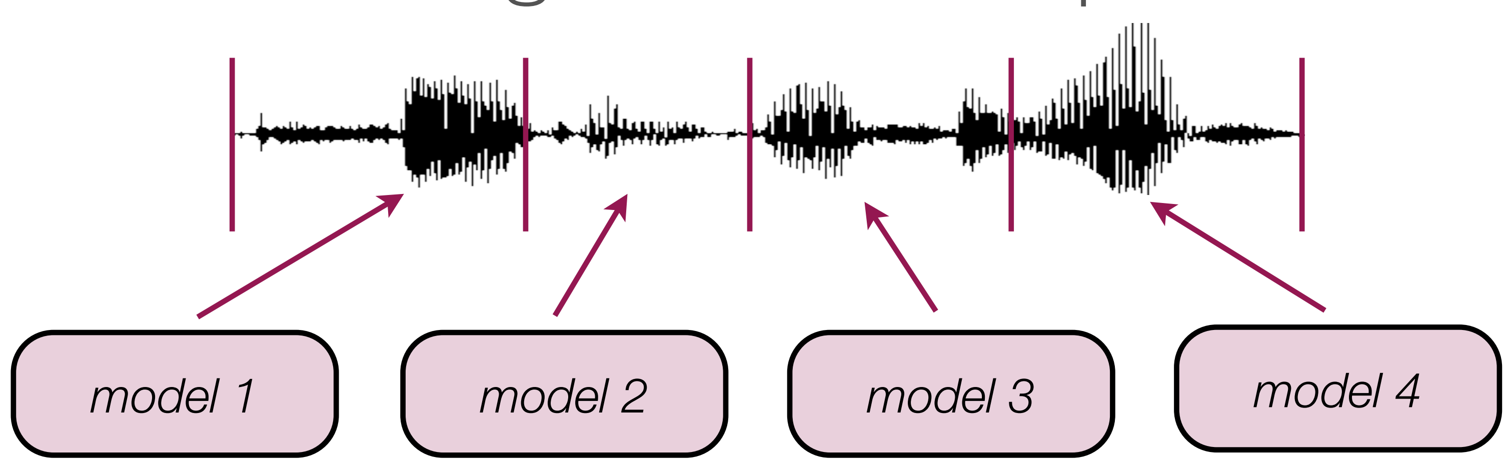


From linguistic specification to a waveform

- **Concatenation** builds up the utterance from units of recorded speech:



- **Generation** uses a model to generate the speech:



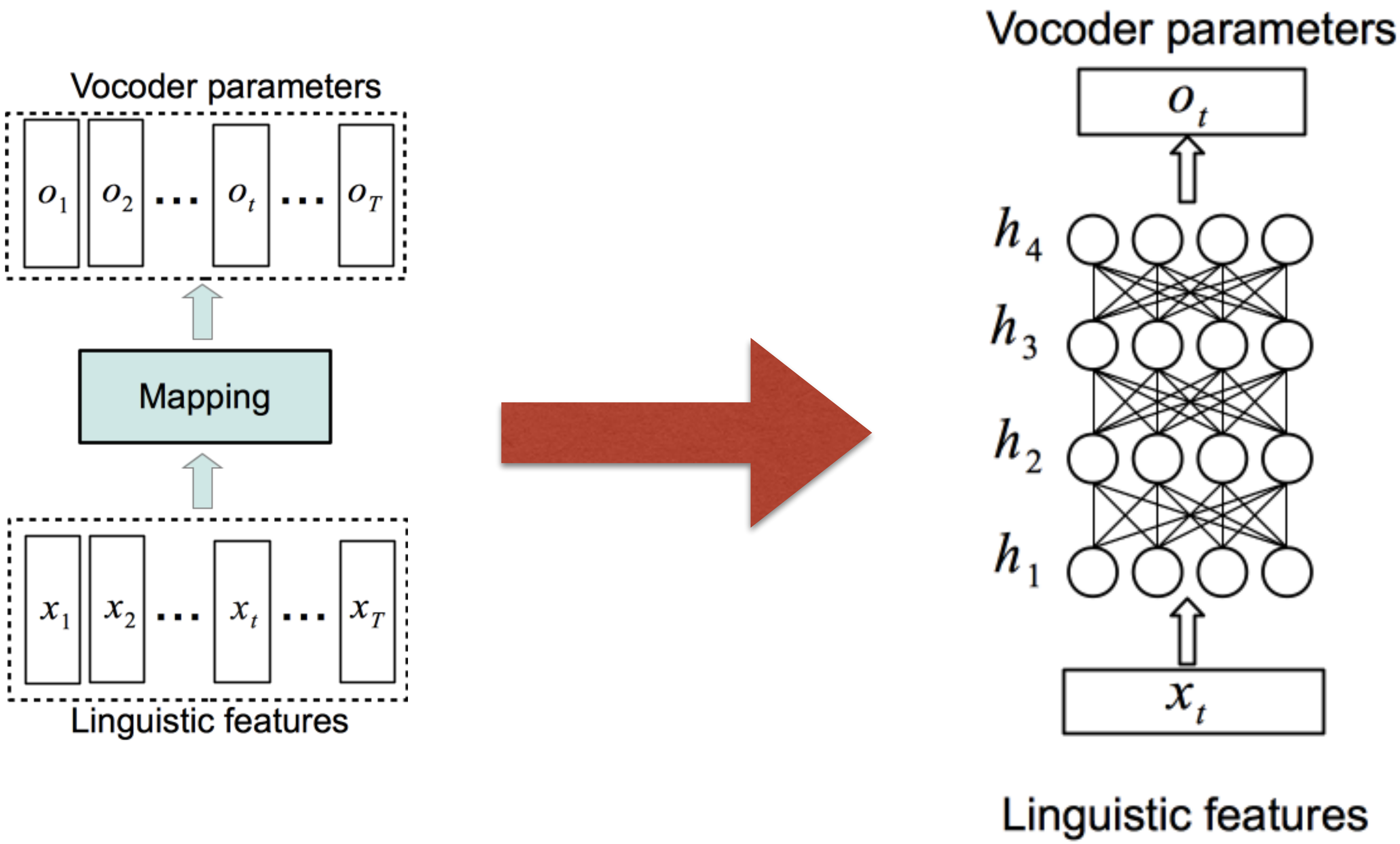
could be a sequence of HMMs, or a single DNN

Synthetic speech created from audiobooks

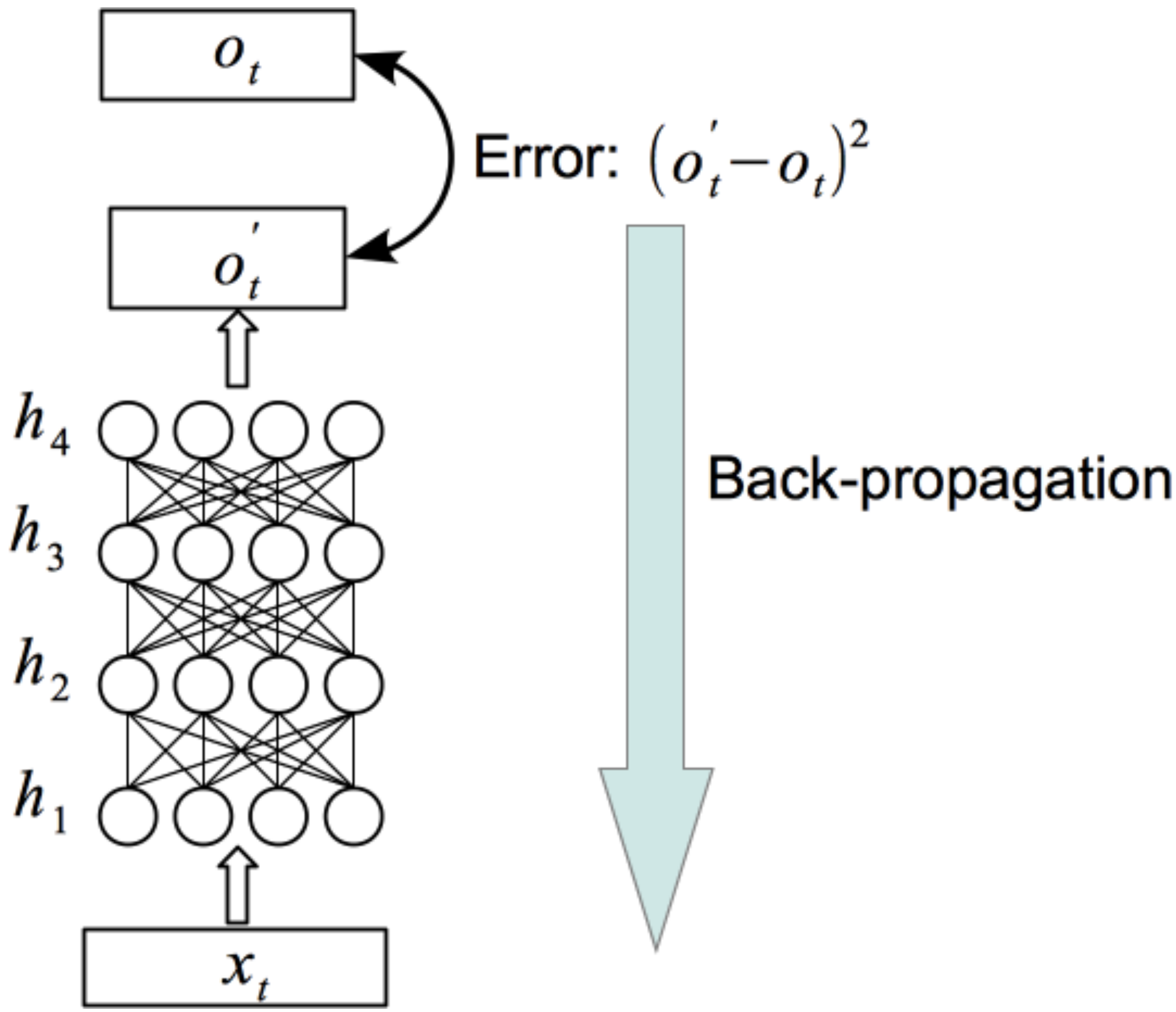
1 paragraph example

Audio credits:Speech and Hearing Research Center, Peking University

DNN speech synthesis



Training



Demos

(Exit slide show mode for demo)

	#1	#2	#3	#4
Natural				
HMM-GV				
DNN				
DNN-LSTM				
BN-DNN				
TE-DNN				
TE-BN-DNN				

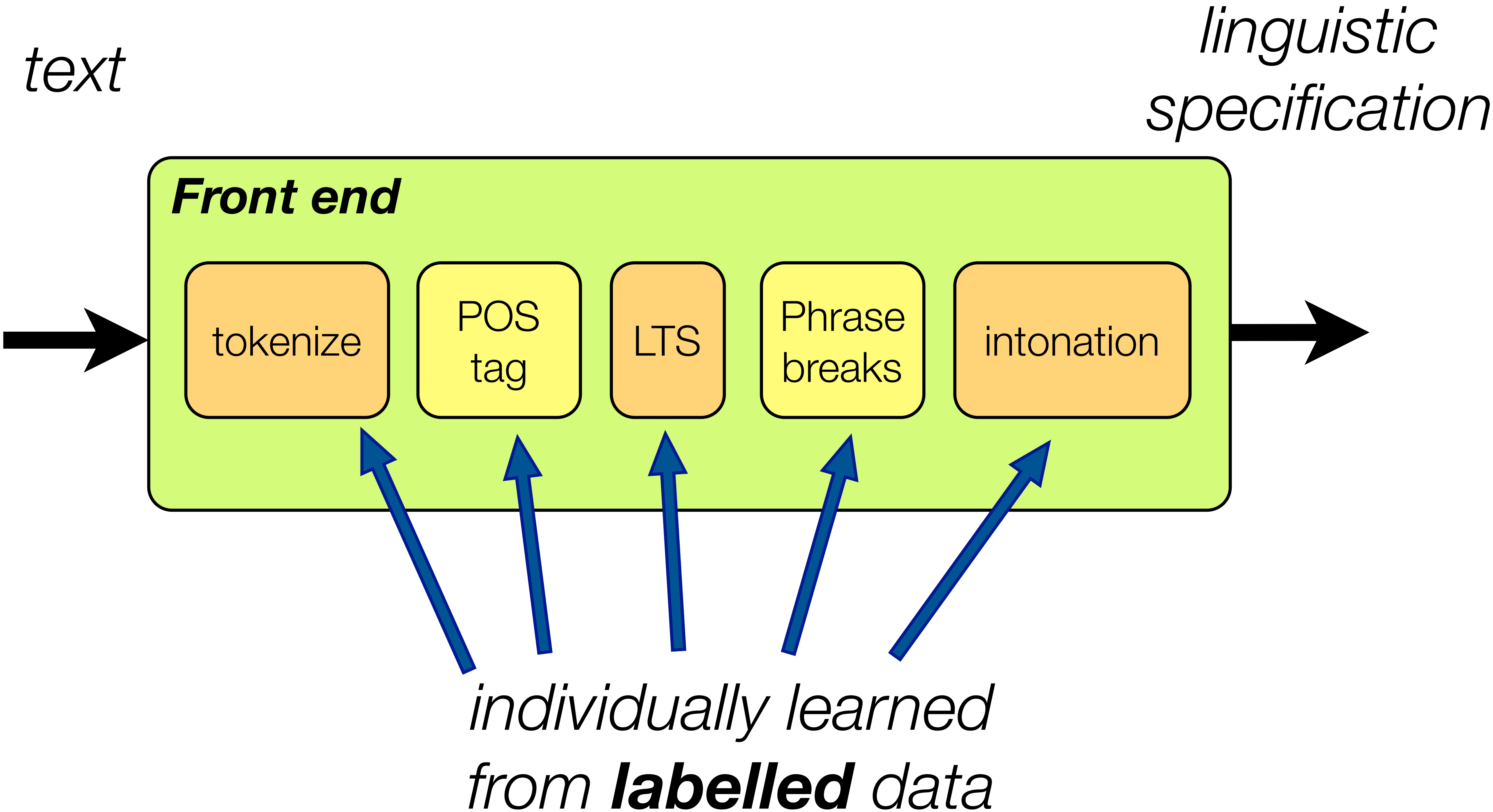
Speech Synthesis: open problem 1

From **input feature engineering** (traditional NLP and knowledge sources)

to

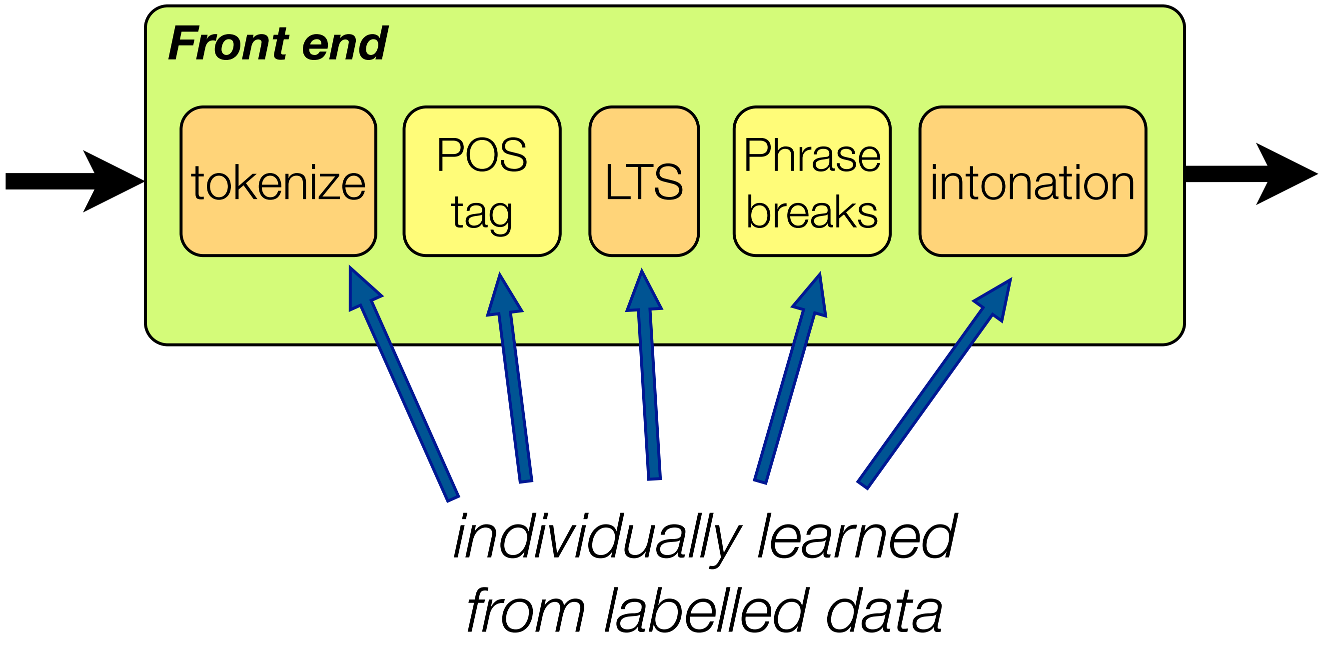
learned-from-data linguistic features

Standard text processing pipeline

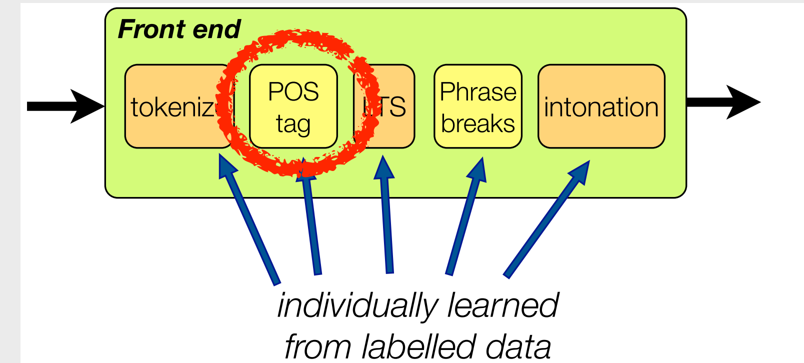


Text processing pipeline

- A chain of **processes**
- Each process is performed by a **model**
- These models are independently trained in a **supervised** fashion on annotated data



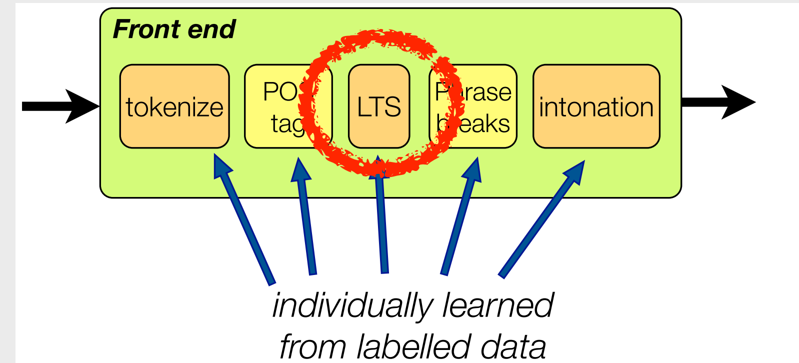
Example process I



- Part-of-speech tagger
- Accuracy is very high
- But
  - trained on **annotated** text data
  - **categories** are designed for text, not speech

```
IN of
DT the
NP McCormick
NP Public
NPSAffairs
NP Institute
IN at
NP U-Mass
NP Boston,
NP Doctor
NP Ed
NP Beard,
VBZsays
DT the
NN push
IN for
VBPdo
PP it
PP yourself
```

Example process 2



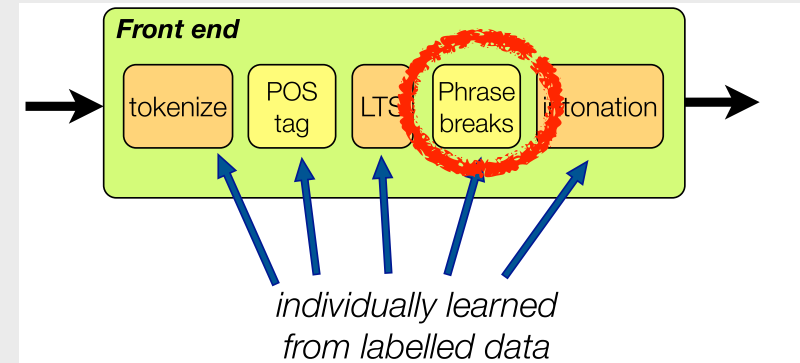
- Pronunciation model
  - dictionary look-up, *plus*
  - letter-to-sound model
- <u>But</u>
  - need deep **knowledge** of the language to design the phoneme set
  - human **expert** must write dictionary

This sequence is the annotated training data for our letter-to-sound predictor

ADWEEK AE1 D W IY0 K
ADWELL AH0 D W EH1 L
ADY EY1 D IY0
ADZ AE1 D Z
AE EY1
AEGEAN IH0 JH IY1 AH0 N
AEGIS IY1 JH AH0 S
AEGON EY1 G AA0 N
AELTUS AE1 L T AH0 S
AENEAS AE1 N IY0 AH0 S
AENEID AH0 N IY1 IH0 D
AEQUITRON EY1 K W IH0 T R AA0 N
AER EH1 R
AERIAL EH1 R IY0 AH0 L
AERIALS EH1 R IY0 AH0 L Z
AERIE EH1 R IY0
AERIEN EH1 R IY0 AH0 N
AERIENS EH1 R IY0 AH0 N Z
AERITALIA EH2 R IH0 T AE1 L Y AH0
AERO EH1 R OW0

A	-
E	EH1
R	R
I	IY0
A	AH0
L	L
S	Z

Example process 3



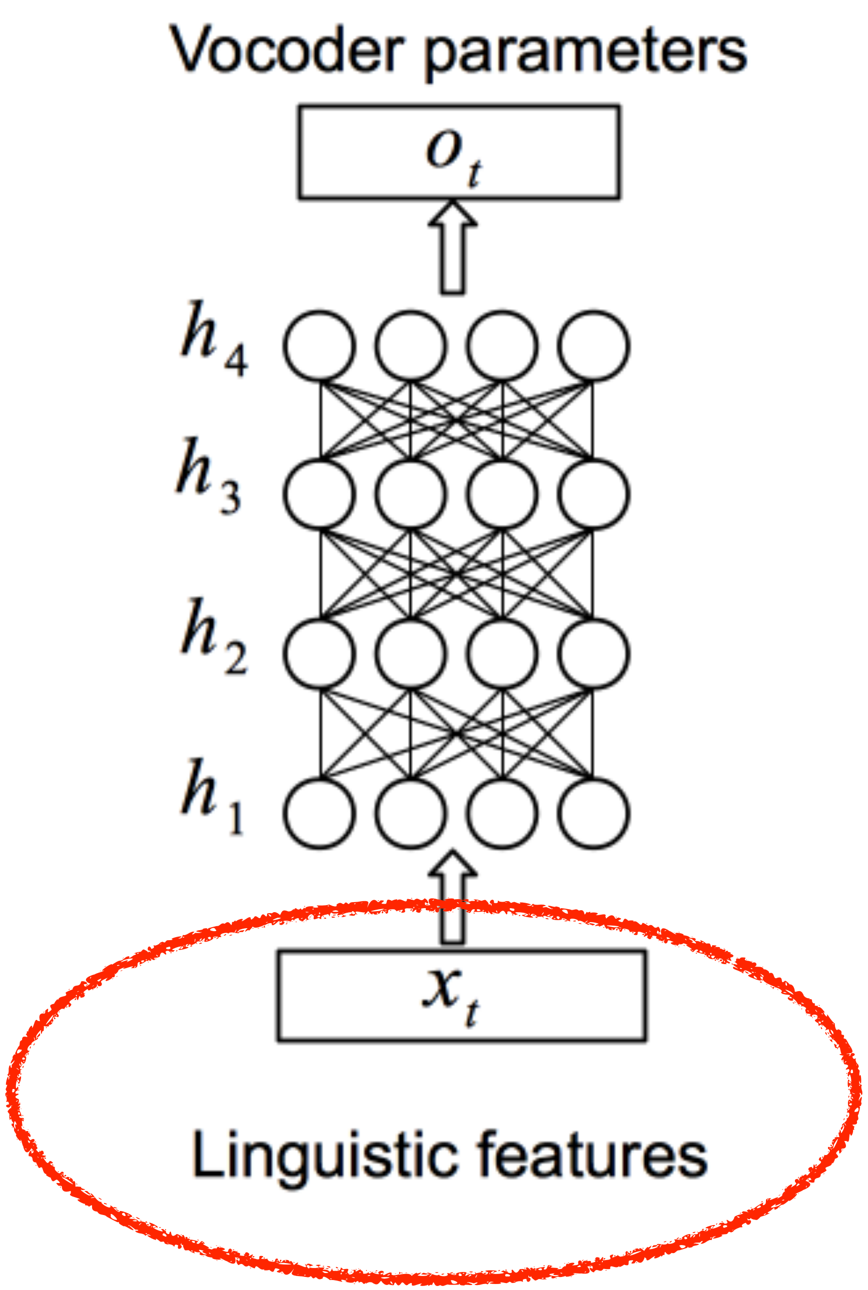
- Phrase-break prediction
  - binary classifier usi[ng] sequence as input
- <u>But</u>
  - trained on **annota[ted]** spoken data
  - therefore very **sm[all]** training set

This sequence is the annotated training data for our phrase break predictor

DT	NB
CD	NB
CD	NB
NN	NB
JJ	NB
NN	B

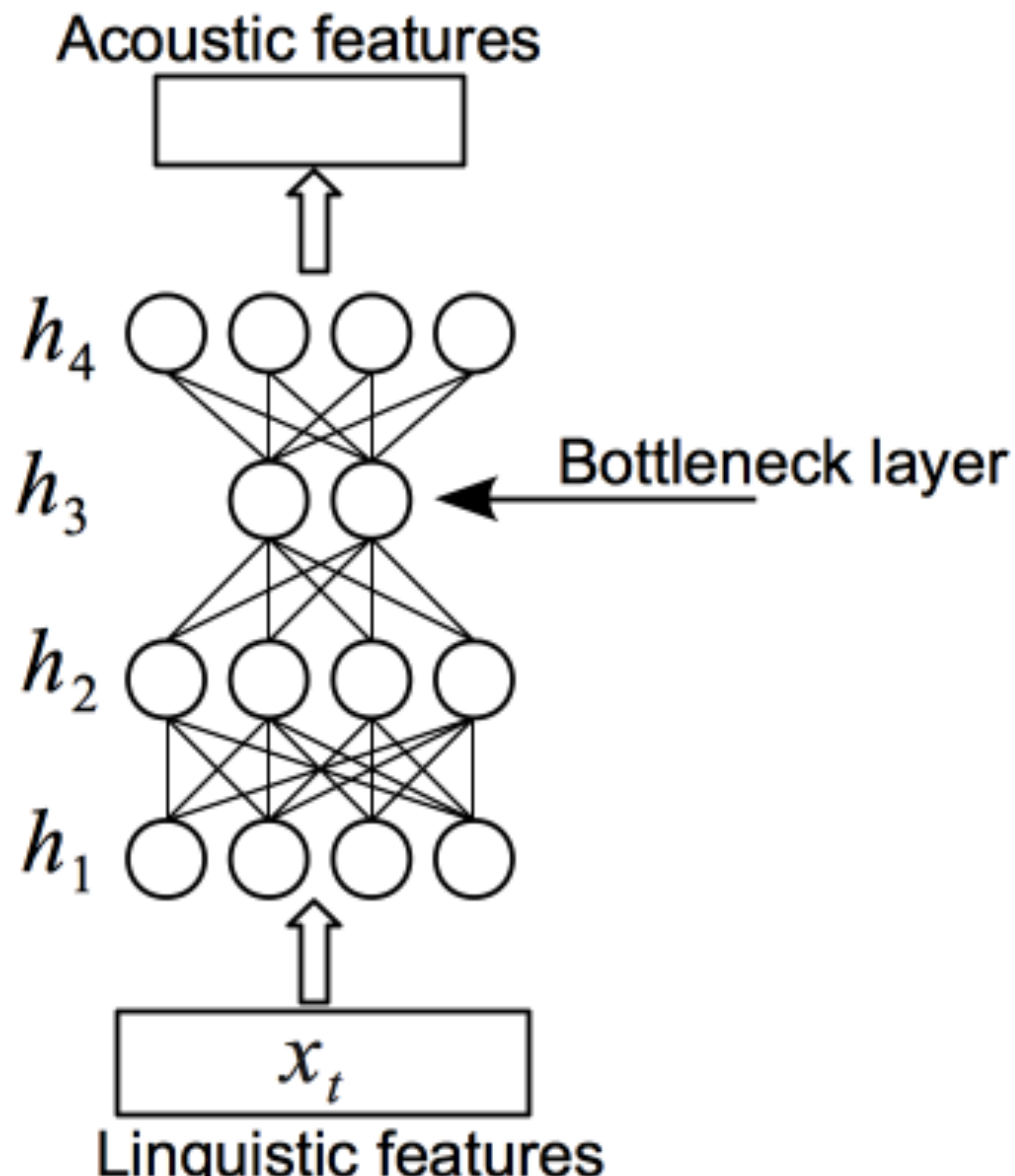
Break !

Representing linguistic features

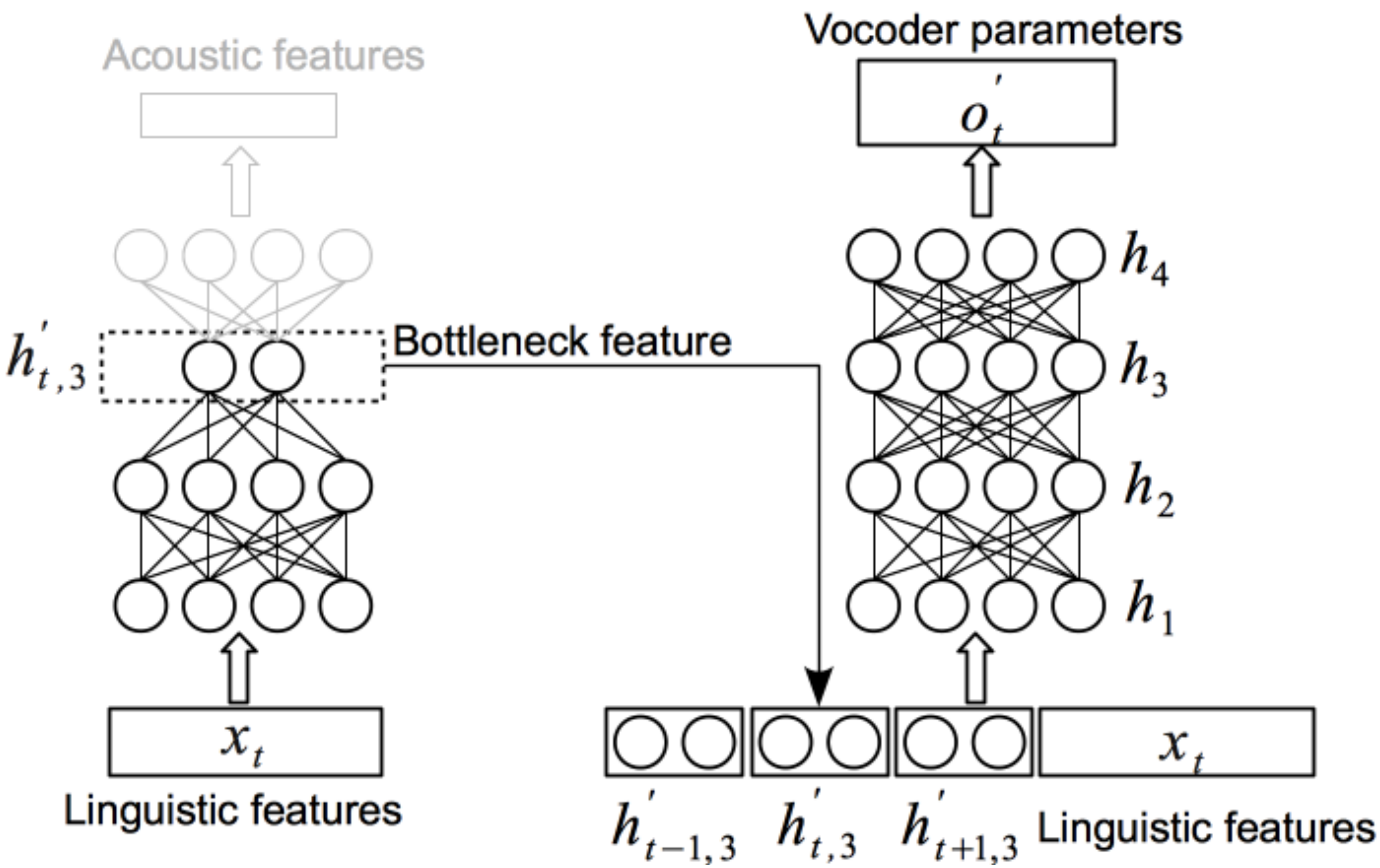


- **Encoding**
  - 1-of-N for phoneme identity, POS, etc
  - binary partitions of the space, e.g. “is this a vowel”
  - positional features
    - within syllable, word, phrase
- **Representing context**
  - include previous & next phonemes, etc
  - some features span the current utterance
- **Problems**
  - sparsity (mostly zeros)
  - noise (errors in linguistic processing)
  - relevance (not all features are predictive of speech)

Learning embeddings of features



Stacking up more context



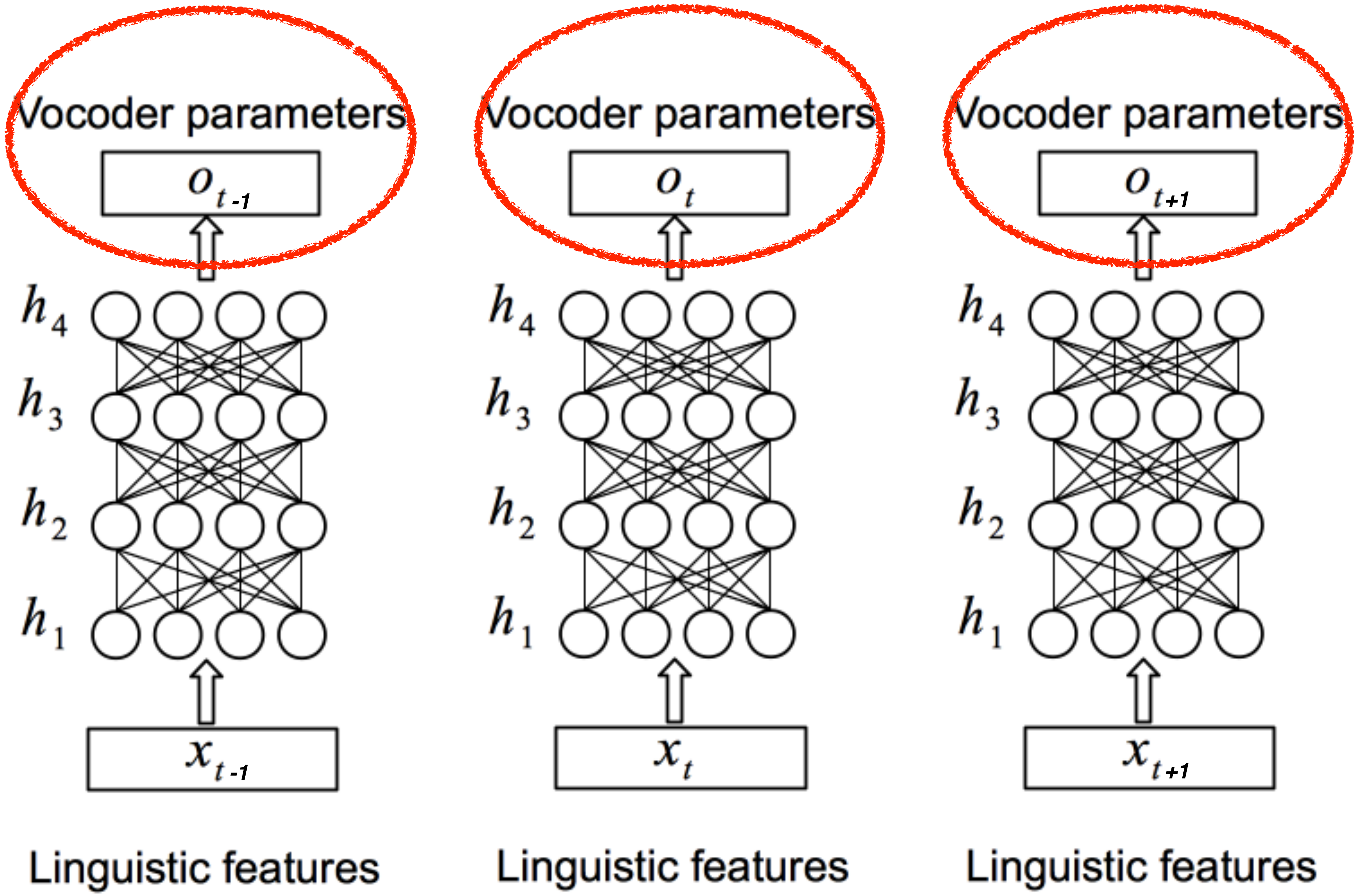
Speech Synthesis: open problem 2

From **frame-by-frame prediction**

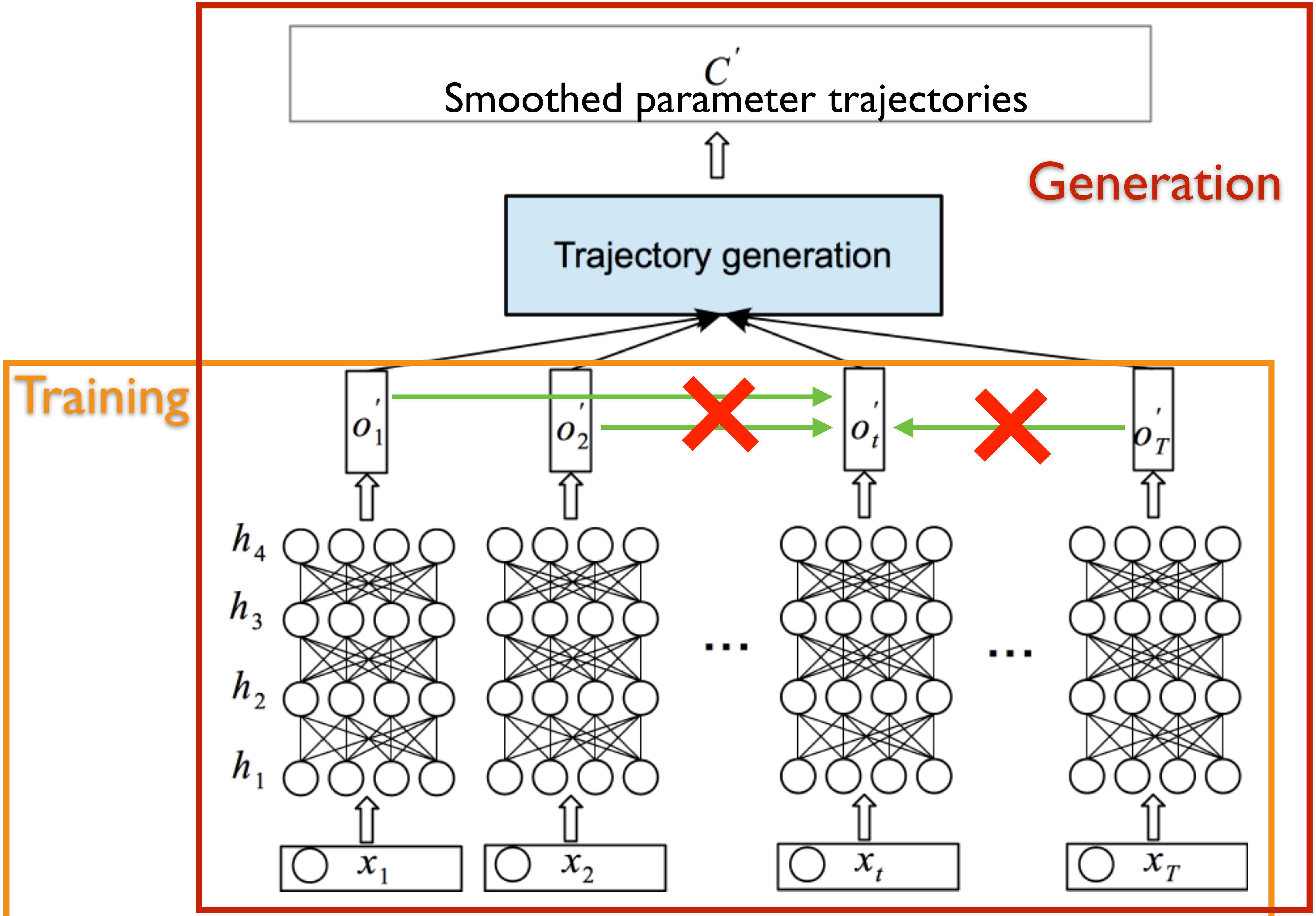
to

trajectory generation

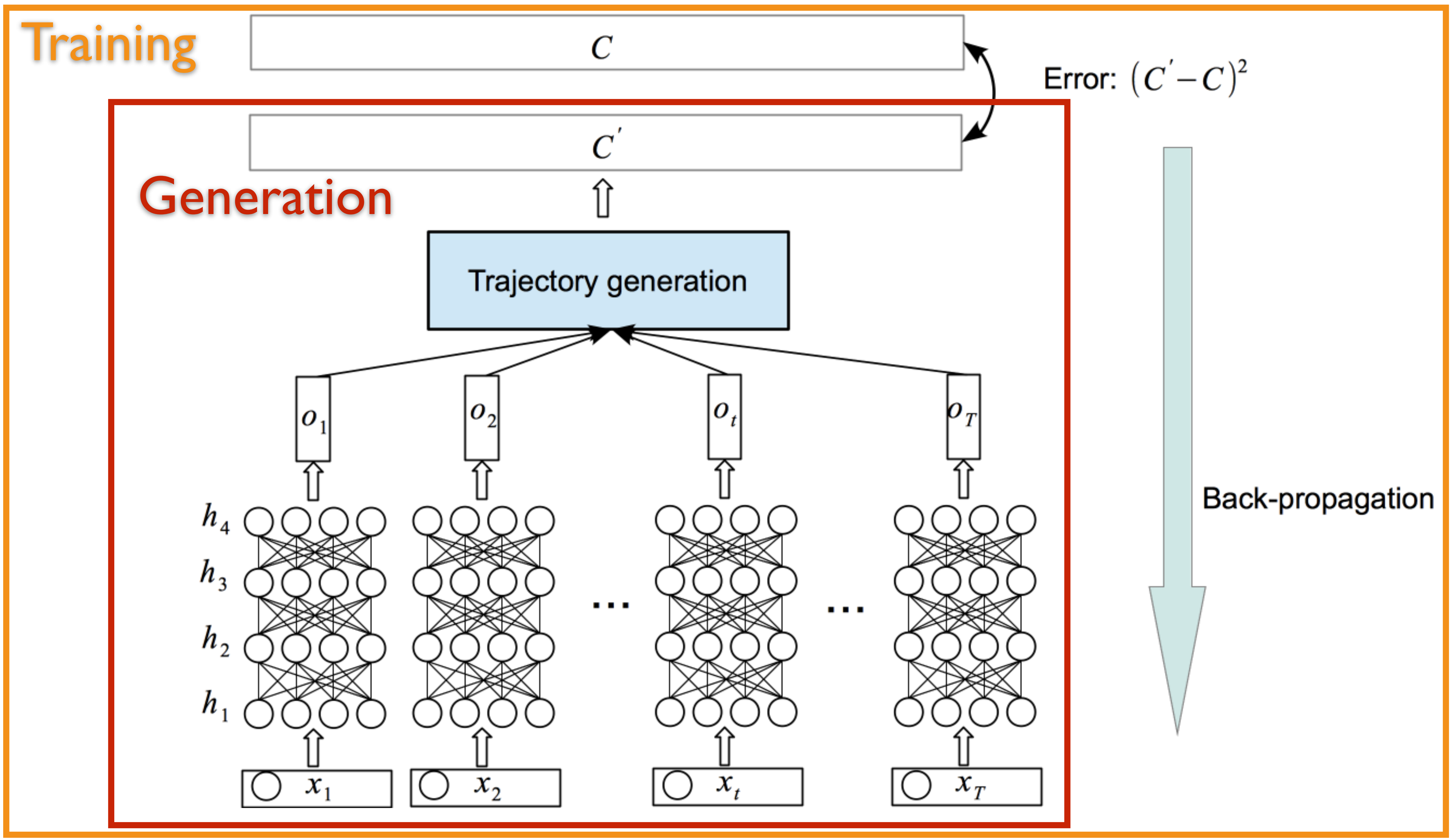
Frame-by-frame prediction



Inconsistency



Trajectory generation



Speech Synthesis: open problem 3

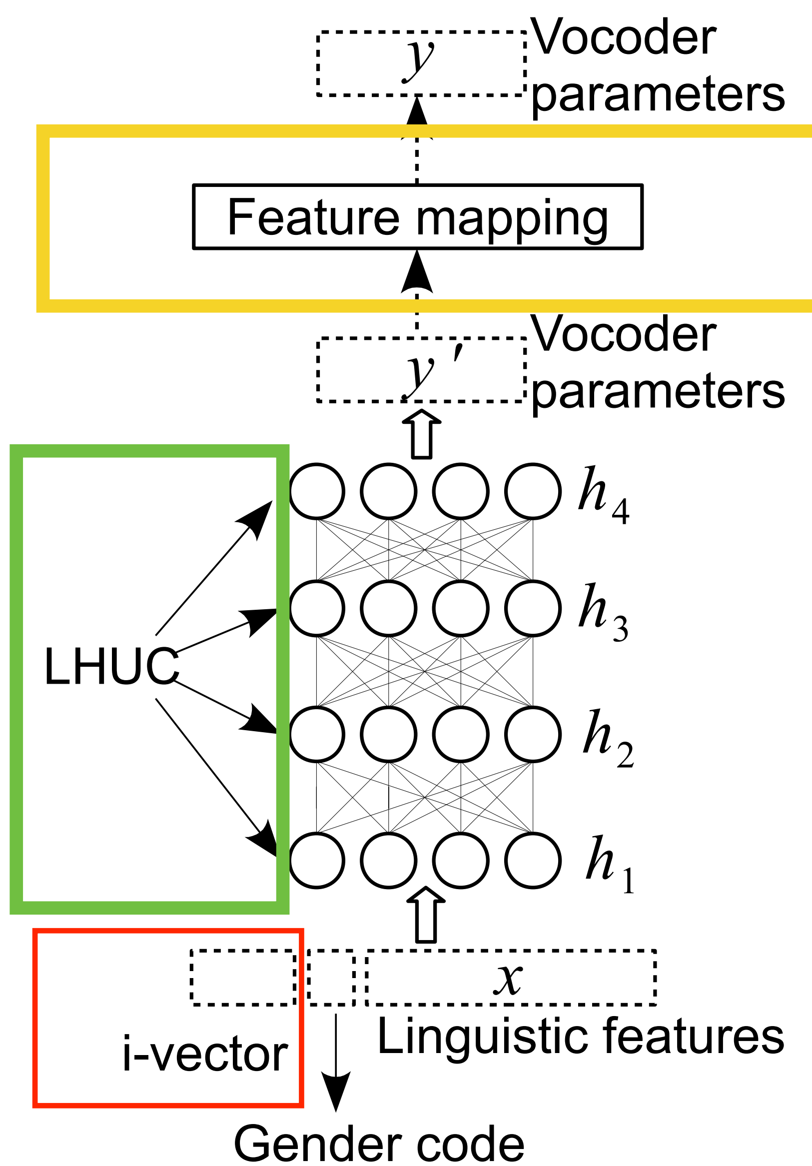
From **speaker-dependent** speech synthesis

to

adaptable and controllable models

<u>Lots of work already on this in the HMM framework, but still remains an open problem for DNNs</u>

Different ways to adapt the DNN



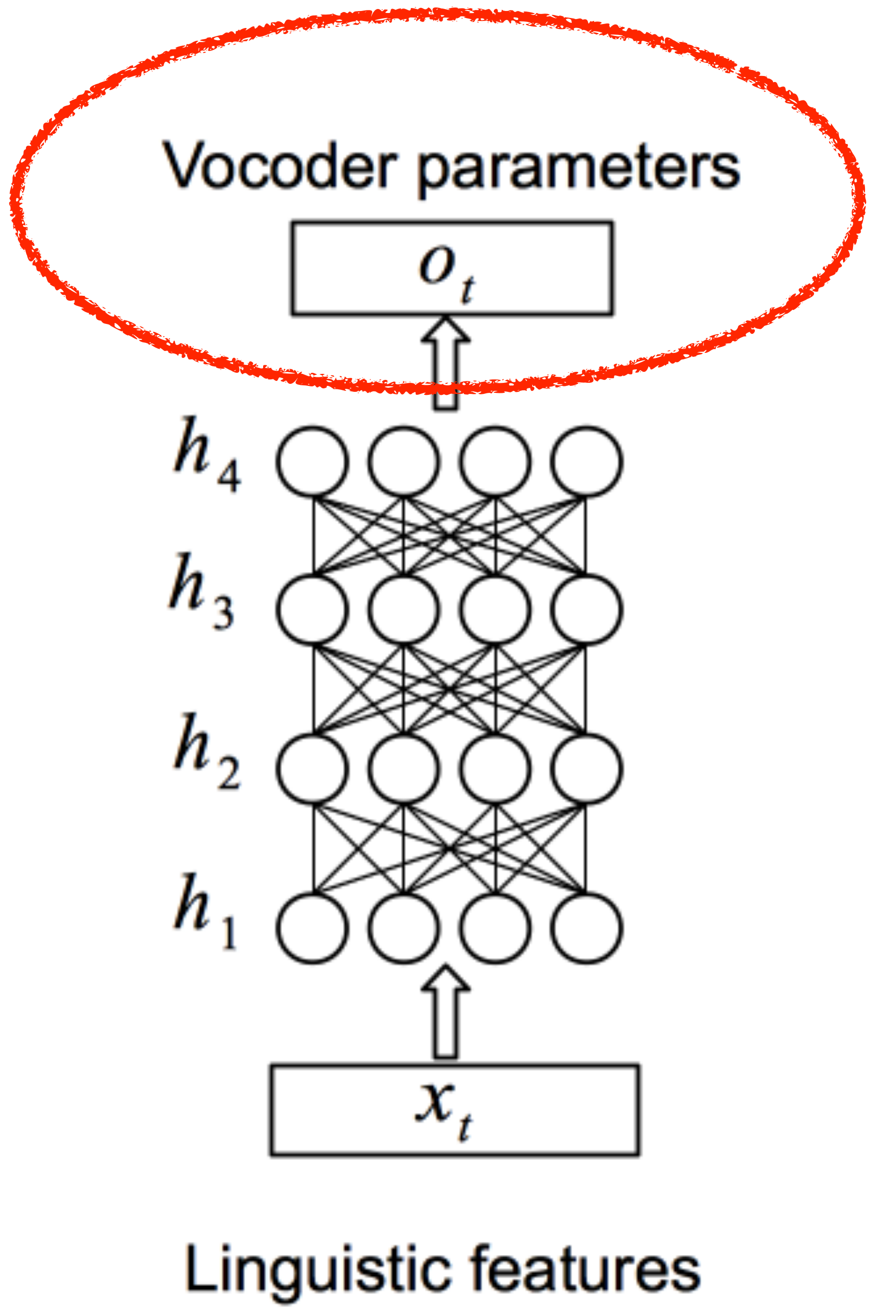
Speech Synthesis: open problem 4

From **output feature engineering** (speech signal modelling, a.k.a vocoding)

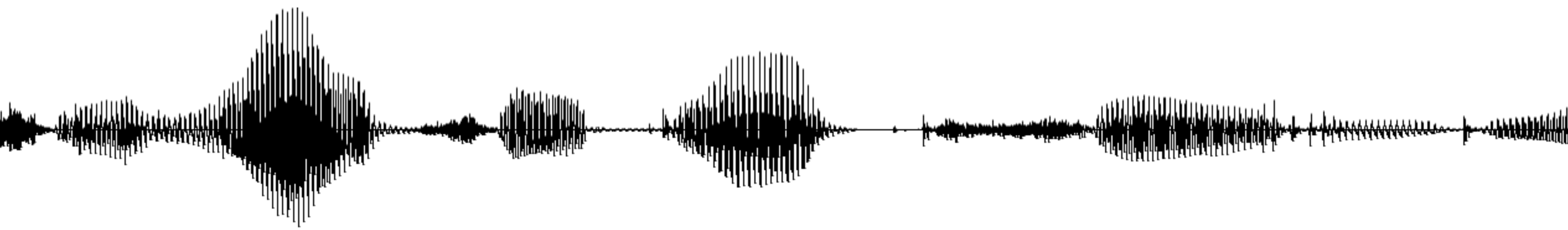
to

learned-from-data speech generation

What to predict?



Direct waveform generation ?



?