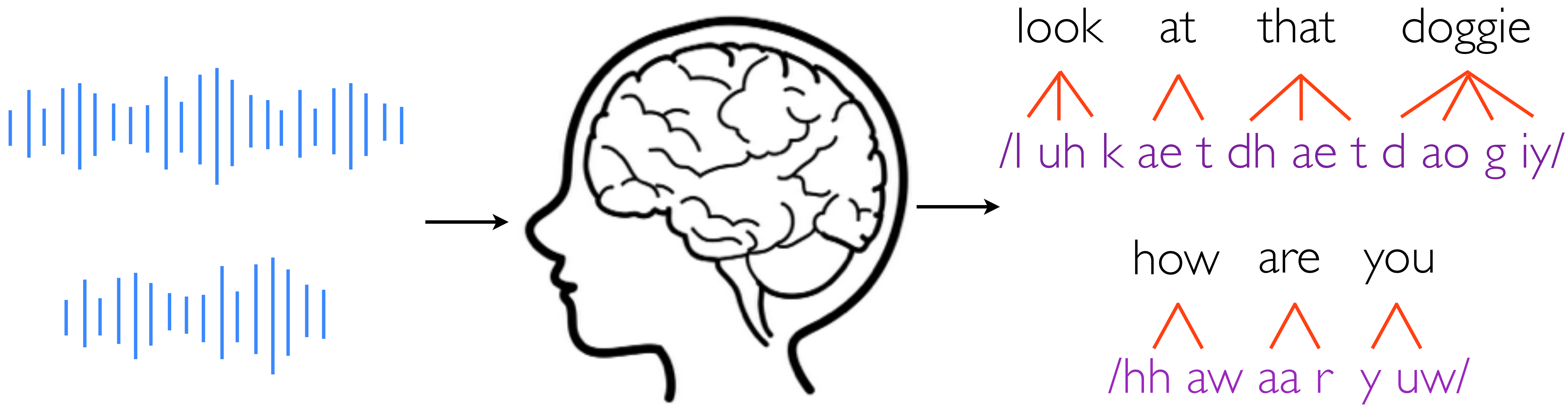
Discovering Linguistic Structures from Speech: Models and Applications

Jackie Lee

Spoken Language Systems Group, CSAIL, MIT
Machine Learning Engineer @ Kite

Problem Overview

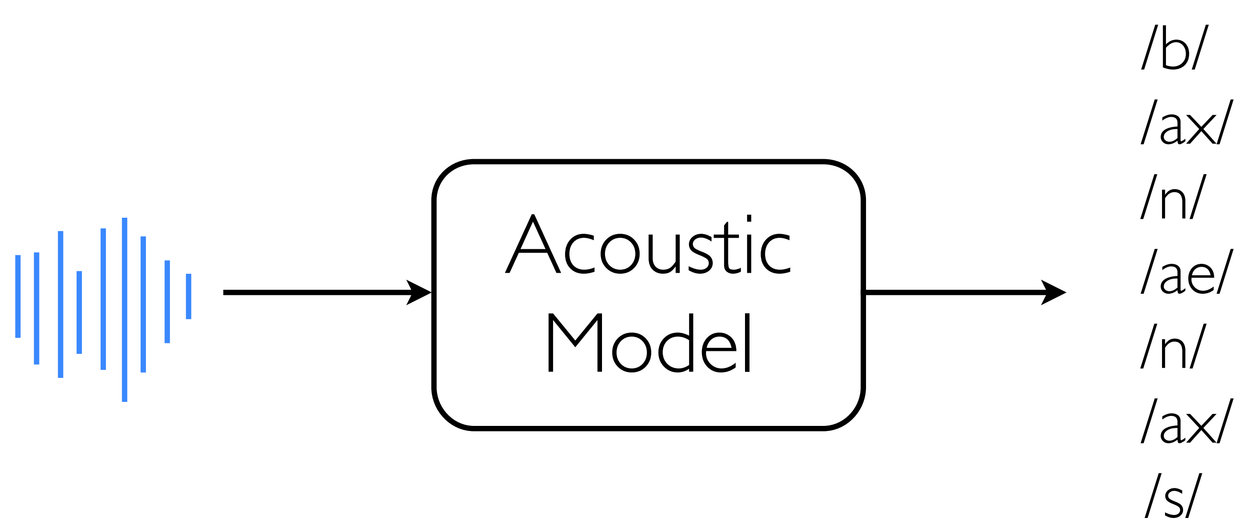
- A task that humans can perform naturally



- Goal
  - Develop computational models for discovering linguistic structures from speech

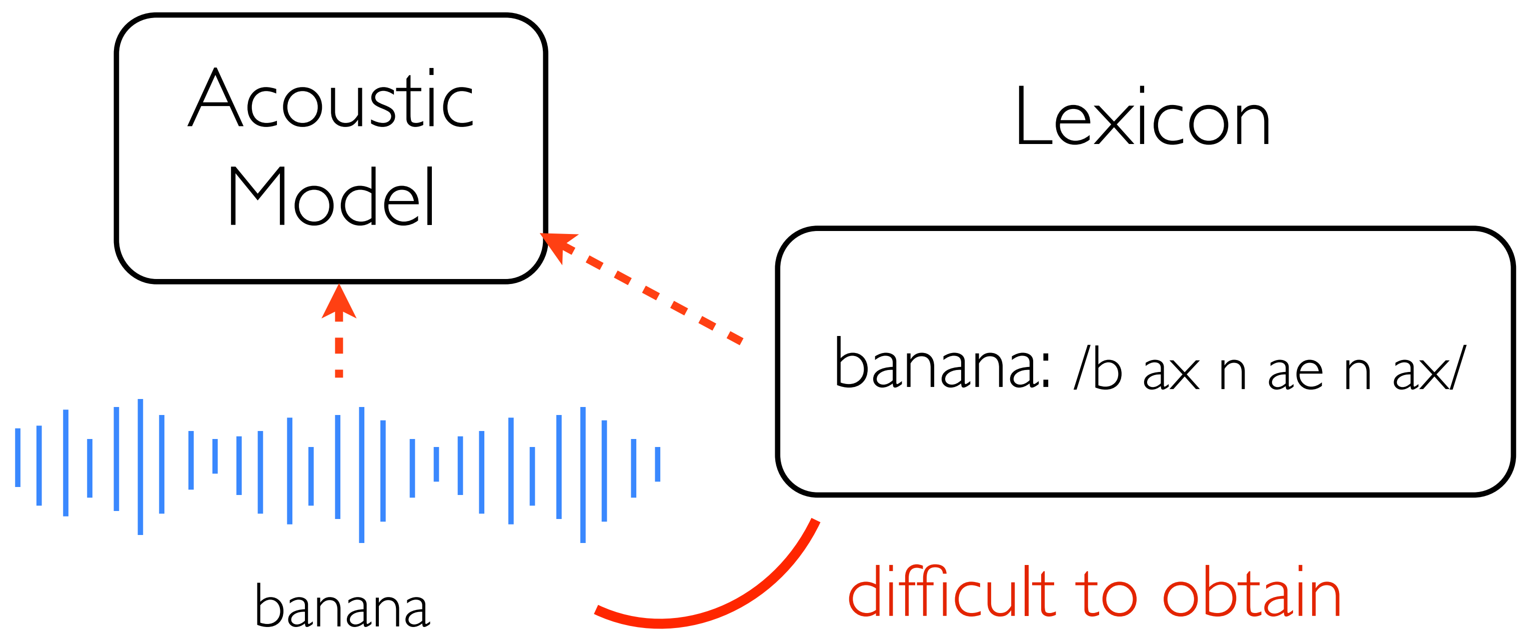
Potential Applications of Discovered Structures

- Unsupervised training of speech recognizers
- Take acoustic model as an example



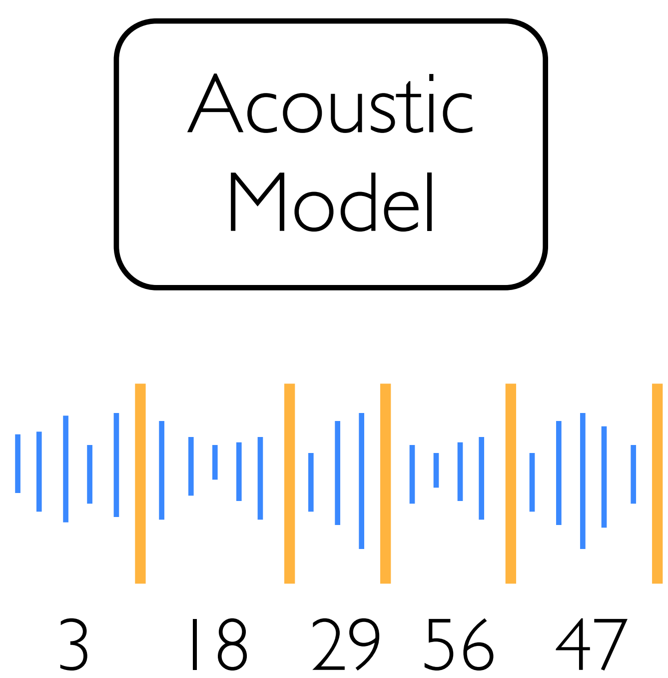
Potential Applications of Discovered Structures

- Unsupervised training of speech recognizers
- Take acoustic model as an example
  - Training requires word transcriptions with a pronunciation lexicon



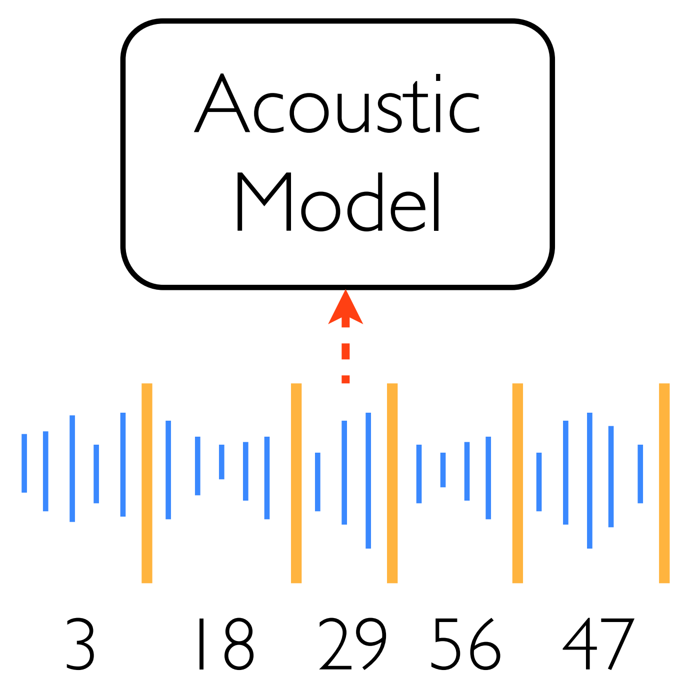
Potential Applications of Discovered Structures

- Unsupervised training of speech recognizers
- Take acoustic model as an example
  - Training requires word transcriptions with a pronunciation lexicon
- Unsupervised phonetic unit discovery



Potential Applications of Discovered Structures

- Unsupervised training of speech recognizers
- Take acoustic model as an example
  - Training requires word transcriptions with a pronunciation lexicon
- Unsupervised phonetic unit discovery
  - Allows learning an acoustic model directly from speech data

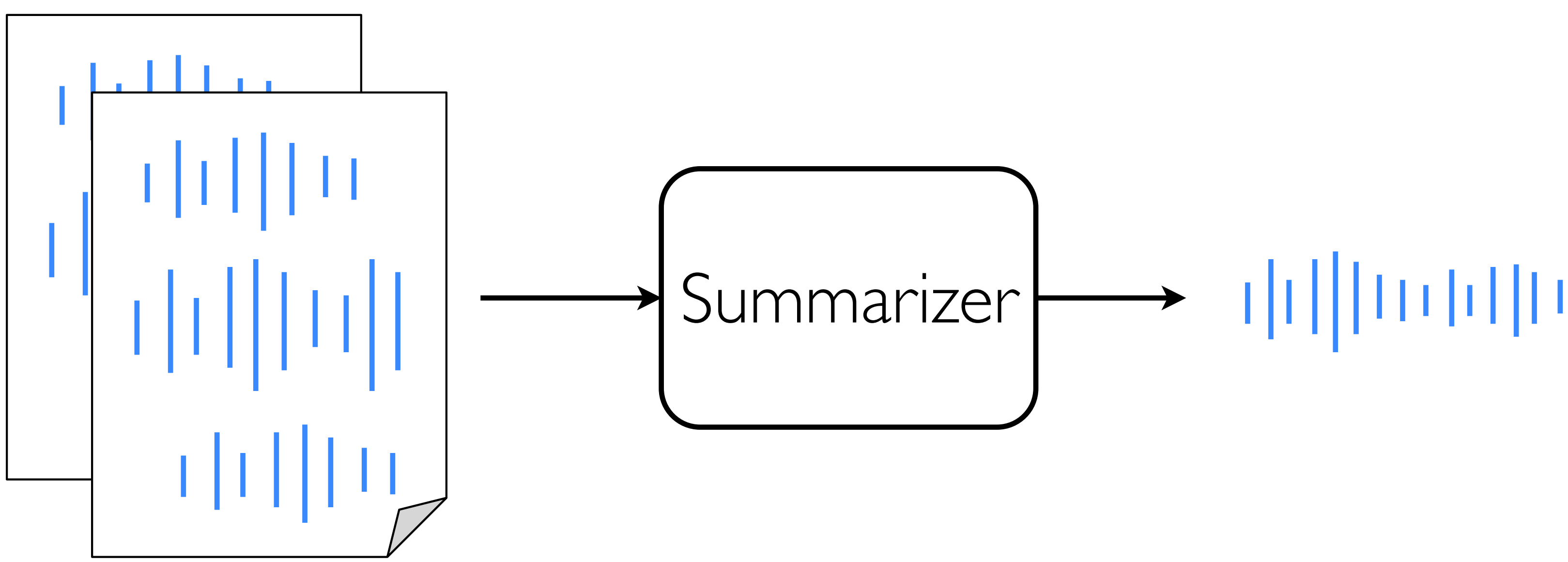


Applications of Higher Level Linguistic Structures

- Sub-word units are useful for representing out-of-vocabulary words

Applications of Higher Level Linguistic Structures

- Sub-word units are useful for representing out-of-vocabulary words
- Unsupervised word discovery
  - Natural language processing on spoken documents without speech recognition

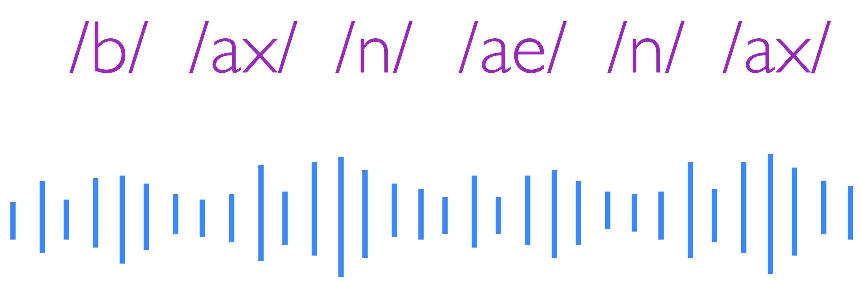


- Connection to the field of Cognitive Science

Outline

Discovering phonetic inventory

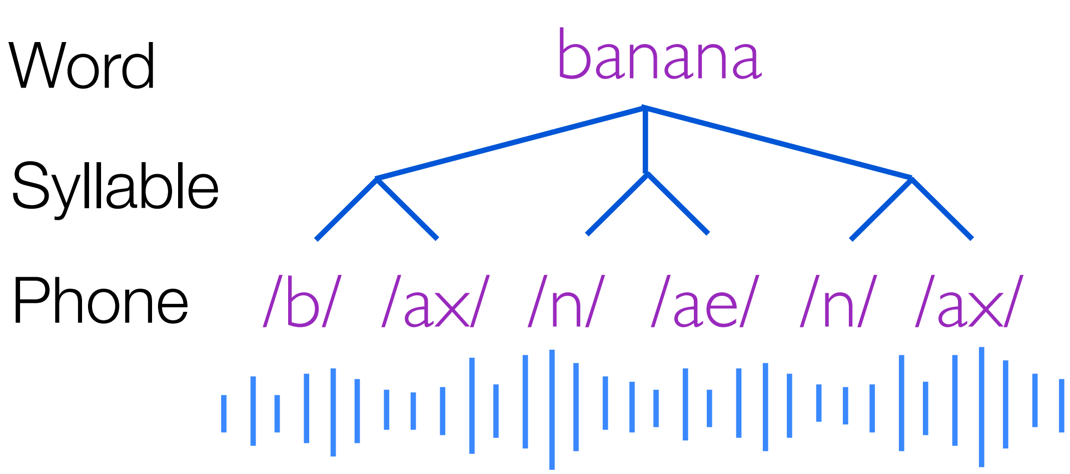
[Lee and Glass, ACL 2012]



Part I of the talk

Discovering hierarchical linguistic structures

[Lee, O'Donnell, and Glass, TACL 2015]



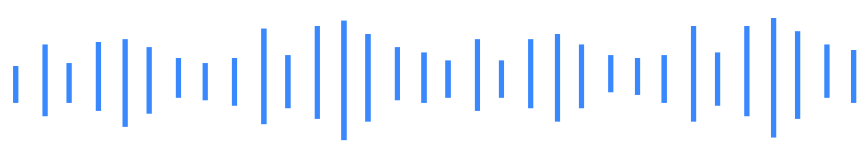
Part II of the talk

Part I: Discovering Phonetic Units from Speech

Discovering phonetic inventory

[Lee and Glass, ACL 2012]

/b/ /ax/ /n/ /ae/ /n/ /ax/



Part I of the talk

Discovering hierarchical linguistic structures

[Lee, O'Donnell, and Glass,TACL 2015

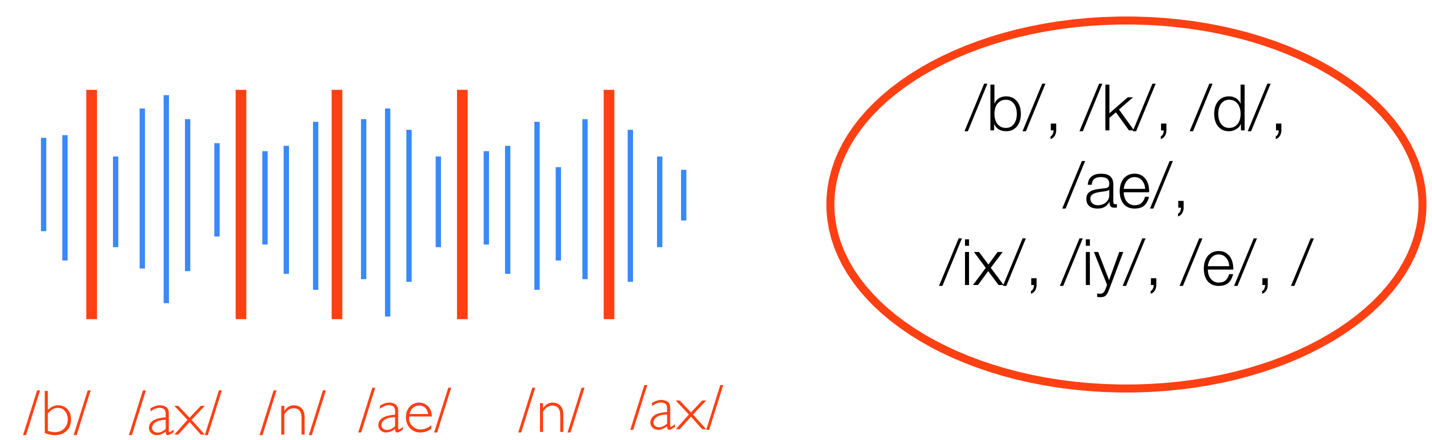
Part II of the talk

Problem Overview

- Find the phone units embedded in the observed speech data

Problem Overview

- Find the phone units embedded in the observed speech data
- Latent variables



  - Phone boundaries
  - Phone labels
  - Phone inventory

Related Work

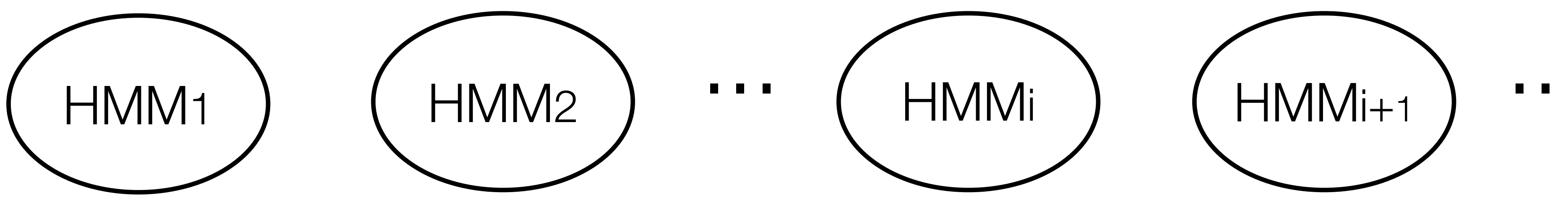
- **Unsupervised acoustic unit discovery and modeling**
  - Towards unsupervised training of speaker independent acoustic models [*Jansen and Church, INTERSPEECH 2011*]
  - Unsupervised hidden Markov modeling of spoken queries for spoken term detection without speech recognition [*Chan et al., INTERSPEECH 2011*]
  - Keyword spotting of arbitrary words using minimal speech resources [*Garcia and Gish, ICASSP 2006*]
  - Toward ALISP: A proposal for automatic language independent speech processing [*Chollet et al., Computational Models of Speech Pattern Processing 1999*]
  - A segment model based approach to speech recognition [*Lee et al., ICASSP1988*]

Generative Story

- A simple explanation of how a spoken utterance is generated

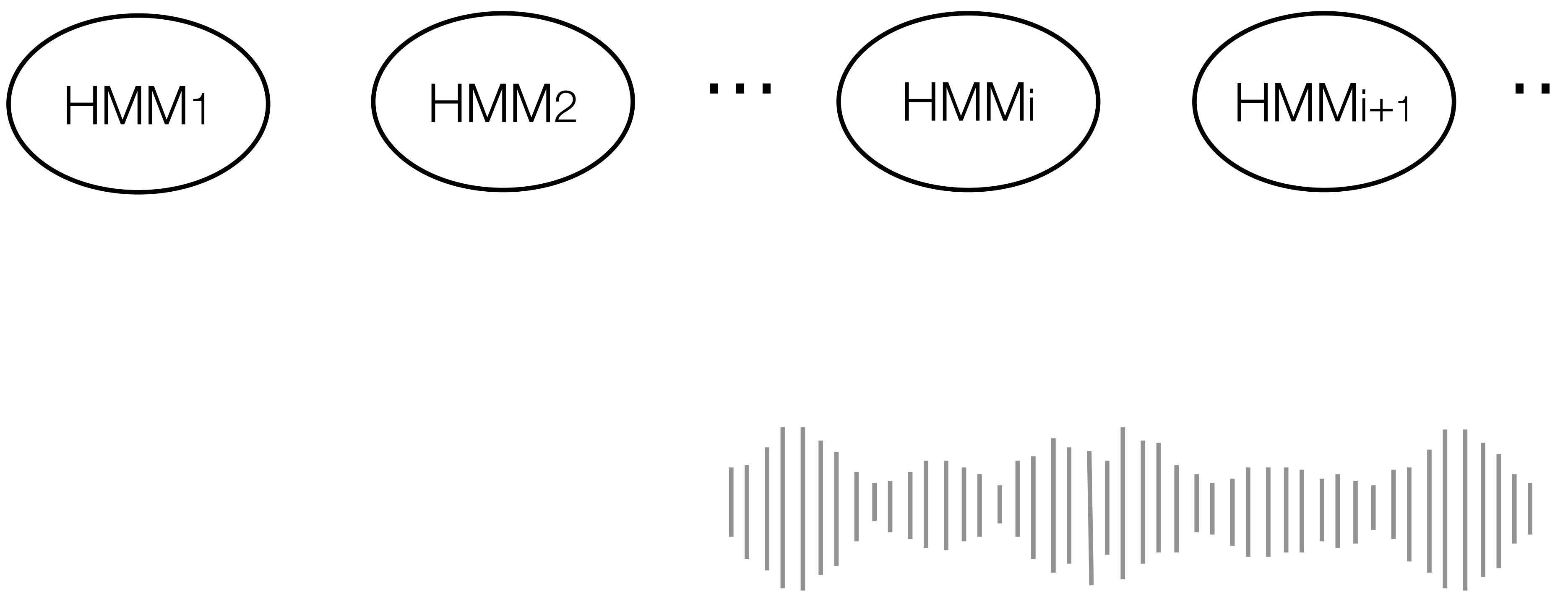
Generative Story

- A simple explanation of how a spoken utterance is generated



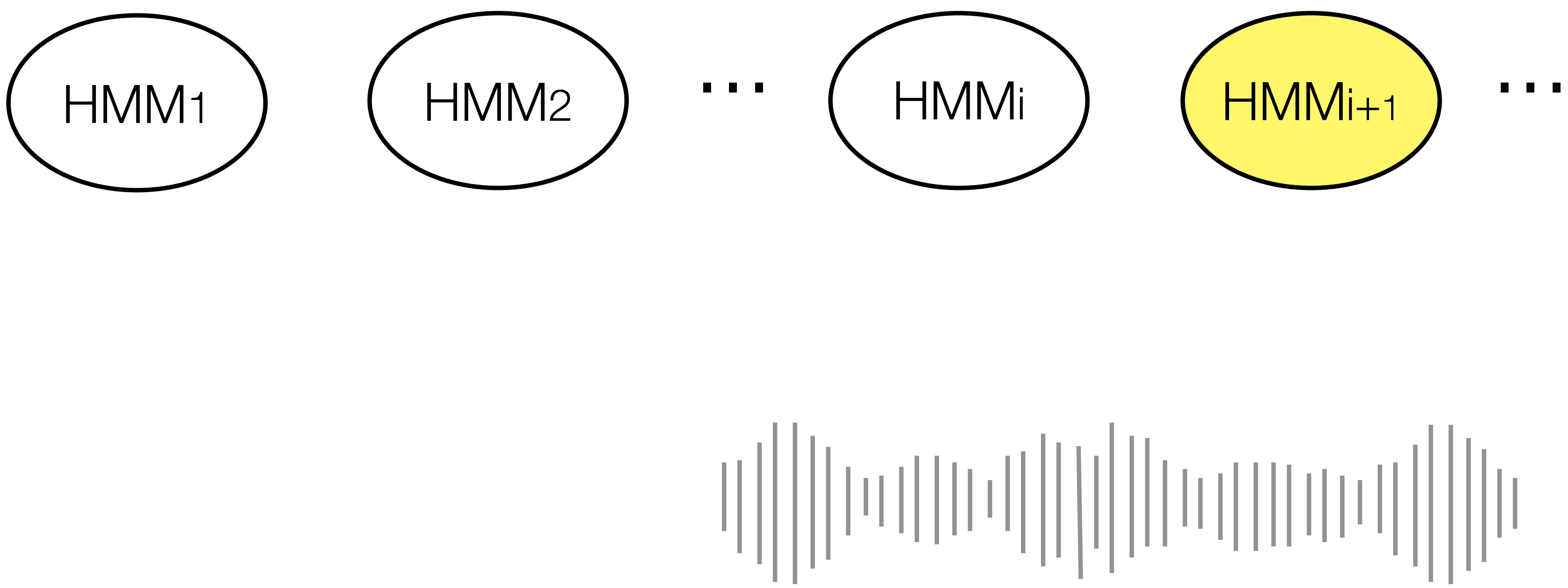
Generative Story

- A simple explanation of how a spoken utterance is generated



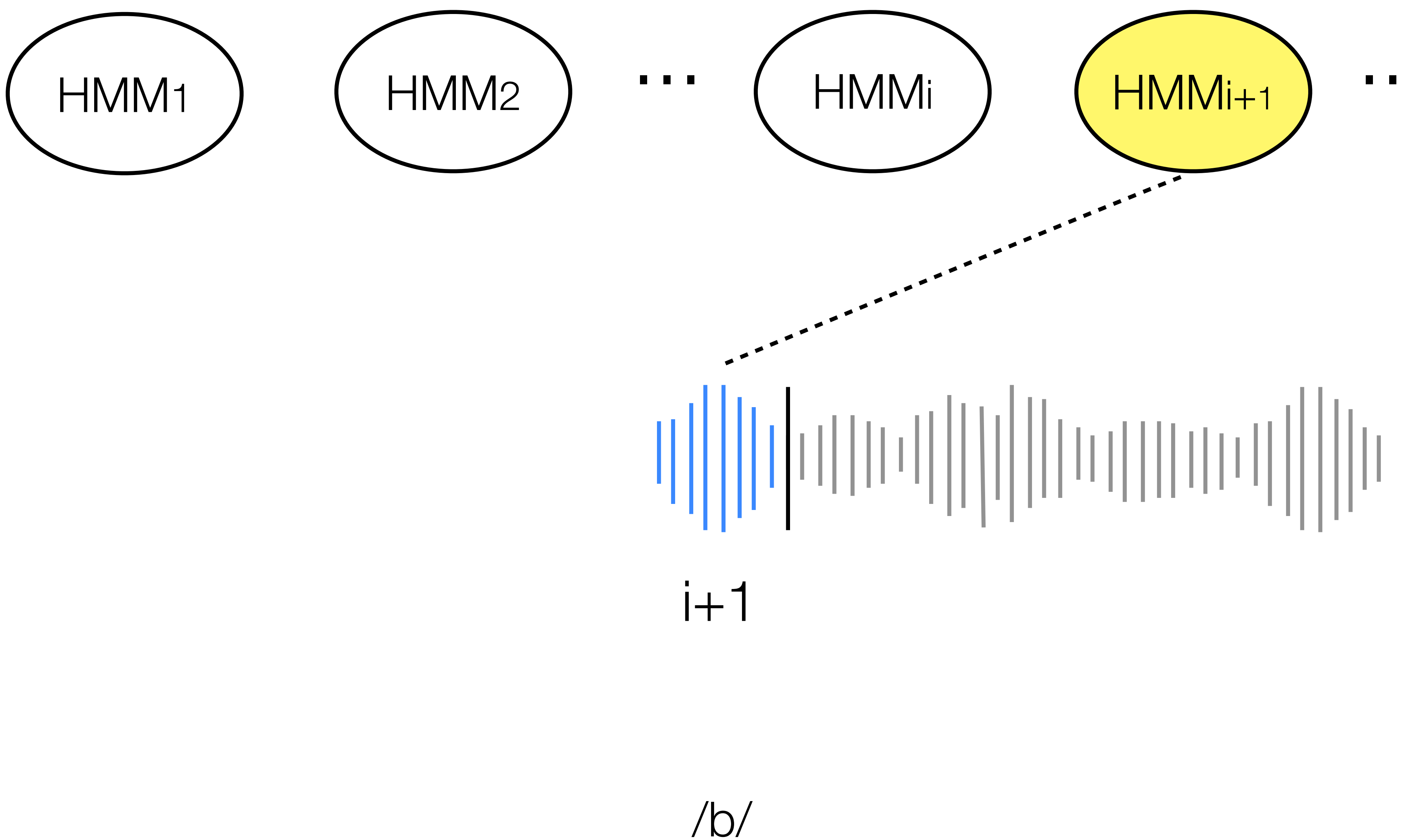
Generative Story

- A simple explanation of how a spoken utterance is generated



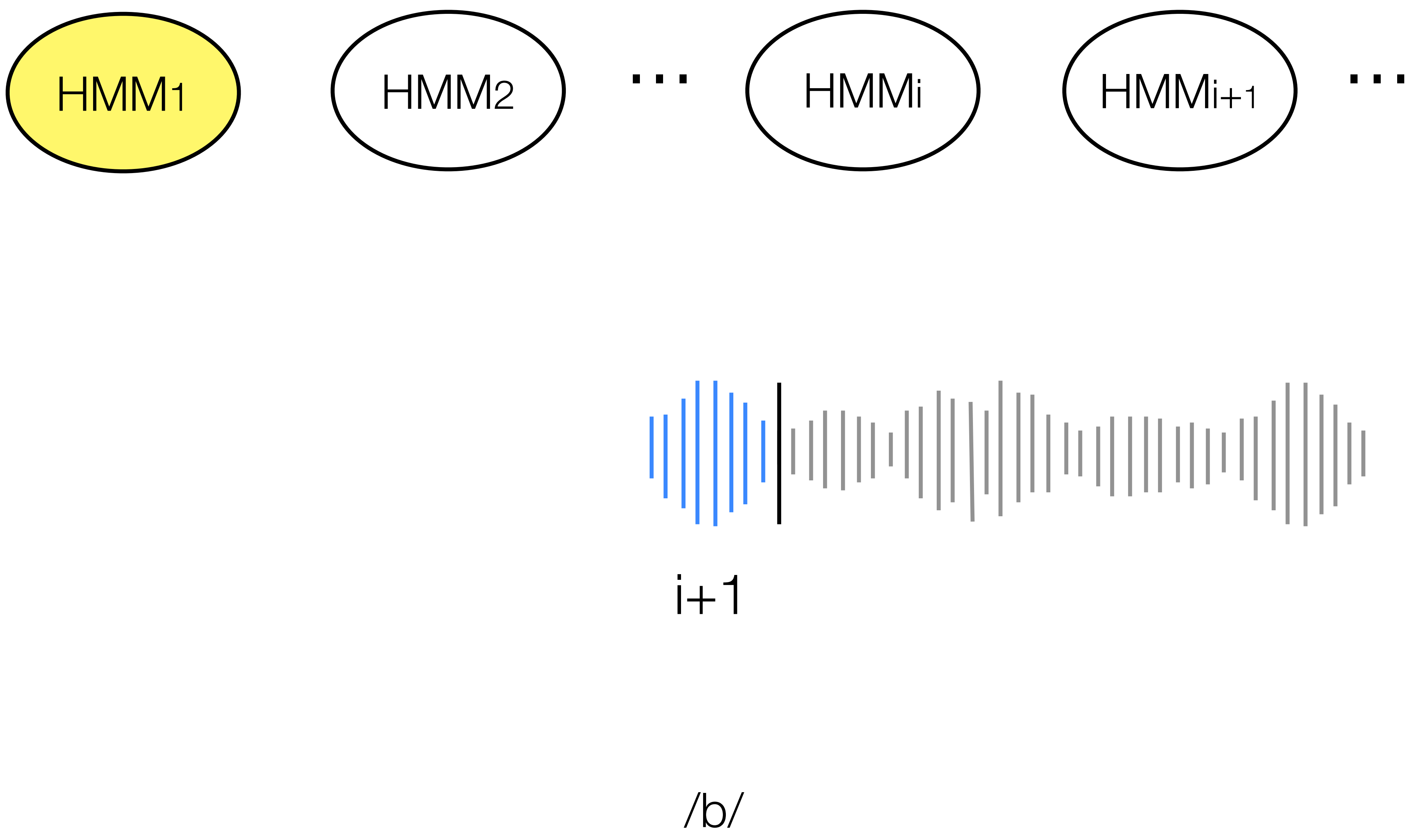
Generative Story

- A simple explanation of how a spoken utterance is generated



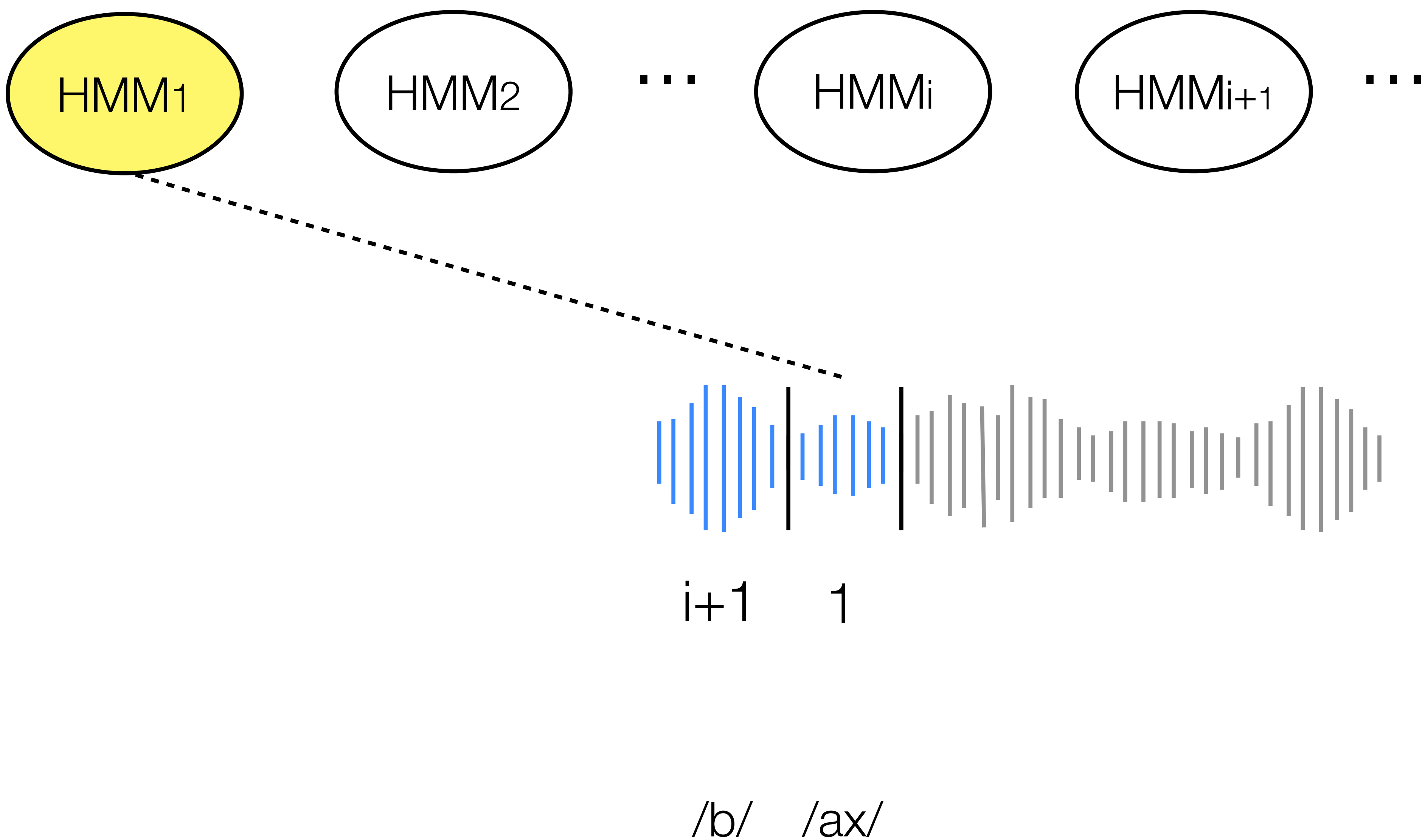
Generative Story

- A simple explanation of how a spoken utterance is generated



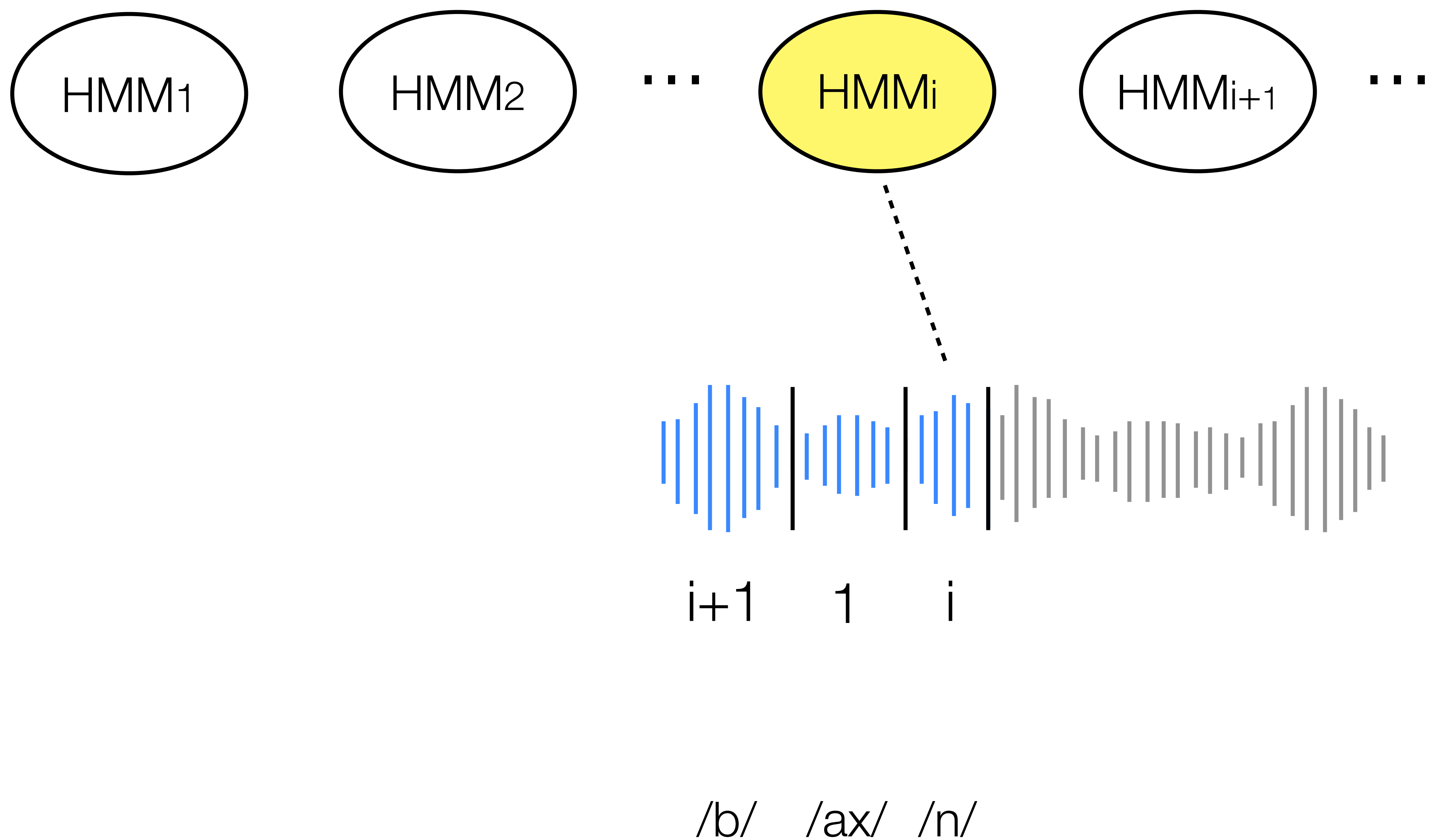
Generative Story

- A simple explanation of how a spoken utterance is generated



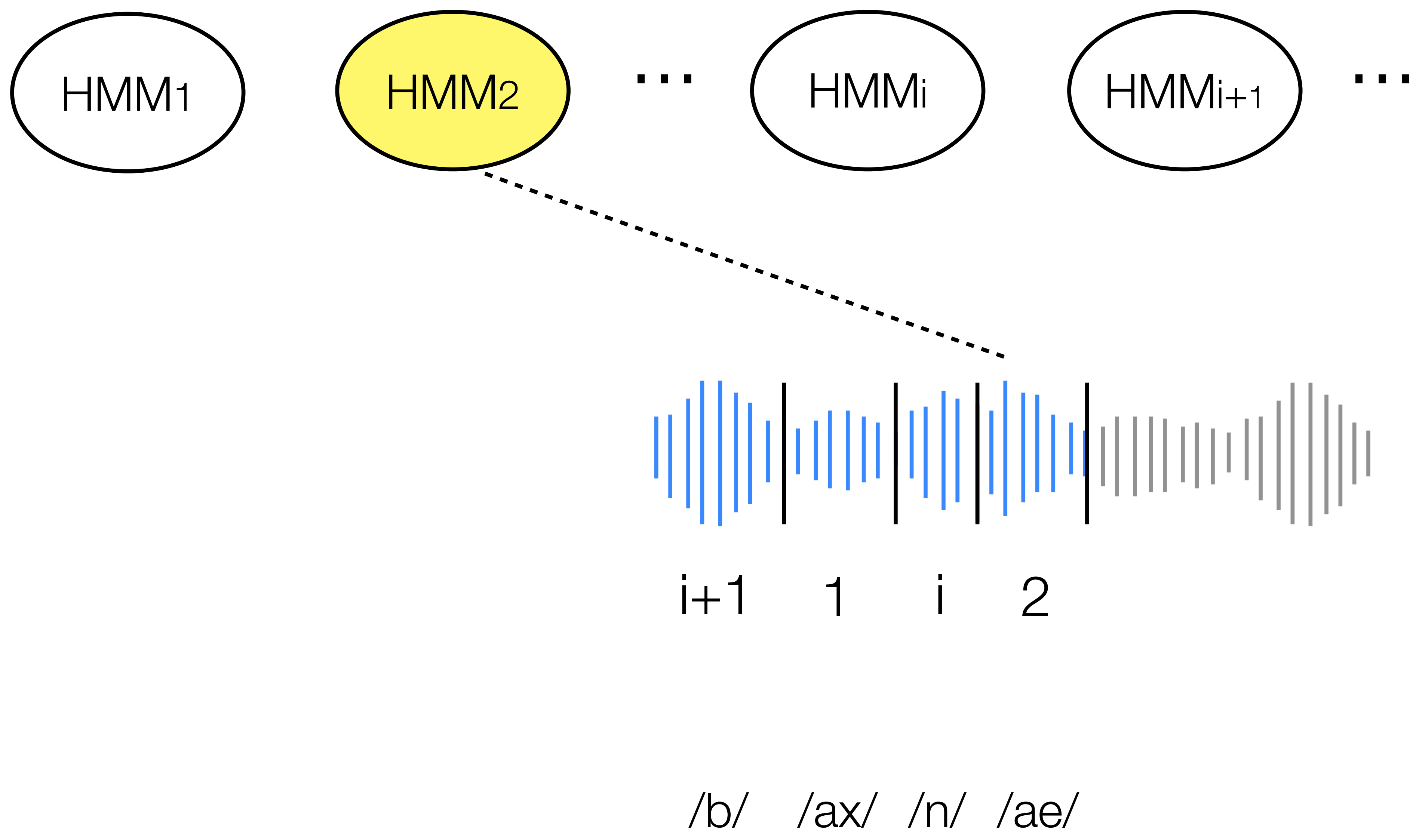
Generative Story

- A simple explanation of how a spoken utterance is generated



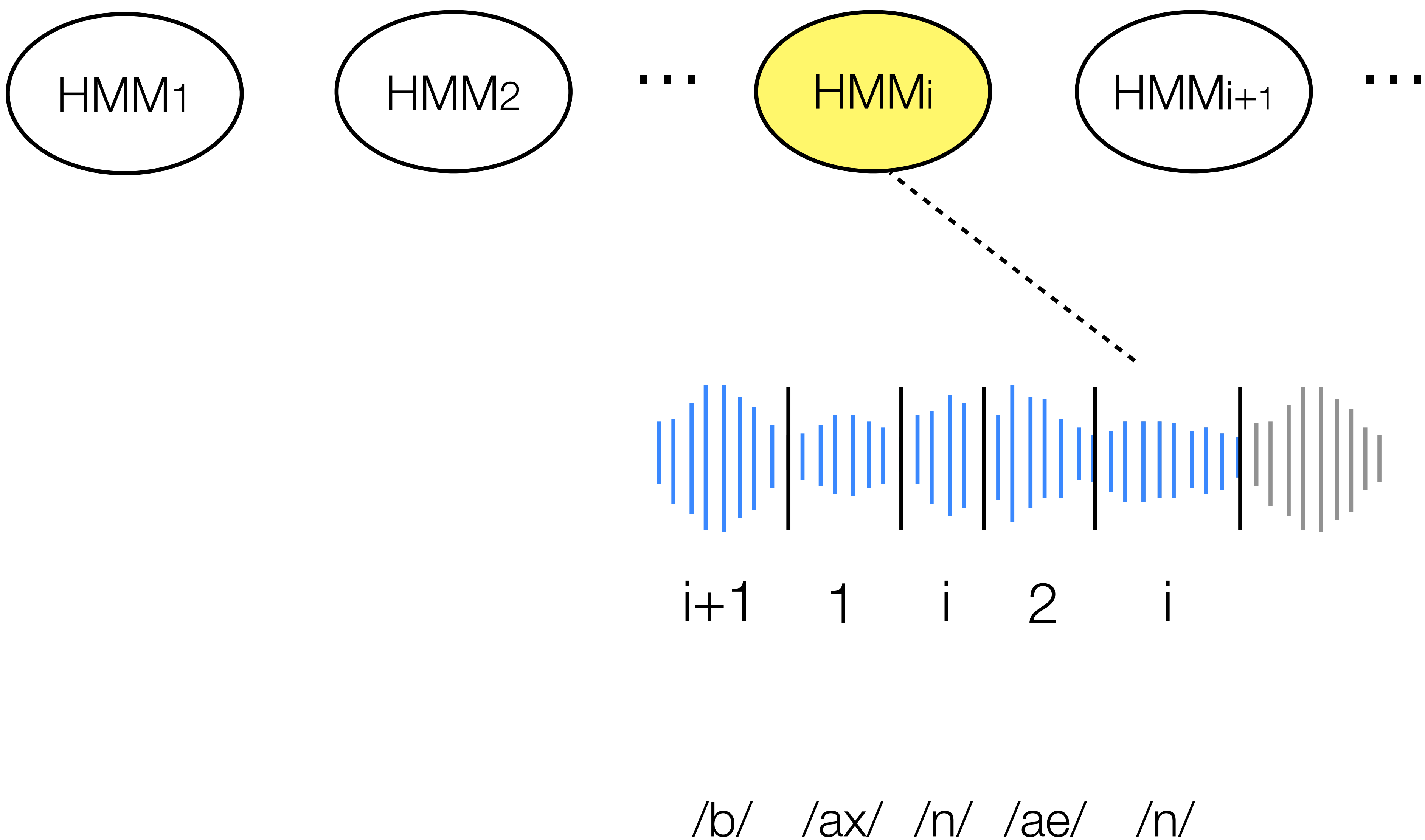
Generative Story

- A simple explanation of how a spoken utterance is generated



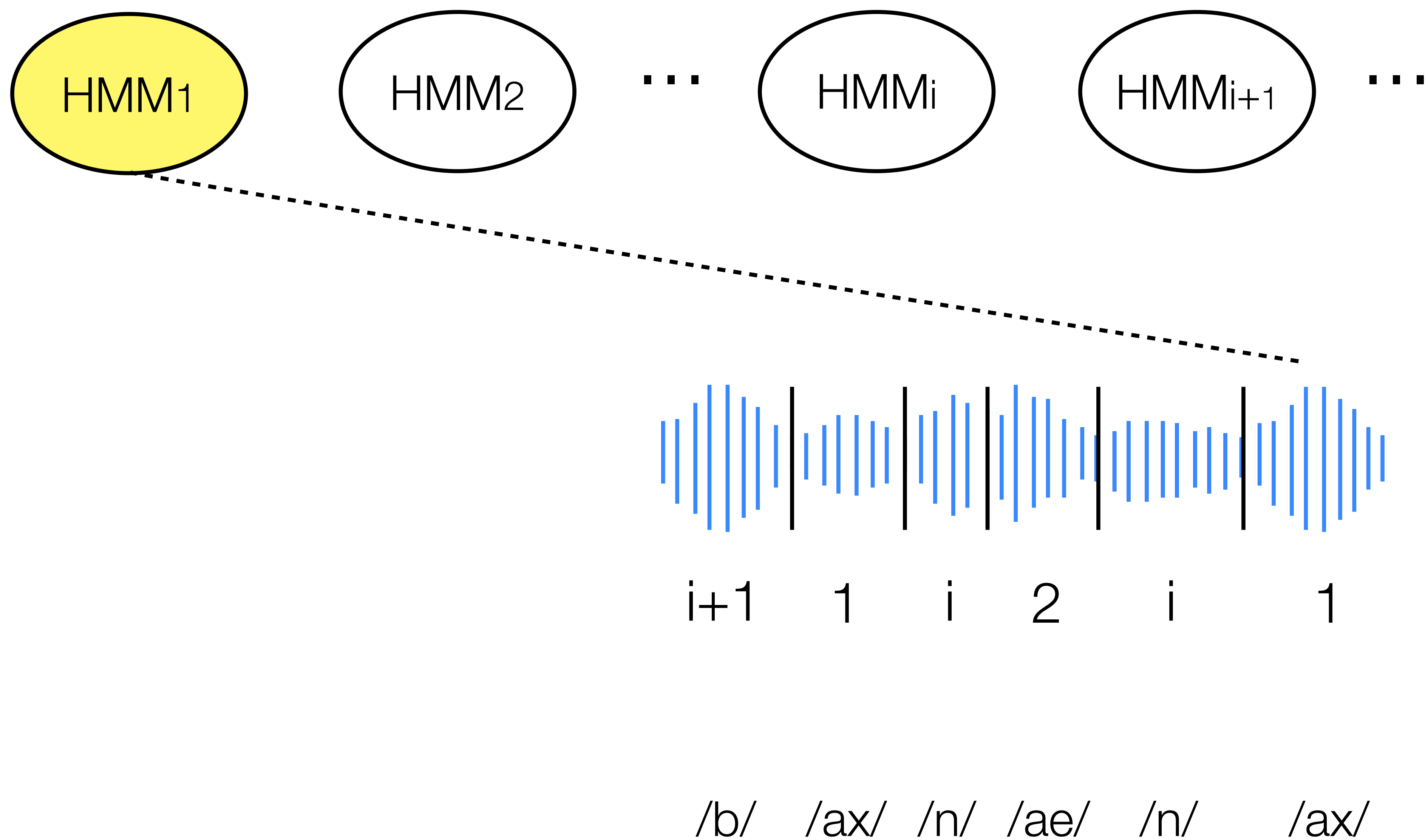
Generative Story

- A simple explanation of how a spoken utterance is generated



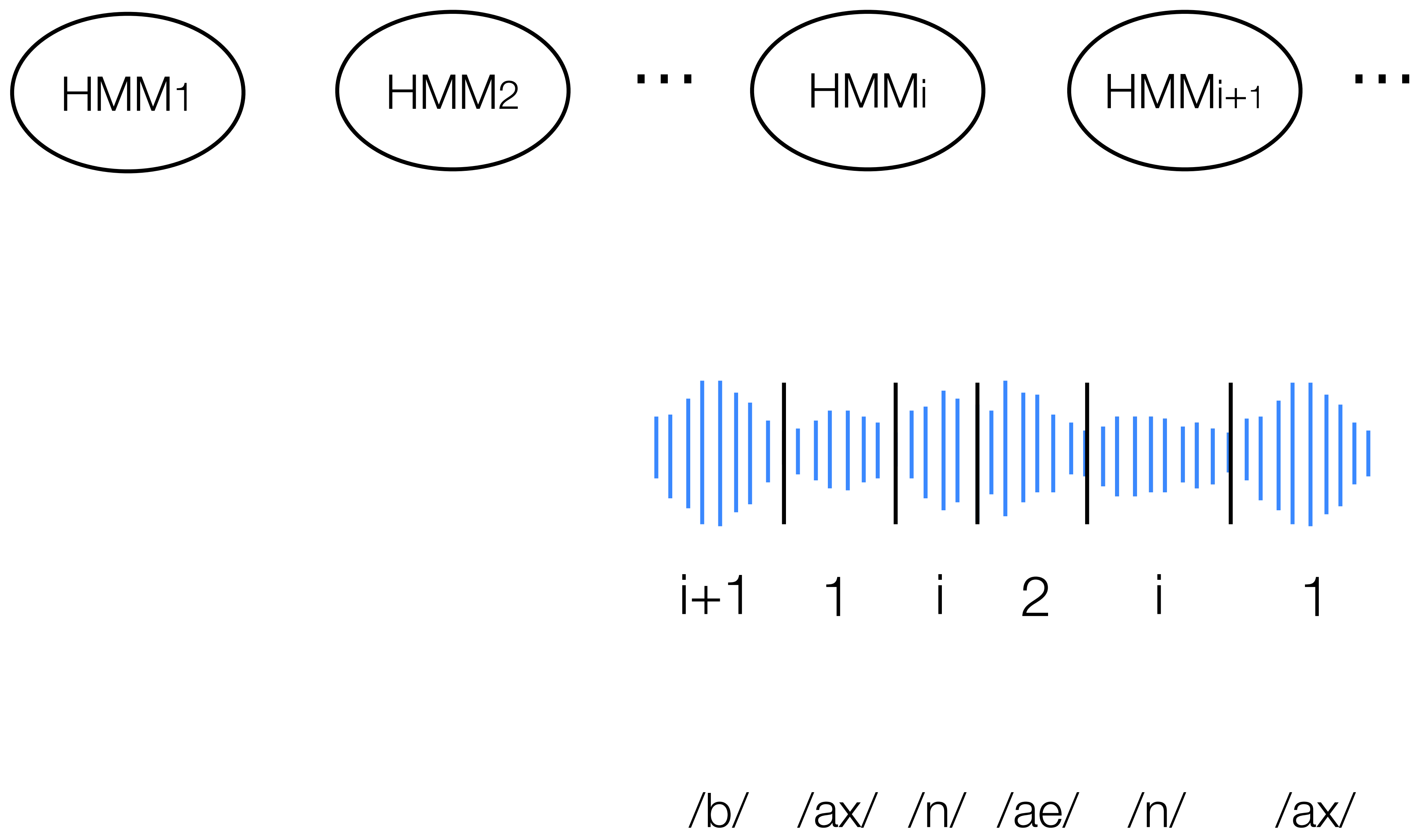
Generative Story

- A simple explanation of how a spoken utterance is generated



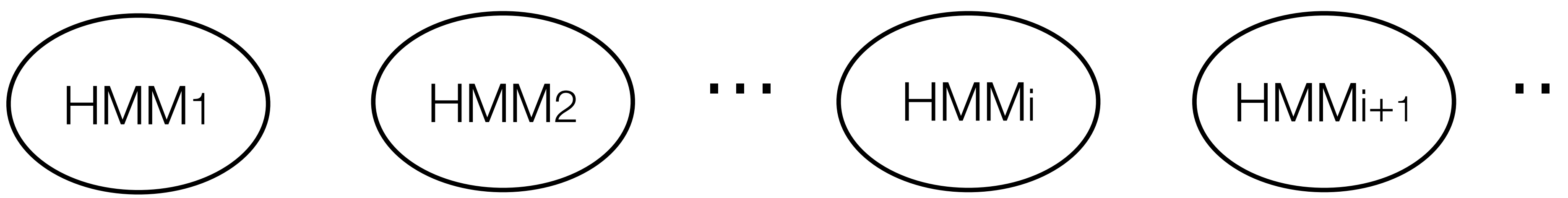
Generative Story

- A simple explanation of how a spoken utterance is generated

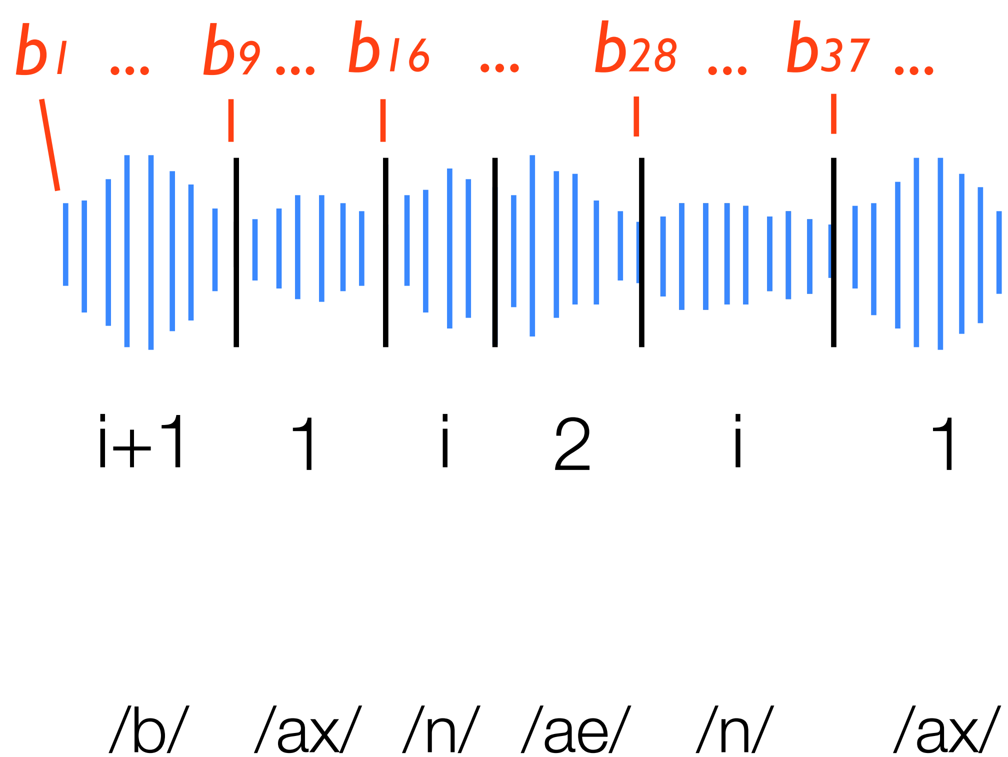


Generative Story

- A simple explanation of how a spoken utterance is generated

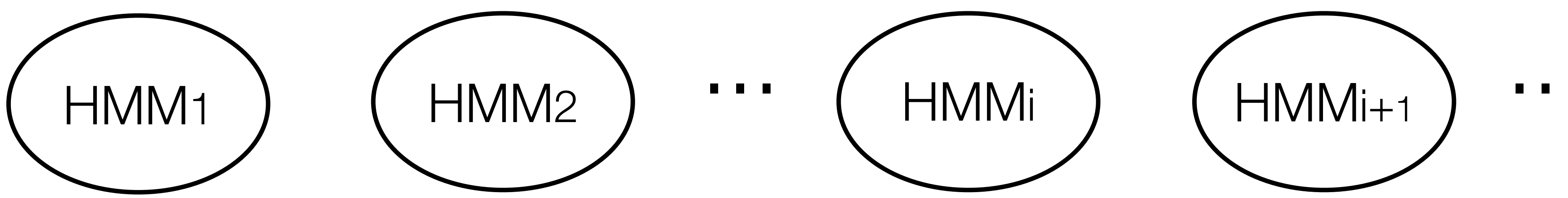


- Main latent variables
  - Phone boundaries (***b***)

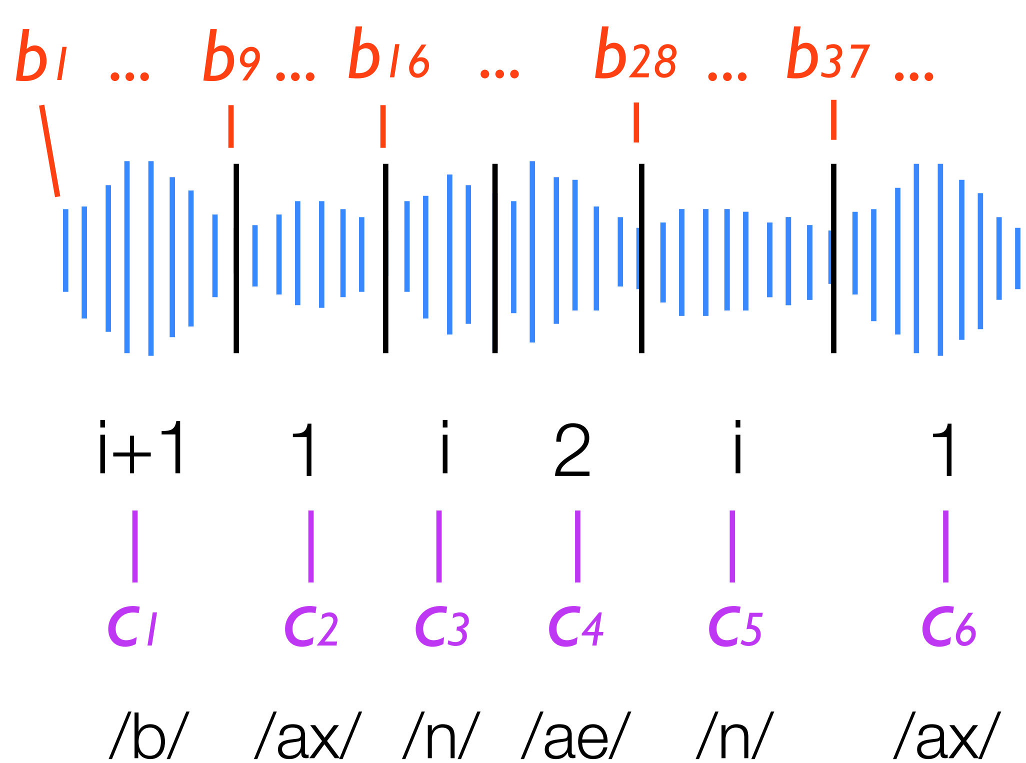


Generative Story

- A simple explanation of how a spoken utterance is generated

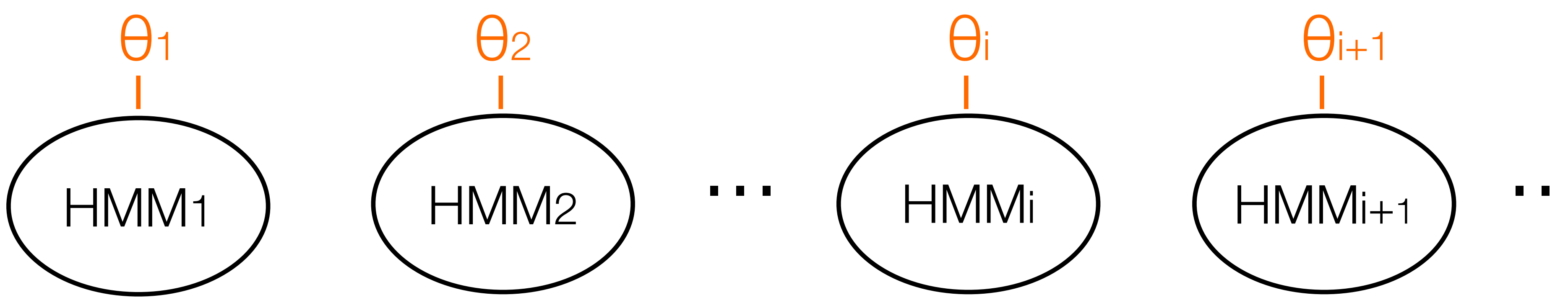


- Main latent variables
  - Phone boundaries (***b***)
  - Phone labels (***c***)

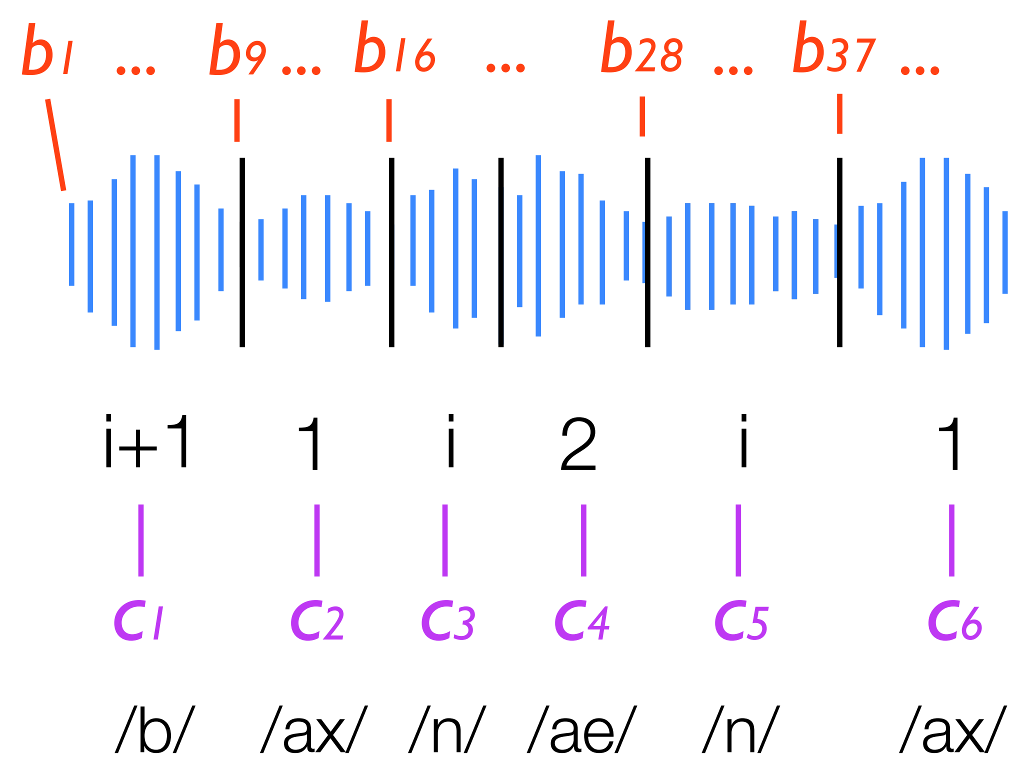


Generative Story

- A simple explanation of how a spoken utterance is generated

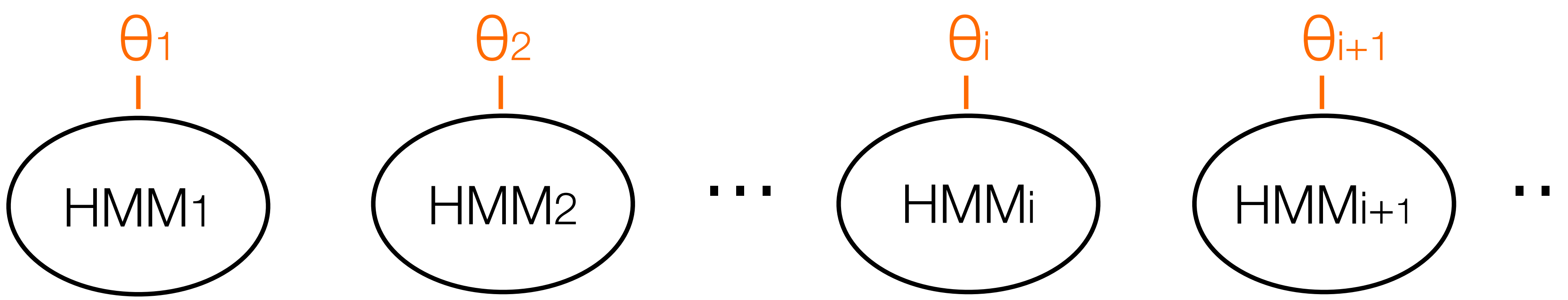


- Main latent variables
  - Phone boundaries (***b***)
  - Phone labels (***c***)
  - HMM parameters (**θ**)



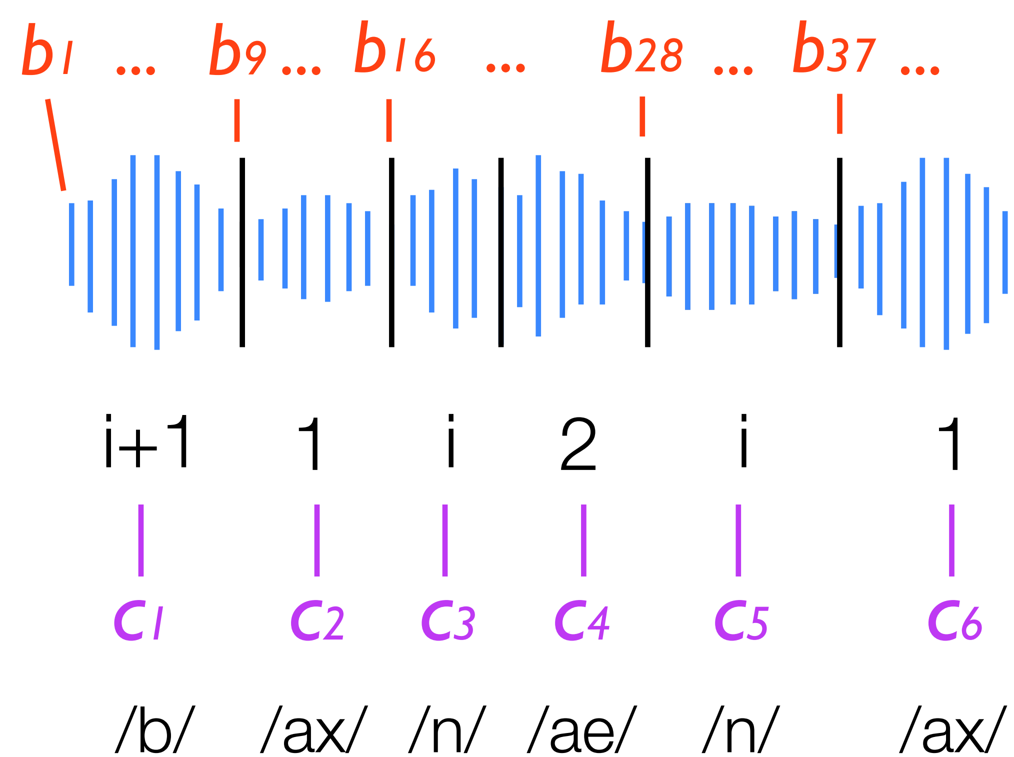
Generative Story

- A simple explanation of how a spoken utterance is generated

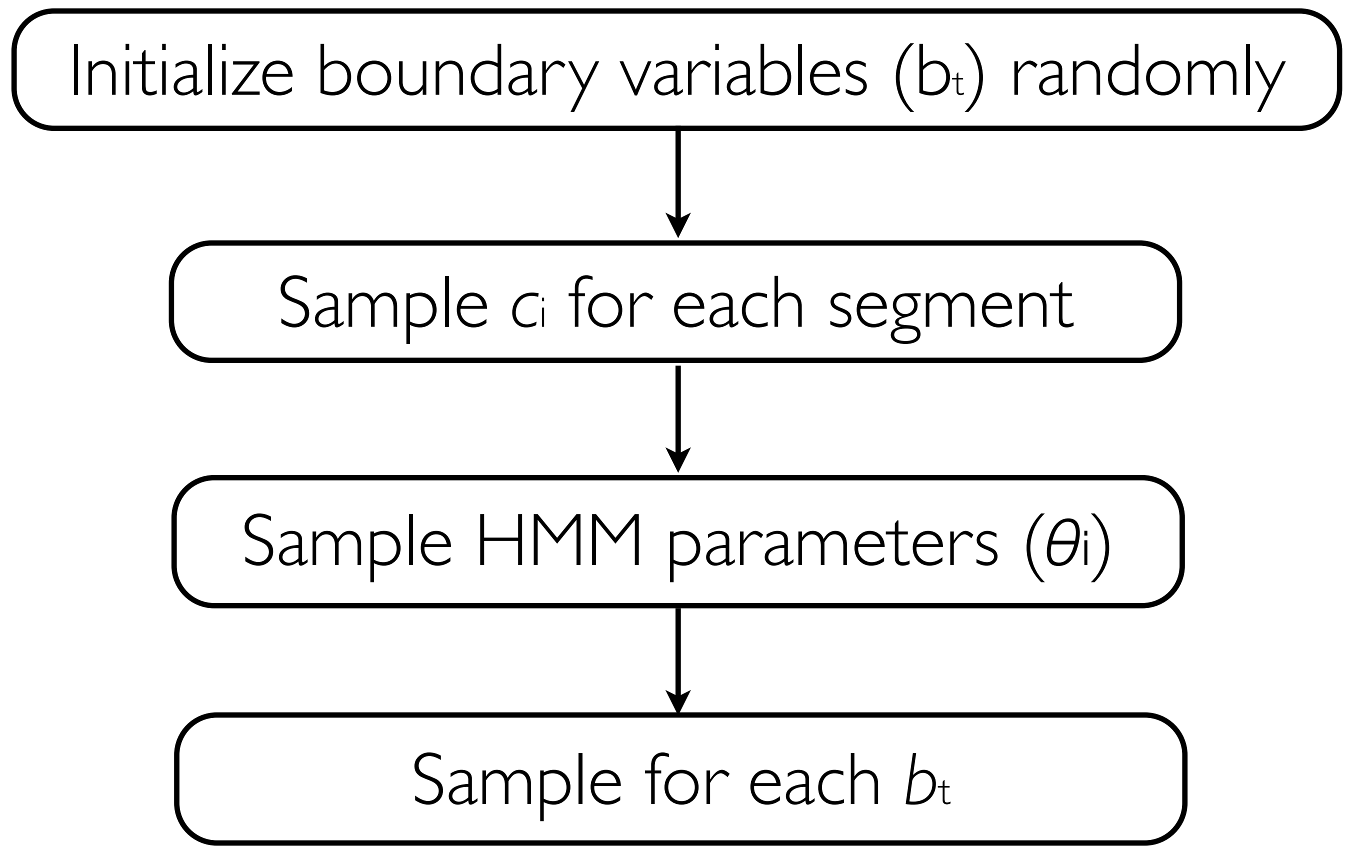


- Main latent variables
  - Phone boundaries (***b***)
  - Phone labels (***c***)
  - HMM parameters (**θ**)
  - # of HMMs (phones)

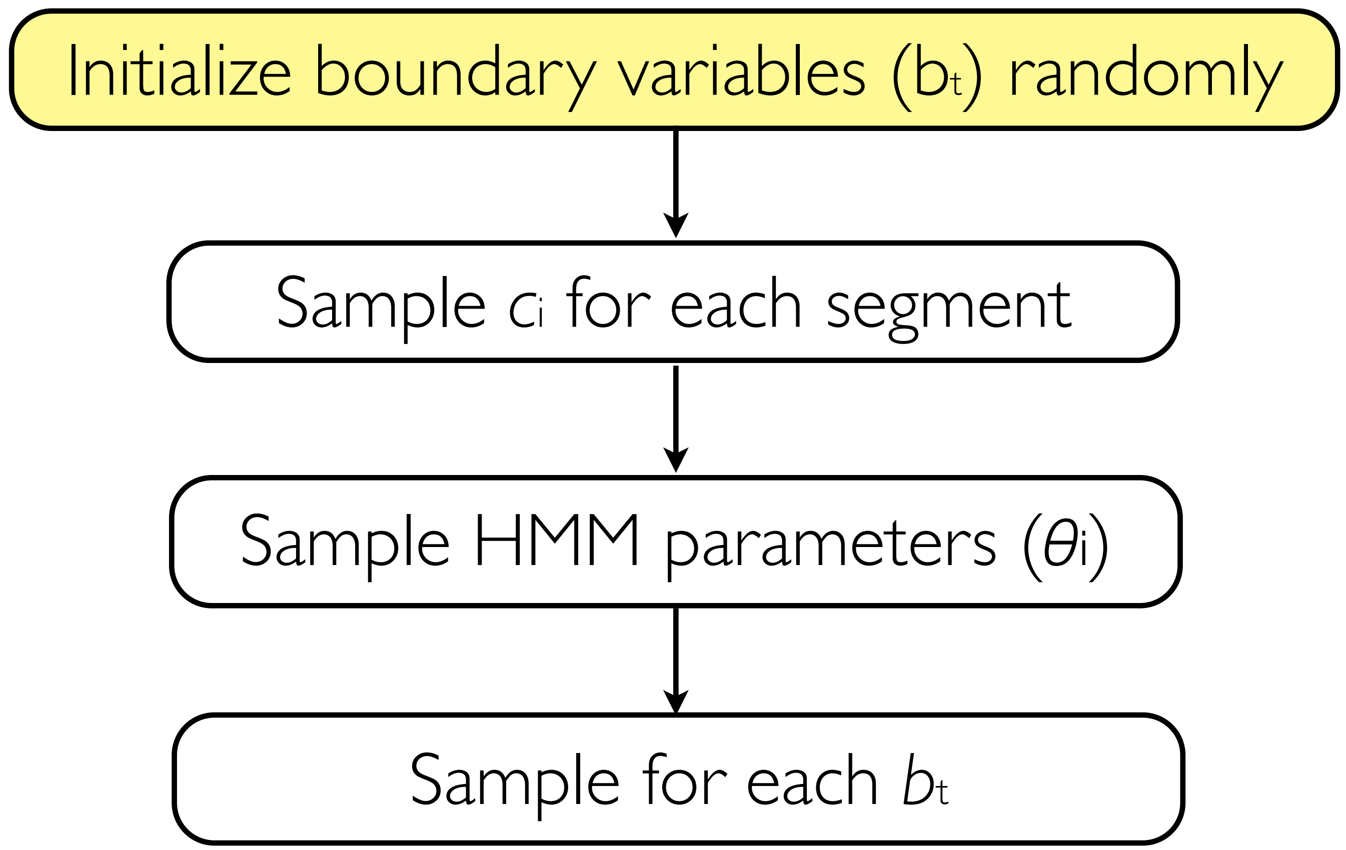
Dirichlet Process



Inference Procedure



Inference Procedure



Inference Procedure

Initialize boundary variables (b_t) randomly

↓

Sample c_i for each segment

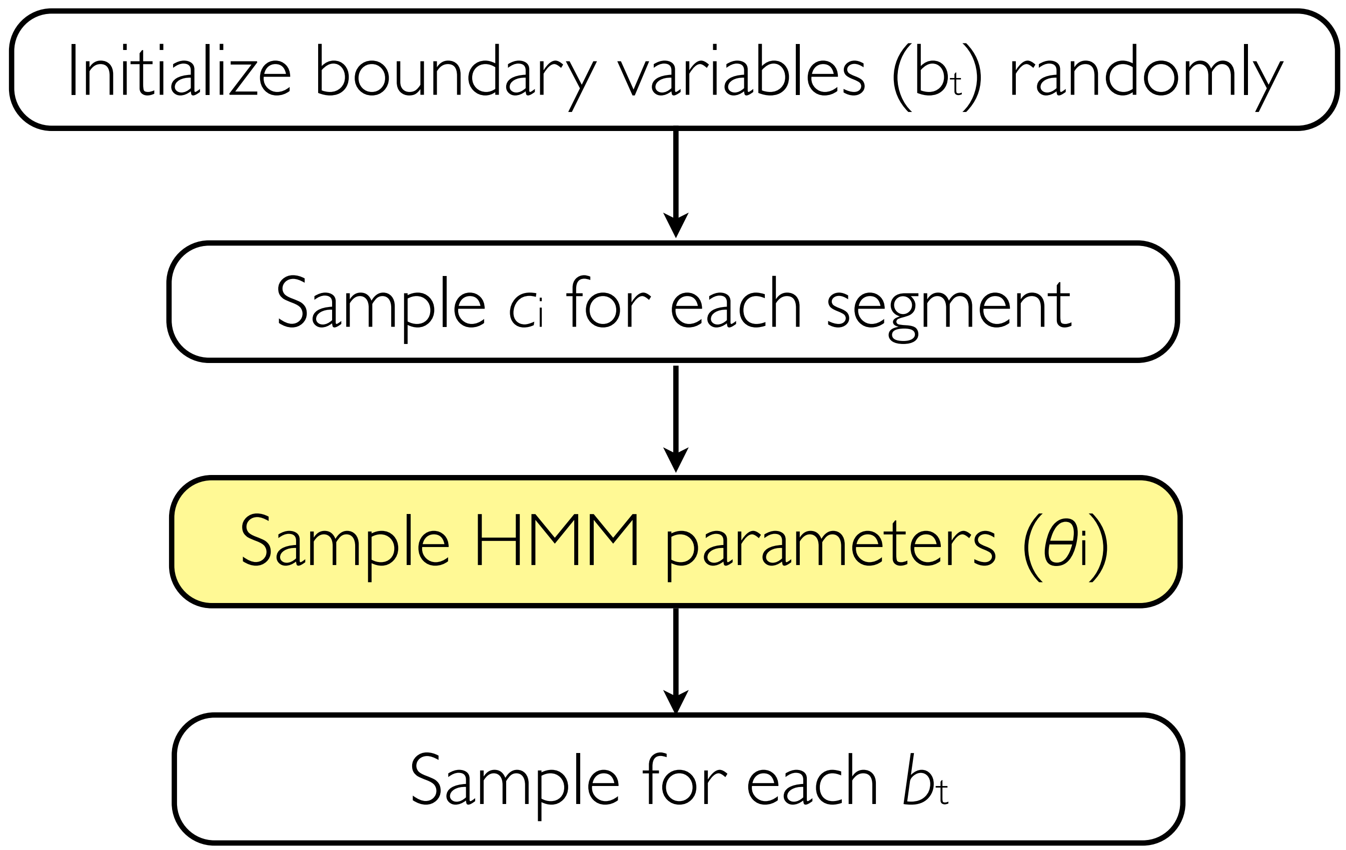
↓

Sample HMM parameters (θ_i)

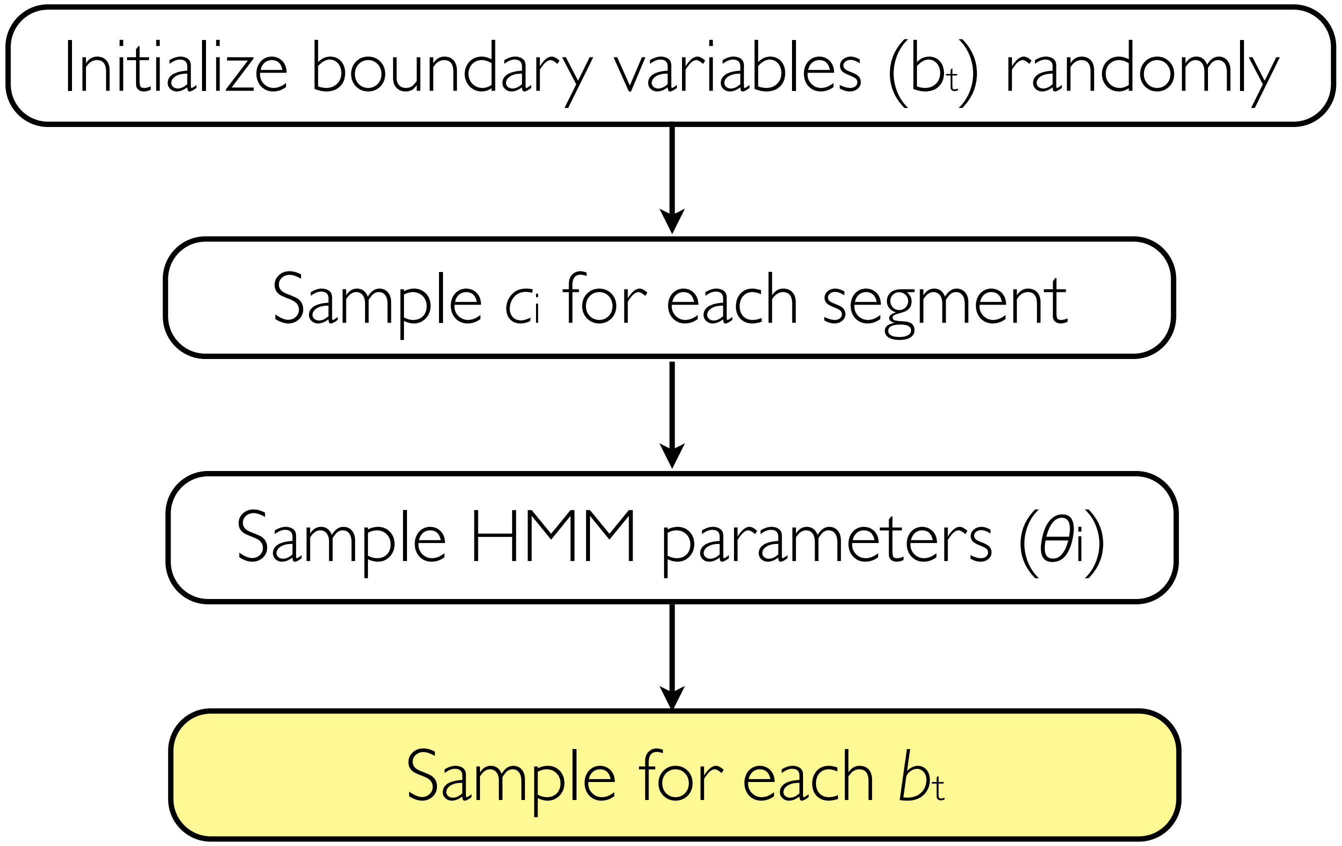
↓

Sample for each b_t

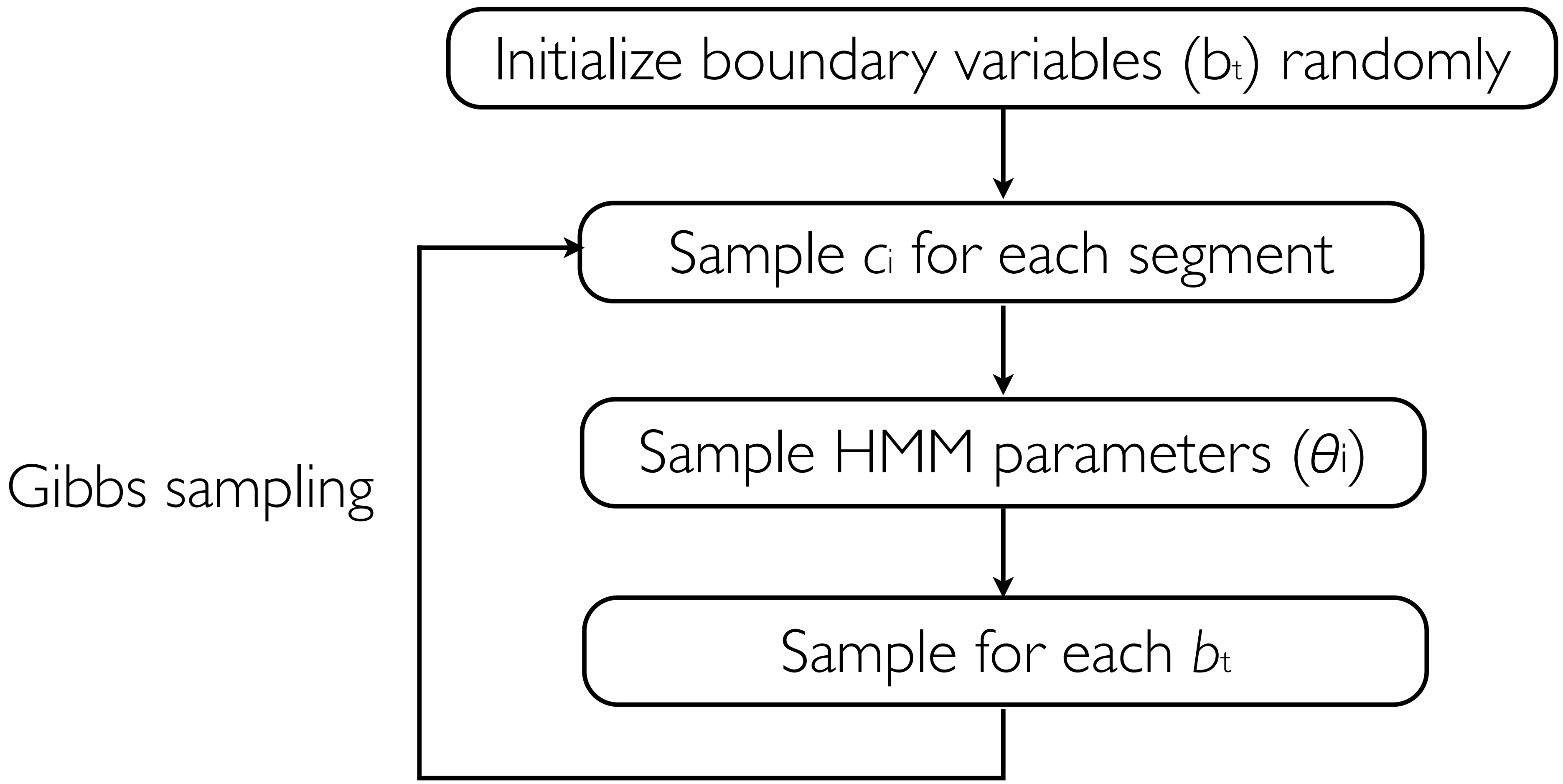
Inference Procedure



Inference Procedure

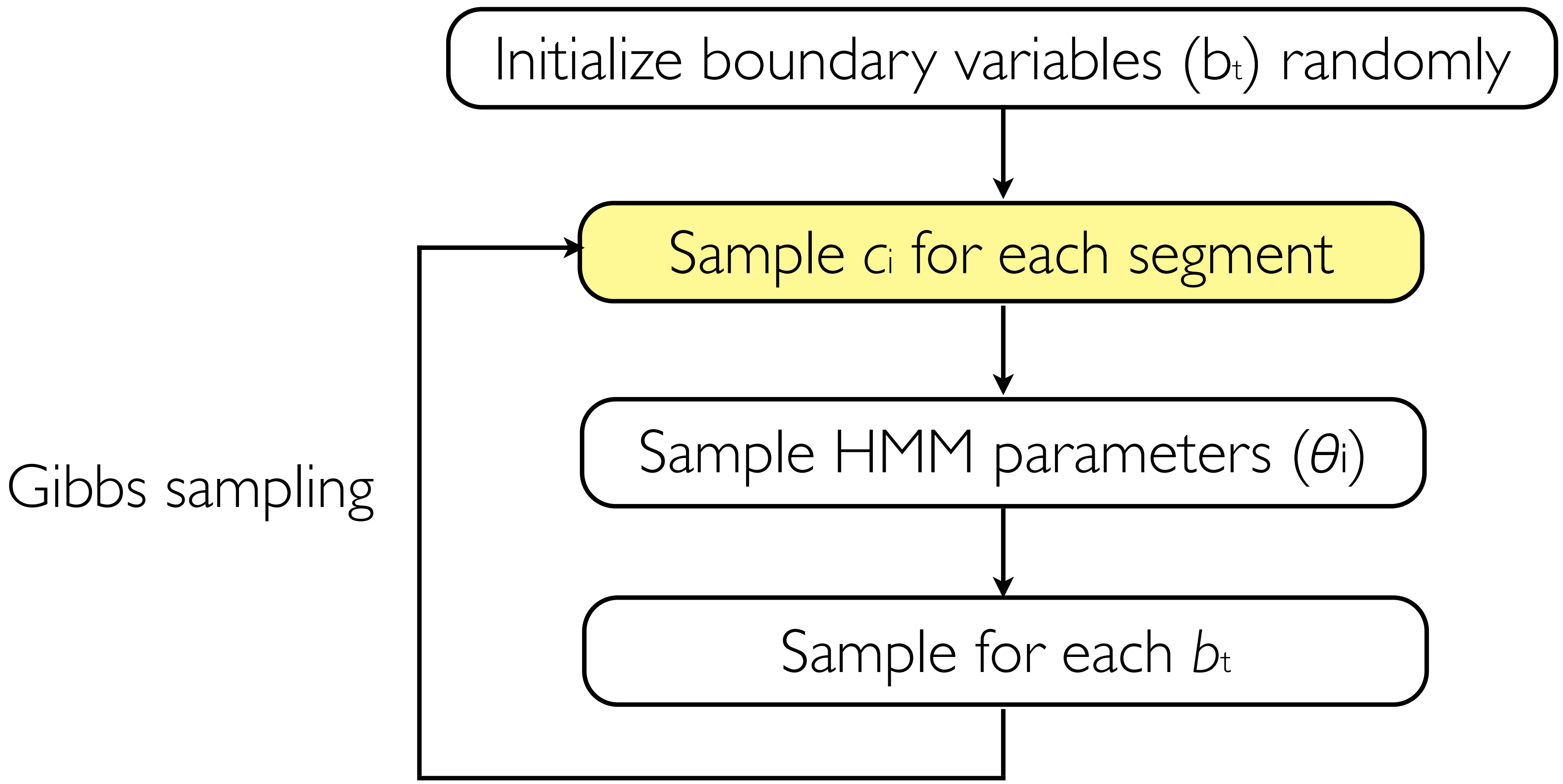


Inference Procedure



- **Iterate n times**
  - n = 20,000 in our experiments

Inference Procedure



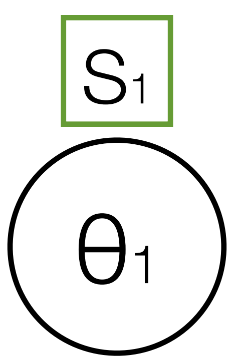
- **Iterate n times**
  - n = 20,000 in our experiments

DP as a Prior for Phone Labels (*c*)

- A Chinese Restaurant Process (CRP) representation
  - Each table is a phonetic unit
  - Each speech segment is a customer $s_i = [x_t, x_{t+1}, ... x_{t+L_i}]$

DP as a Prior for Phone Labels (c)

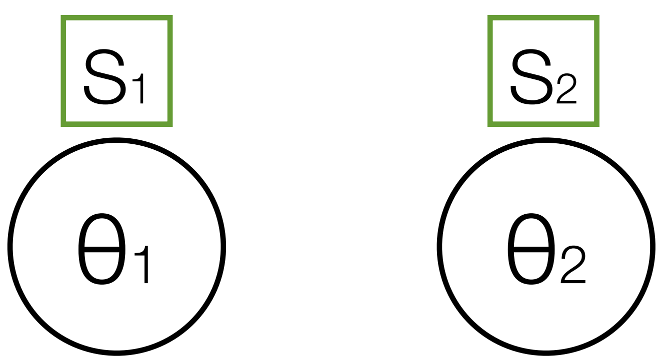
- A Chinese Restaurant Process (CRP) representation
  - Each table is a phonetic unit
  - Each speech segment is a customer $s_i = [x_t, x_{t+1}, \ldots x_{t+L_i}]$



$c_1 = 1$

DP as a Prior for Phone Labels (c)

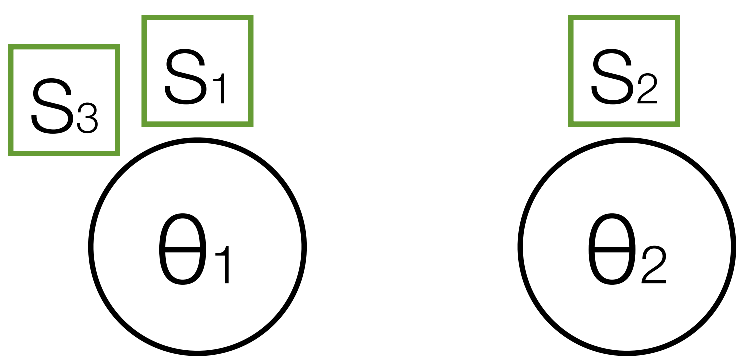
- A Chinese Restaurant Process (CRP) representation
  - Each table is a phonetic unit
  - Each speech segment is a customer $s_i = [x_t, x_{t+1}, \ldots x_{t+L_i}]$



$c_1 = 1$ $c_2 = 2$

DP as a Prior for Phone Labels (c)

- A Chinese Restaurant Process (CRP) representation
  - Each table is a phonetic unit
  - Each speech segment is a customer $s_i = [x_t, x_{t+1}, \ldots x_{t+L_i}]$



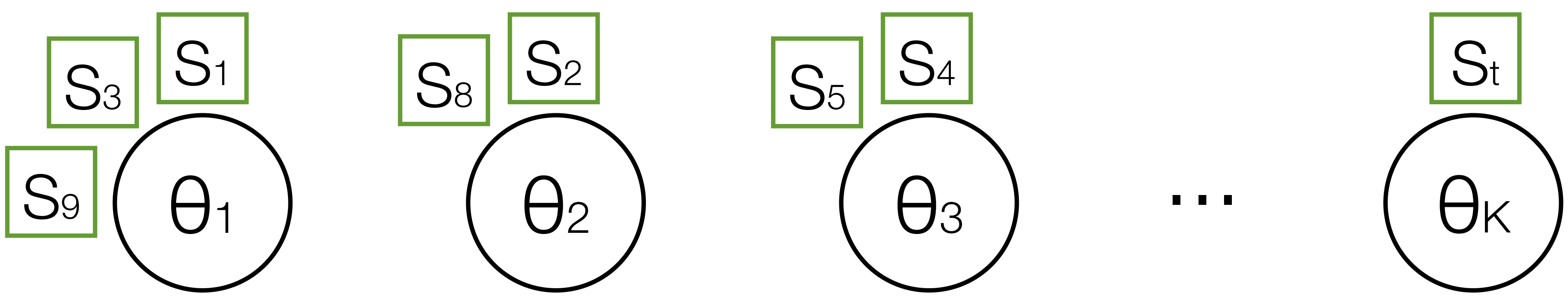
$c_1 = 1$

$c_3 = 1$

$c_2 = 2$

DP as a Prior for Phone Labels (c)

- A Chinese Restaurant Process (CRP) representation
  - Each table is a phonetic unit
  - Each speech segment is a customer $s_i = [x_t, x_{t+1}, \ldots x_{t+L_i}]$

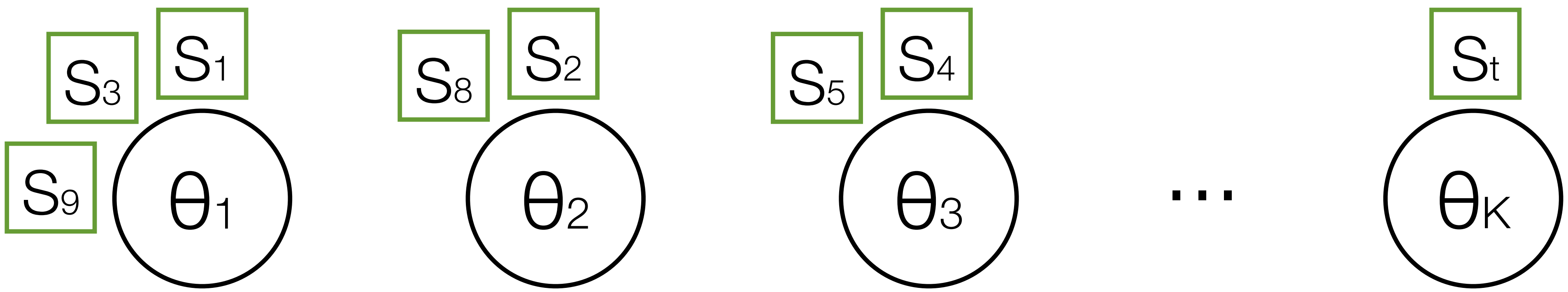


$c_1 = 1$ $c_2 = 2$ $c_4 = 3$ $c_t = K$

$c_3 = 1$ $c_8 = 2$ $c_5 = 3$

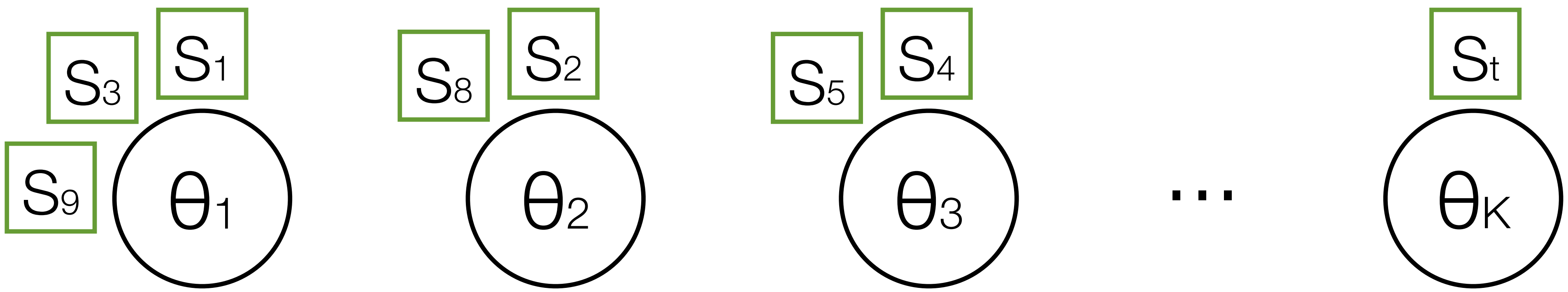
$c_9 = 1$

Posterior Distribution for c_i



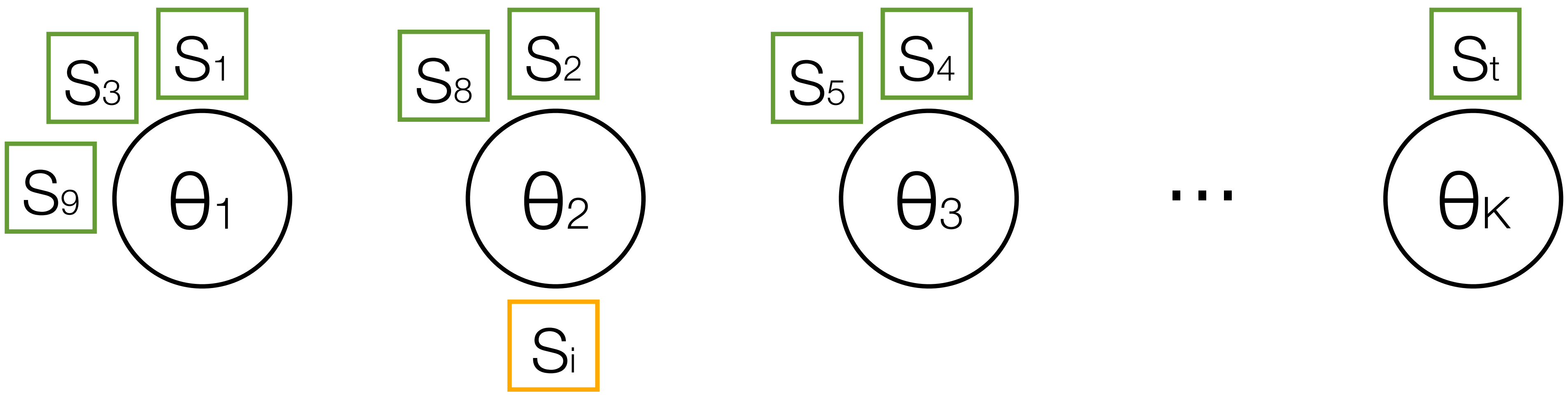
- For a new segment (s_i), the posterior probability distribution of c_i :

Posterior Distribution for c_i



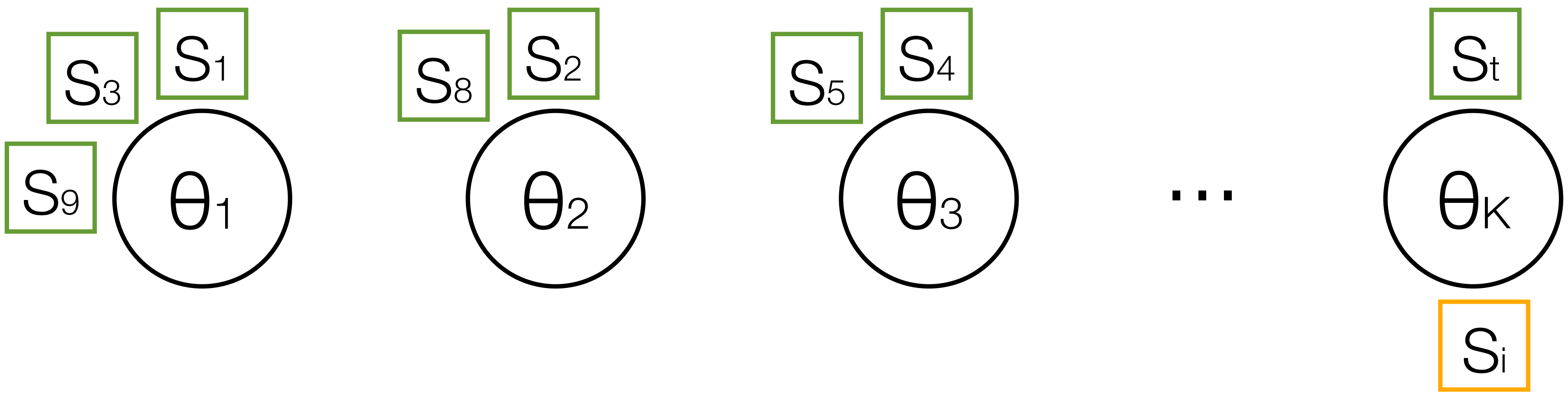
- For a new segment (s_i), the posterior probability distribution of c_i :
  - s_i sits at an occupied table ⟶ s_i is not a new phone

Posterior Distribution for c_i



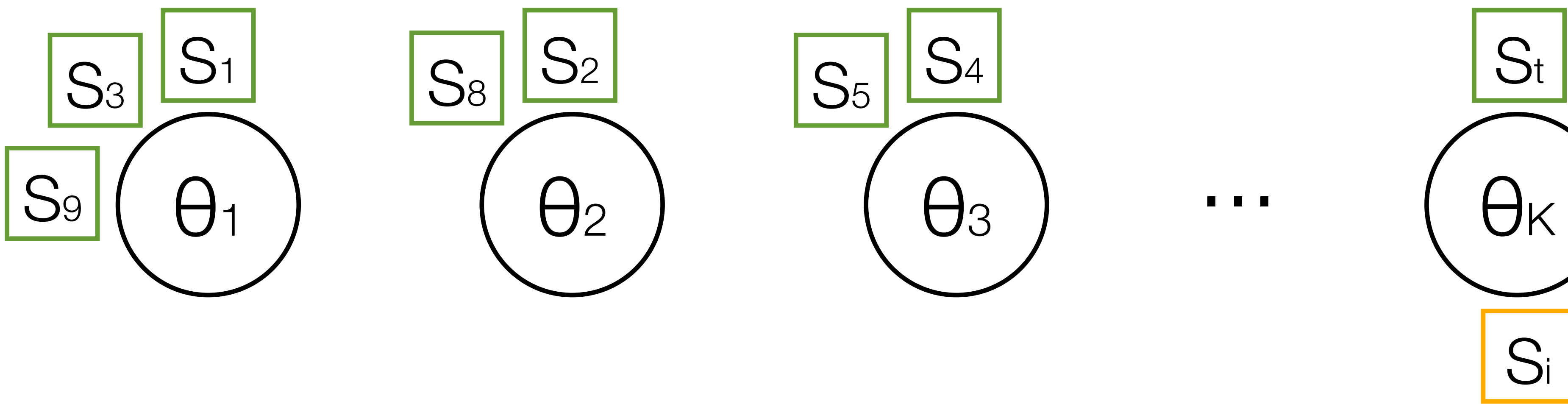
- For a new segment (s_i), the posterior probability distribution of c_i :
  - s_i sits at an occupied table ⟶ s_i is not a new phone

Posterior Distribution for c_i



- For a new segment (s_i), the posterior probability distribution of c_i :
  - s_i sits at an occupied table ⟶ s_i is not a new phone

Posterior Distribution for c_i



- For a new segment (s_i), the posterior probability distribution of c_i :

  - s_i sits at an occupied table $\longrightarrow$ s_i is not a new phone

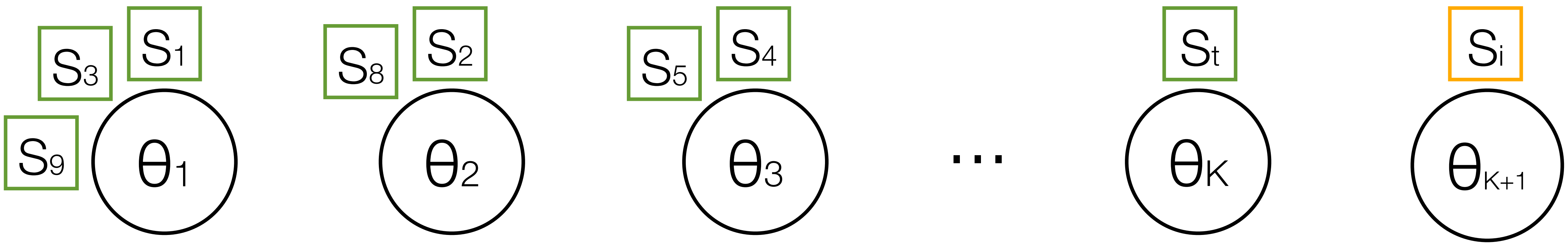
$$\underbrace{p(c_i = k, 1 \le k \le K \mid \cdots)}_{\text{posterior probability}} \propto \underbrace{\frac{n_k}{N+\alpha}}_{\text{DP prior}} \underbrace{p(s_i \mid \theta_k)}_{\text{likelihood}}$$

n_k : number of customers at table k

N : number of costumers seen so far

α : concentration parameter of DP

Posterior Distribution for c_i

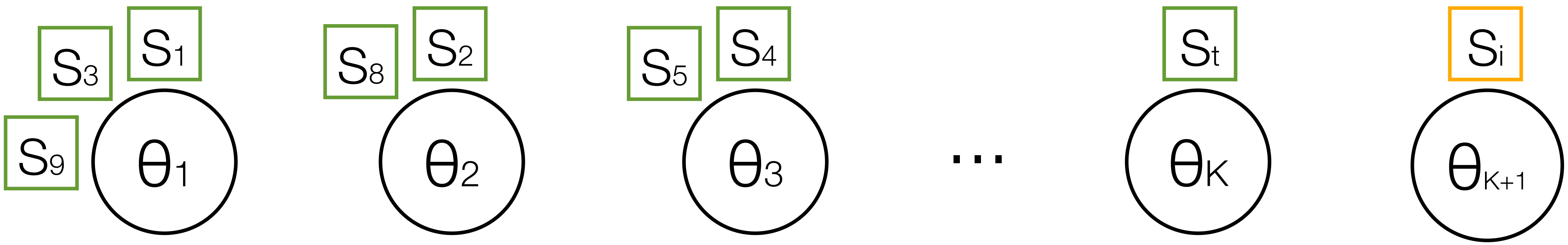


- **For a new segment (s_i), the posterior probability distribution of c_i :**
  - s_i sits at an occupied table $\longrightarrow$ s_i is not a new phone

$$p(c_i = k, 1 \le k \le K \mid \cdots) \propto \frac{n_k}{N + \alpha} p(s_i \mid \theta_k)$$

  - s_i opens a new table $\longrightarrow$ s_i is a new phone

Posterior Distribution for c_i



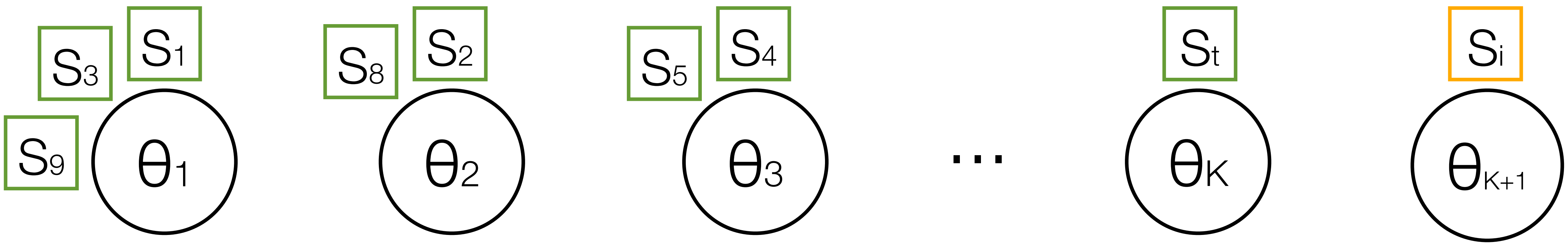
- For a new segment (s_i), the posterior probability distribution of c_i :
  - s_i sits at an occupied table $\longrightarrow$ s_i is not a new phone

$$p(c_i = k, 1 \le k \le K \mid \cdots) \propto \frac{n_k}{N+\alpha} p(s_i \mid \theta_k)$$

  - s_i opens a new table $\longrightarrow$ s_i is a new phone

$$p(c_i = K+1 \mid \cdots) \propto \frac{\alpha}{N+\alpha} \int_\theta p(s_i \mid \theta) d\theta$$

Posterior Distribution for c_i



- **For a new segment (s_i), the posterior probability distribution of c_i :**
  - s_i sits at an occupied table $\longrightarrow$ s_i is not a new phone

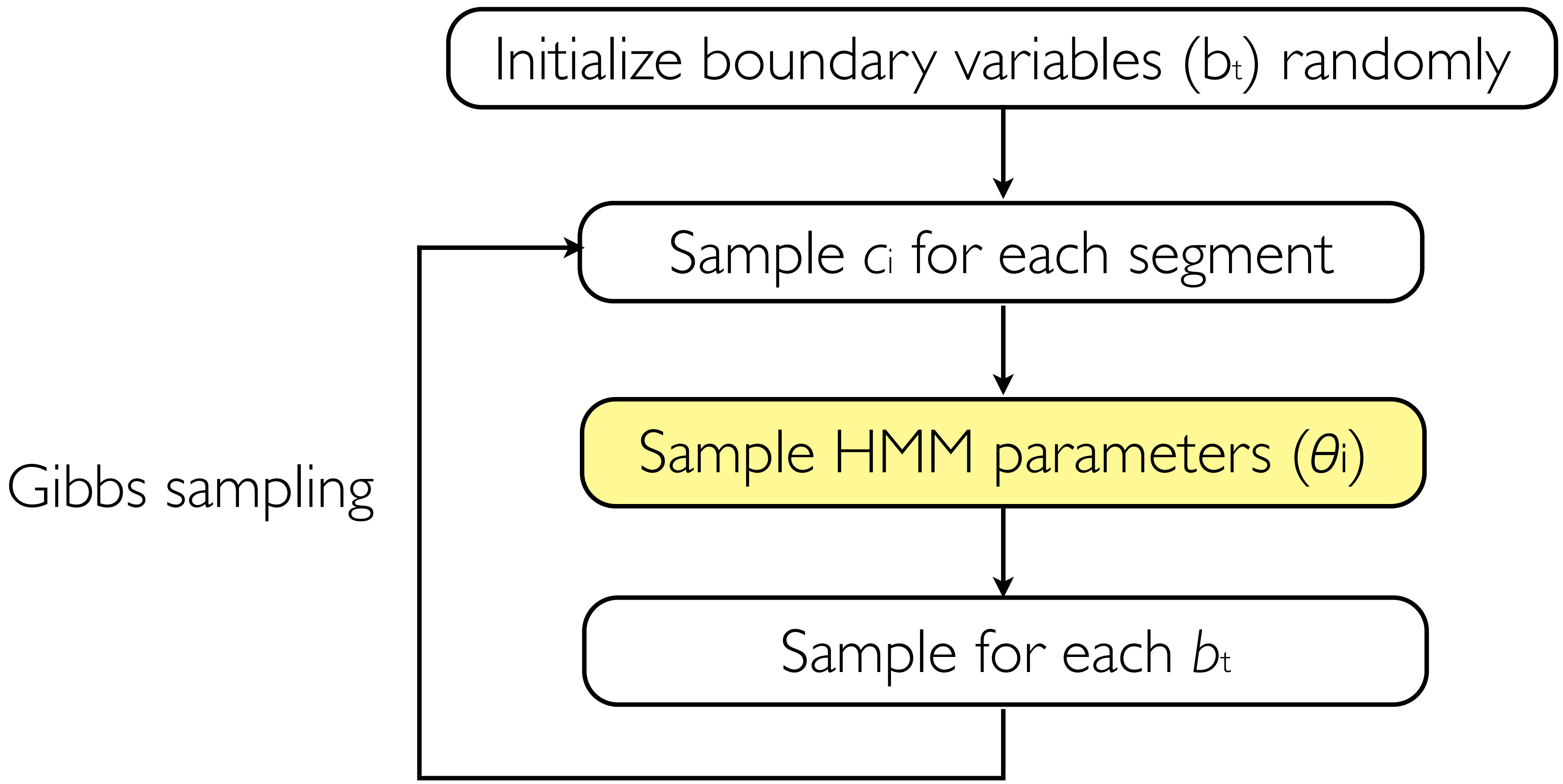
$$p(c_i = k, 1 \leq k \leq K \mid \cdots) \propto \frac{n_k}{N+\alpha} p(s_i \mid \theta_k)$$

  - s_i opens a new table $\longrightarrow$ s_i is a new phone

$$p(c_i = K+1 \mid \cdots) \propto \frac{\alpha}{N+\alpha} \int_\theta p(s_i \mid \theta) d\theta$$

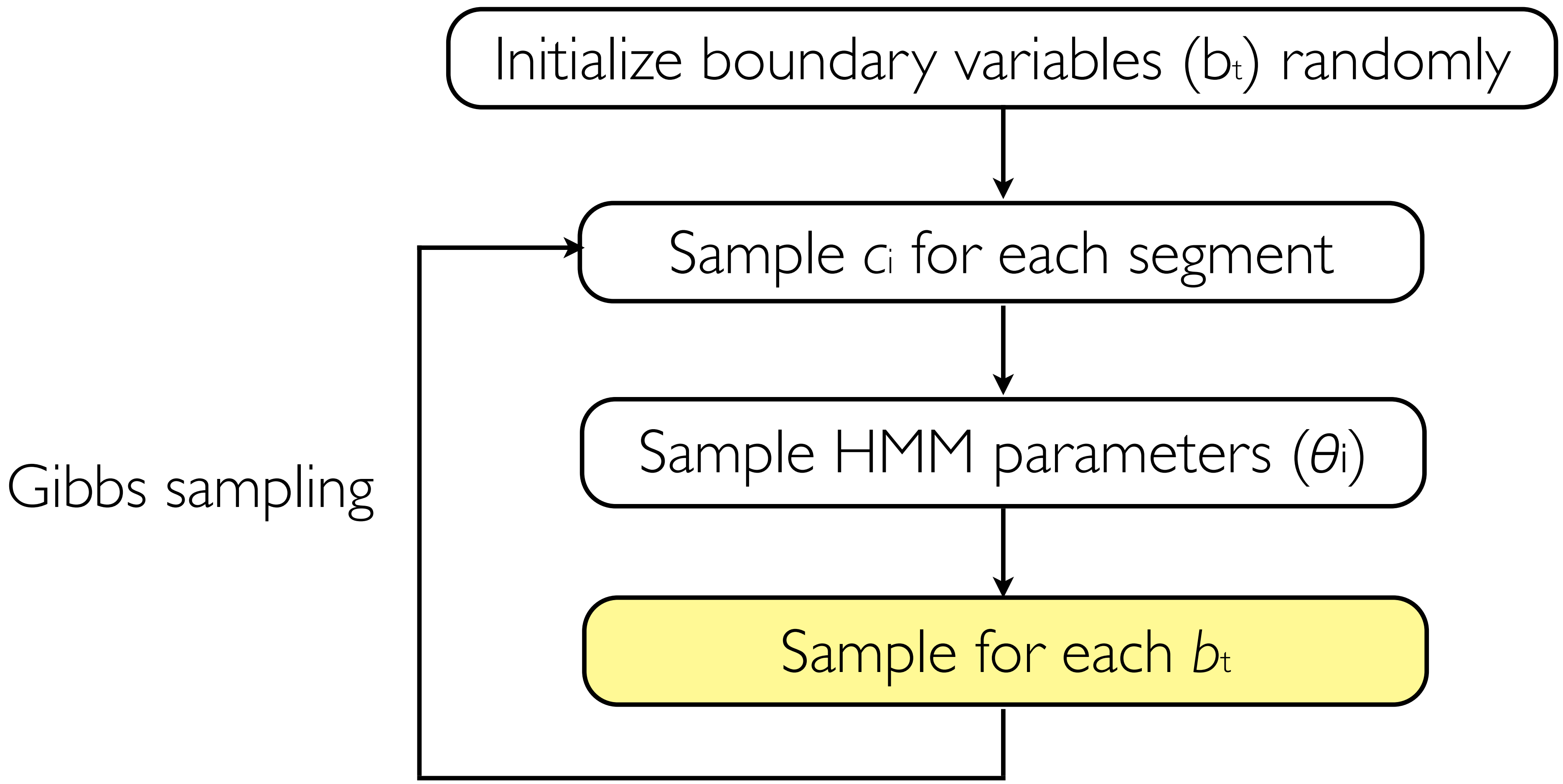
Generate a sample for c_i

Inference Procedure



- **Iterate n times**
  - n = 20,000 in our experiments

Inference Procedure



- **Iterate n times**
  - n = 20,000 in our experiments

Experiments

- **Data set**
  - TIMIT corpus
  - Multi-speaker, clean read speech, 16kHz sampling rate

Experiments

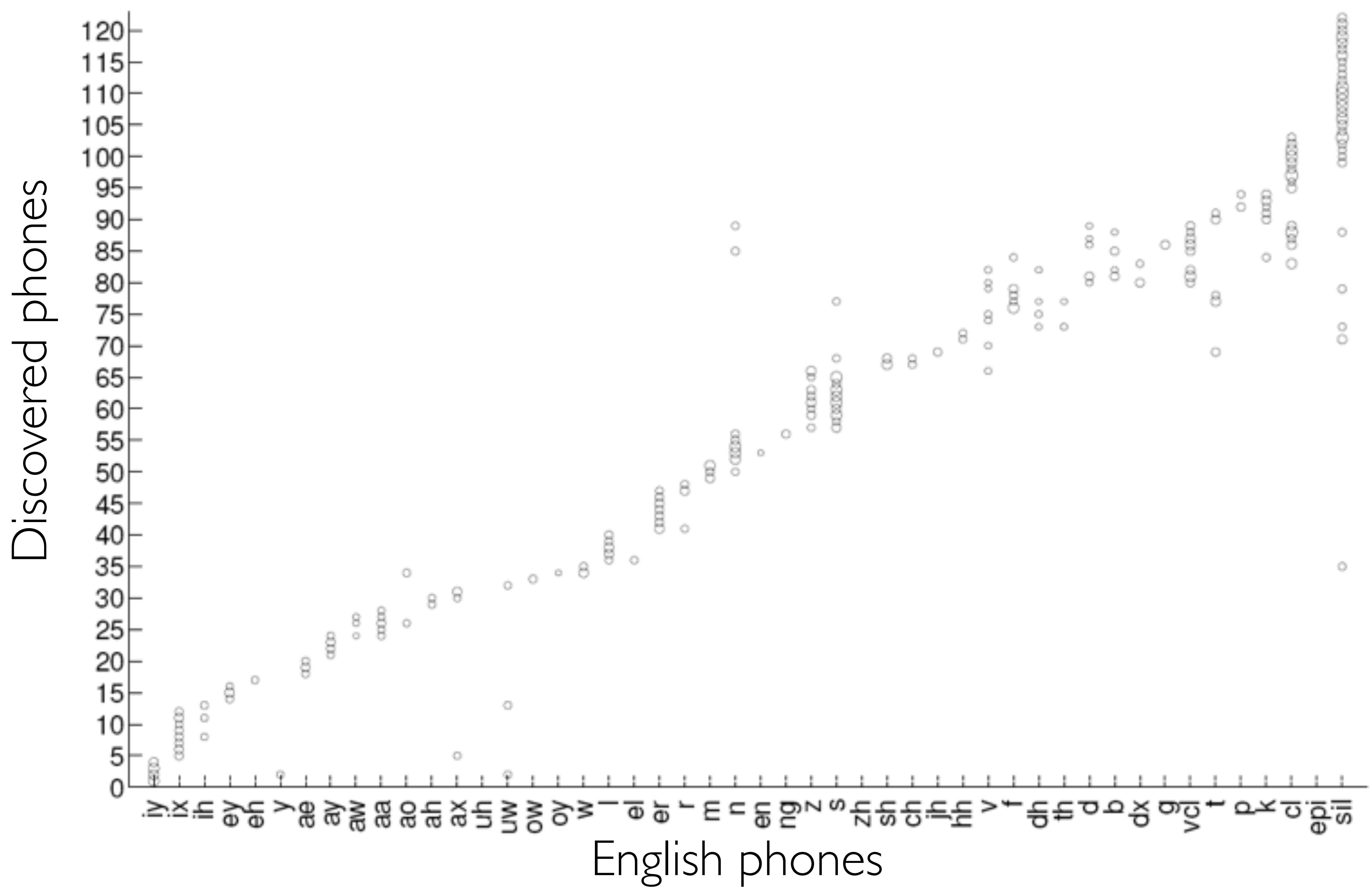
- **Data set**
  - TIMIT corpus
  - Multi-speaker, clean read speech, 16kHz sampling rate
- **Qualitative assessment**
  - Correlation between induced phone units and English phones
  - Results learned from 3696 utterances

Experiments

- **Data set**
  - TIMIT corpus
  - Multi-speaker, clean read speech, 16kHz sampling rate
- **Qualitative assessment**
  - Correlation between induced phone units and English phones
  - Results learned from 3696 utterances
- **Quantitative assessments**
  - Phone segmentation
  - (Query-by-example spoken term detection)

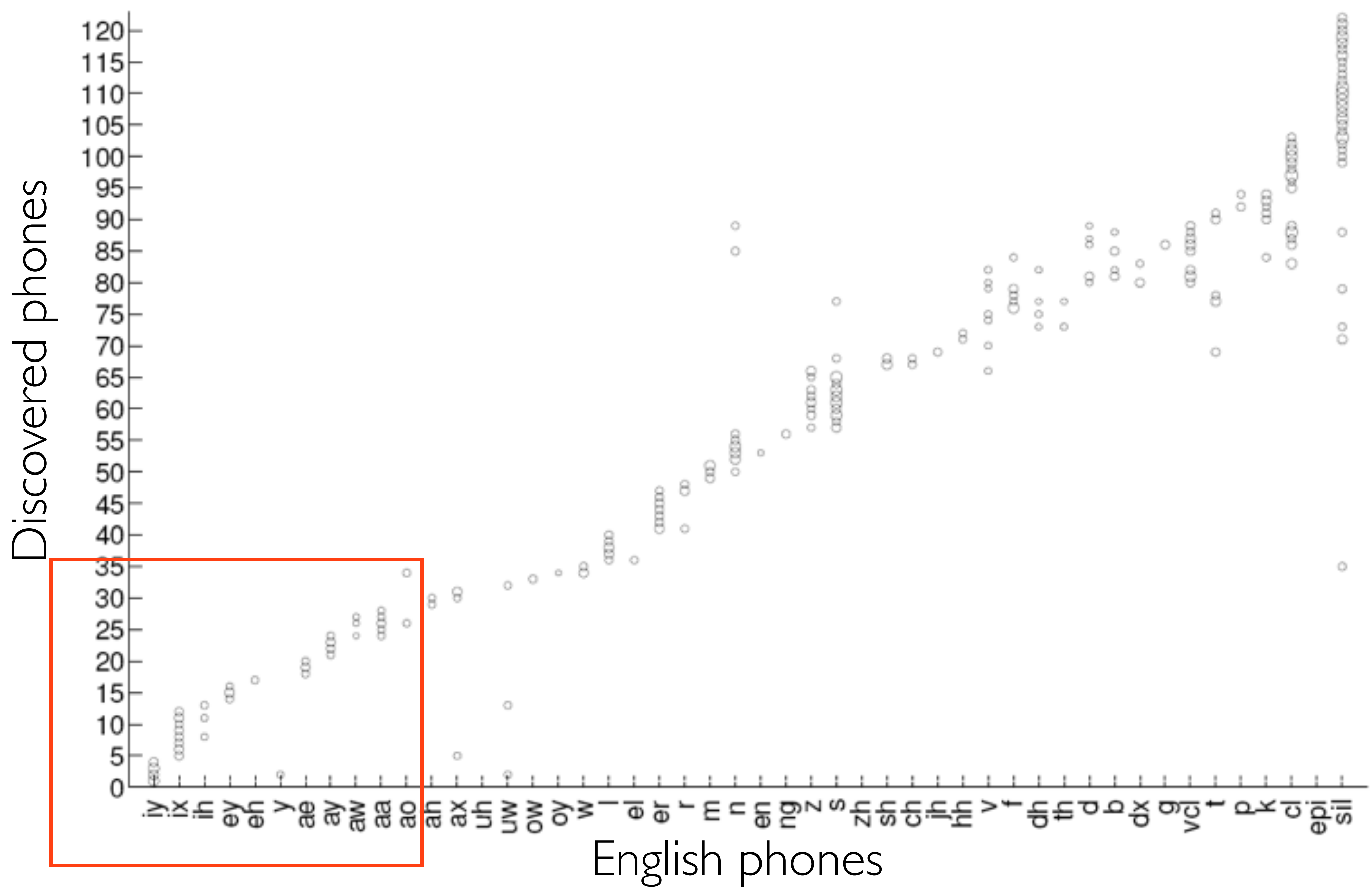
Discovered Phone Units -- 3696 utterances

- 123 phone units discovered from 3696 TIMIT utterances
  - A fine correlation between discovered phones and English phones



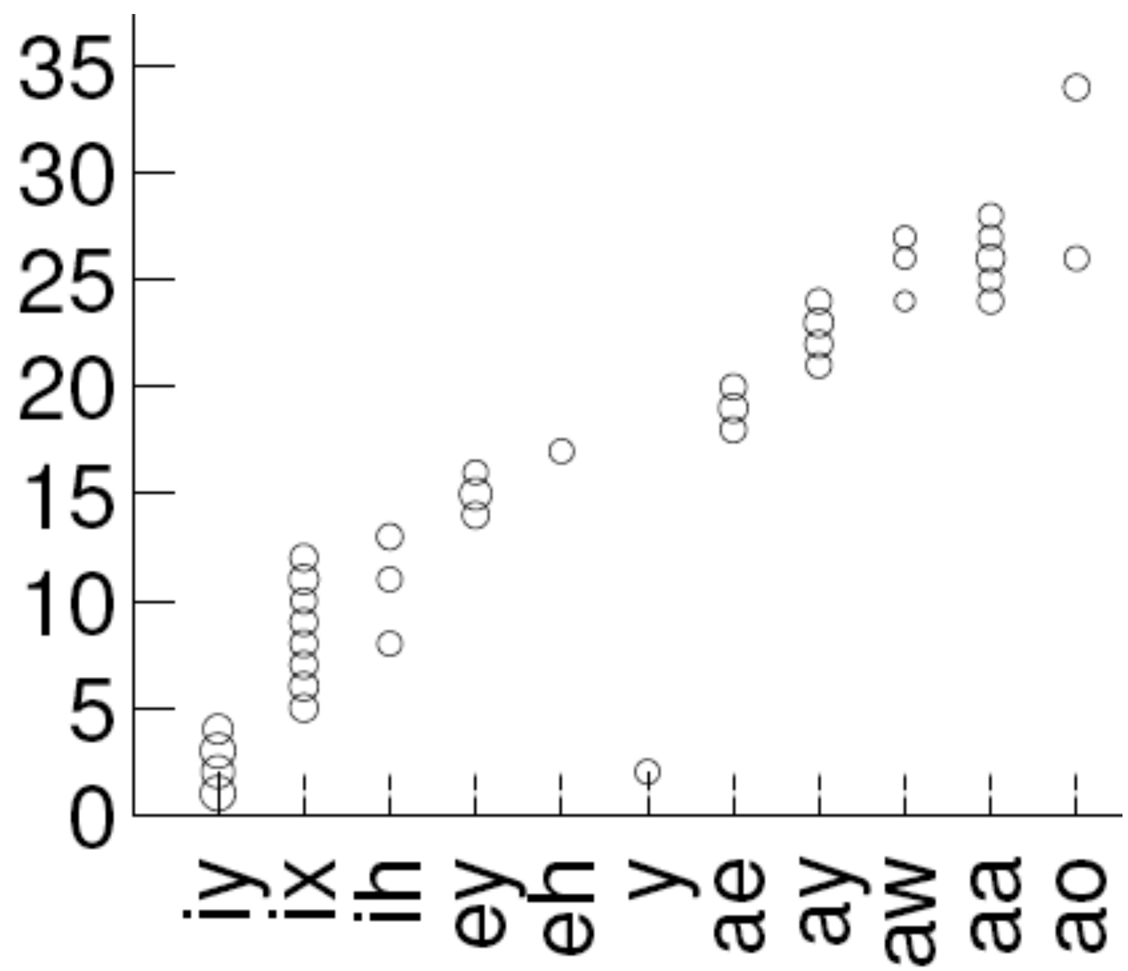
Discovered Phone Units -- 3696 utterances

- 123 phone units discovered from 3696 TIMIT utterances
  - A fine correlation between discovered phones and English phones



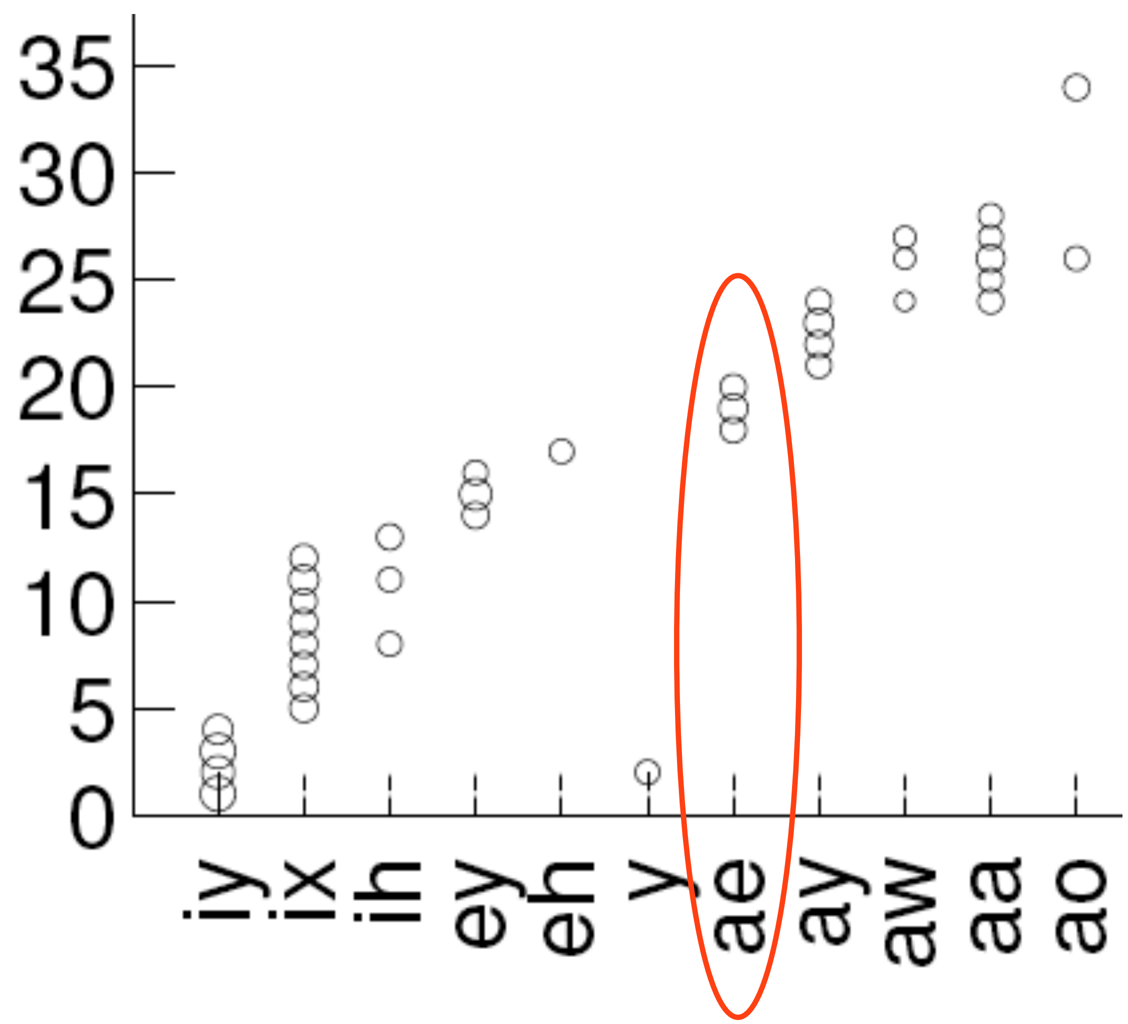
Discovered Phone Units -- 3696 utterances

- 123 phone units discovered from 3696 TIMIT utterances
  - A fine correlation between discovered phones and English phones



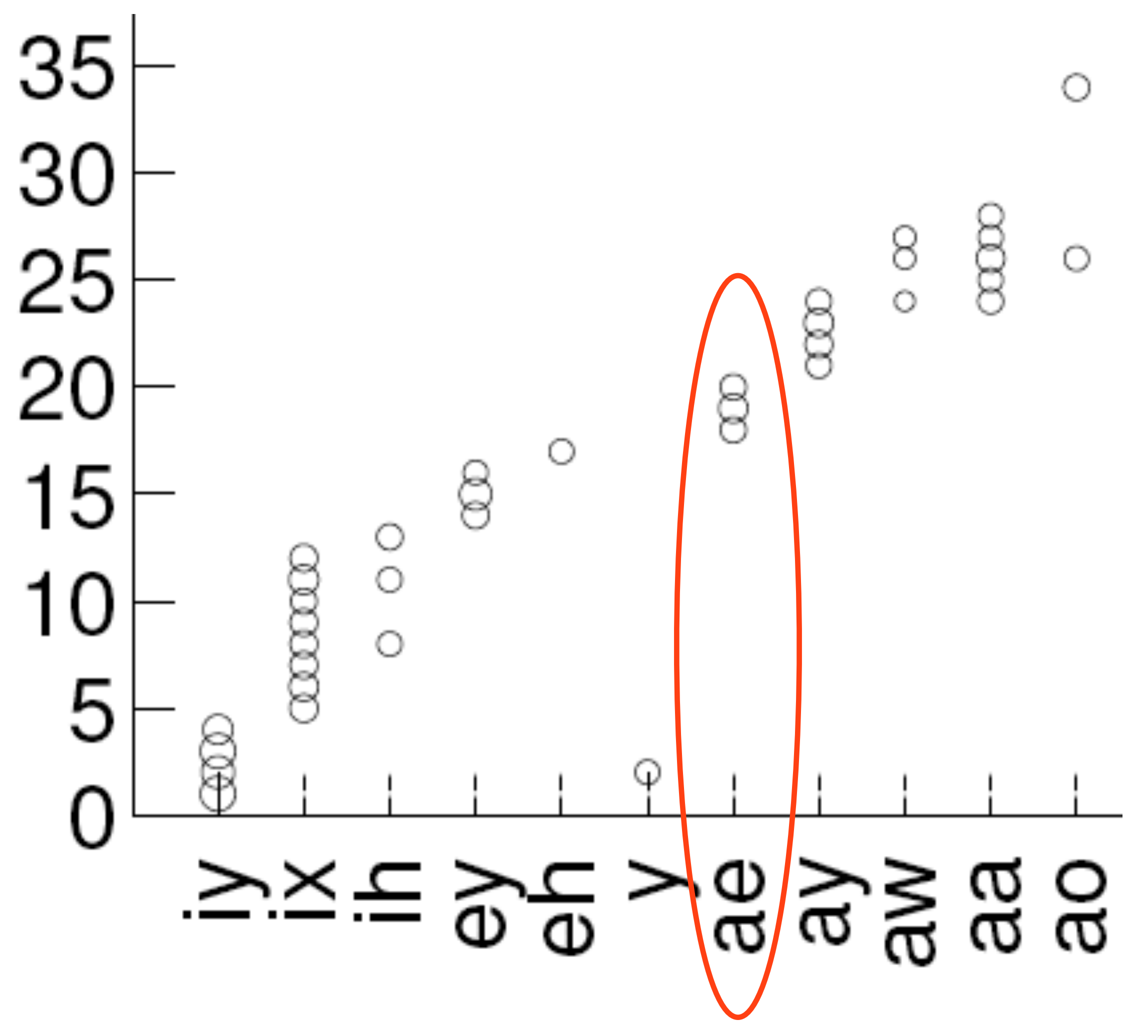
Discovered Phone Units -- 3696 utterances

- 123 phone units discovered from 3696 TIMIT utterances
  - A fine correlation between discovered phones and English phones



Discovered Phone Units -- 3696 utterances

- 123 phone units discovered from 3696 TIMIT utterances
  - A fine correlation between discovered phones and English phones



Context-dependent:
/ae/ + /m/, /n/
/ae/ + stops

Phone Segmentation

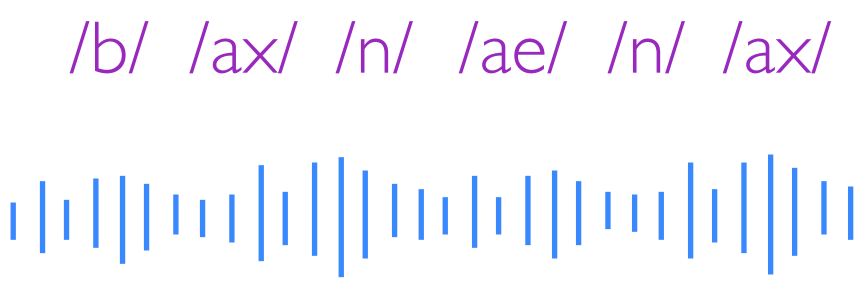
- TIMIT training portion

	Recall	Precision	F-score
Dusan et al. (unsupervised)	75.2	66.8	70.8
Qiao et al. (semi-supervised)	77.5	76.3	76.9
Our model (unsupervised)	76.2	76.4	76.3

Part I: Discovering Phonetic Units from Speech

Discovering phonetic inventory

[Lee and Glass, ACL 2012]



- DP mixture models with HMMs
  - Discovered phonetic units are highly correlated with standard phones
  - Achieves phone segmentation performance similar to the semi-supervised baseline

Discovering hierarchical linguistic structures

[Lee, O'donnell, and Glass, TACL 2015

Part II of the talk

Part II: Discovering Hierarchical Linguistic Structures

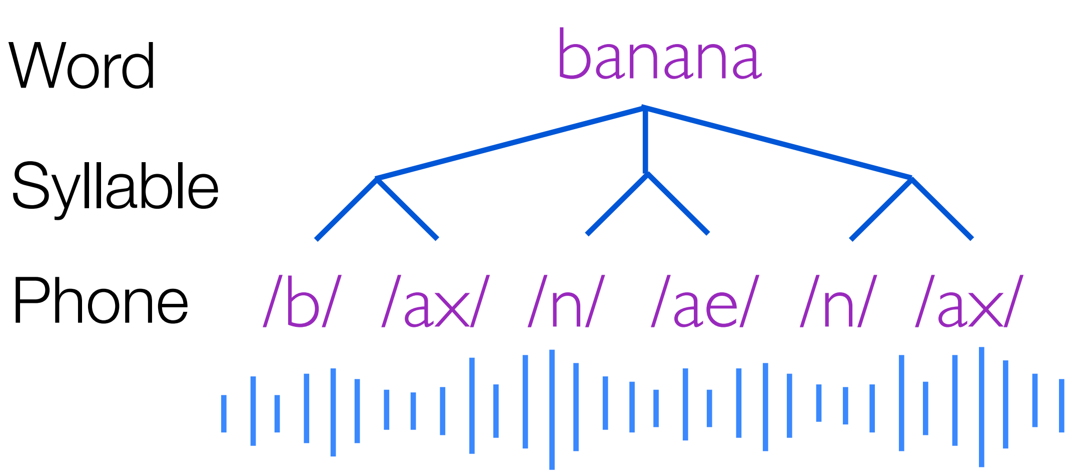
Discovering phonetic inventory

[Lee and Glass, ACL 2012]

- DP mixture models with HMMs
  - Discovered phonetic units are highly correlated with standard phones
  - Achieves phone segmentation performance similar to the semi-supervised baseline

Discovering hierarchical linguistic structures

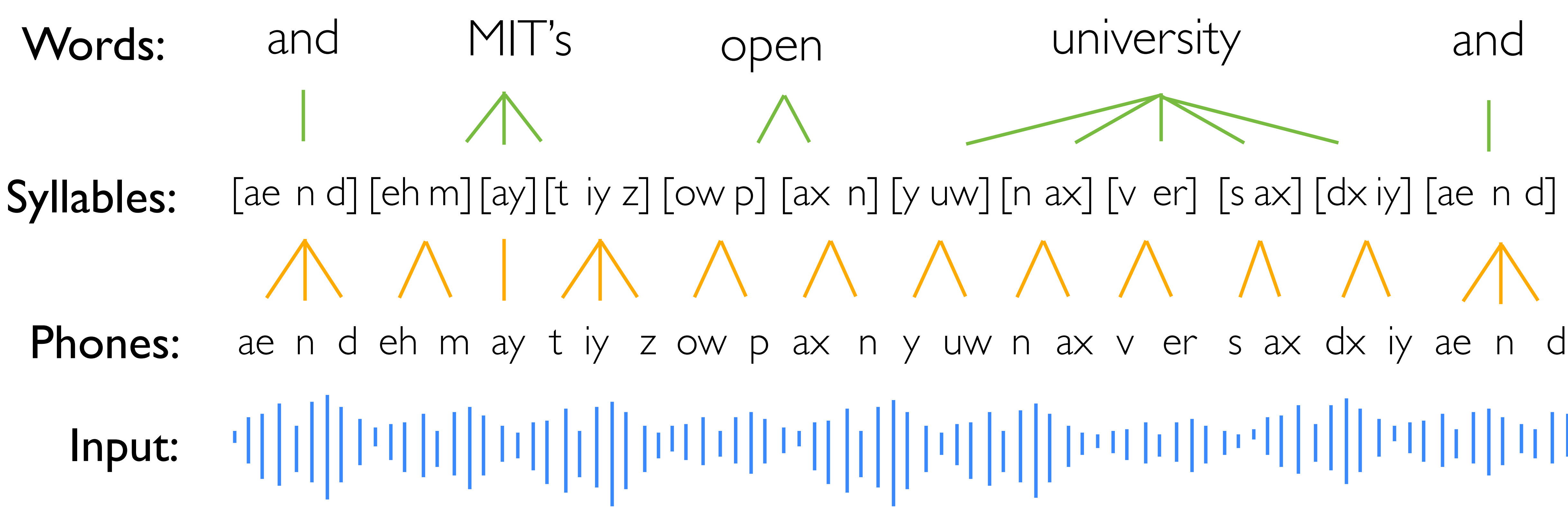
[Lee, O'Donnell, and Glass, TACL 2015]



Part II of the talk

Problem Overview

- Discover hierarchical linguistic structures from speech
  - Phone-like, syllable-like and word-like units



Related Work

- **Spoken term discovery**
  - Unsupervised patter discovery in speech [*Park and Glass, IEEE Trans., 2008*]
  - Unsupervised speech processing with applications to query-by-example spoken term detection [*Zhang, Ph.D. Thesis 2013*]
  - Towards spoken term discovery at scale with zero resources [*Jansen et al., INTERSPEECH 2010*]
- **Word segmentation on phone transcripts of spoken utterances**
  - A Bayesian framework for word segmentation: Exploring the effects of context [*Goldwater et al., Cognition 2009*]
  - Bayesian unsupervised word segmentation with nested Pitman-Yor language modeling [*Mochihashi et al., ACL 2009*]
  - Using adaptor grammars to identify synergies in the unsupervised acquisition of linguistic structure [*Johnson, ACL-HLT 2008*]

Spoken Term Discovery

- Discover speech segments that correspond to words

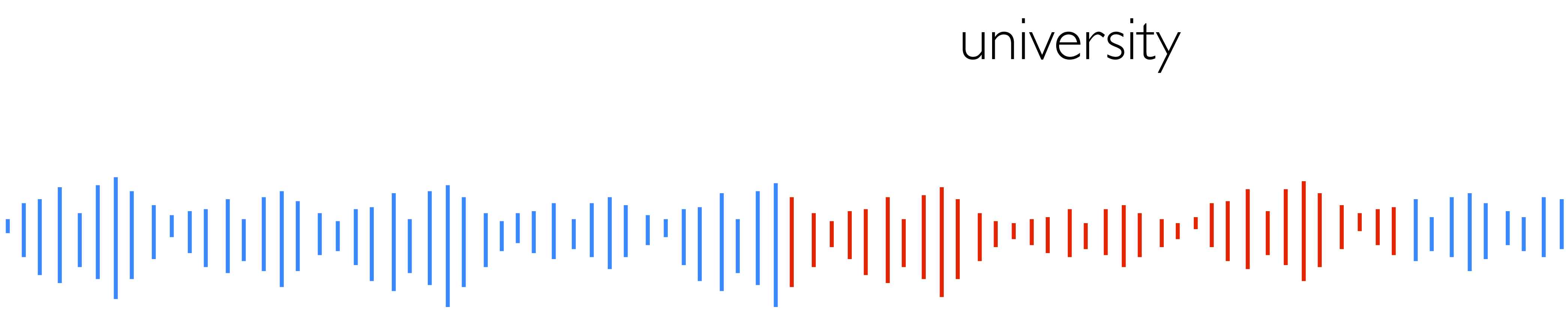
Input: 

[Park and Glass, IEEE Trans., 2008] [Zhang, Ph.D. Thesis 2013] [Jansen et al., INTERSPEECH 2010]

Spoken Term Discovery

- Discover speech segments that correspond to words

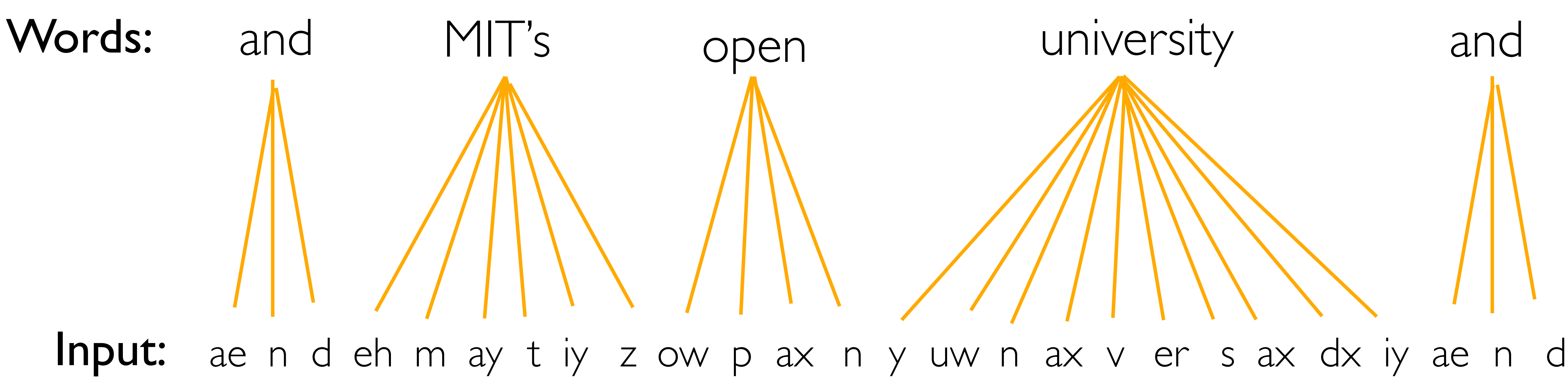
Input:



[Park and Glass, IEEE Trans., 2008] [Zhang, Ph.D. Thesis 2013] [Jansen et al., INTERSPEECH 2010]

Word Segmentation on Phone Transcripts

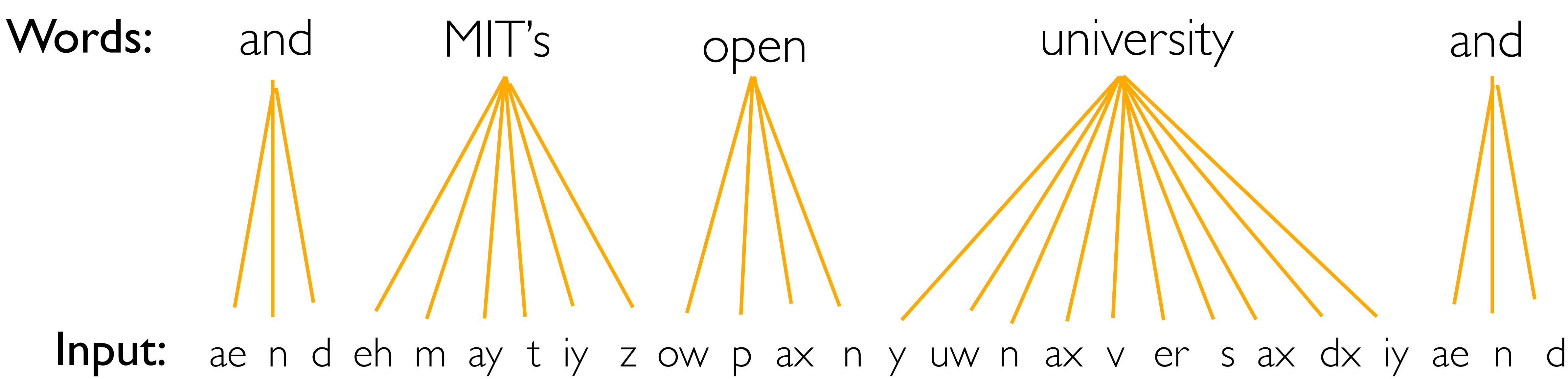
- Model words as sequences of phones



[*Goldwater et al., ACL 2006*] [*Brent and Cartwrite, Cognition 1996*] [*Mochihashi et al., ACL 2009*]

Word Segmentation on Phone Transcripts

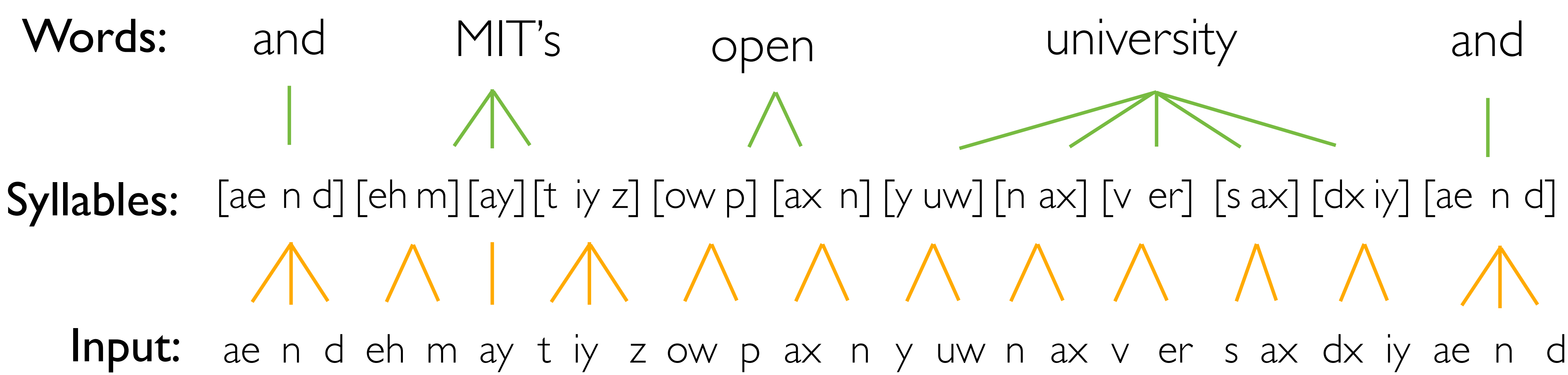
- Model words as sequences of phones
- Modeling more levels of structures improves word segmentation
  - Word → Syllables   Syllable → Phones



[Johnson, ACL-HLT 2008] [Johnson et al., NAACL-HLT, 2009]

Word Segmentation on Phone Transcripts

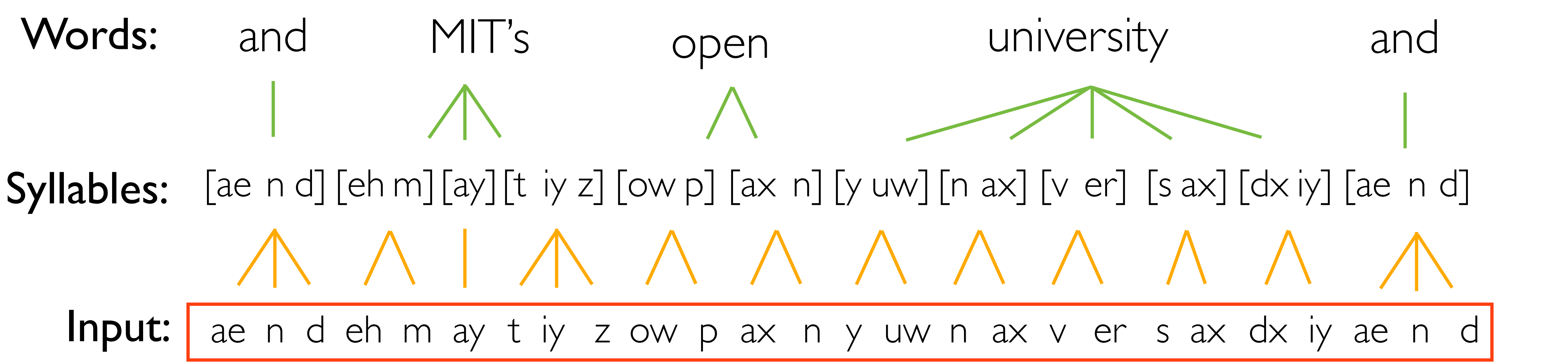
- Model words as sequences of phones
- Modeling more levels of structures improves word segmentation
  - Word → Syllables   Syllable → Phones



[Johnson, ACL-HLT 2008] [Johnson et al., NAACL-HLT, 2009]

Word Segmentation on Phone Transcripts

- Model words as sequences of phones
- Modeling more levels of structures improves word segmentation
  - Word → Syllables   Syllable → Phones
- Adaptor grammars is an effective tool for learning rich structures



only learns from symbolic input

[Johnson, ACL-HLT 2008] [Johnson et al., NAACL-HLT, 2009]

Model Overview

- **Integrate adaptor grammars and the phone discovery model**
  - To discover rich linguistic structures from speech
- **Three components in the model**

Adaptor grammars

Noisy-channel model

Phone discovery model

Model Overview

- Integrate adaptor grammars and the phone discovery model
  - To discover rich linguistic structures from speech
- Three components in the model

Adaptor grammars

Noisy-channel model

Phone discovery model

A nonparametric Bayesian extension of probabilistic context-free grammars (PCFGs)

PCFG Example

An example PCFG for generating phone sequences

PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
		...	

PCFG Generative Process

Sen

PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/

...

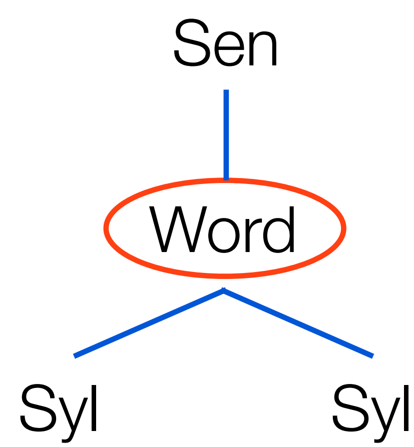
PCFG Generative Process

PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/

...

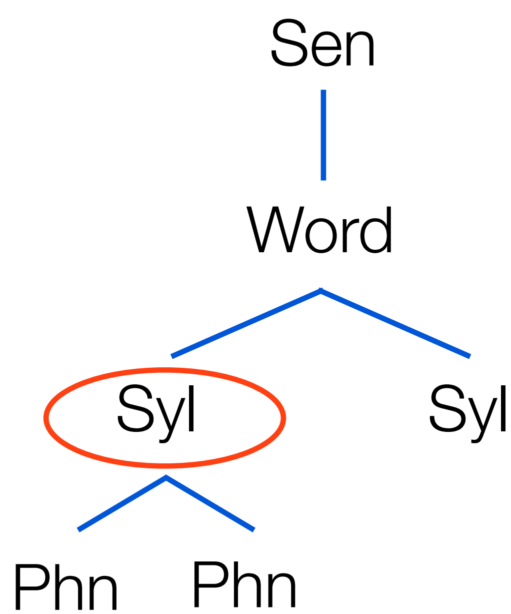
PCFG Generative Process



PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
	...		

PCFG Generative Process

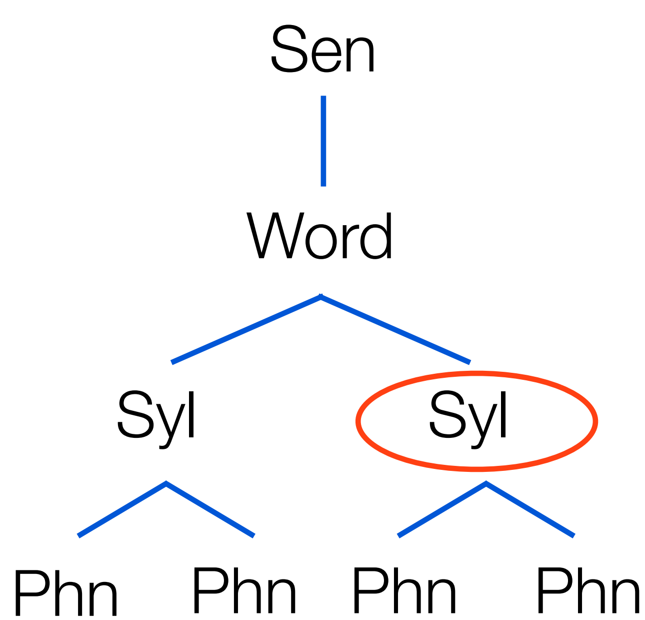


PCFG

0.5	Sen	→	Word Word
0.5	Sen	→	Word
0.7	Word	→	Syl Syl
0.3	Word	→	Syl
1.0	Syl	→	Phn Phn
0.1	Phn	→	/ax/
0.05	Phn	→	/n/
0.1	Phn	→	/ow/
0.1	Phn	→	/p/

...

PCFG Generative Process

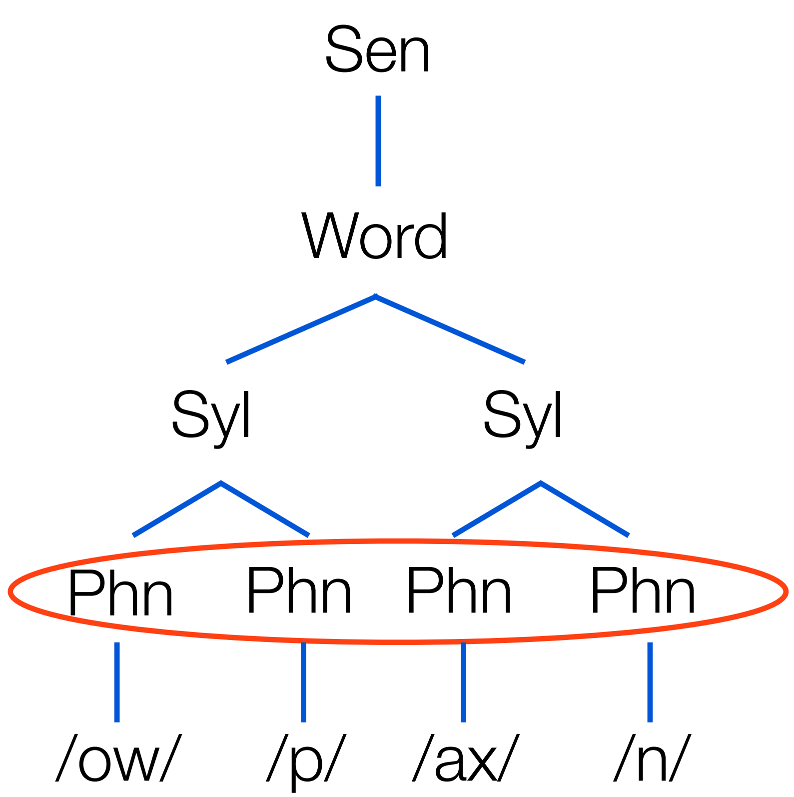


PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/

...

PCFG Generative Process

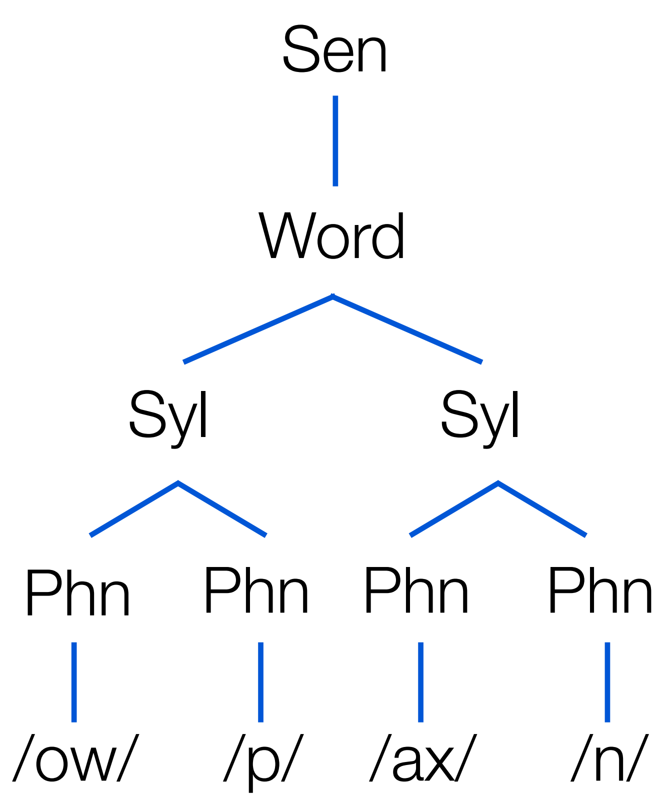


PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/

...

PCFG Generative Process



PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/

...

Adaptor Grammars

- A PCFG +

PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
		...	

Adaptor Grammars

- A PCFG + cached subtrees for adapted nonterminals

PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/

...

Cached subtrees

Syl:

Adaptor Grammars

- A PCFG + cached subtrees for adapted nonterminals

Key idea:
Adaptor grammars memorize reusable structures

PCFG

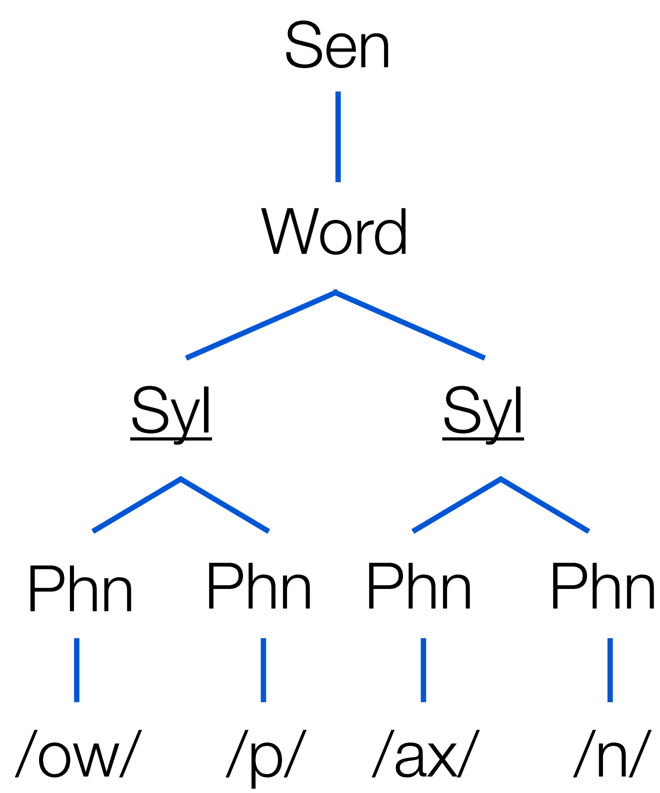
0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	<u>Syl</u> <u>Syl</u>
0.3	Word	$\longrightarrow$	<u>Syl</u>
1.0	<u>Syl</u>	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
	...		

Cached subtrees

<u>Syl</u>:

Adaptor Grammars Generative Process

- Assume a current parse



PCFG

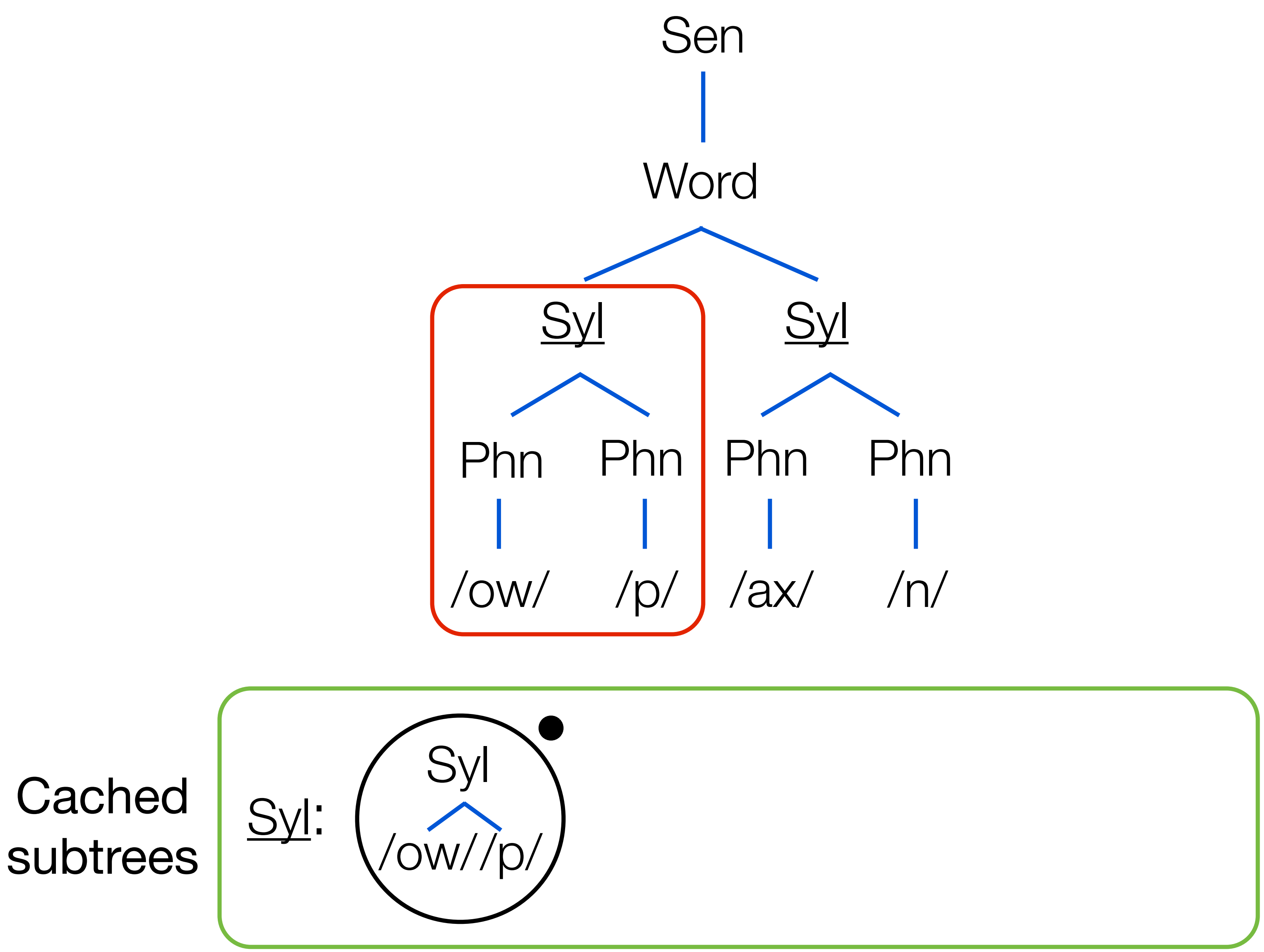
0.5	Sen	⟶	Word Word
0.5	Sen	⟶	Word
0.7	Word	⟶	Syl Syl
0.3	Word	⟶	Syl
1.0	Syl	⟶	Phn Phn
0.1	Phn	⟶	/ax/
0.05	Phn	⟶	/n/
0.1	Phn	⟶	/ow/
0.1	Phn	⟶	/p/
	...		

Cached subtrees

Syl:

Adaptor Grammars Generative Process

- Cache subtrees for adapted nonterminals



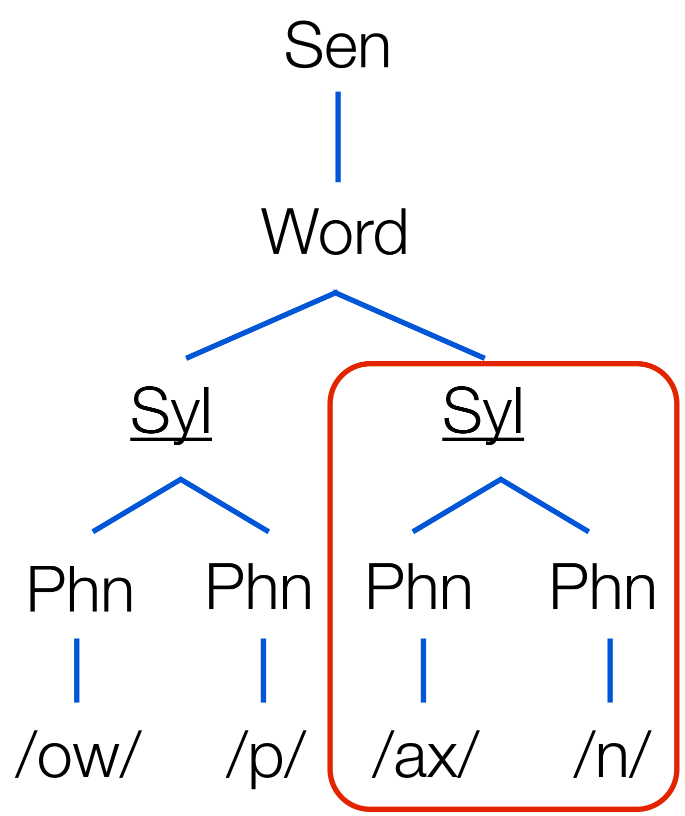
PCFG

0.5	Sen	⟶	Word Word
0.5	Sen	⟶	Word
0.7	Word	⟶	Syl Syl
0.3	Word	⟶	Syl
1.0	Syl	⟶	Phn Phn
0.1	Phn	⟶	/ax/
0.05	Phn	⟶	/n/
0.1	Phn	⟶	/ow/
0.1	Phn	⟶	/p/

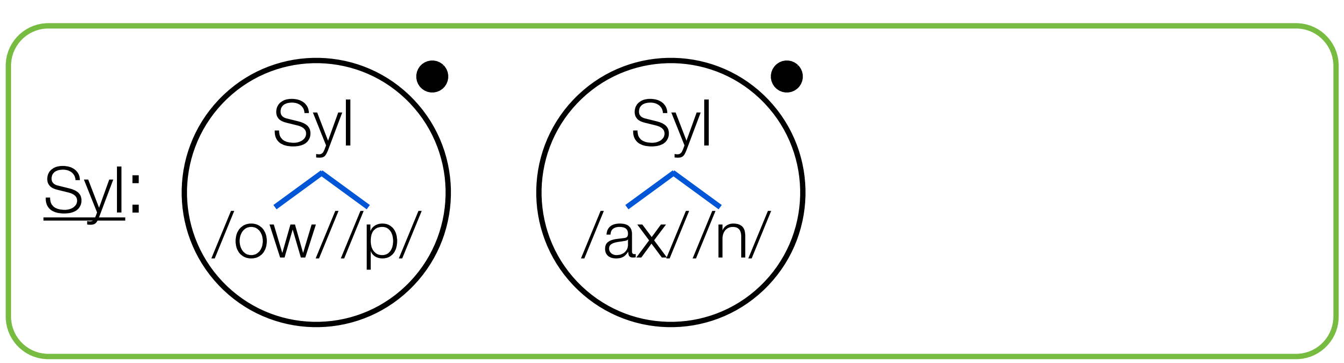
...

Adaptor Grammars Generative Process

- Cache subtrees for adapted nonterminals



Cached subtrees



PCFG

0.5	Sen	⟶	Word Word
0.5	Sen	⟶	Word
0.7	Word	⟶	Syl Syl
0.3	Word	⟶	Syl
1.0	Syl	⟶	Phn Phn
0.1	Phn	⟶	/ax/
0.05	Phn	⟶	/n/
0.1	Phn	⟶	/ow/
0.1	Phn	⟶	/p/

...

Adaptor Grammars Generative Process

- Generate a new parse

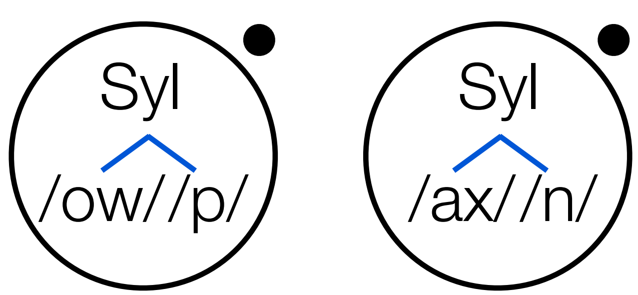
Sen

PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
	...		

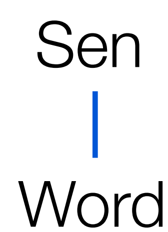
Cached subtrees

Syl:



Adaptor Grammars Generative Process

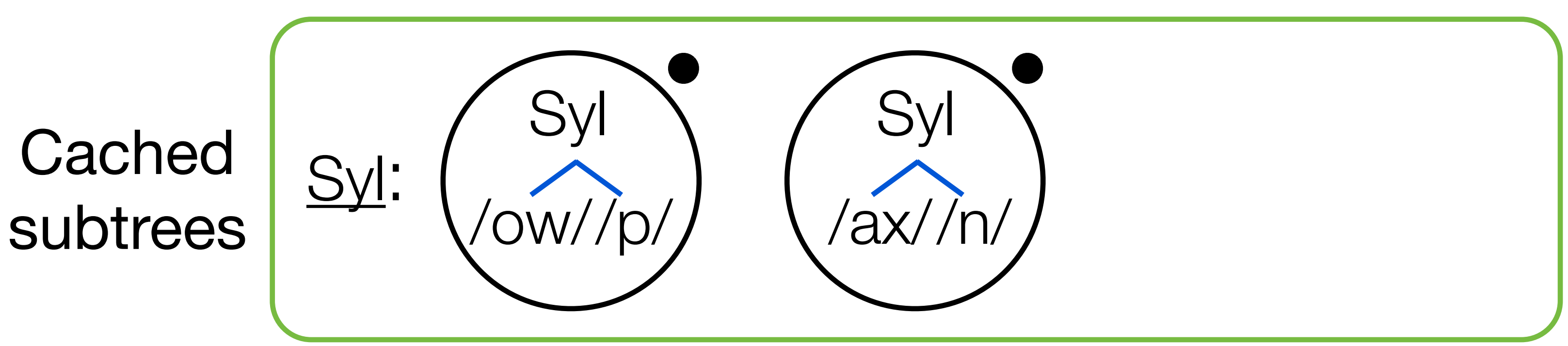
- Expand regular nonterminals using PCFG rules



PCFG

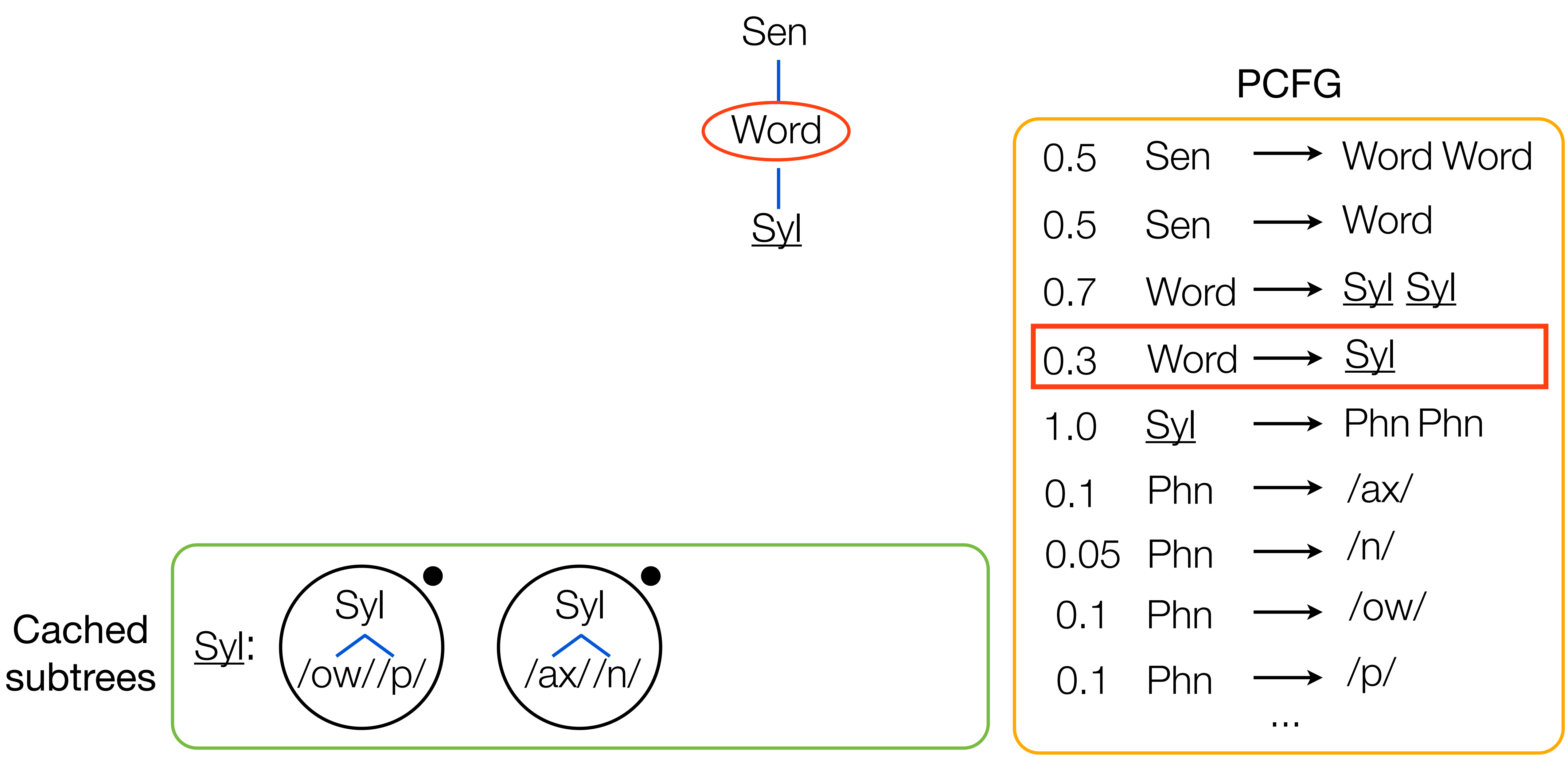
0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/

...



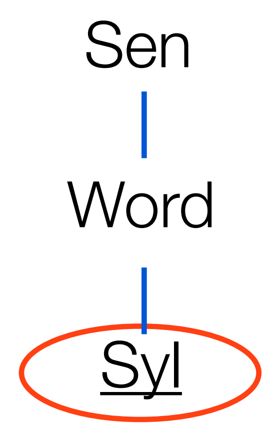
Adaptor Grammars Generative Process

- Expand regular nonterminals using PCFG rules



Adaptor Grammars Generative Process

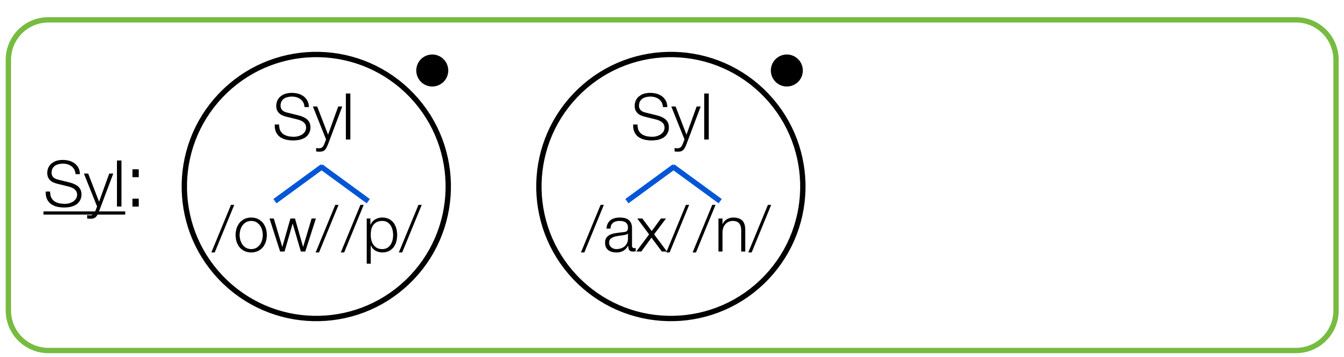
- Expand adapted nonterminals
  - Reuse a cached subtree



PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
		...	

Cached subtrees

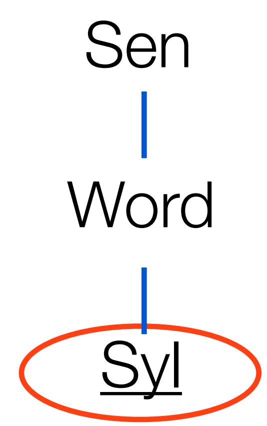


Adaptor Grammars Generative Process

- Expand adapted nonterminals
  - Reuse a cached subtree

$$p\left(\text{Syl} \rightarrow \text{/ow/ /p/}\right) = \frac{1}{2 + \alpha_{\text{syl}}}$$

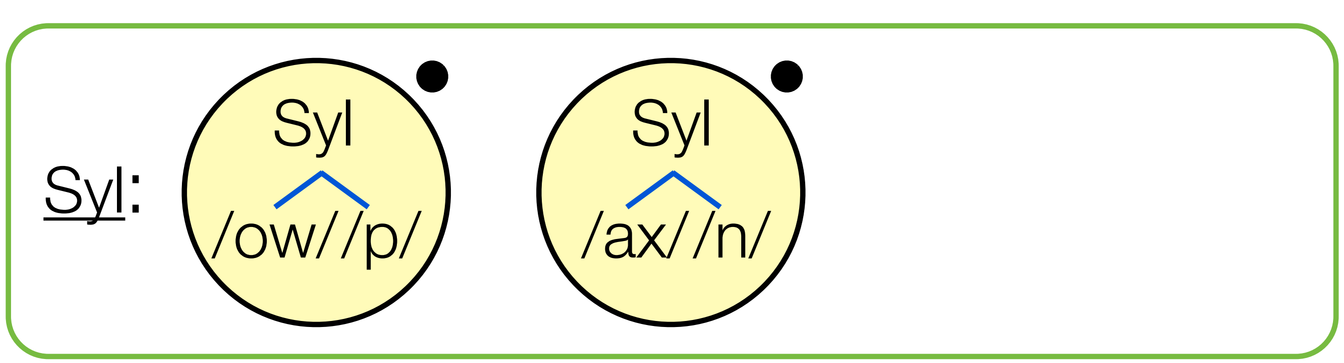
$$p\left(\text{Syl} \rightarrow \text{/ax/ /n/}\right) = \frac{1}{2 + \alpha_{\text{syl}}}$$



PCFG

0.5	Sen	⟶	Word Word
0.5	Sen	⟶	Word
0.7	Word	⟶	Syl Syl
0.3	Word	⟶	Syl
1.0	Syl	⟶	Phn Phn
0.1	Phn	⟶	/ax/
0.05	Phn	⟶	/n/
0.1	Phn	⟶	/ow/
0.1	Phn	⟶	/p/
	...		

Cached subtrees



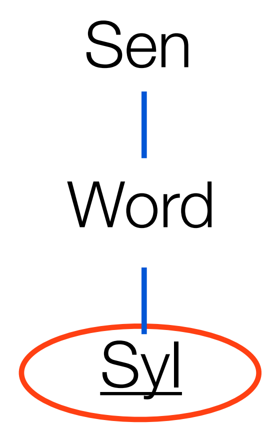
Adaptor Grammars Generative Process

- Expand adapted nonterminals
  - Reuse a cached subtree

$$p\left(\begin{array}{c}\text{Syl}\\ \text{/ow/ /p/}\end{array}\right) = \frac{1}{2+\alpha_{\text{syl}}}$$

$$p\left(\begin{array}{c}\text{Syl}\\ \text{/ax/ /n/}\end{array}\right) = \frac{1}{2+\alpha_{\text{syl}}}$$

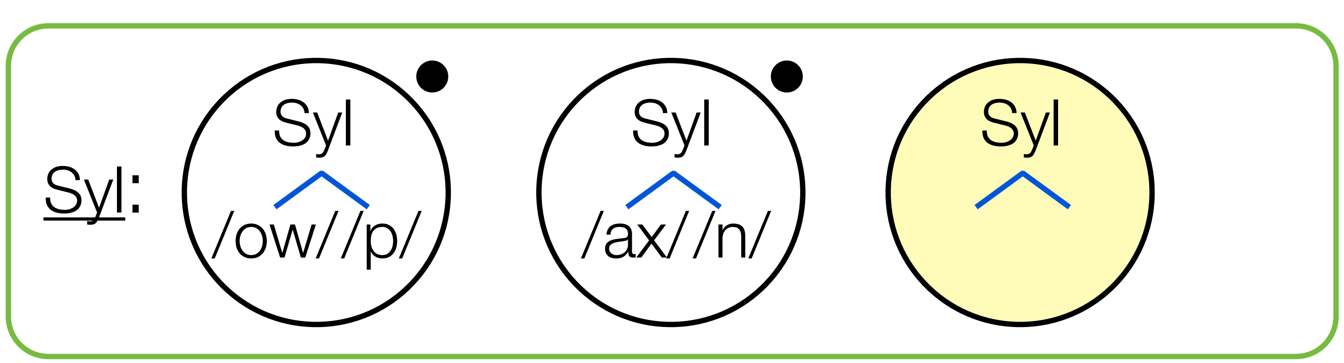
  - Create and store a new subtree



PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
		...	

Cached subtrees



Adaptor Grammars Generative Process

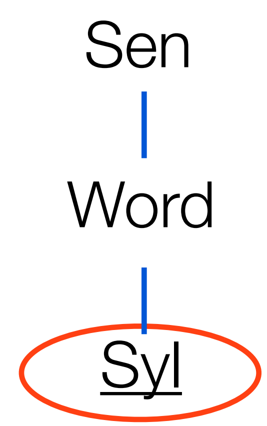
- Expand adapted nonterminals
  - Reuse a cached subtree

$$p\left(\begin{array}{c}\text{Syl}\\ \wedge\\ /\text{ow}/\ /\text{p}/\end{array}\right) = \frac{1}{2+\alpha_{\text{syl}}}$$

$$p\left(\begin{array}{c}\text{Syl}\\ \wedge\\ /\text{ax}/\ /\text{n}/\end{array}\right) = \frac{1}{2+\alpha_{\text{syl}}}$$

  - Create and store a new subtree

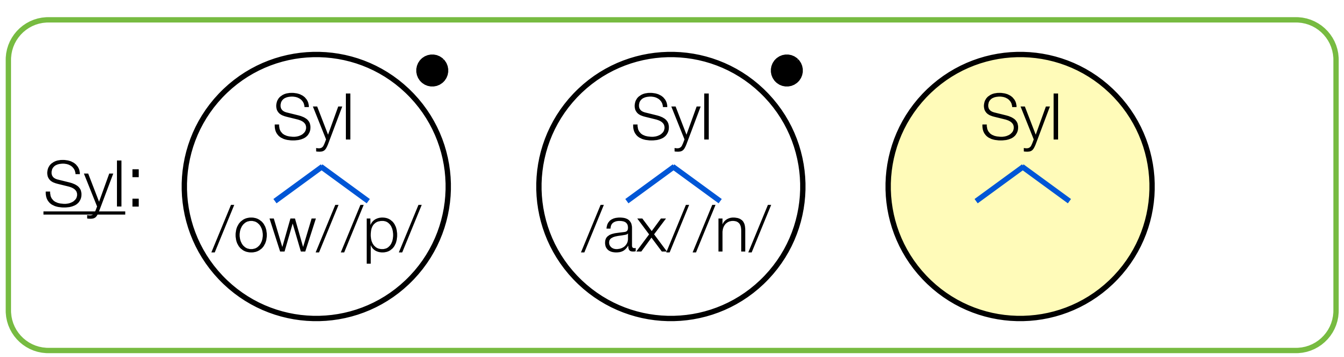
$$p\left(\begin{array}{c}\text{Syl}\\ \wedge\end{array}\right) = \frac{\alpha_{\text{syl}}}{2+\alpha_{\text{syl}}}$$



PCFG

0.5	Sen	⟶	Word Word
0.5	Sen	⟶	Word
0.7	Word	⟶	Syl Syl
0.3	Word	⟶	Syl
1.0	Syl	⟶	Phn Phn
0.1	Phn	⟶	/ax/
0.05	Phn	⟶	/n/
0.1	Phn	⟶	/ow/
0.1	Phn	⟶	/p/
	...		

Cached subtrees

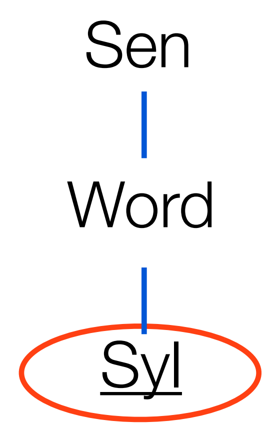


Adaptor Grammars Generative Process

- Expand adapted nonterminals
  - Reuse a cached subtree

$$p\left(\begin{array}{c}\text{Syl}\\ \text{/ow/ /p/}\end{array}\right) = \frac{1}{2+\alpha_{\text{syl}}}$$

$$p\left(\begin{array}{c}\text{Syl}\\ \text{/ax/ /n/}\end{array}\right) = \frac{1}{2+\alpha_{\text{syl}}}$$



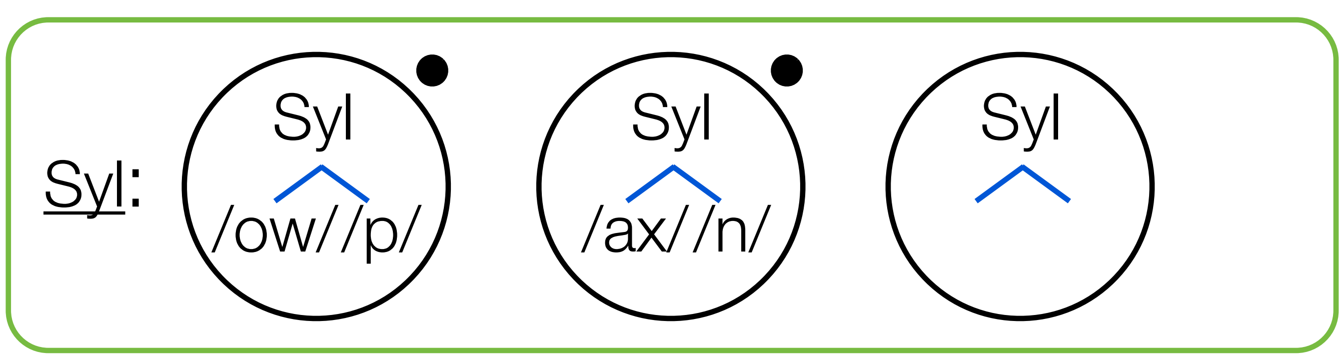
  - Create and store a new subtree

$$p\left(\text{Syl}\right) = \frac{\alpha_{\text{syl}}}{2+\alpha_{\text{syl}}}$$

PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	<u>Syl</u> <u>Syl</u>
0.3	Word	$\longrightarrow$	<u>Syl</u>
1.0	<u>Syl</u>	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
	...		

Cached subtrees



Adaptor Grammars Generative Process

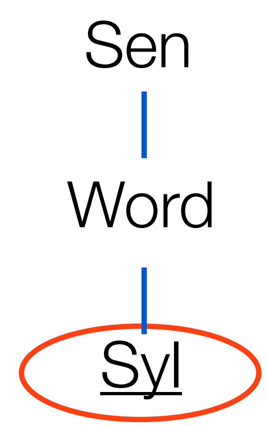
- Expand adapted nonterminals
  - Reuse a cached subtree

$$p\left(\begin{array}{c}\text{Syl}\\ /ow/\ /p/\end{array}\right) = \frac{1}{2+\alpha_{syl}}$$

$$p\left(\begin{array}{c}\text{Syl}\\ /ax/\ /n/\end{array}\right) = \frac{1}{2+\alpha_{syl}}$$

  - Create and store a new subtree

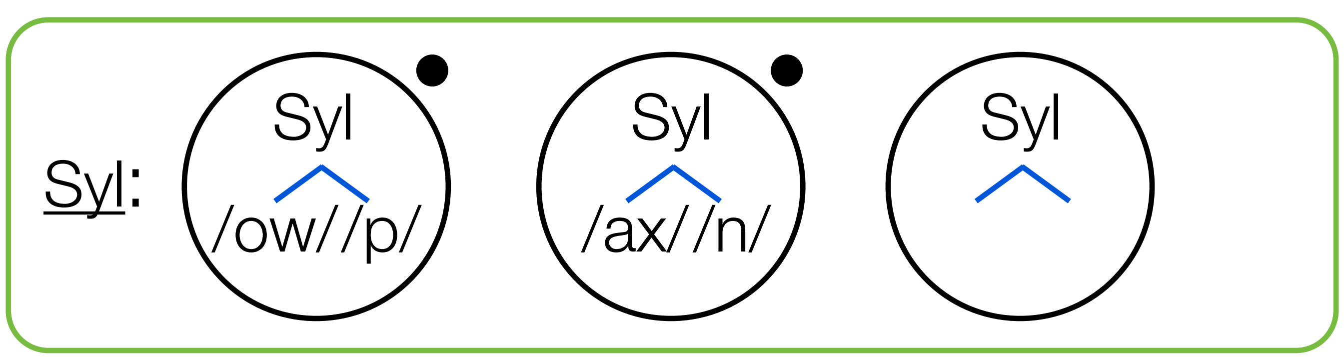
$$p\left(\text{Syl}\right) = \frac{\alpha_{syl}}{2+\alpha_{syl}}$$



PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
	...		

Cached subtrees



Adaptor Grammars Generative Process

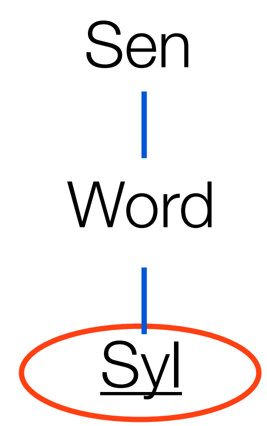
- Expand adapted nonterminals
  - Reuse a cached subtree

$$p\left(\underset{/ow/\ /p/}{\text{Syl}}\right) = \frac{1}{2 + \alpha_{\text{syl}}}$$

$$\boxed{p\left(\underset{/ax/\ /n/}{\text{Syl}}\right) = \frac{1}{2 + \alpha_{\text{syl}}}}$$

  - Create and store a new subtree

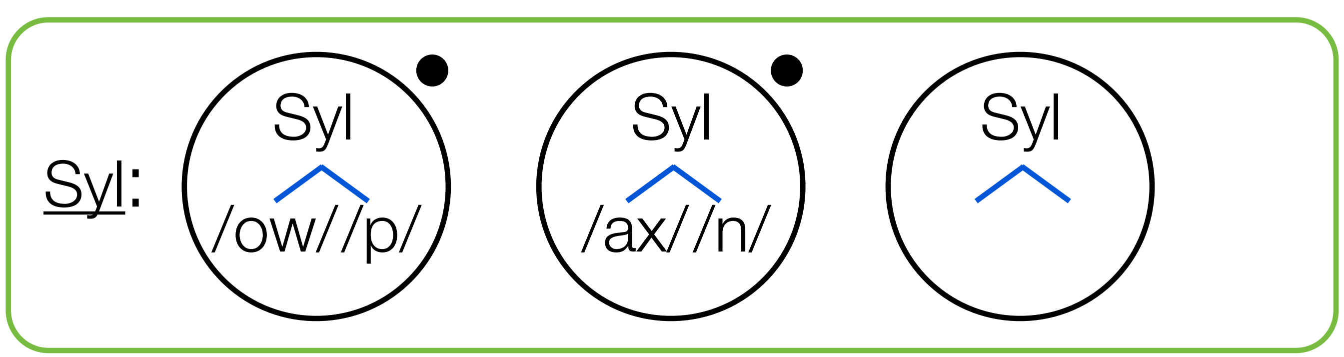
$$p\left(\text{Syl}\right) = \frac{\alpha_{\text{syl}}}{2 + \alpha_{\text{syl}}}$$



PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
	...		

Cached subtrees



Adaptor Grammars Generative Process

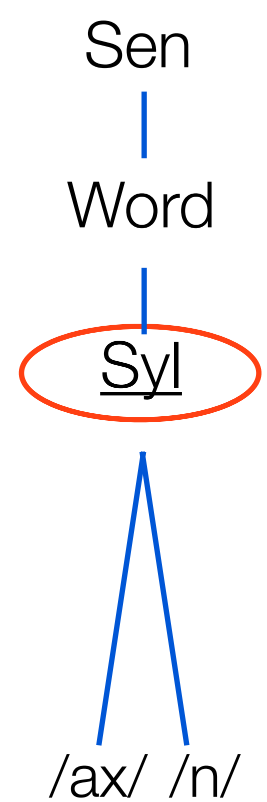
- Expand adapted nonterminals
  - Reuse a cached subtree

$$p\left(\begin{array}{c}\text{Syl}\\ /\text{ow}/\ /\text{p}/\end{array}\right) = \frac{1}{2+\alpha_{\text{syl}}}$$

$$\boxed{p\left(\begin{array}{c}\text{Syl}\\ /\text{ax}/\ /\text{n}/\end{array}\right) = \frac{1}{2+\alpha_{\text{syl}}}}$$

  - Create and store a new subtree

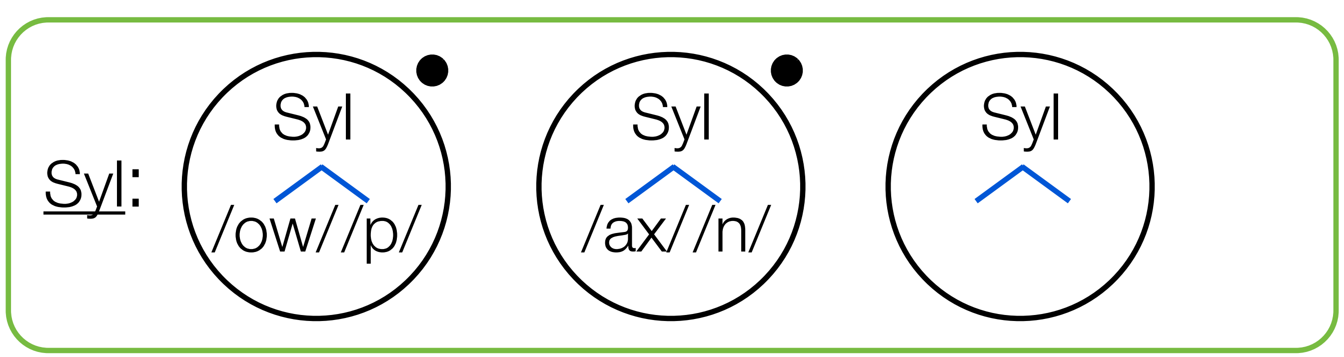
$$p\left(\text{Syl}\right) = \frac{\alpha_{\text{syl}}}{2+\alpha_{\text{syl}}}$$



PCFG

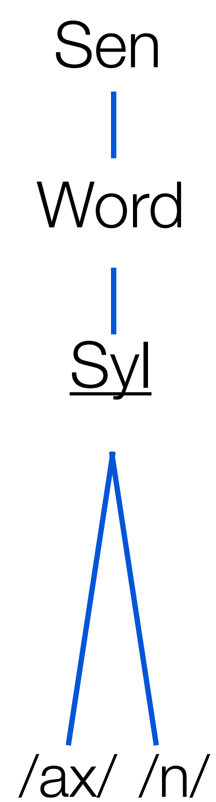
0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
1.0	Syl	$\longrightarrow$	Phn Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
0.1	Phn	$\longrightarrow$	/ow/
0.1	Phn	$\longrightarrow$	/p/
	...		

Cached subtrees



Adaptor Grammars Generative Process

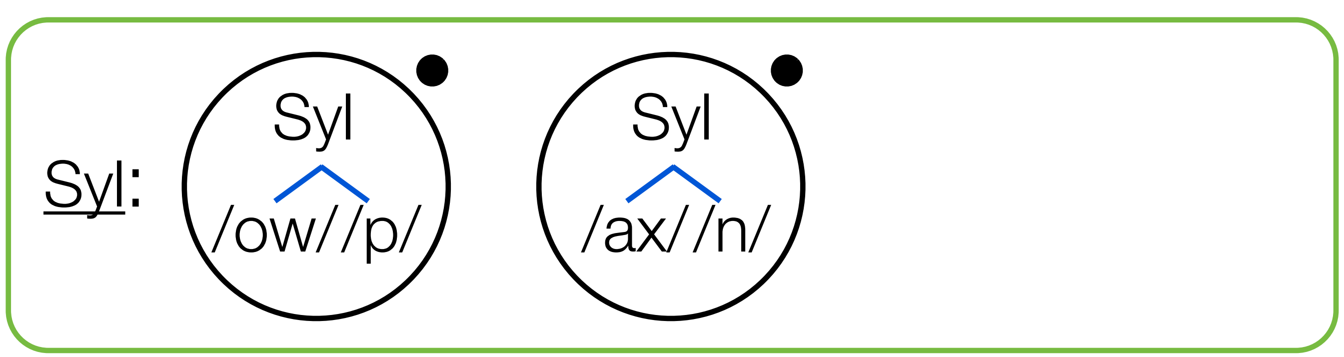
- Expand adapted nonterminals



PCFG

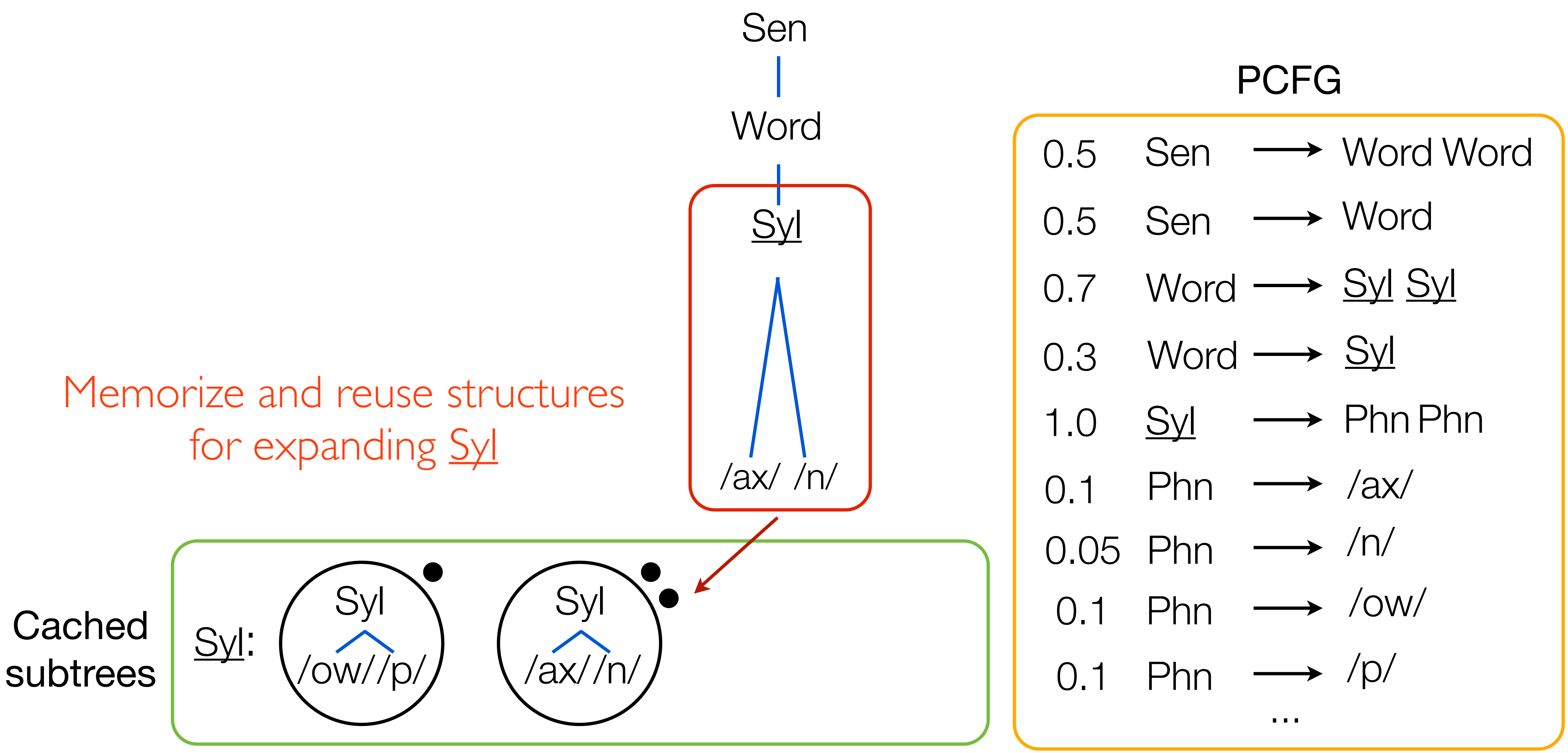
0.5	Sen	⟶	Word Word
0.5	Sen	⟶	Word
0.7	Word	⟶	Syl Syl
0.3	Word	⟶	Syl
1.0	Syl	⟶	Phn Phn
0.1	Phn	⟶	/ax/
0.05	Phn	⟶	/n/
0.1	Phn	⟶	/ow/
0.1	Phn	⟶	/p/
	...		

Cached subtrees



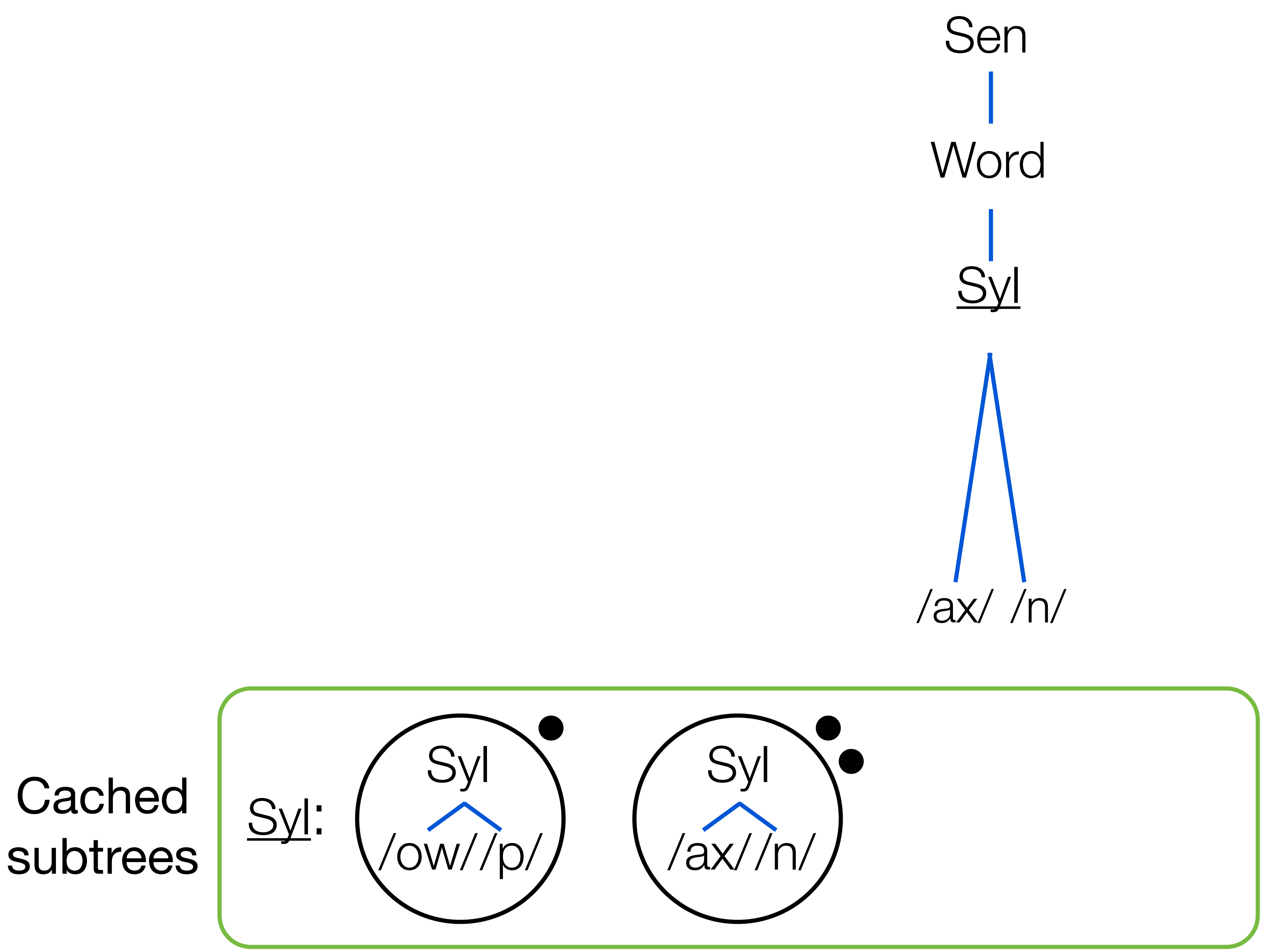
Adaptor Grammars Generative Process

- Cache subtrees for adapted nonterminals



For Our Problem

- The phone inventory is unknown
  - Terminal symbols should be discovered phonetic unit ids



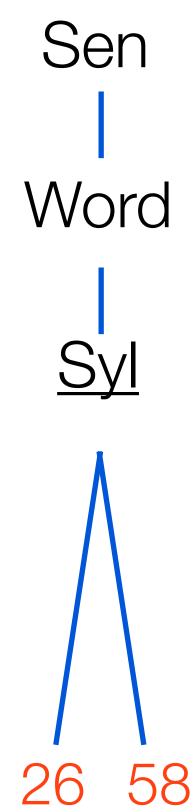
PCFG

0.5	Sen	⟶	Word Word
0.5	Sen	⟶	Word
0.7	Word	⟶	Syl Syl
0.3	Word	⟶	Syl
1.0	Syl	⟶	Phn Phn
0.1	Phn	⟶	/ax/
0.05	Phn	⟶	/n/
0.1	Phn	⟶	/ow/
0.1	Phn	⟶	/p/

...

For Our Problem

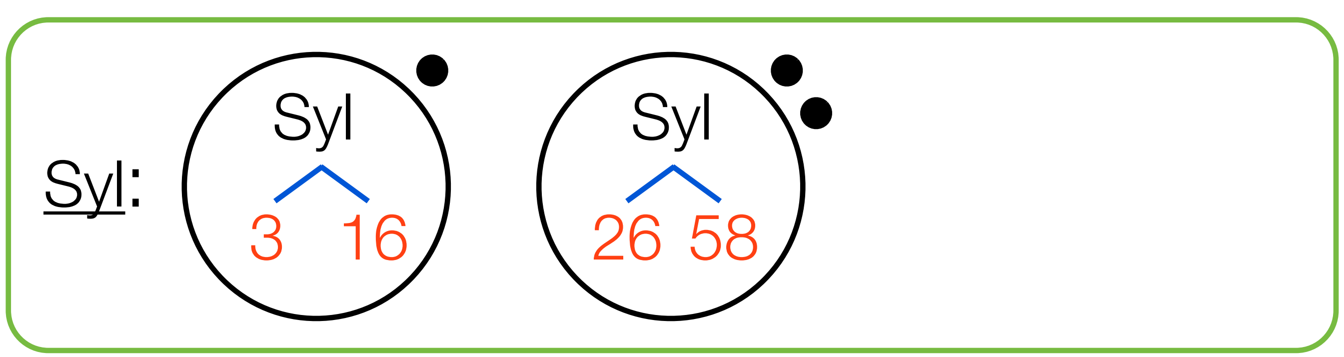
- The phone inventory is unknown
  - Terminal symbols should be discovered phonetic unit ids



PCFG

0.5	Sen	⟶	Word Word
0.5	Sen	⟶	Word
0.7	Word	⟶	<u>Syl</u> <u>Syl</u>
0.3	Word	⟶	<u>Syl</u>
1.0	<u>Syl</u>	⟶	Phn Phn
0.1	Phn	⟶	26
0.05	Phn	⟶	58
0.1	Phn	⟶	3
0.1	Phn	⟶	16
	...		

Cached subtrees



Model Overview

- Integrate adaptor grammars and the phone discovery model
  - To discover rich linguistic structures from speech
- Three components in the model

Adaptor grammars

Noisy-channel model

Phone discovery model

Model Overview

- Integrate adaptor grammars and the phone discovery model
  - To discover rich linguistic structures from speech
- Three components in the model

Adaptor grammars

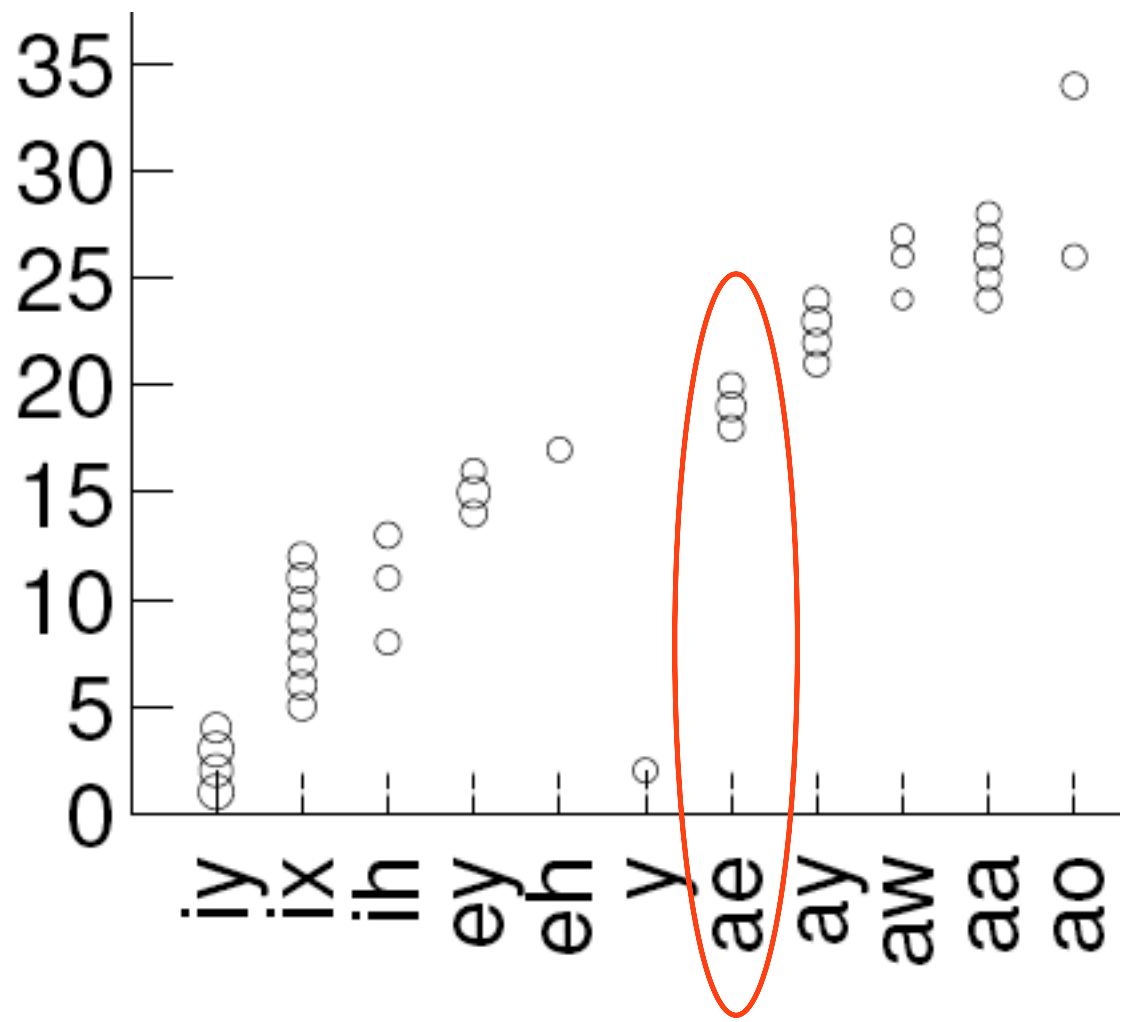
Noisy-channel model

Phone discovery model

First part of the talk

Recall

- A standard phone may map to multiple discovered units
- Various phone sequences for a word type



Recall

- A standard phone may map to multiple discovered units
- Various phone sequences for a word type

/k/	/ae/	/t/		/k/	/ae/	/t/
49	58	32		49	26	32

Recall

- A standard phone may map to multiple discovered units
- Various phone sequences for a word type
- These variations must be collapsed for lexicon learning

/k/ /ae/ /t/   /k/ /ae/ /t/

49 58 32   49 26 32

Collapse the variations by
using a noisy-channel model

Model Overview

- Integrate adaptor grammars and the phone discovery model
  - To discover rich linguistic structures from speech
- Three components in the model

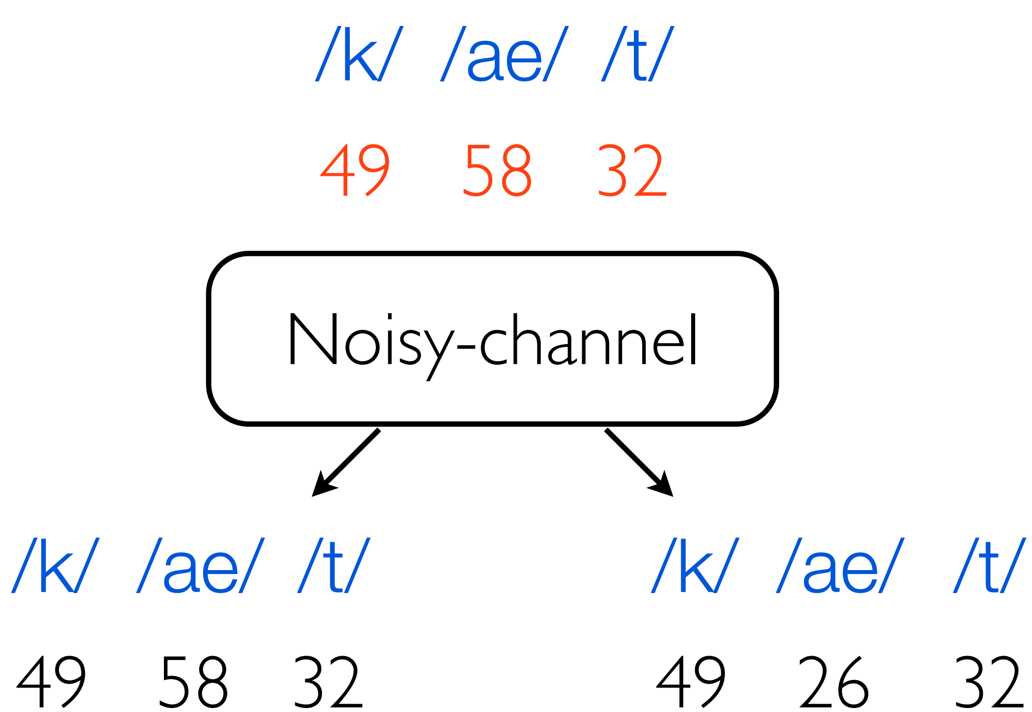
Adaptor grammars

Noisy-channel model — Regularize the phonetic variations

Phone discovery model

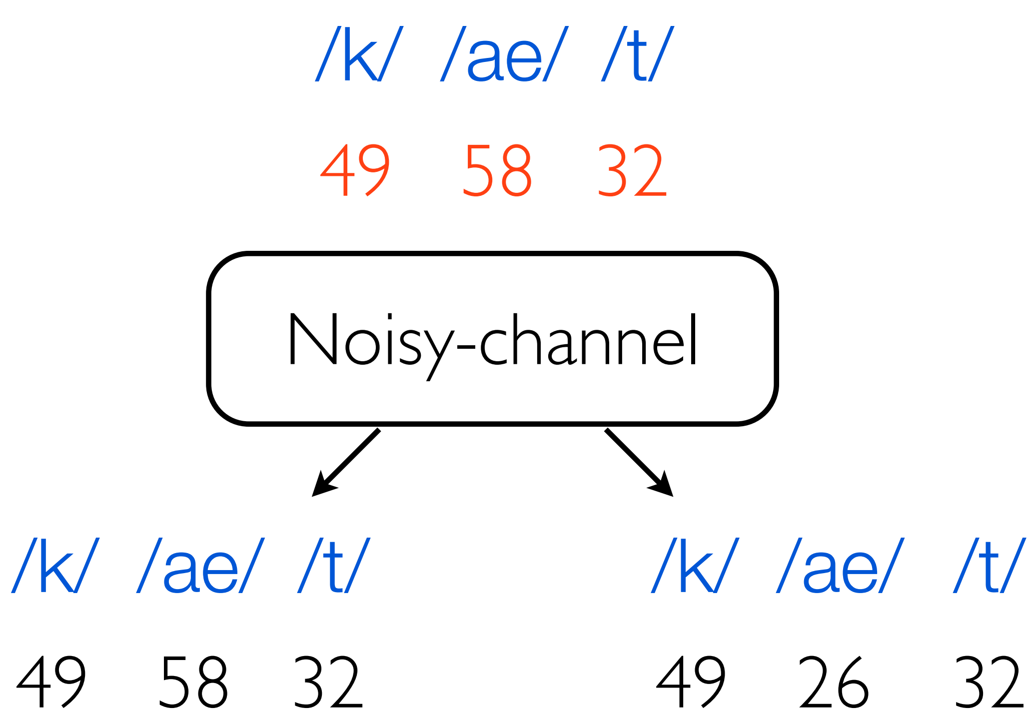
Noisy-channel Model

- Assume the phonetic variations are outcomes of a noisy-channel



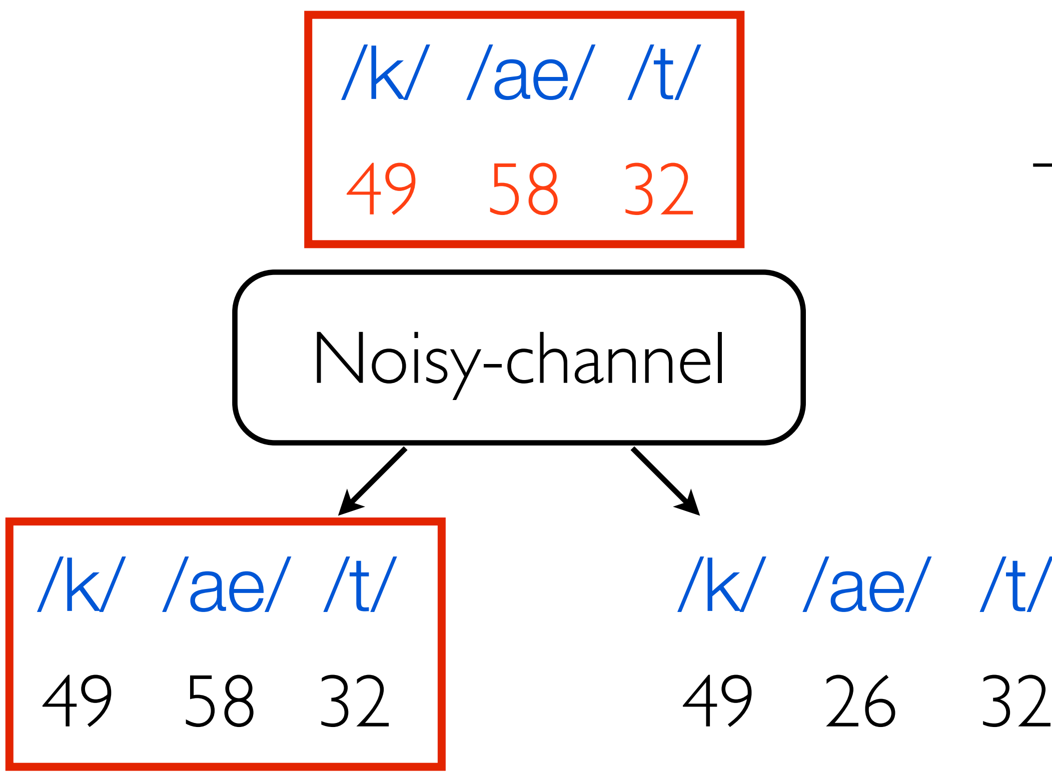
Noisy-channel Model

- Assume the phonetic variations are outcomes of a noisy-channel
- Formulate the noisy-channel model as a set of edit operations
  - Substitution, deletion, insertion, and exact-match



Noisy-channel Model

- Assume the phonetic variations are outcomes of a noisy-channel
- Formulate the noisy-channel model as a set of edit operations
  - Substitution, deletion, insertion, and exact-match



Three exact-match operations

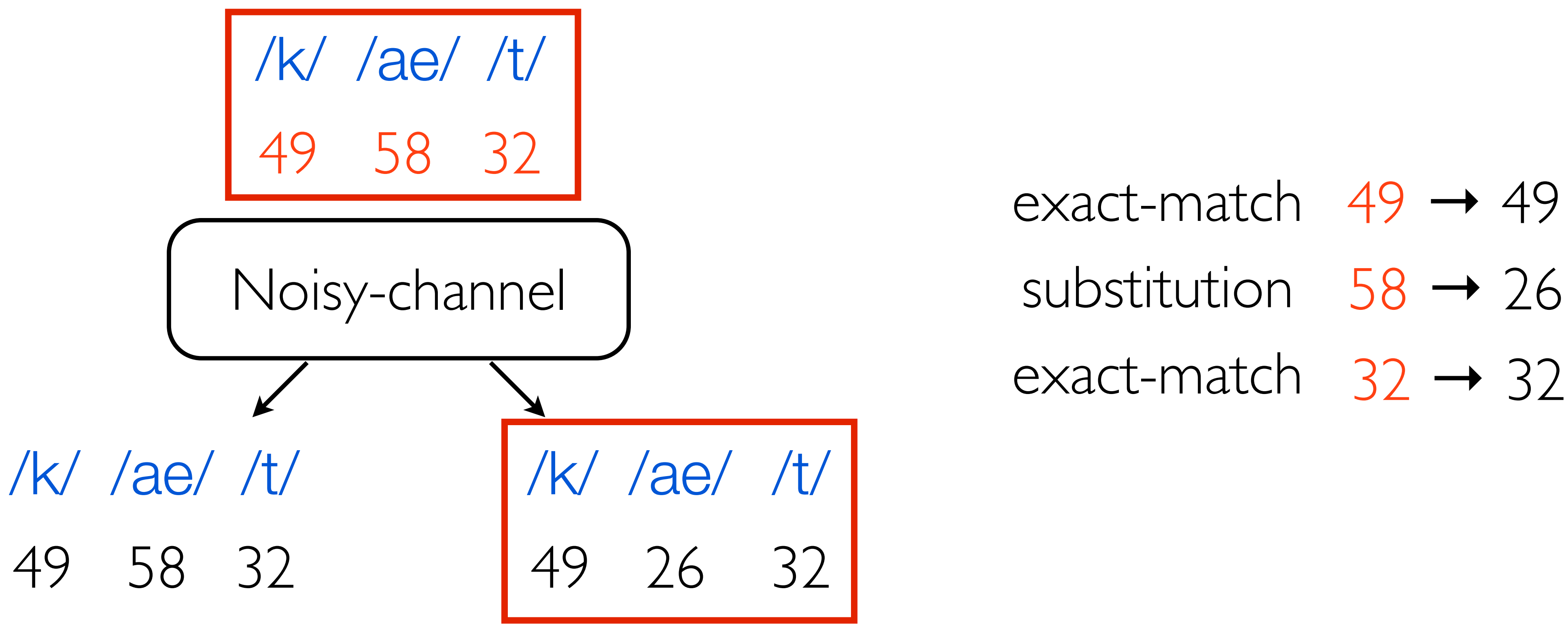
49 → 49

58 → 58

32 → 32

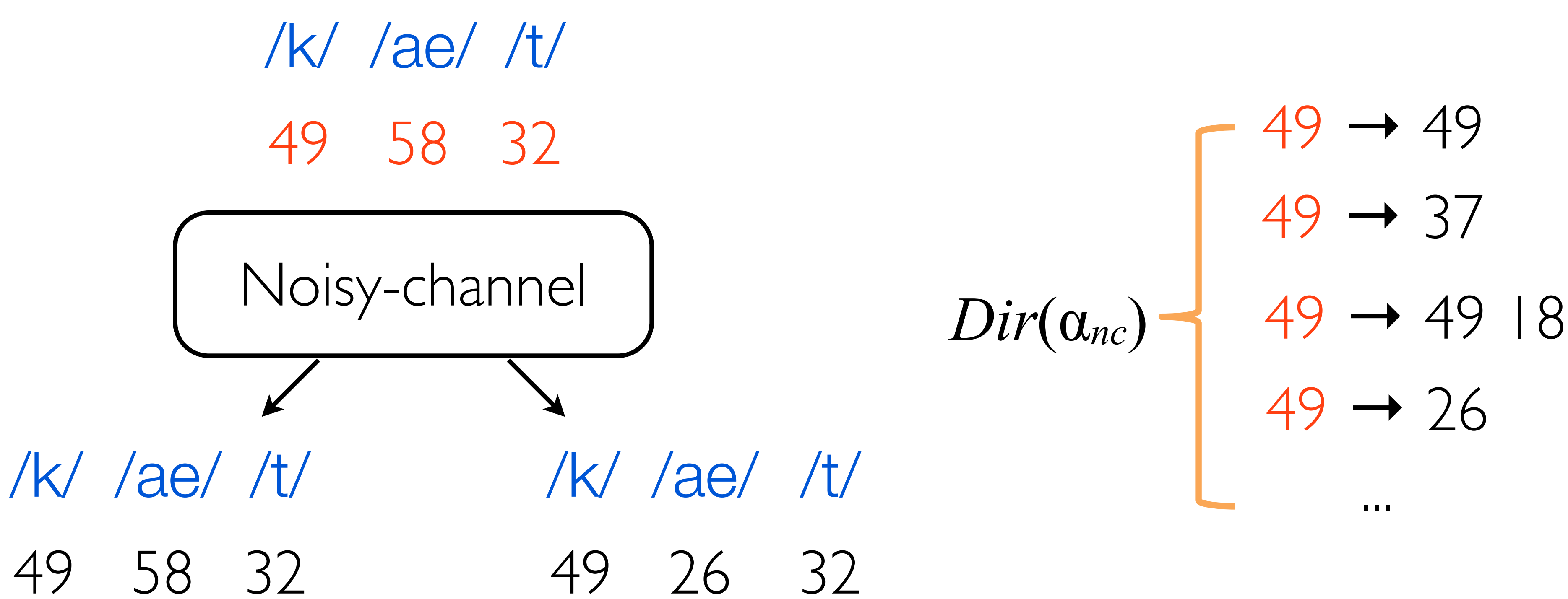
Noisy-channel Model

- Assume the phonetic variations are outcomes of a noisy-channel
- Formulate the noisy-channel model as a set of edit operations
  - Substitution, deletion, insertion, and exact-match



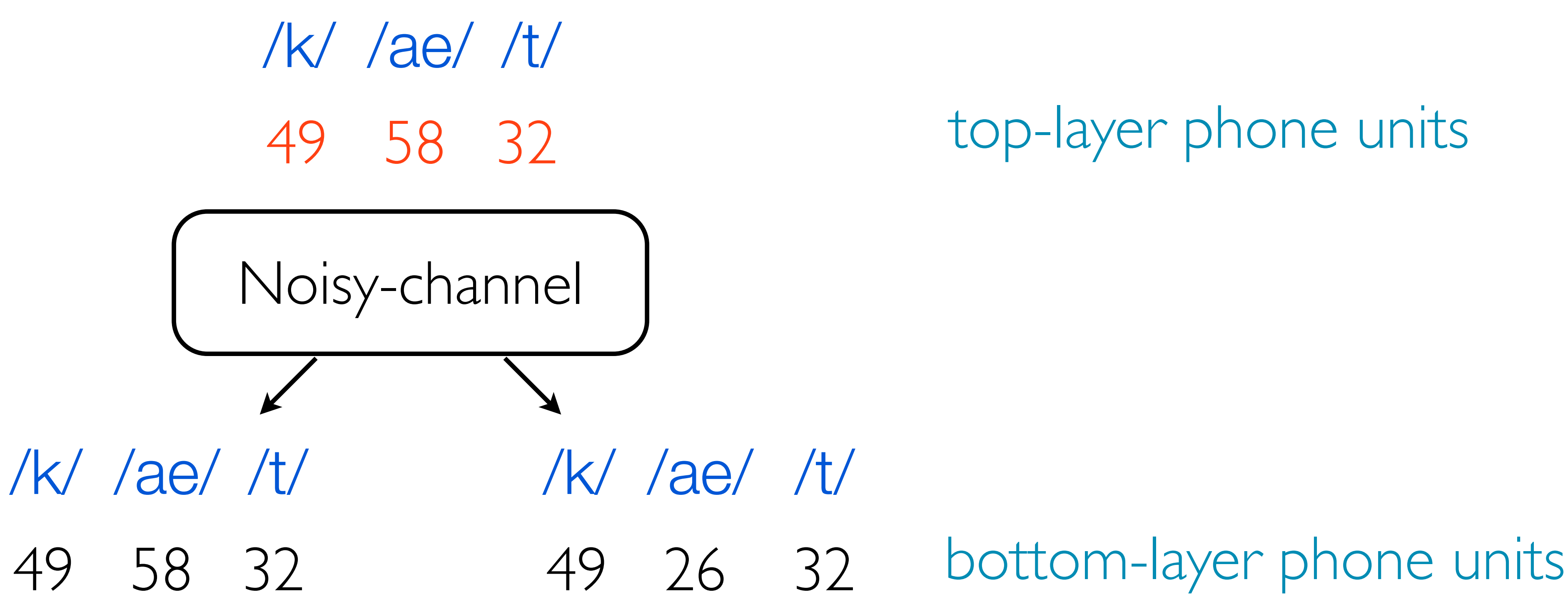
Noisy-channel Model

- Assume the phonetic variations are outcomes of a noisy-channel
- Formulate the noisy-channel model as a set of edit operations
  - Substitution, deletion, insertion, and exact-match



Noisy-channel Model

- Assume the phonetic variations are outcomes of a noisy-channel
- Formulate the noisy-channel model as a set of edit operations
  - Substitution, deletion, insertion, and exact-match



Model Overview

- Integrate adaptor grammars and the phone discovery model
  - To discover rich linguistic structures from speech
- Three components in the model

Adaptor grammars

Noisy-channel model

Phone discovery model

Model Overview

- Integrate adaptor grammars and the phone discovery model
  - To discover rich linguistic structures from speech
- Three components in the model

Adaptor grammars

Noisy-channel model

Phone discovery model

Generative Process

- Generate a parse from adaptor grammars

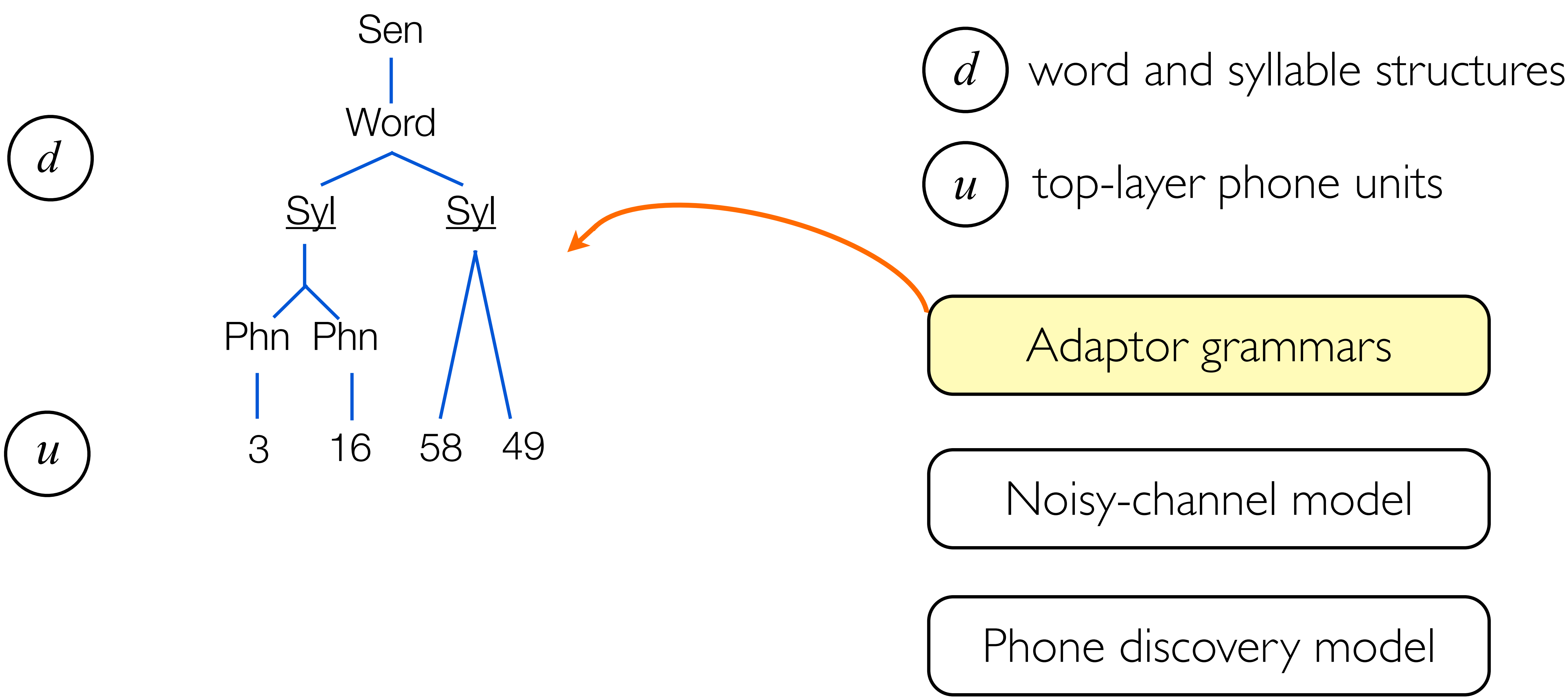
Adaptor grammars

Noisy-channel model

Phone discovery model

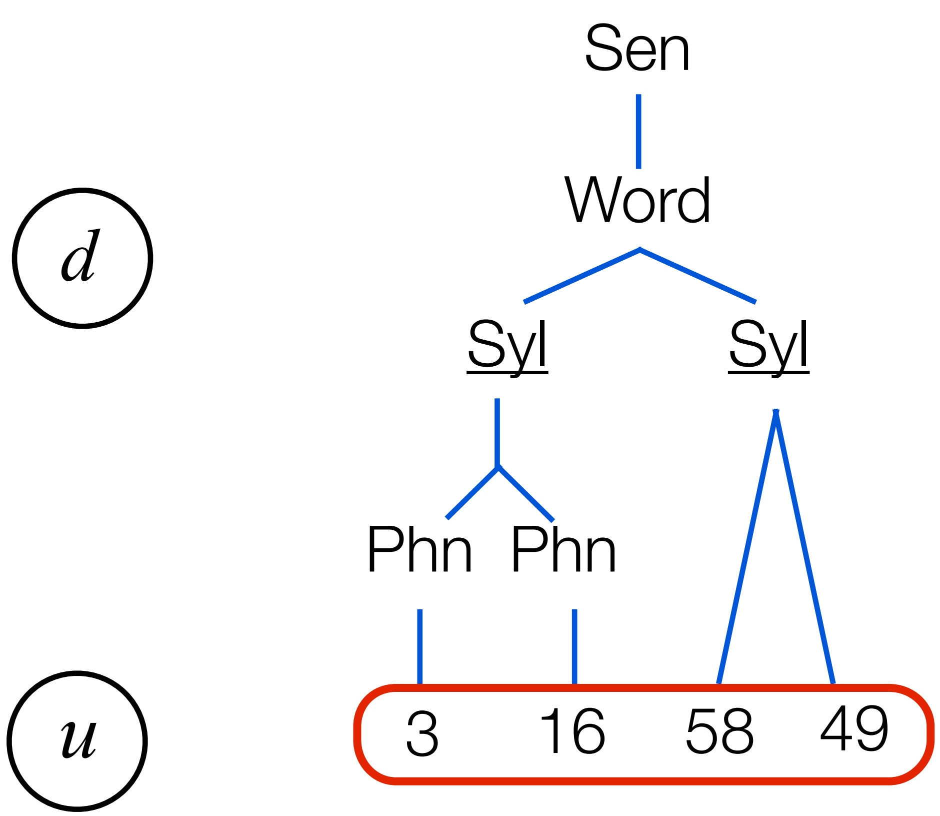
Generative Process

- Generate a parse from adaptor grammars



Generative Process

- Generate phonetic variations



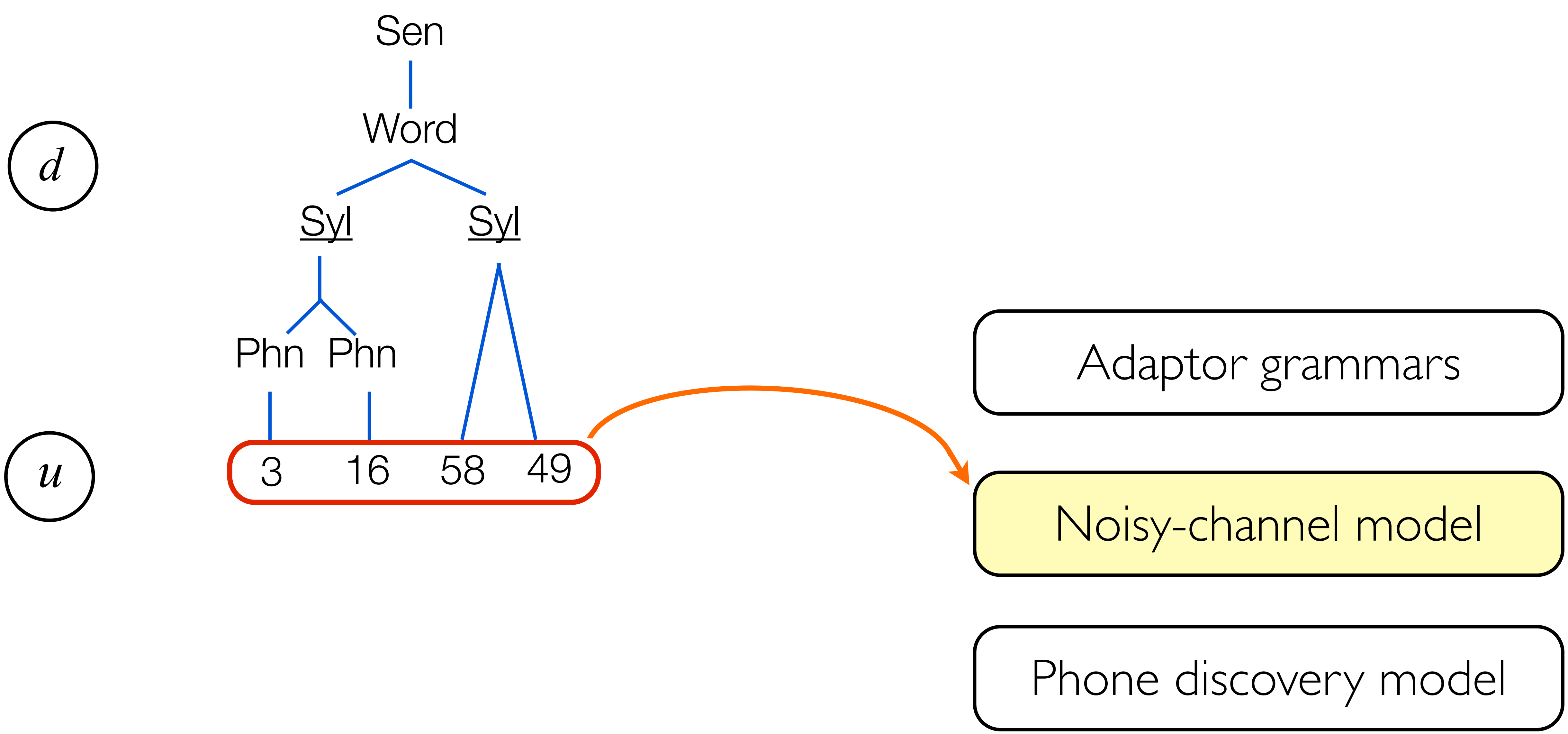
Adaptor grammars

Noisy-channel model

Phone discovery model

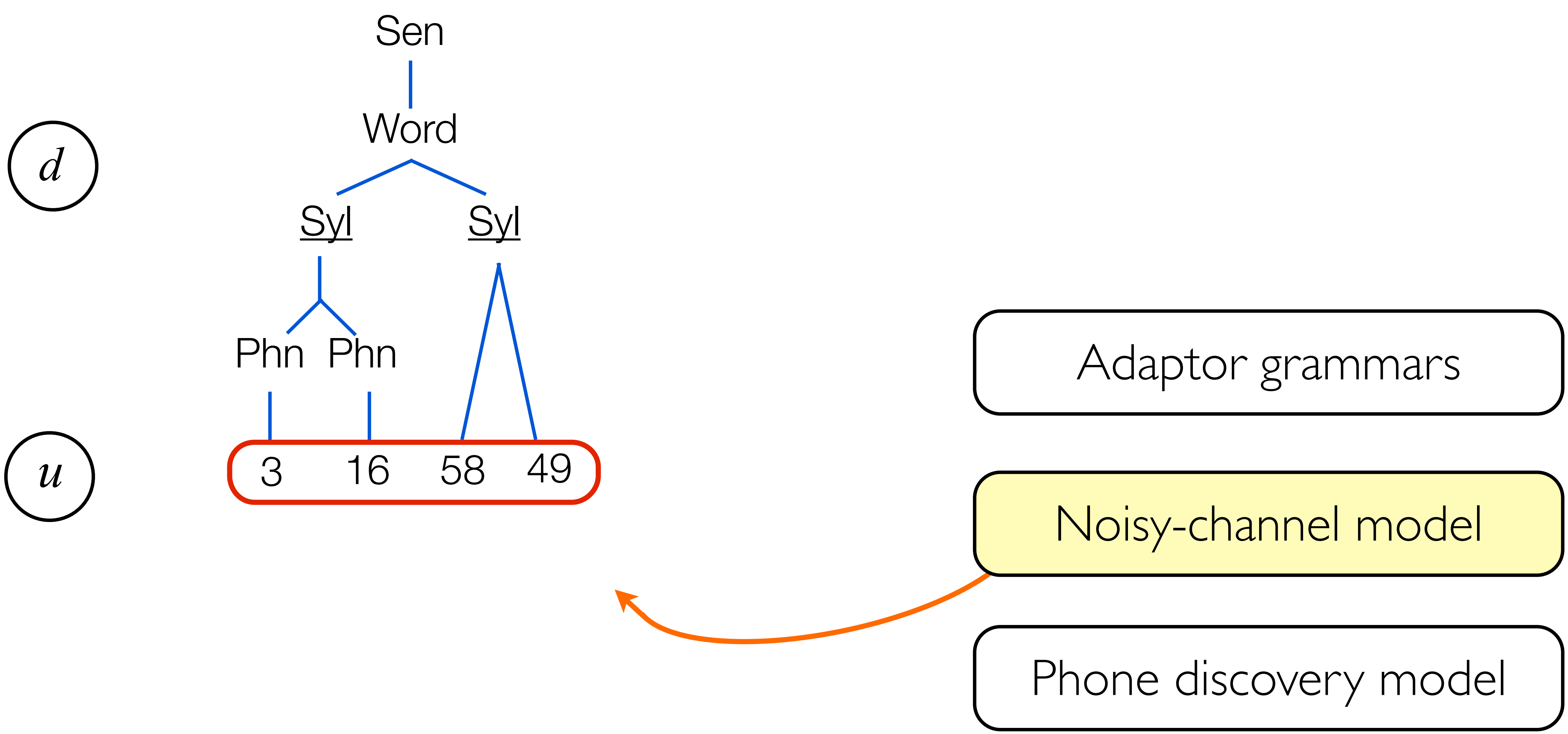
Generative Process

- Generate phonetic variations



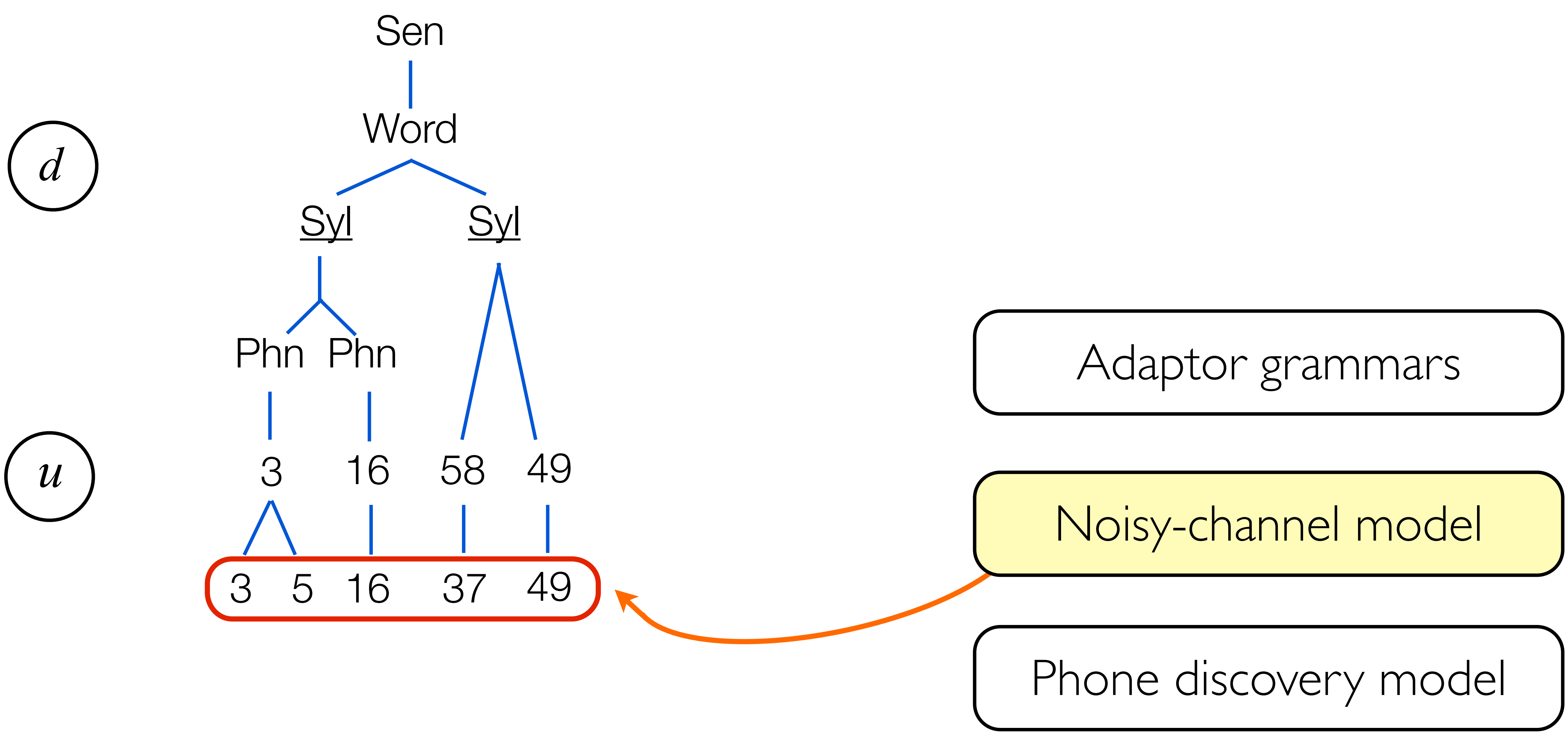
Generative Process

- Generate phonetic variations



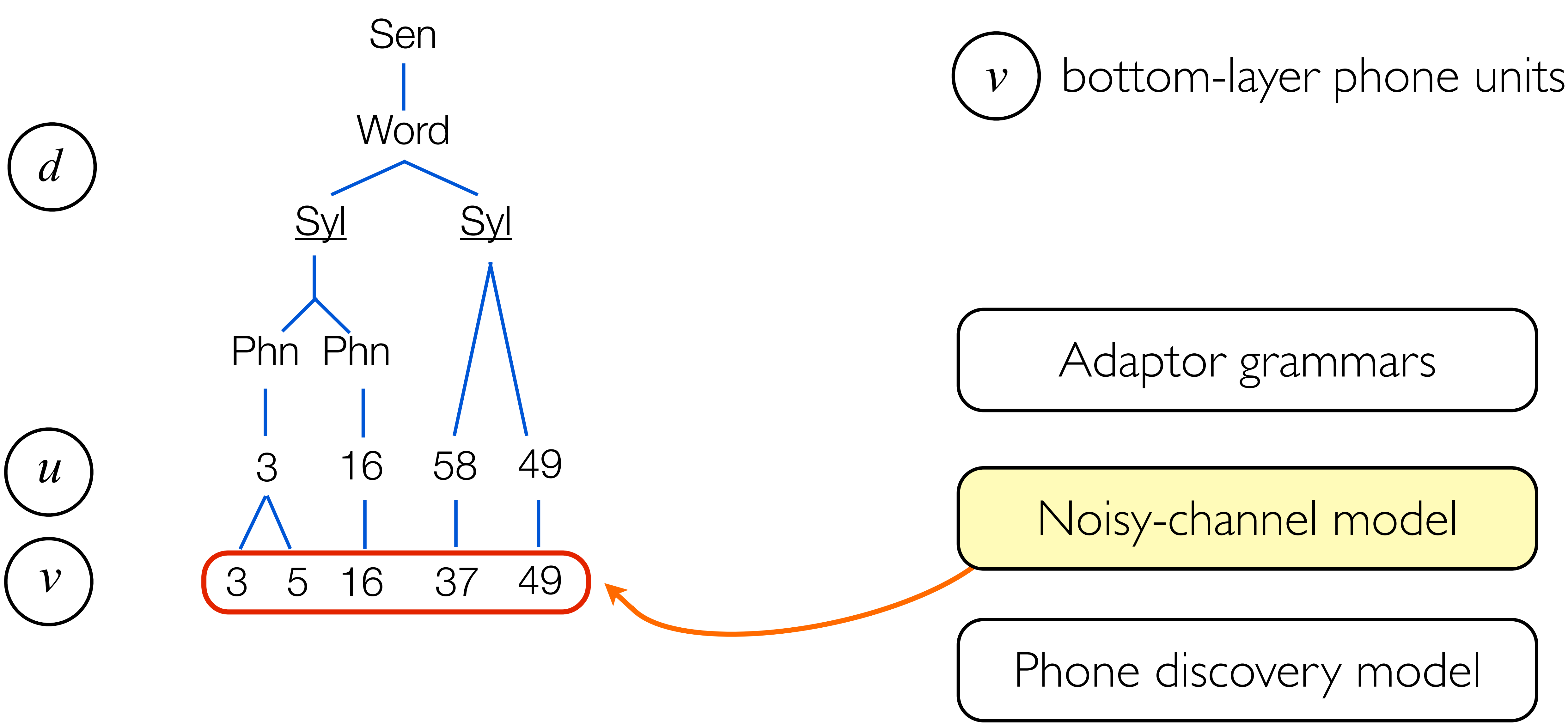
Generative Process

- Generate phonetic variations



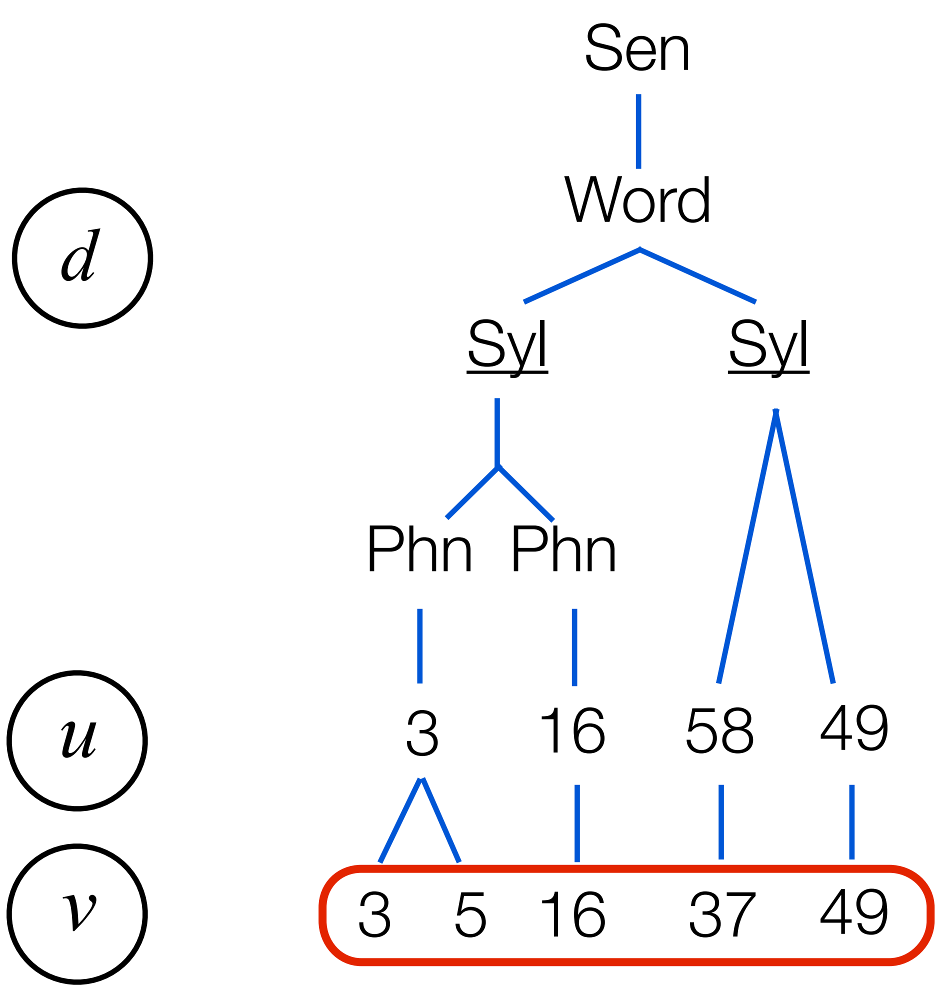
Generative Process

- Generate phonetic variations



Generative Process

- Generate speech data



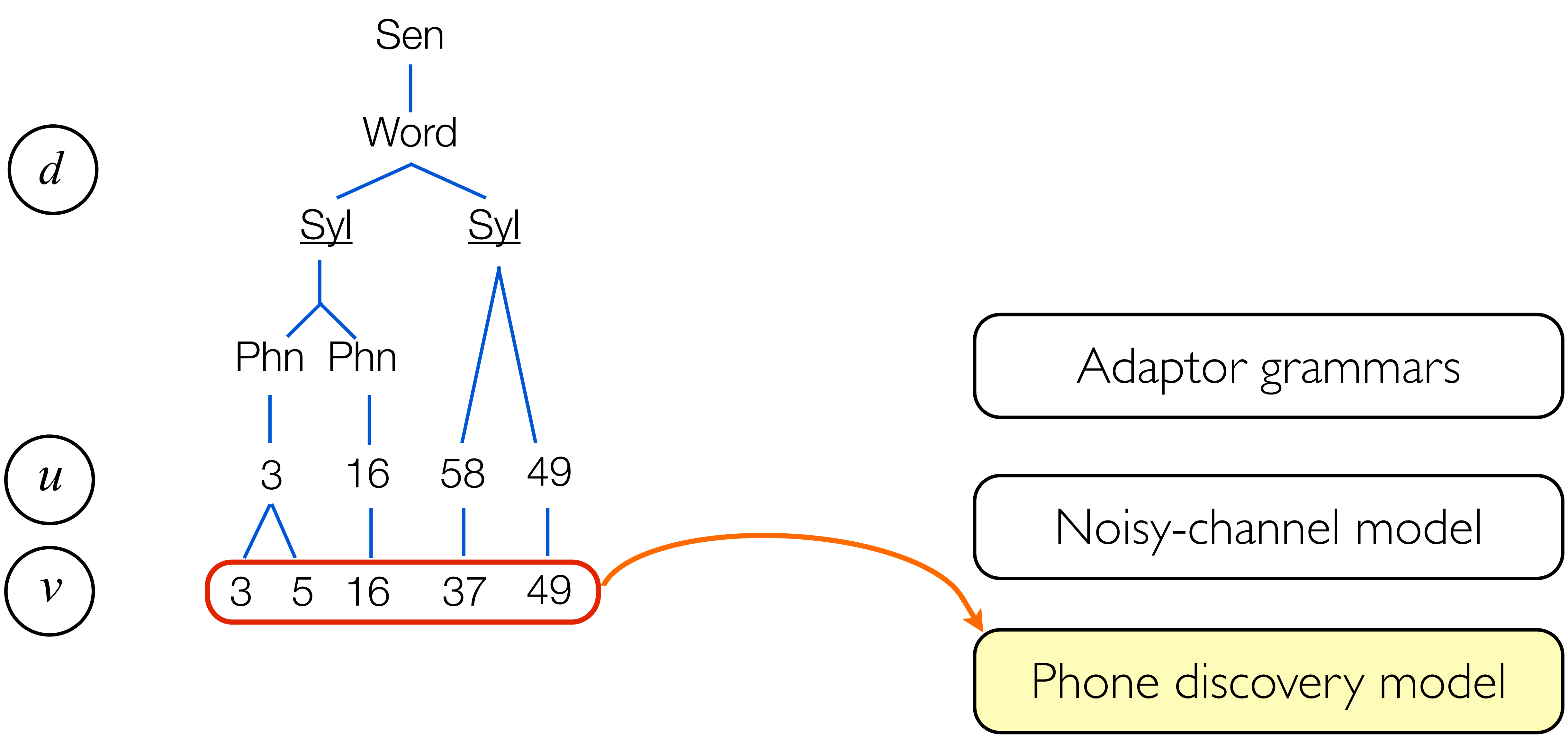
Adaptor grammars

Noisy-channel model

Phone discovery model

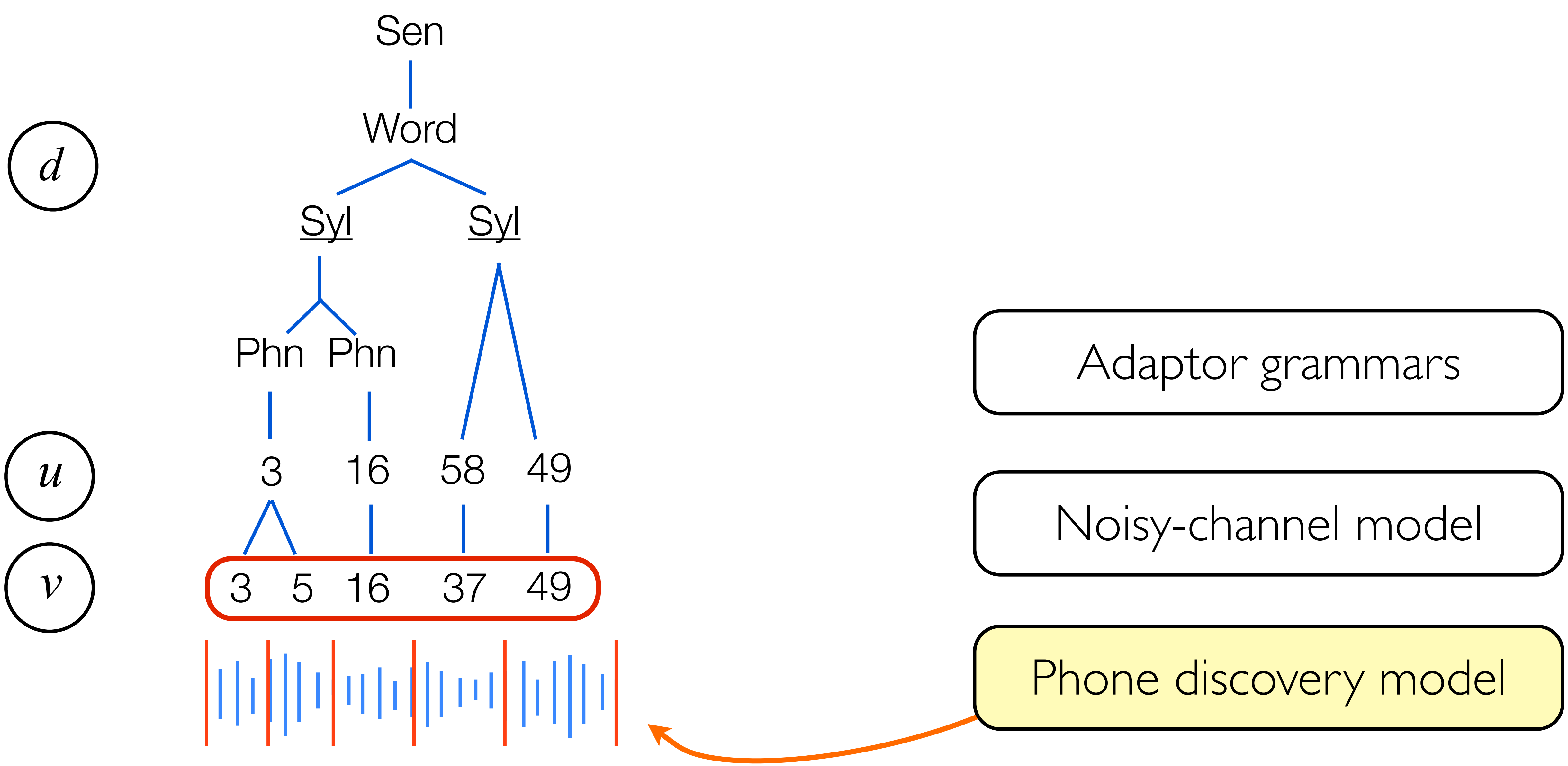
Generative Process

- Generate speech data



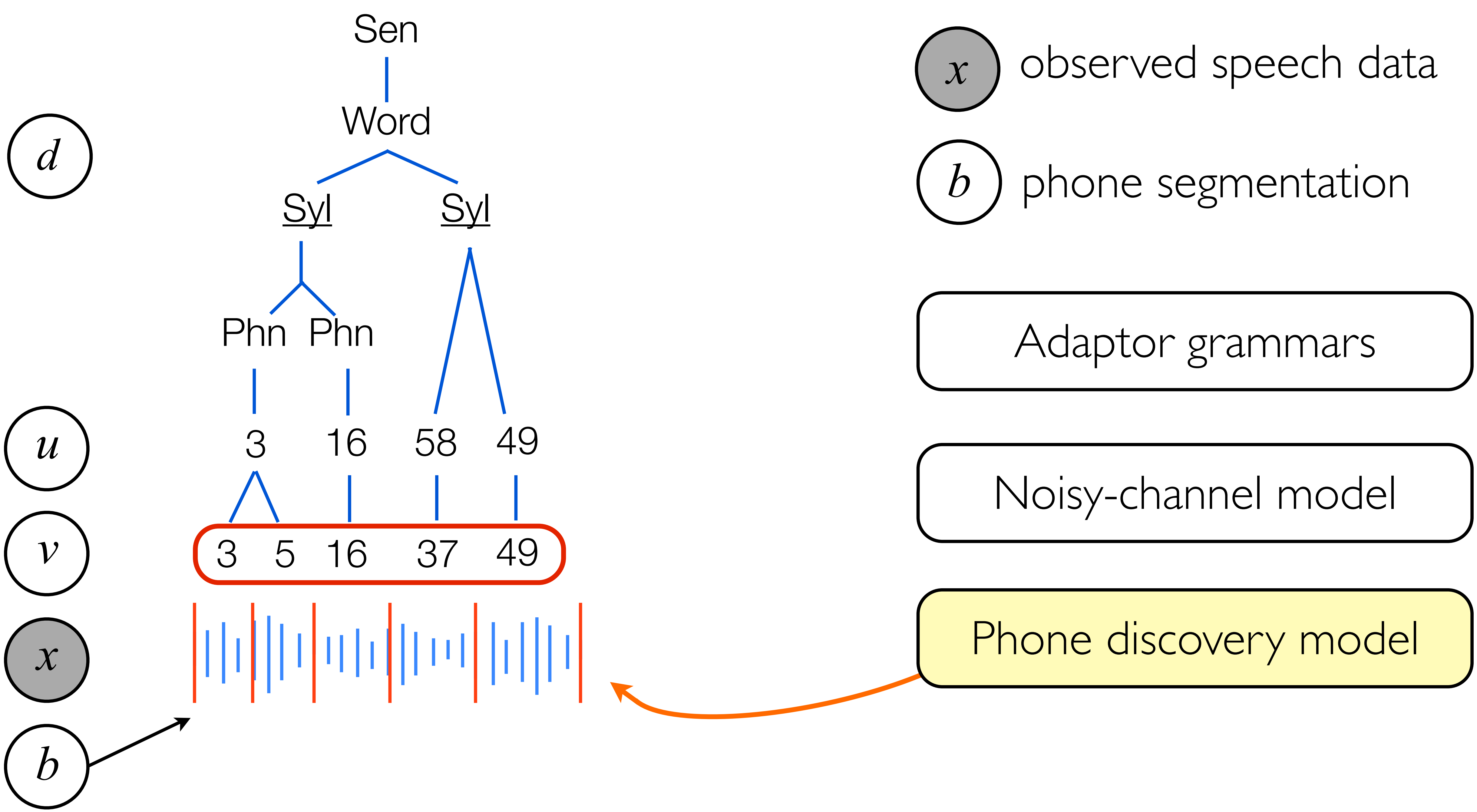
Generative Process

- Generate speech data

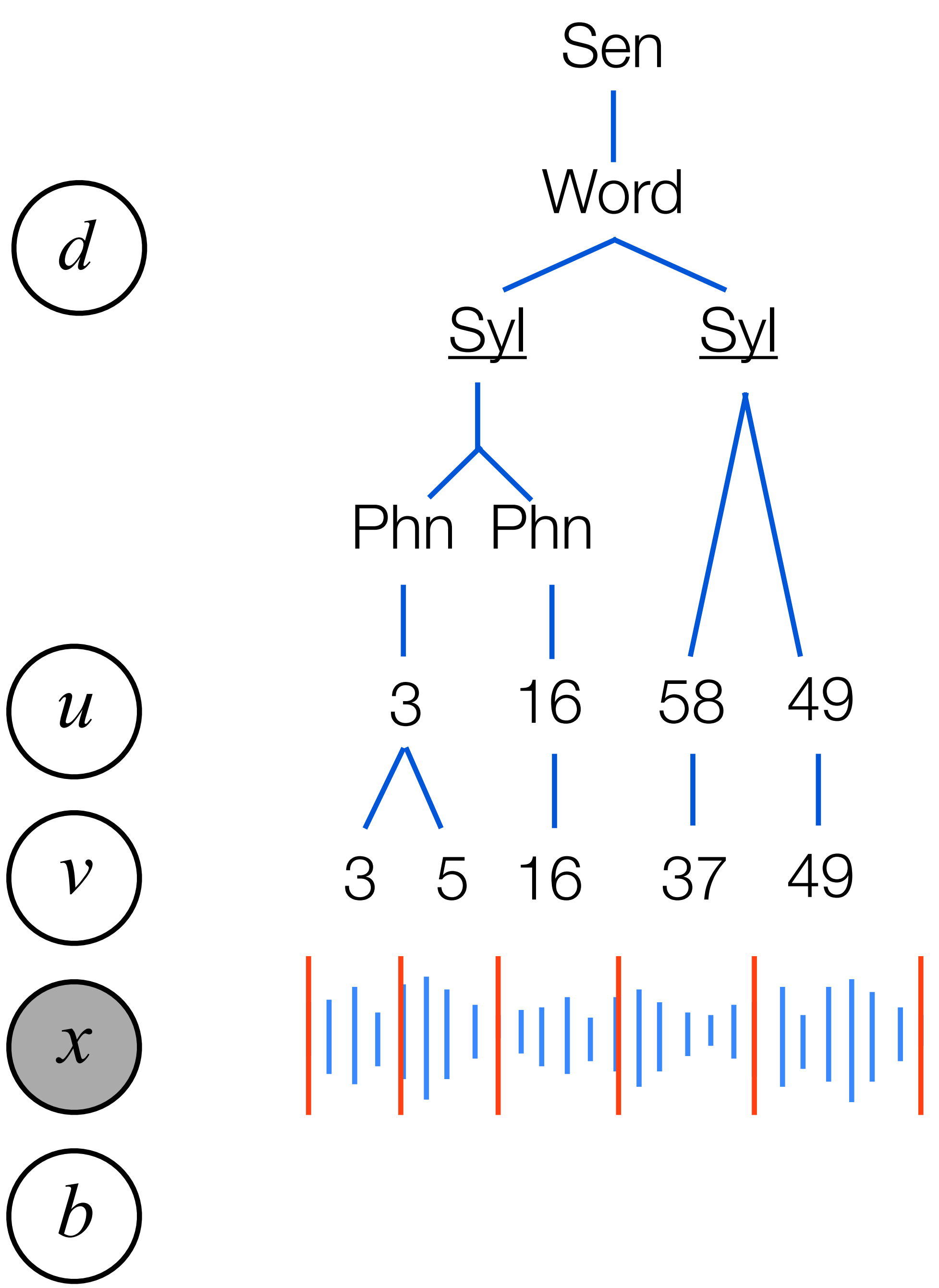


Generative Process

- Generate speech data



Generative Process

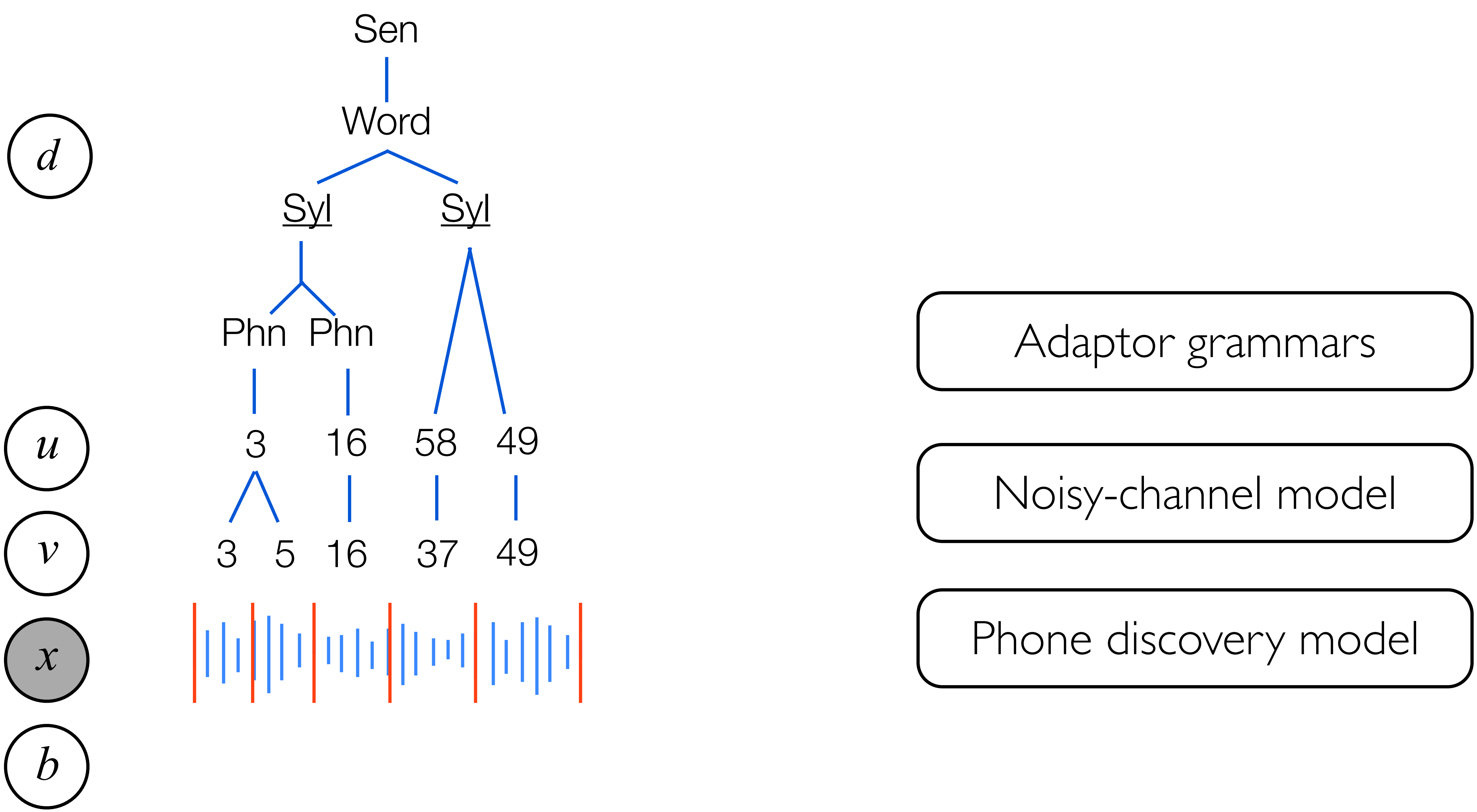


Adaptor grammars

Noisy-channel model

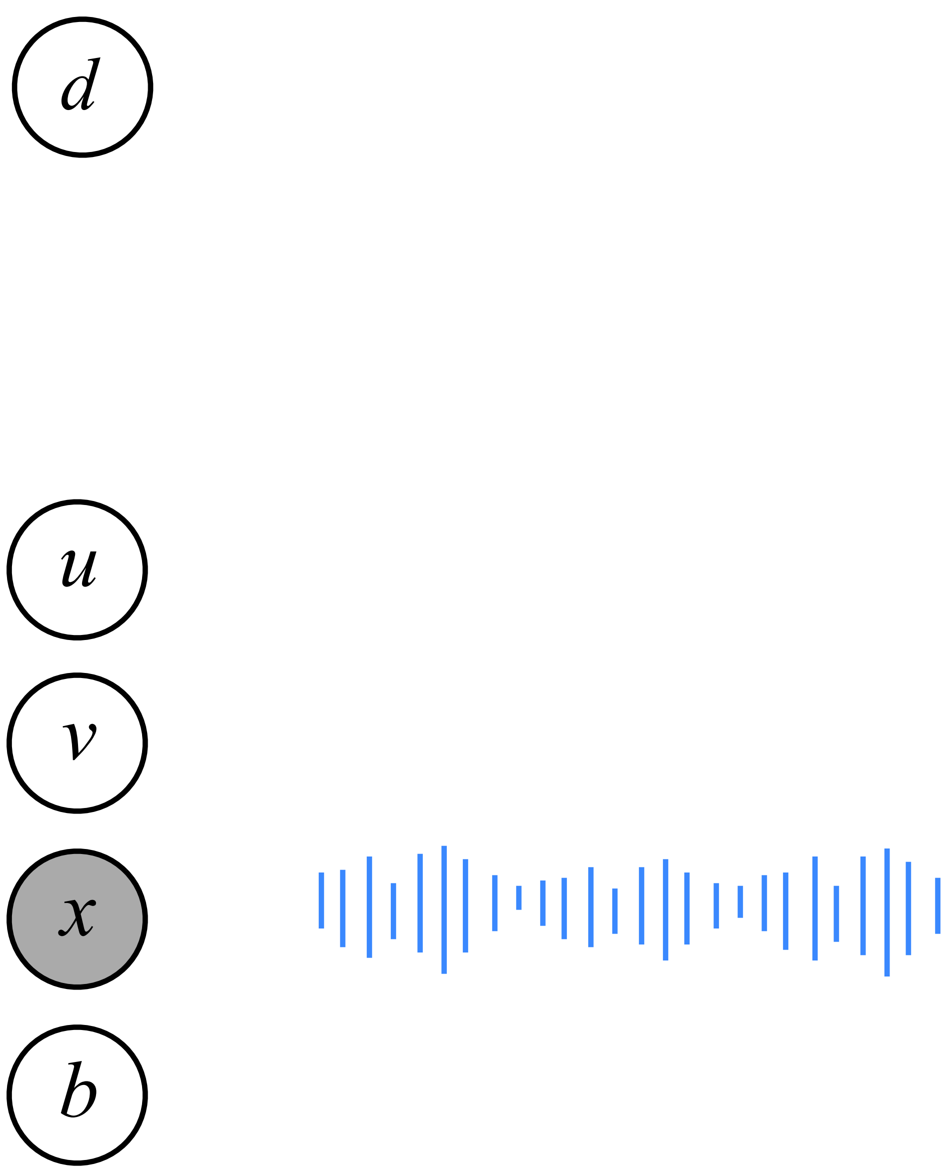
Phone discovery model

Inference


d
u
v
x
b
Sen
Word
Syl
Syl
Phn
Phn
3
16
58
49
3
5
16
37
49
Adaptor grammars
Noisy-channel model
Phone discovery model

Inference

- Only speech data are observed



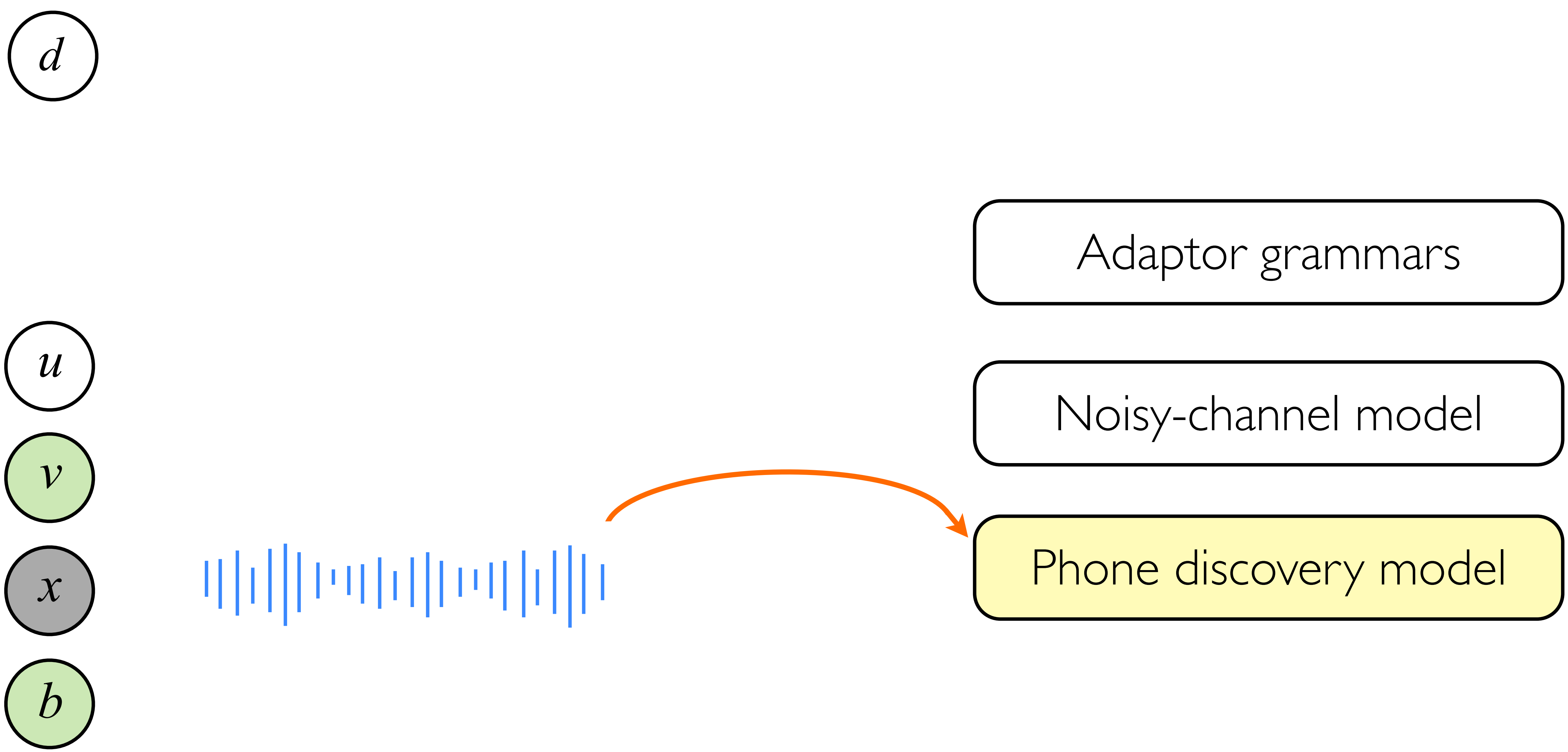
Adaptor grammars

Noisy-channel model

Phone discovery model

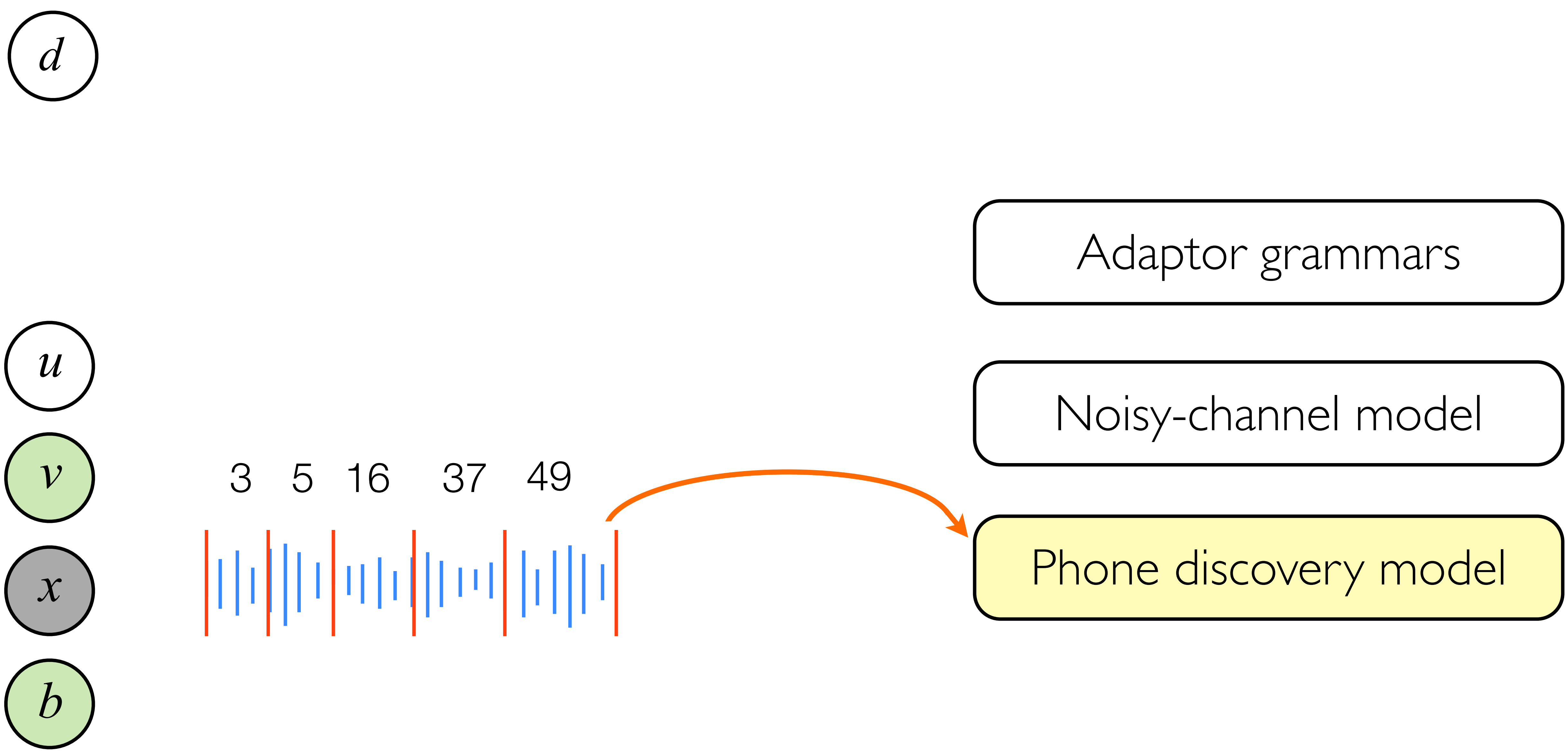
Initialization

- Initialize v and b using the phonetic discovery model



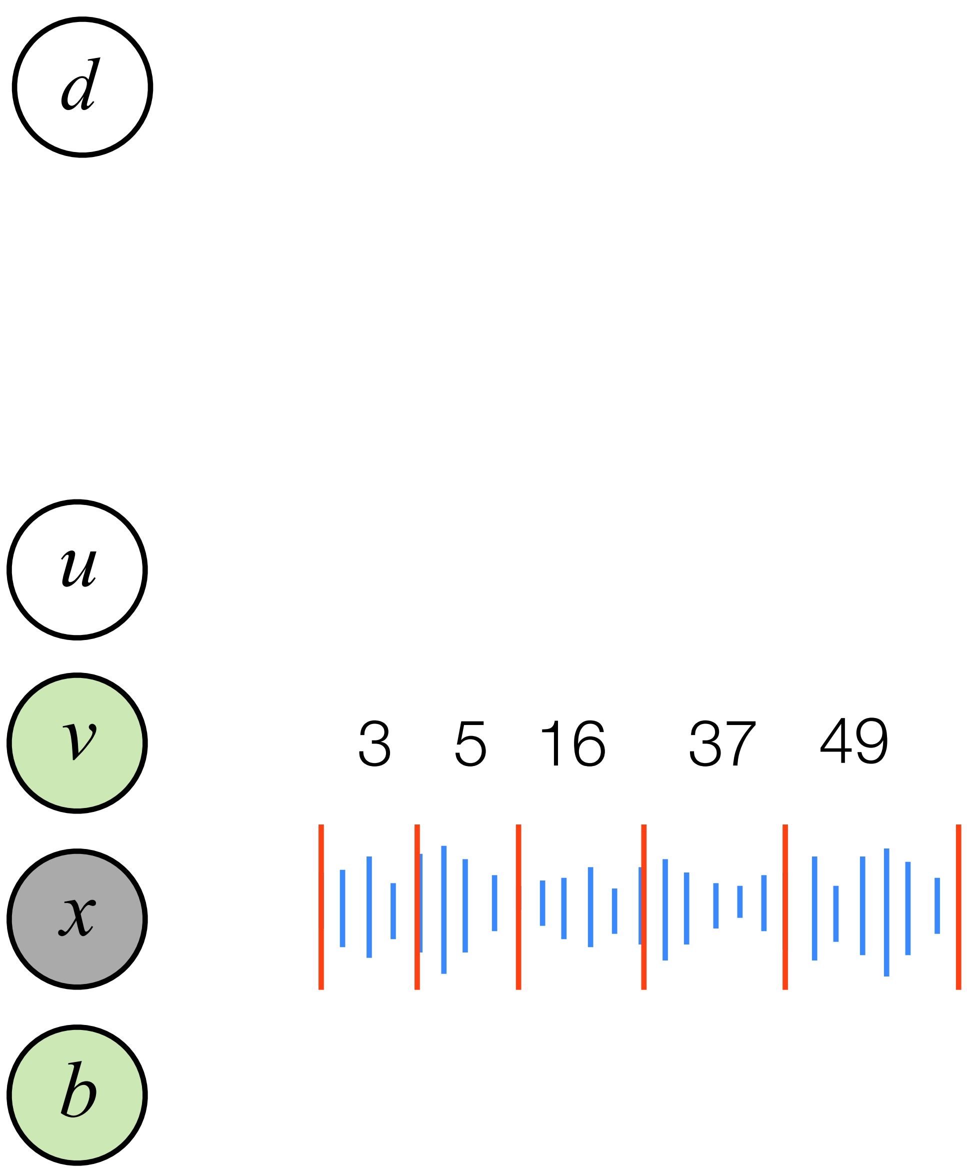
Initialization

- Initialize v and b using the phonetic discovery model



Inference

- Given v and b sample d and u



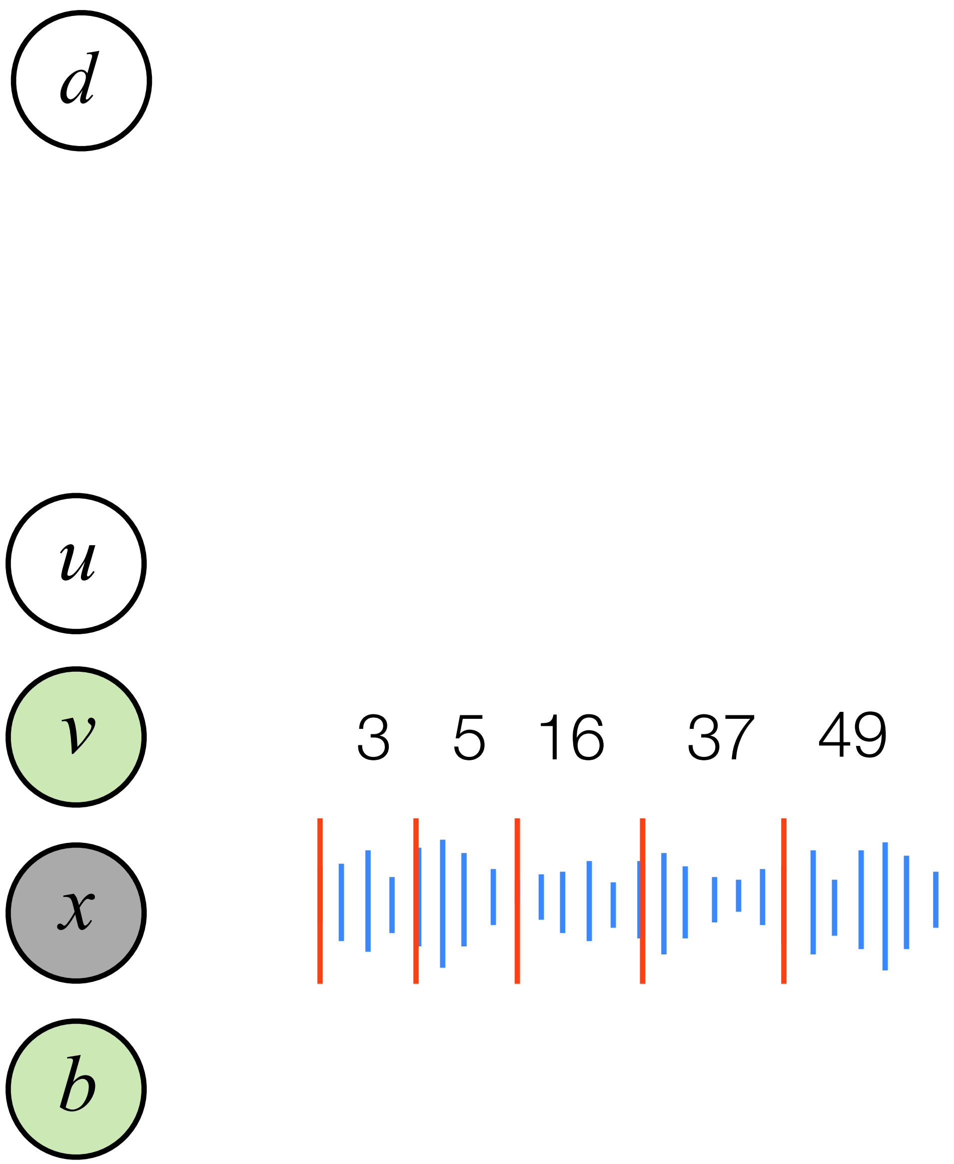
Adaptor grammars

Noisy-channel model

Phone discovery model

Inference

- Given v and b sample d and u



Metropolis-Hastings algorithm

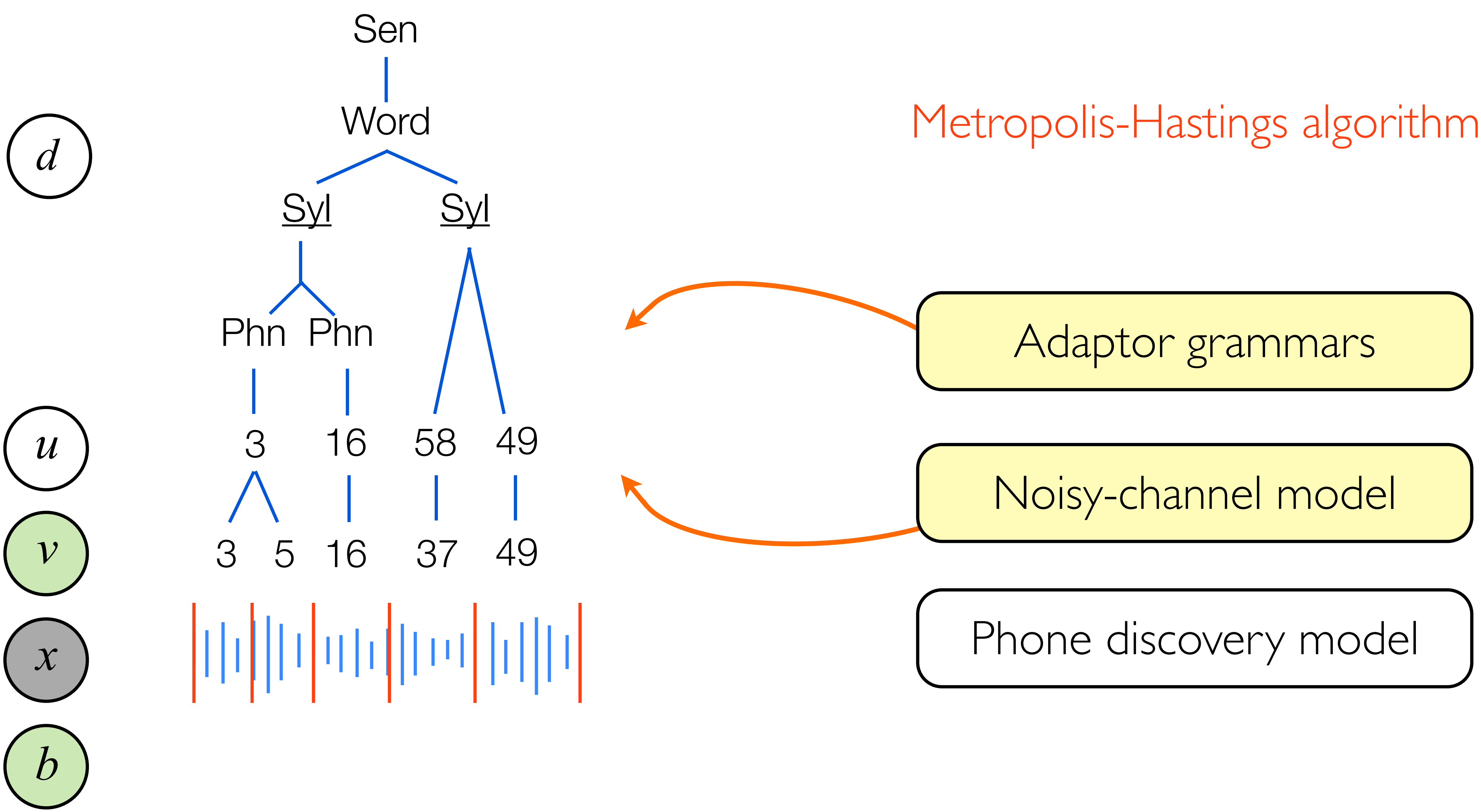
Adaptor grammars

Noisy-channel model

Phone discovery model

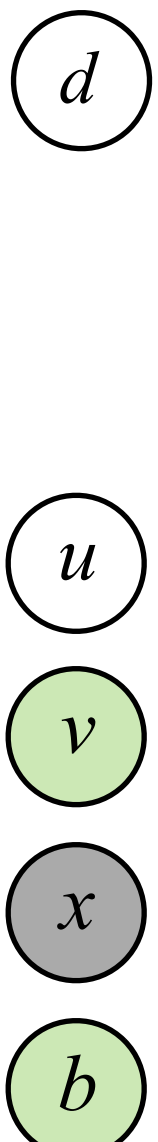
Inference

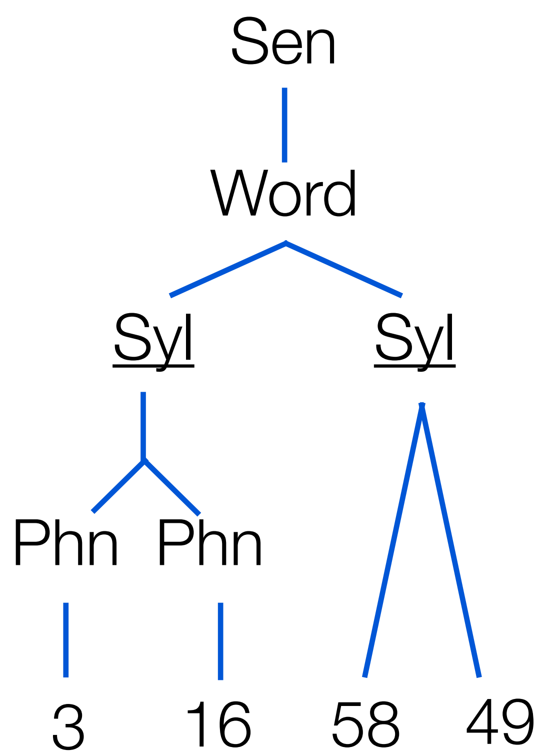
- Given v and b sample d and u



Inference

- Given d and u resample v and b





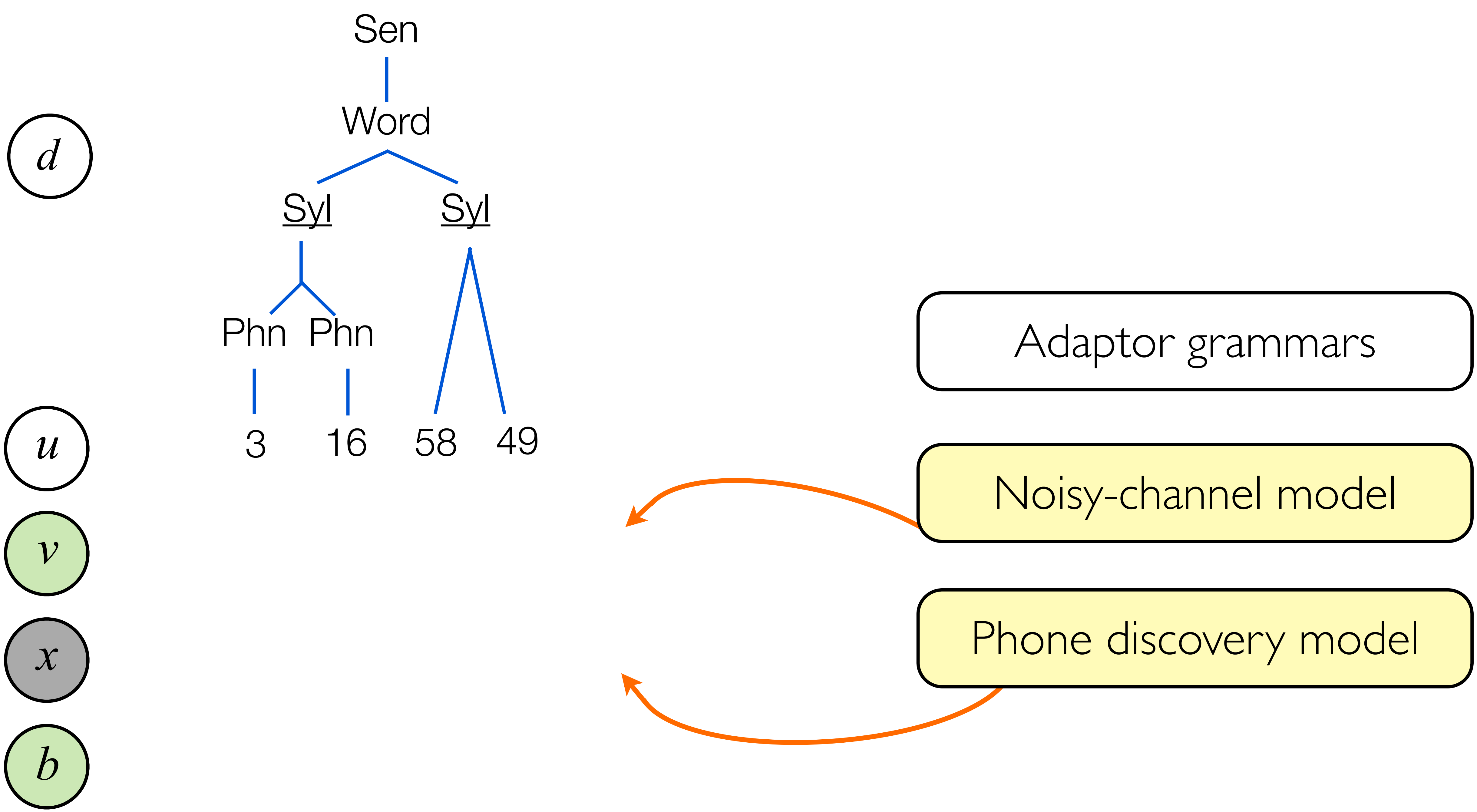
Adaptor grammars

Noisy-channel model

Phone discovery model

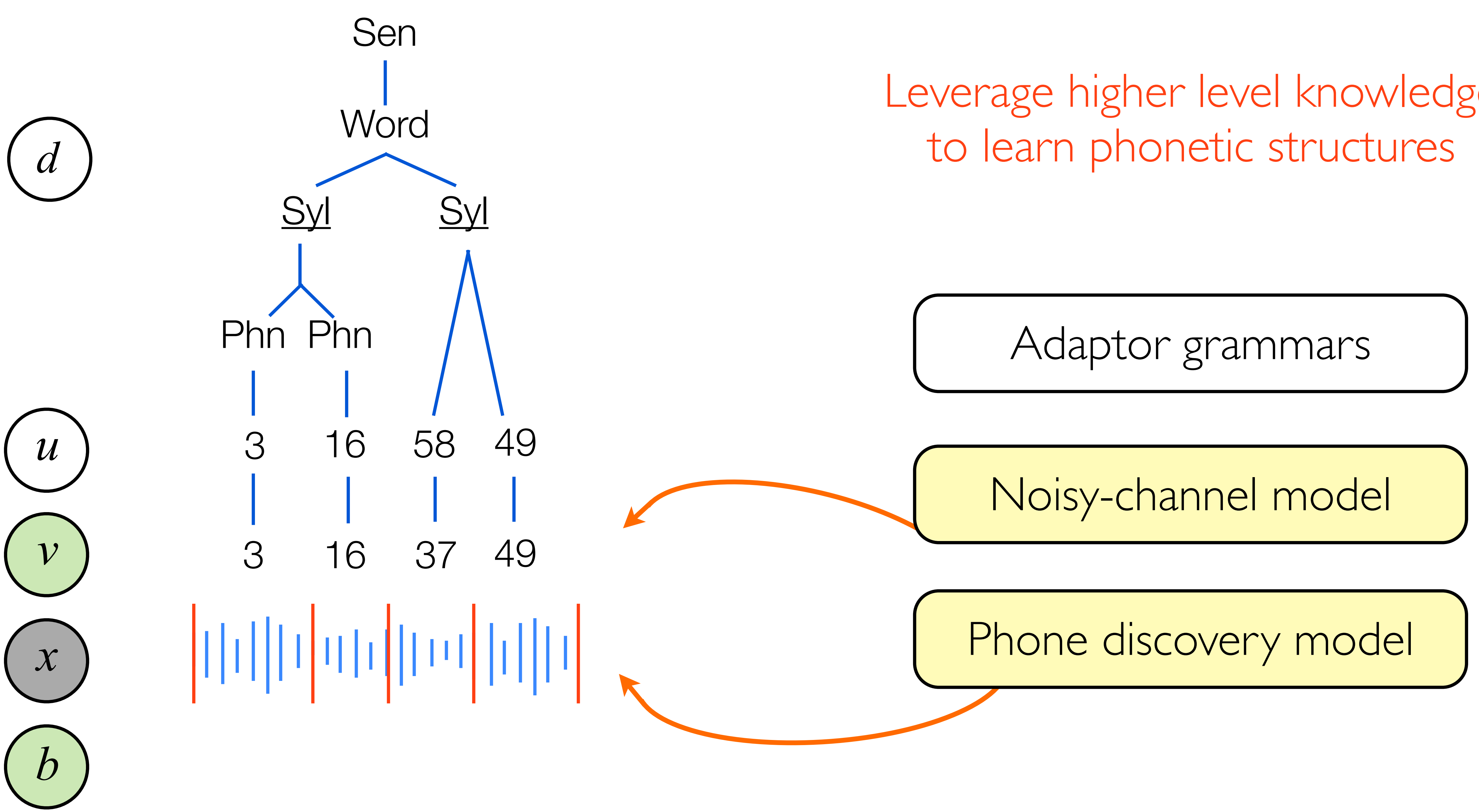
Inference

- Given d and u resample v and b



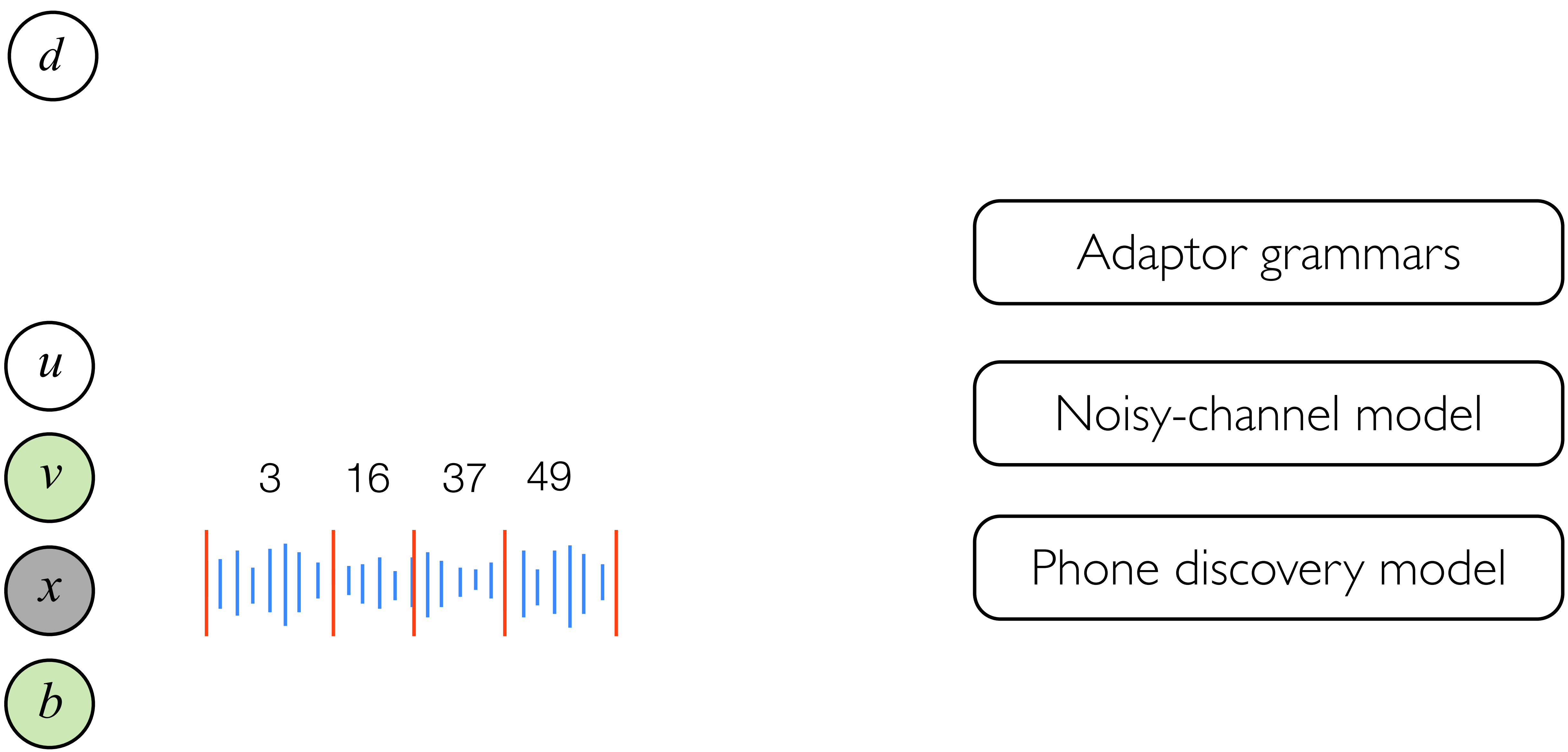
Inference

- Given d and u resample v and b



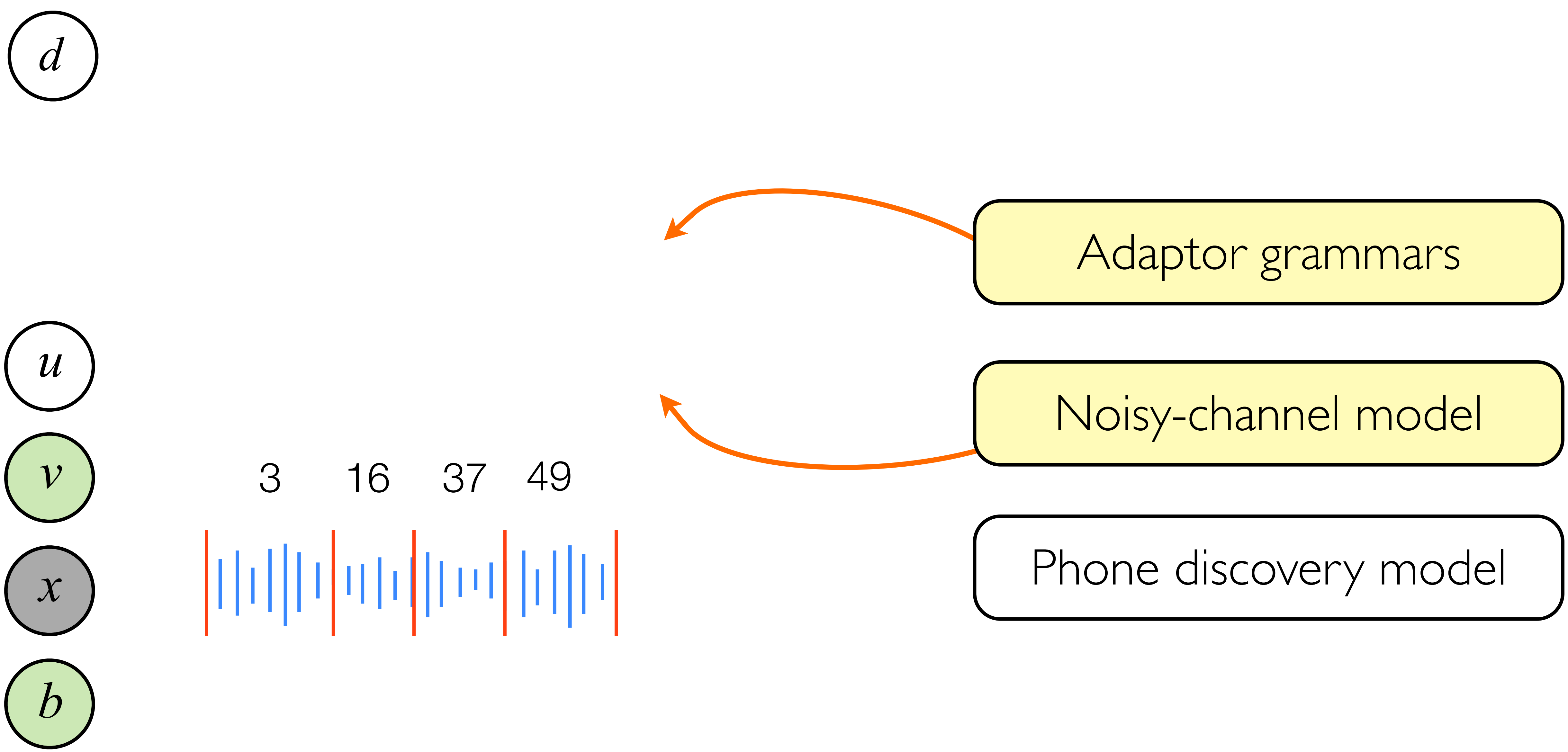
Inference

- Given v and b sample d and u



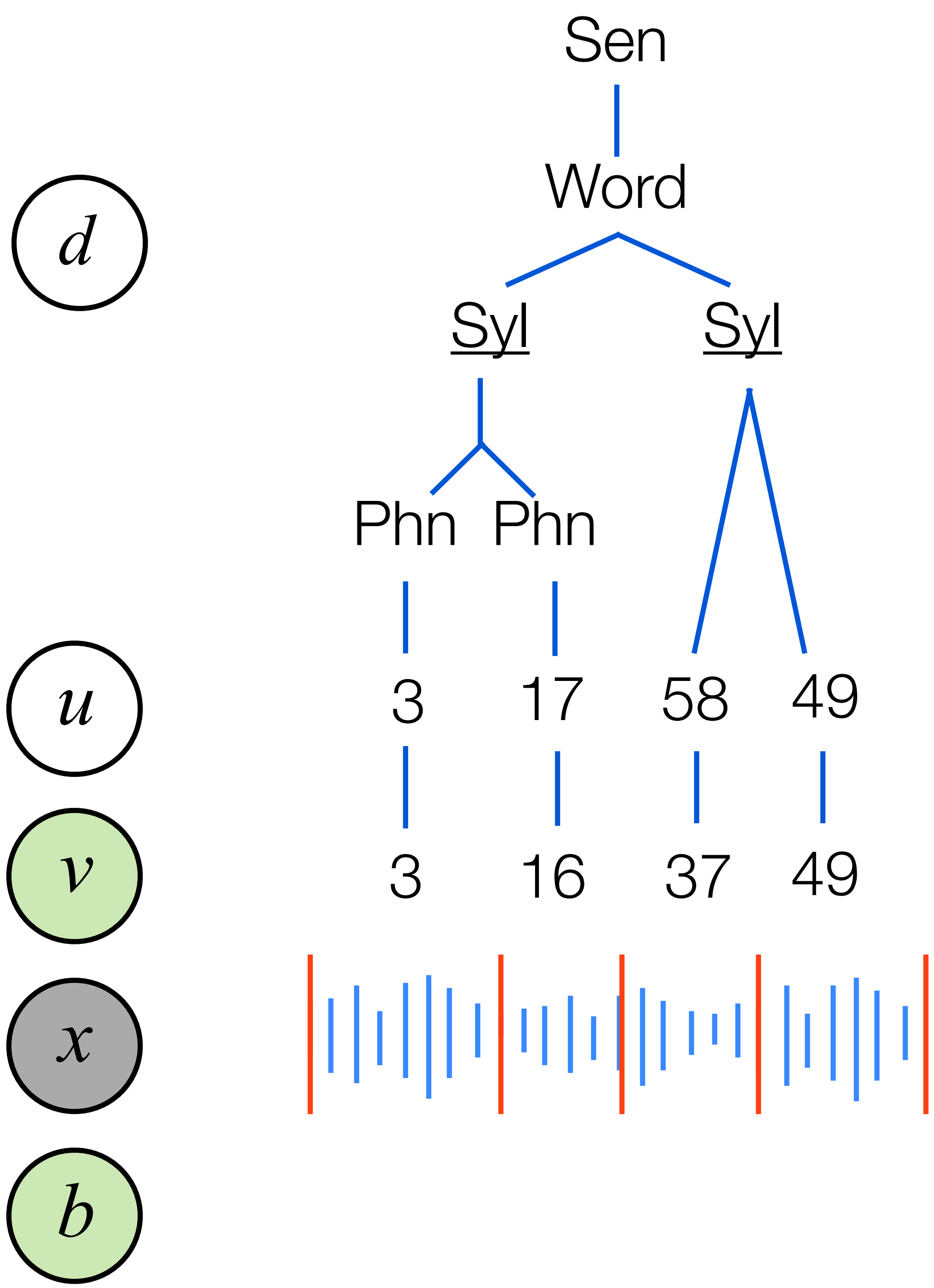
Inference

- Given v and b sample d and u

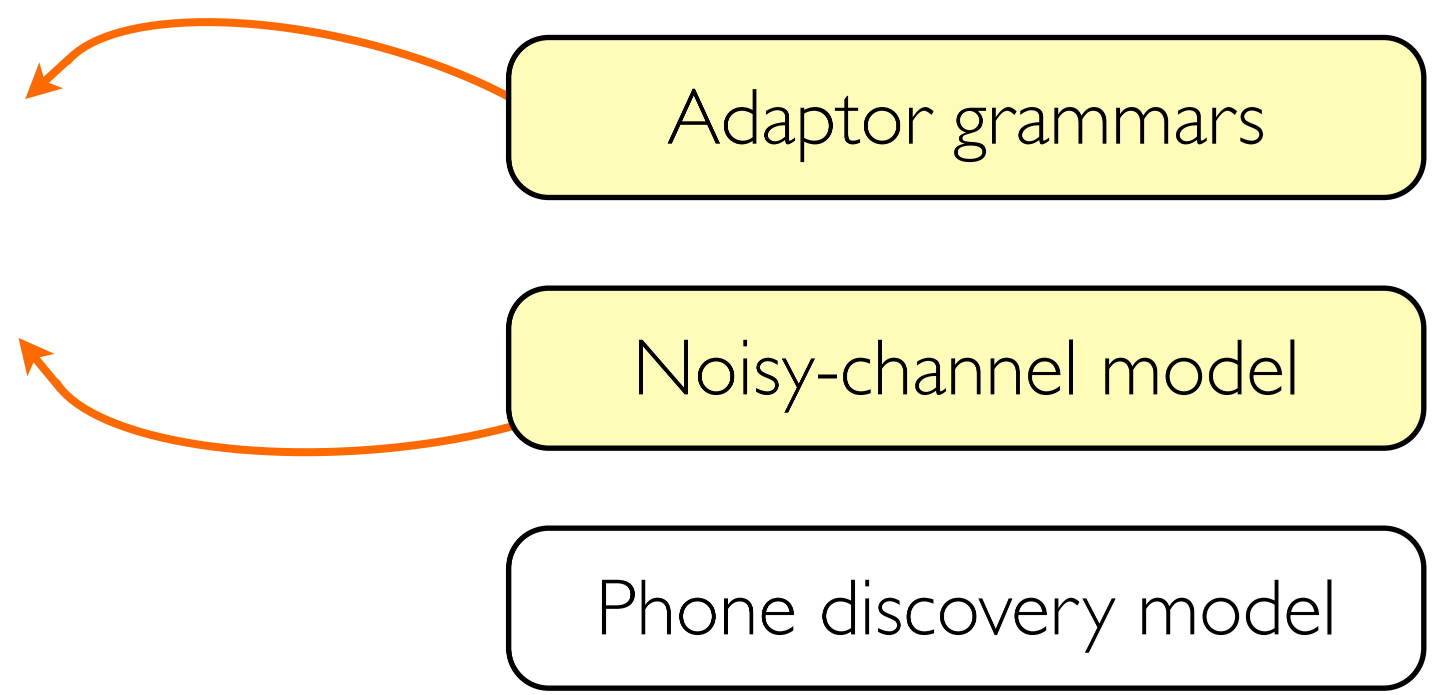


Inference

- Given v and b sample d and u



Alternate between the two steps



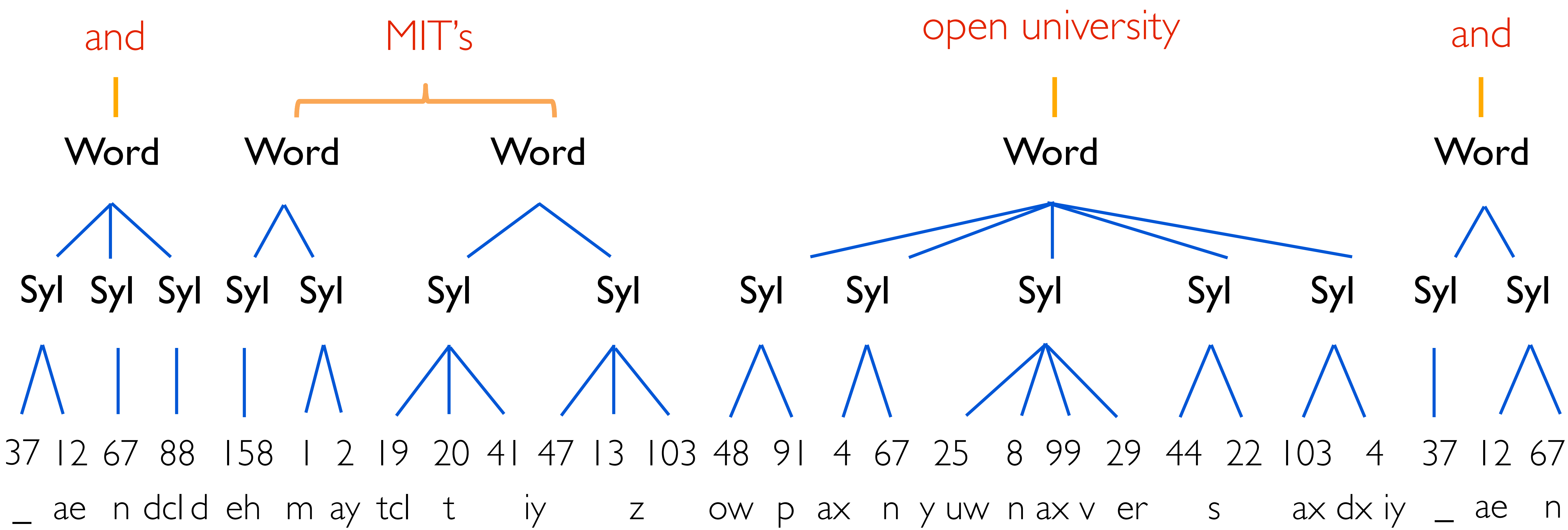
Experimental Setup

- **MIT Lecture Corpus**
  - The six lectures evaluated in [*Park and Glass, IEEE Trans. 2008*]
  - Each lecture contains ~1 hour of speech data by a single speaker
  - Each lecture contains a set of subject-specific keywords
- **Qualitative assessment**
  - Sentence and word parses
  - Analysis on the discovered hierarchical linguistic structures
- **Quantitative assessment**
  - Coverage of subject-specific keywords
  - (Word and phone segmentation)

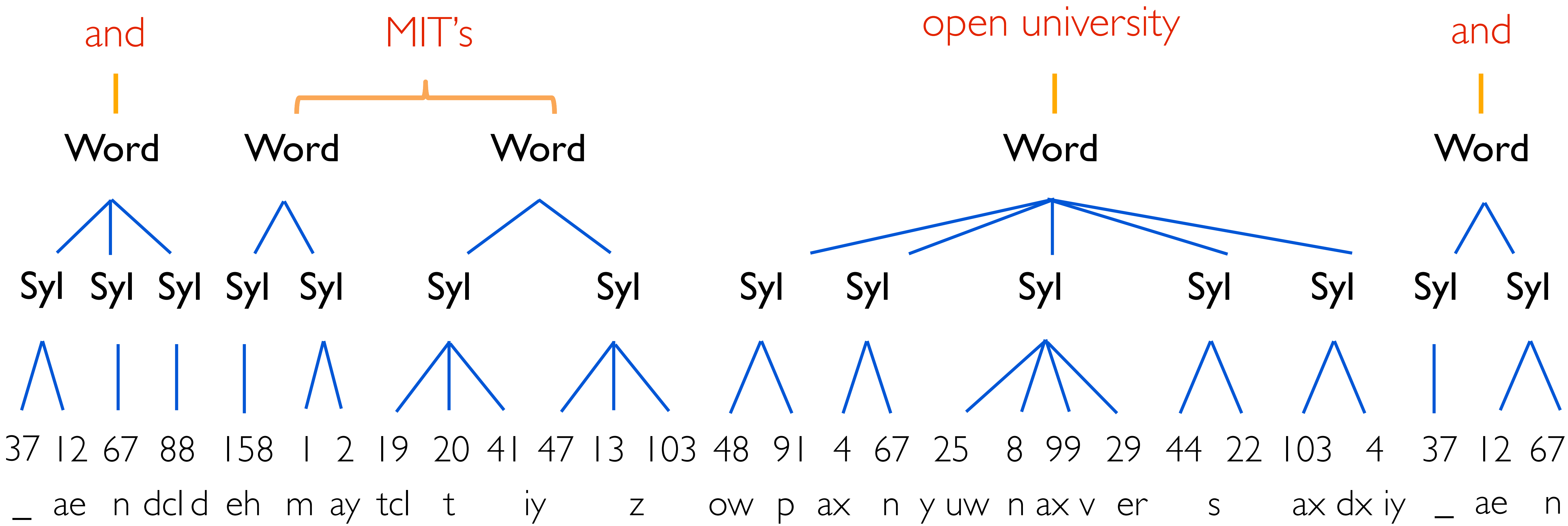
Parse of a Full Sentence

37 12 67 88 158 1 2 19 20 41 47 13 103 48 91 4 67 25 8 99 29 44 22 103 4 37 12 67

Parse of a Full Sentence



Parse of a Full Sentence



MIT's only occurs 3 times in the lecture

open and university almost always appear together in the lecture

Word Parses

- Two instances of “collaboration”

Word Parses

- Two instances of “collaboration”

u	50	137	28	16	18	31	43	6	7	30	50	137	28	16	18	31	43	6	7	30
v	91	106	28	16	18	29	43	6	7	30	50	106	28	16	18	29	43	41	7	30

Word Parses

- Two instances of "collaboration"
  - Noisy-channel model regularizes the bottom-layer phone units

collaboration

[50 137] [28 16] [18 31 43] [6 7 30]

kcl k el ae bcl ax r ey sh en

u	50	137	28	16	18	31	43	6	7	30	50	137	28	16	18	31	43	6	7	30
v	91	106	28	16	18	29	43	6	7	30	50	106	28	16	18	29	43	41	7	30

Word Parses

- Two instances of "collaboration"
  - Noisy-channel model regularizes the bottom-layer phone units
  - Highly reusable sub-word structures

collaboration

[50 137] [28 16] [18 31 43] [6 7 30]

kcl k el ae bcl ax r ey sh en

u	50	137	28	16	18	31	43	6	7	30
v	91	106	28	16	18	29	43	6	7	30

u	50	137	28	16	18	31	43	6	7	30
v	50	106	28	16	18	29	43	41	7	30

Structure Reuse

- Examples of reusing [6 7 30]

collaboration

[50 137]	[28 16]	[18 31 43]	**[6 7 30]**
kcl k	el ae	bcl ax r	ey sh en

reservation

[1 158]	[70 23]	[34 99]	**[6 7 30]**
r eh	z	er v	ey sh en

innovation

[67]	[1 27]	[99]	**[6 7 30]**
ih	n ax	v	ey sh en

globalization

[106 48]	[18 31]	[147 13]	**[6 7 30]**
gcl g l ow	bcl ax l	ax z	ey sh en

foundation

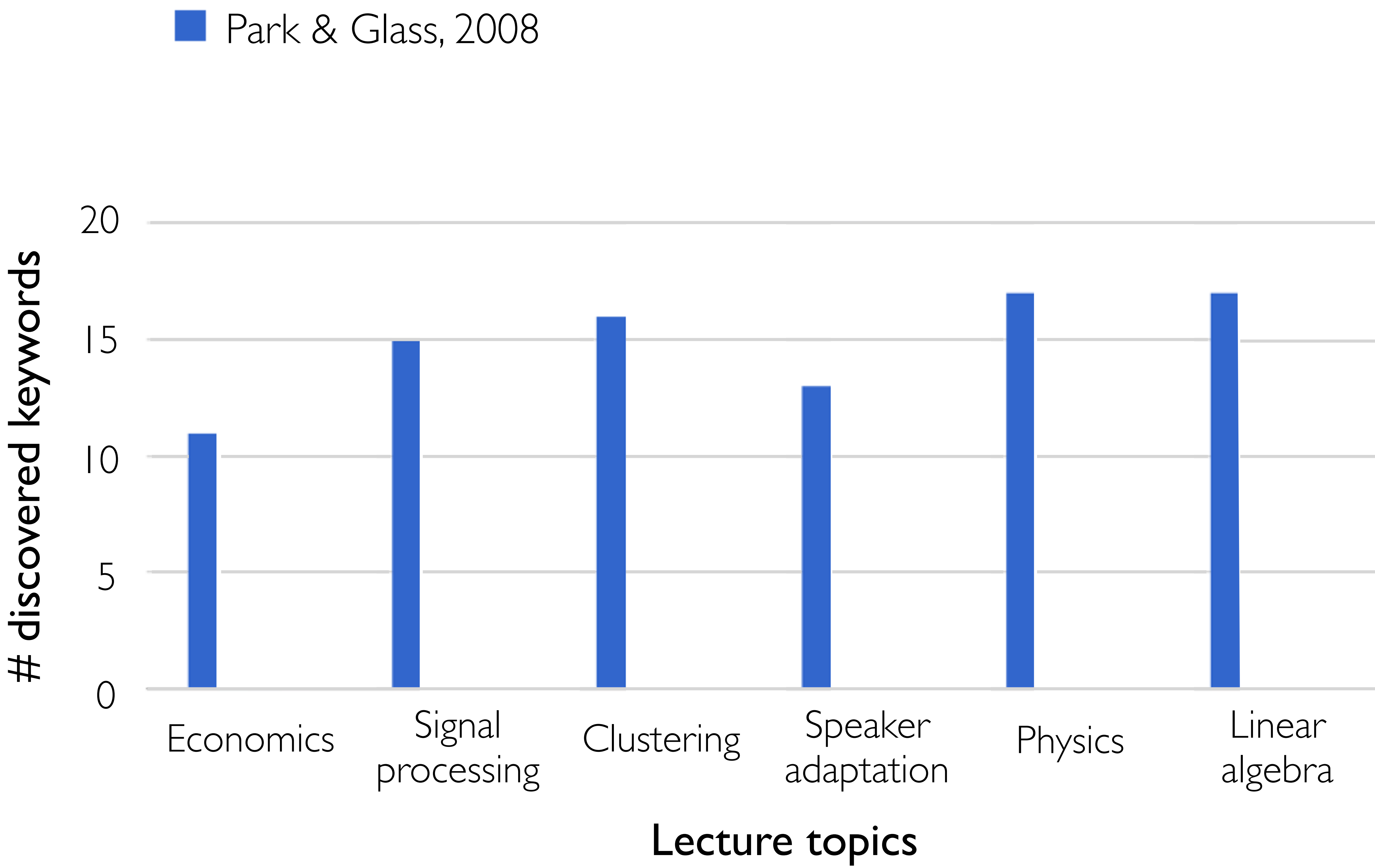
[22 46 8]	**[6 7 30]**
f aw n dcl d	ey sh en

Subject-specific Keywords

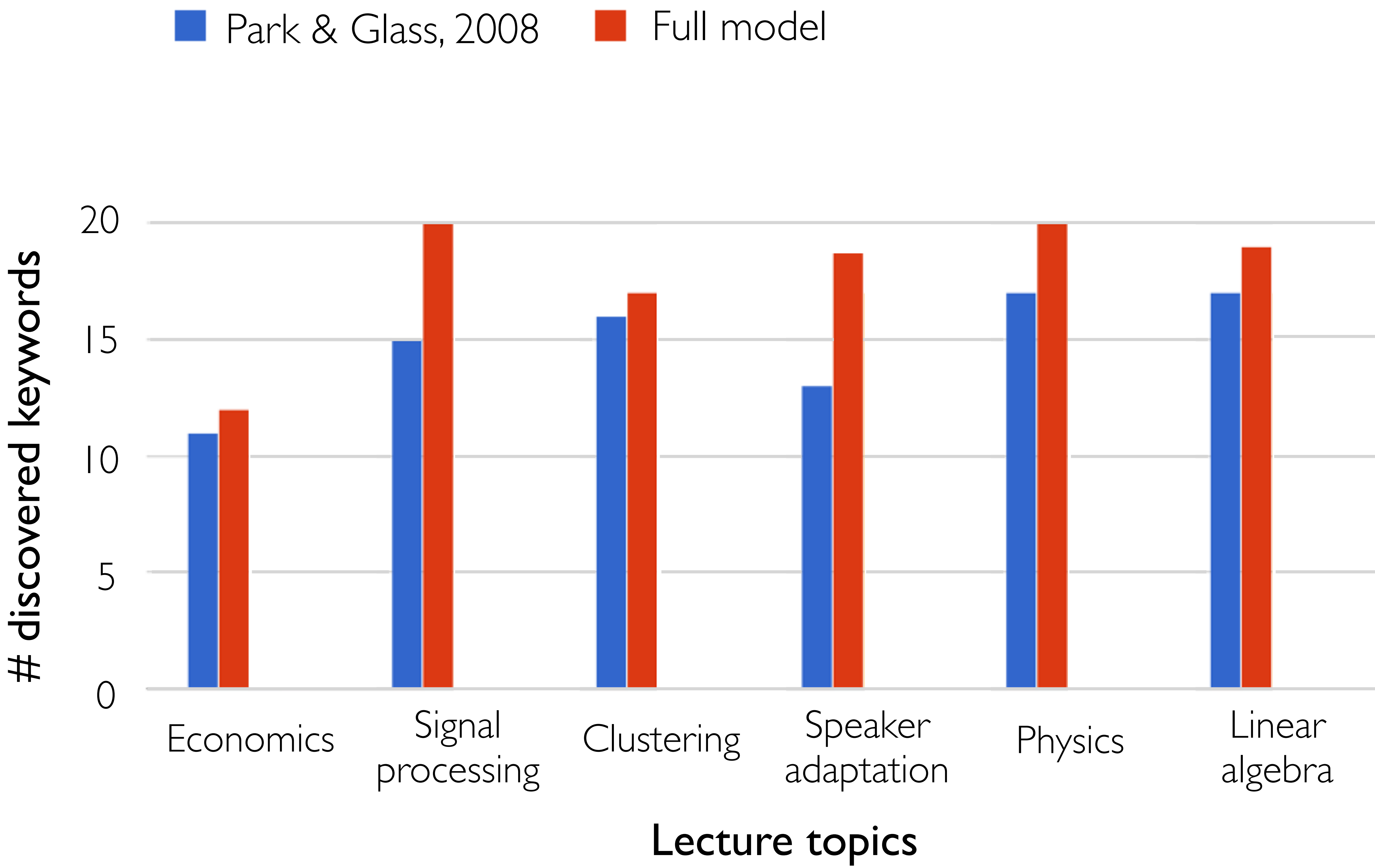
- **Term Frequency Inverse Document Frequency (TFIDF) scores**
  - The top 20 words for each lecture [*Park and Glass, IEEE Trans. 2008*]
- **Keyword examples**
  - From the seminar about the book "The world is flat" by Thomas Friedman

1.	flat	6.	flattener	11.	airline	16.	huge
2.	globalization	7.	dollar	12.	thousand	17.	create
3.	collaboration	8.	China	13.	outsourcing	18.	convergence
4.	India	9.	southwest	14.	really	19.	connect
5.	era	10.	argue	15.	platform	20.	chapter

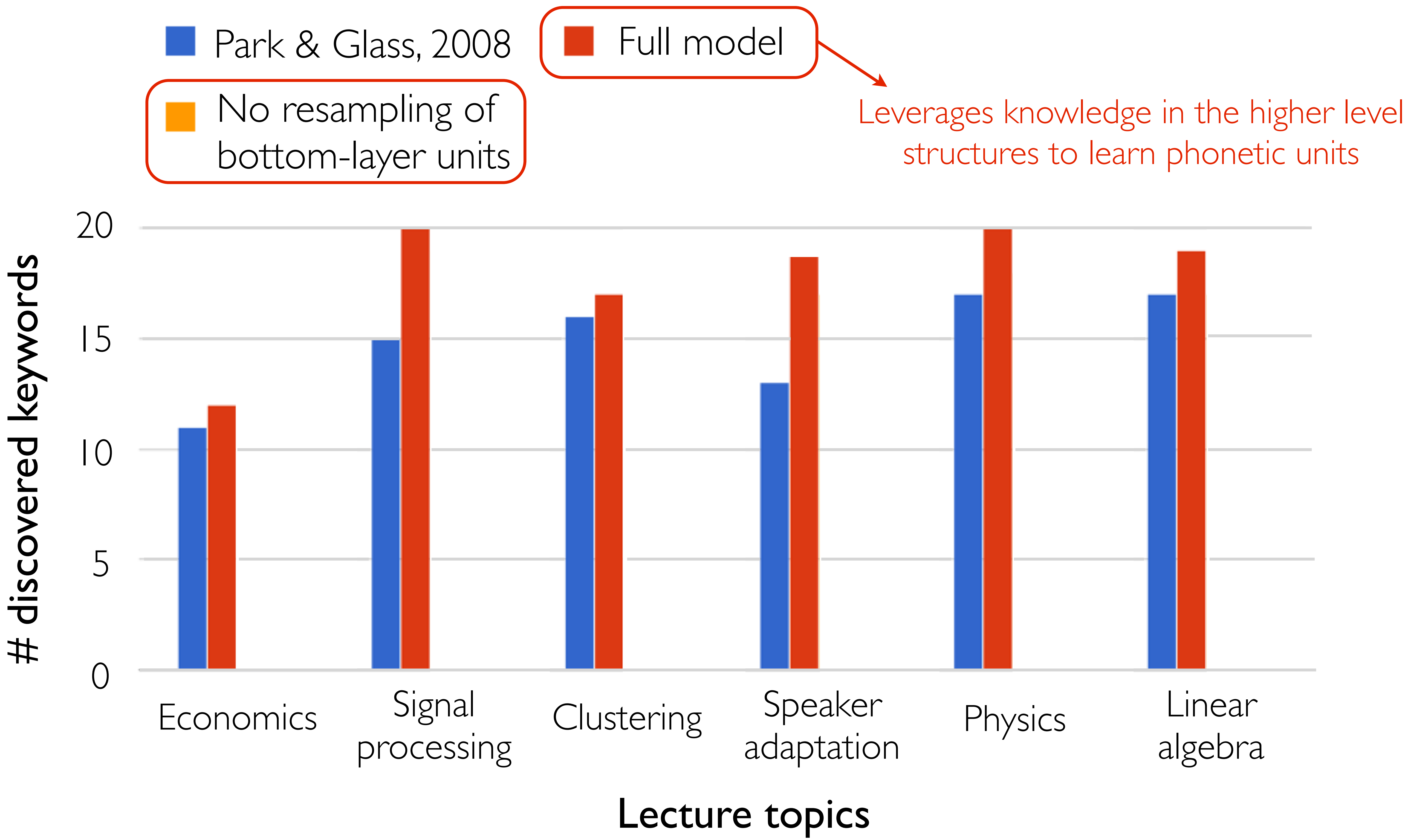
Coverage of Keywords



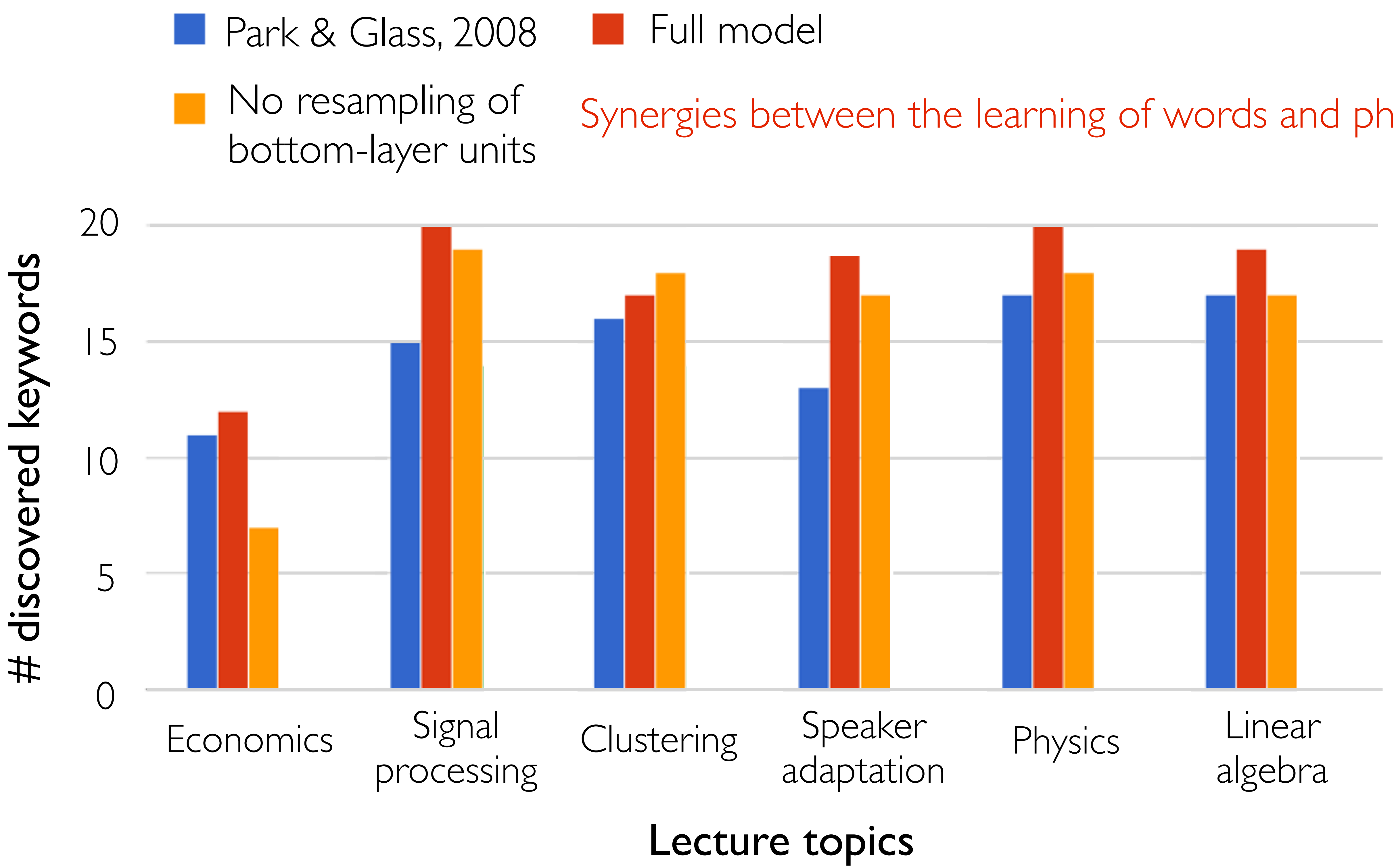
Coverage of Keywords



Coverage of Keywords

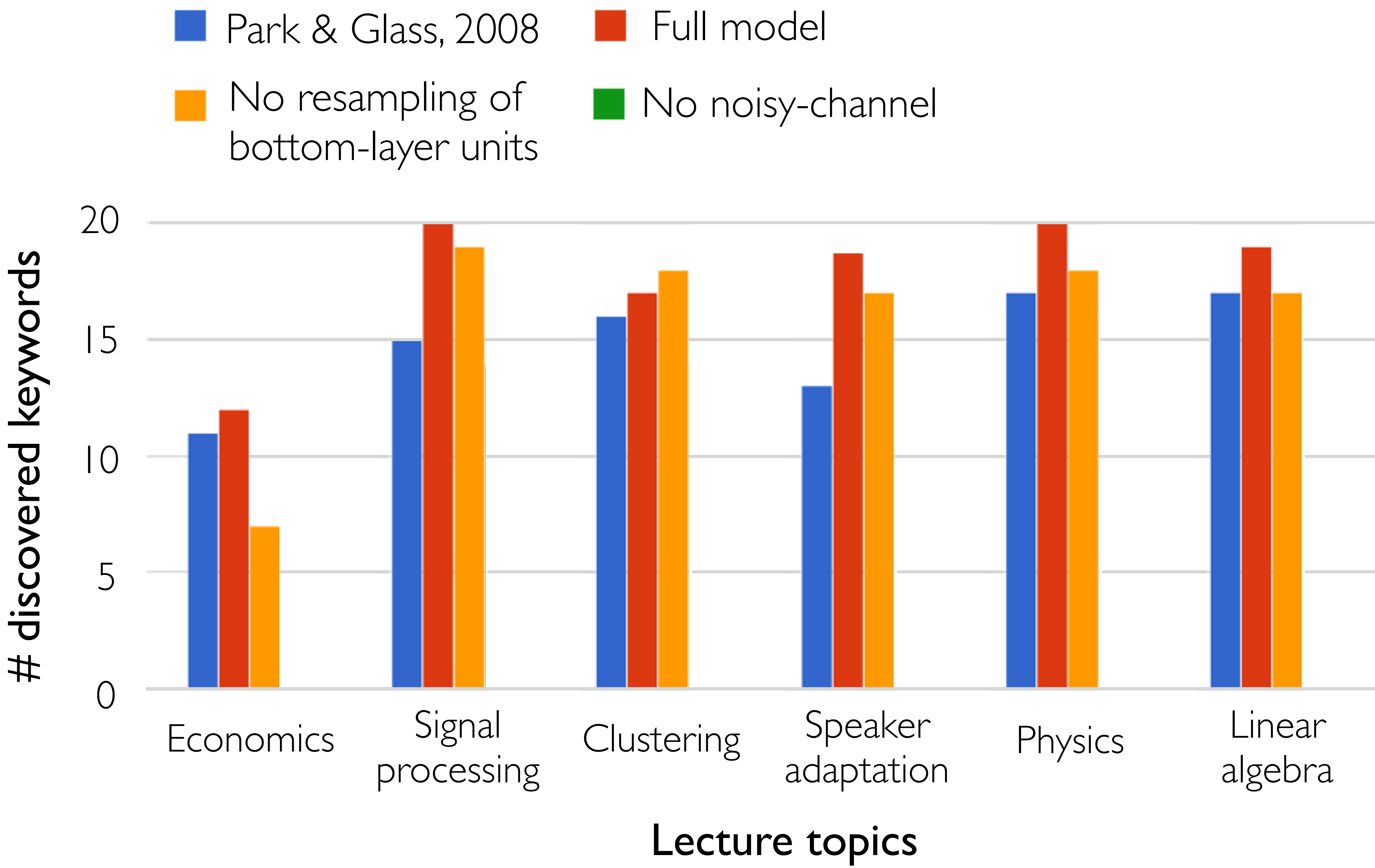


Coverage of Keywords

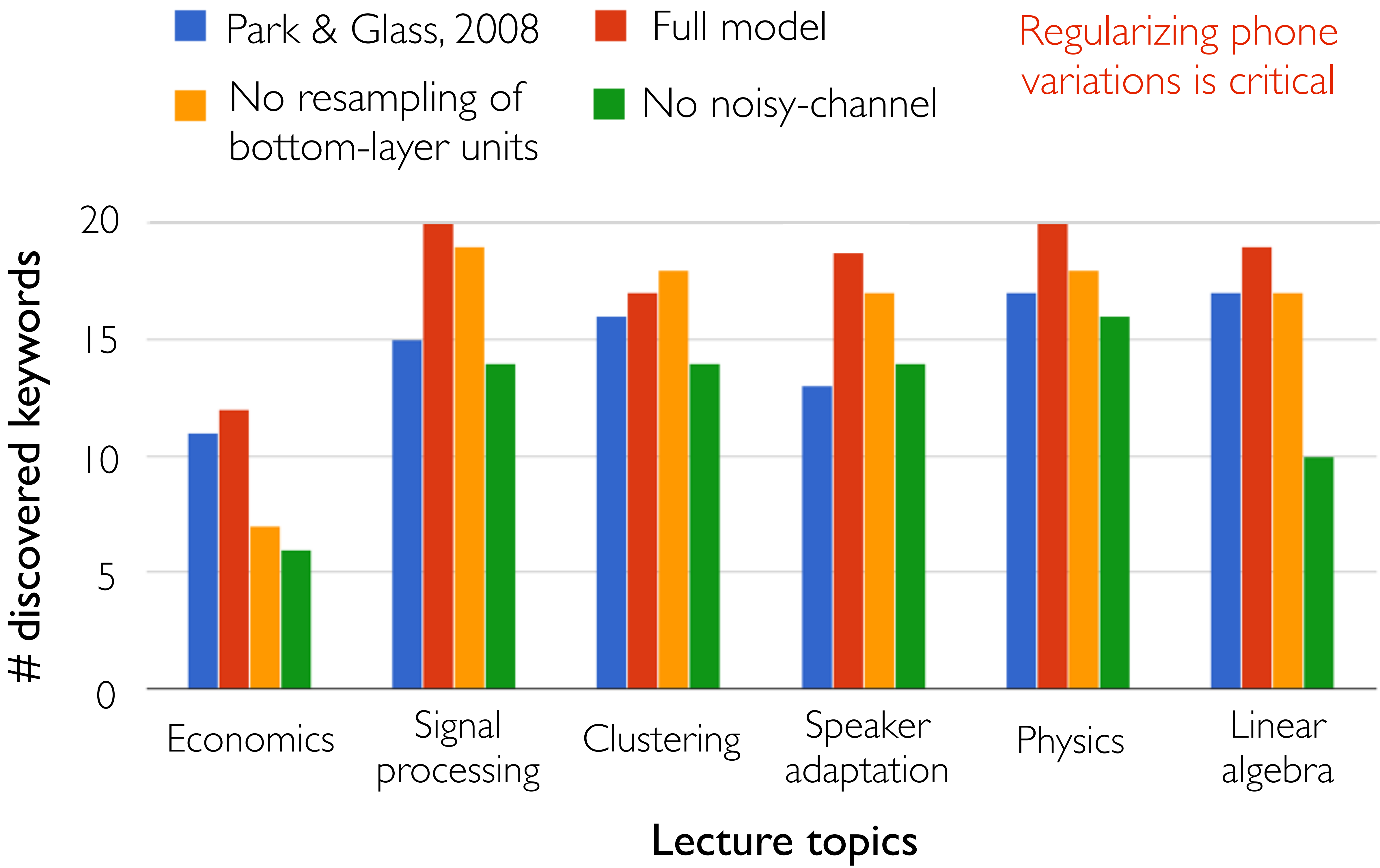


Synergies between the learning of words and phones

Coverage of Keywords



Coverage of Keywords



Conclusion

- Two models for discovering linguistic structures from speech

Conclusion

- Two models for discovering linguistic structures from speech

Discovering phonetic inventory

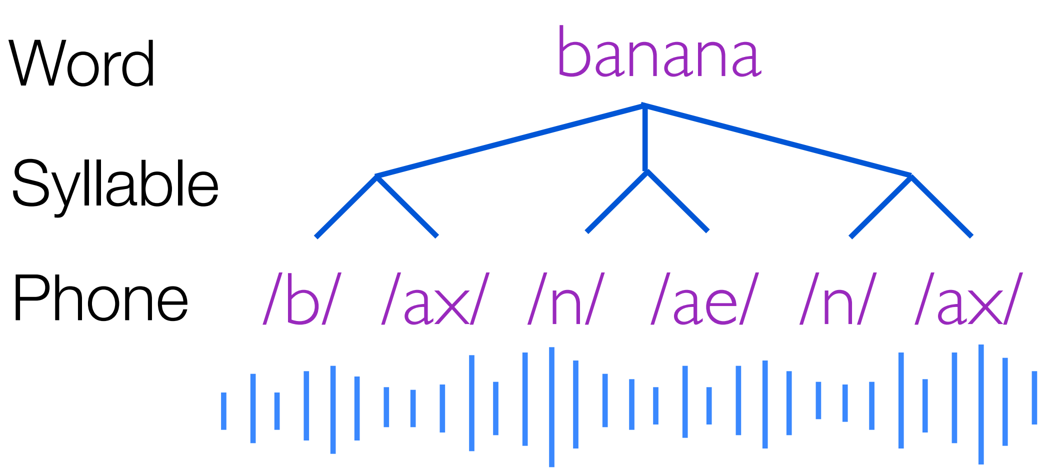
- DP mixture models with HMMs
  - Discovered phonetic units are highly correlated with standard phones

Conclusion

- Two models for discovering linguistic structures from speech

Discovering phonetic inventory

Discovering hierarchical linguistic structures



- DP mixture models with HMMs
  - Discovered phonetic units are highly correlated with standard phones

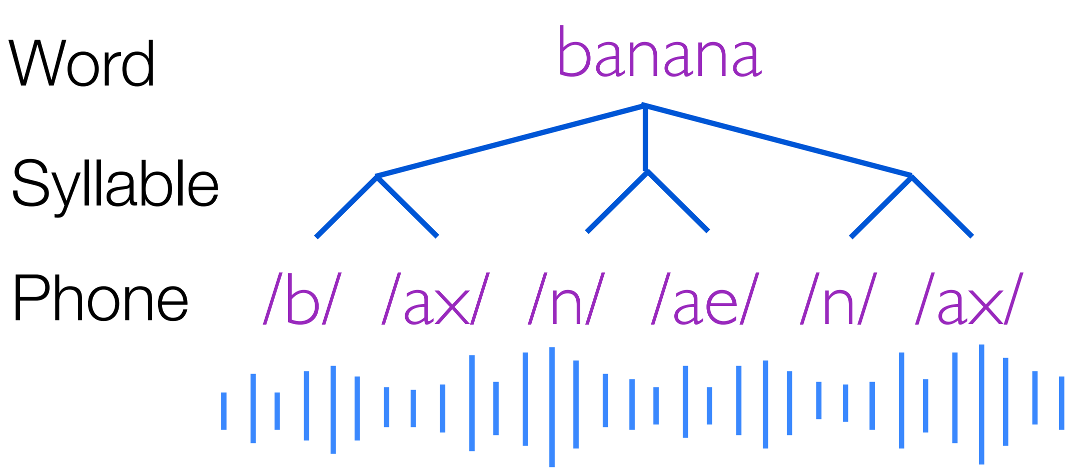
Conclusion

- Two models for discovering linguistic structures from speech

Discovering phonetic inventory

- DP mixture models with HMMs
  - Discovered phonetic units are highly correlated with standard phones

Discovering hierarchical linguistic structures



- Integrate adaptor grammars with the phone discovery model

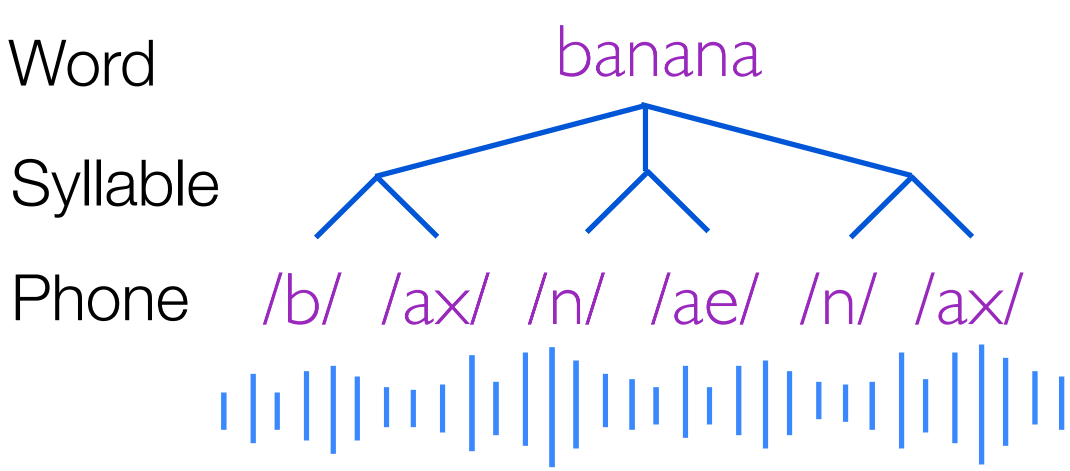
Conclusion

- Two models for discovering linguistic structures from speech

Discovering phonetic inventory

- DP mixture models with HMMs
  - Discovered phonetic units are highly correlated with standard phones

Discovering hierarchical linguistic structures

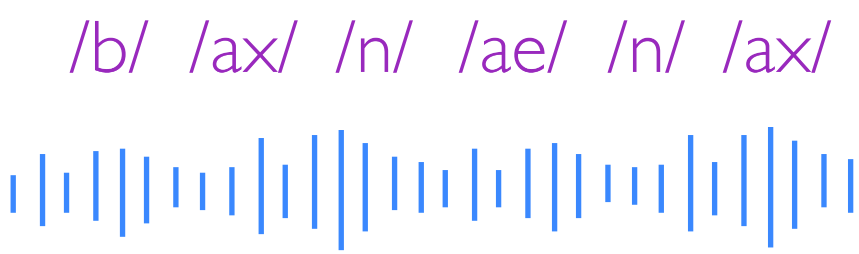


- Integrate adaptor grammars with the phone discovery model
  - Noisy-channel model is critical for learning lexical units

Conclusion

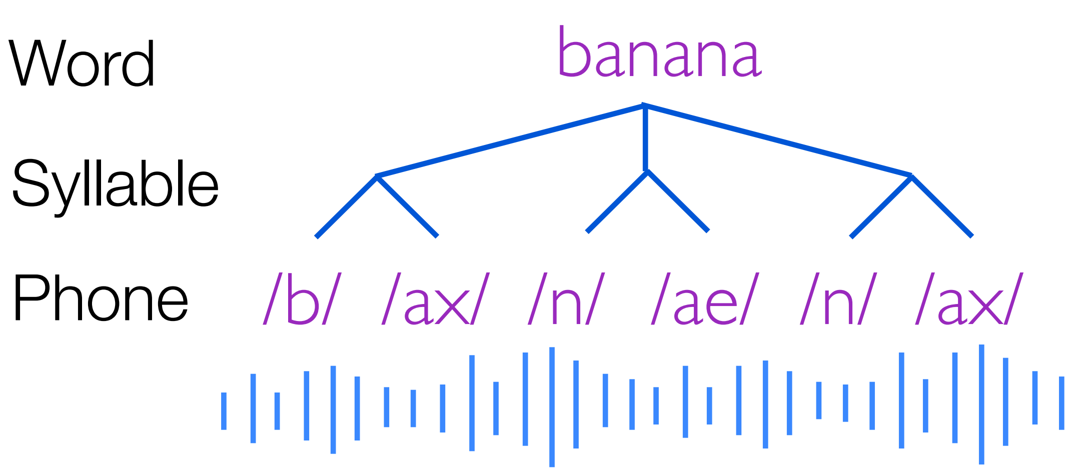
- Two models for discovering linguistic structures from speech

Discovering phonetic inventory



- DP mixture models with HMMs
  - Discovered phonetic units are highly correlated with standard phones

Discovering hierarchical linguistic structures

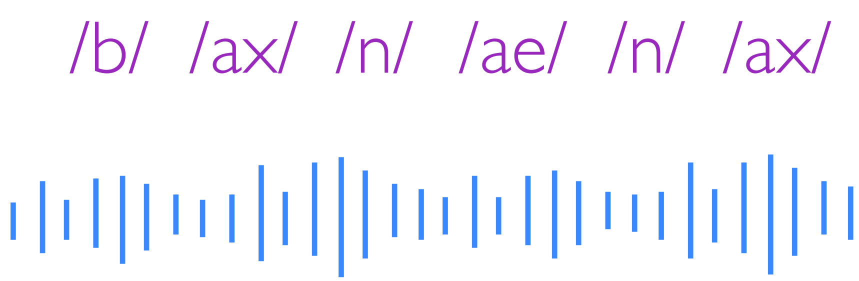


- Integrate adaptor grammars with the phone discovery model
  - Noisy-channel model is critical for learning lexical units
  - Synergies between word and phone learning

Models and Applications

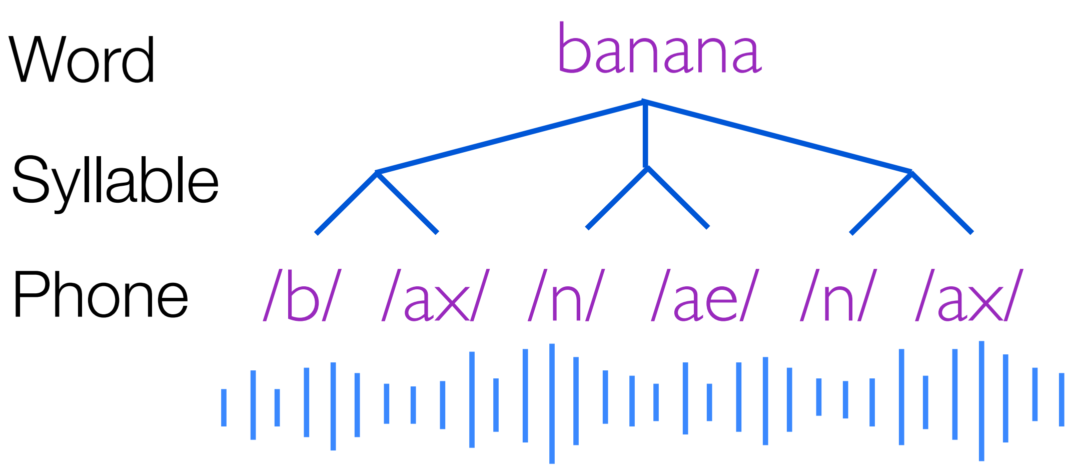
Discovering phonetic inventory

[Lee and Glass, ACL 2012]



Discovering hierarchical linguistic structures

[Lee, O'Donnell, and Glass, TACL 2015]



Models and Applications

Discovering phonetic inventory

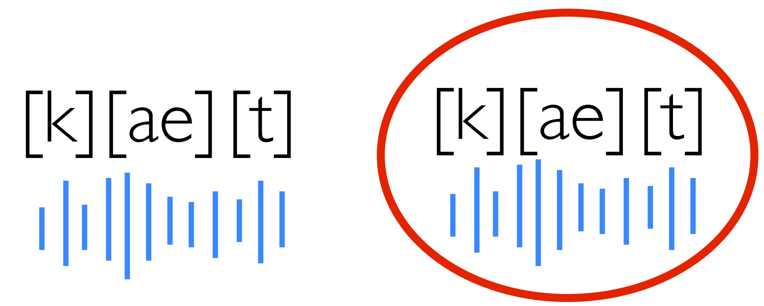
[Lee and Glass, ACL 2012]

/b/ /ax/ /n/ /ae/ /n/ /ax/

One-shot learning spoken words

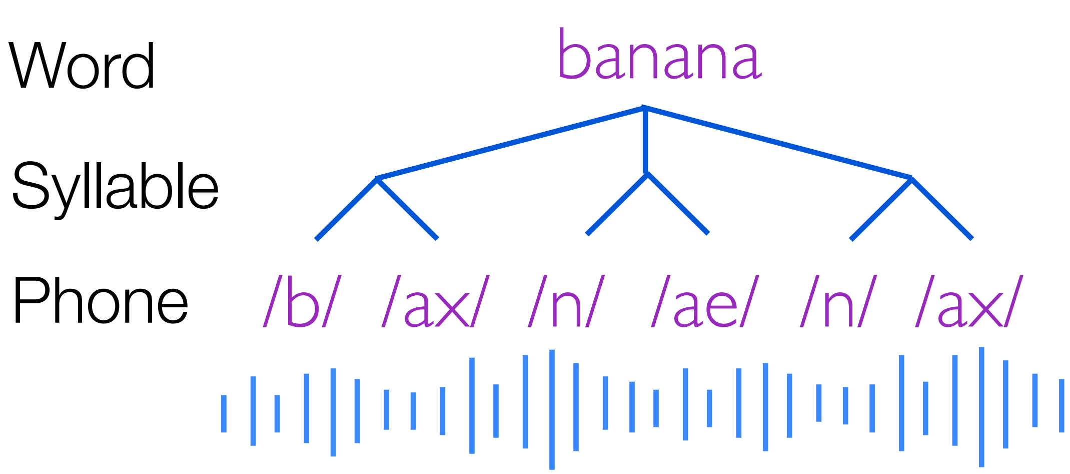
[Lake, Lee*, Glass, Tenenbaum, CogSci 2014]*

** share first authorship*



Discovering hierarchical linguistic structures

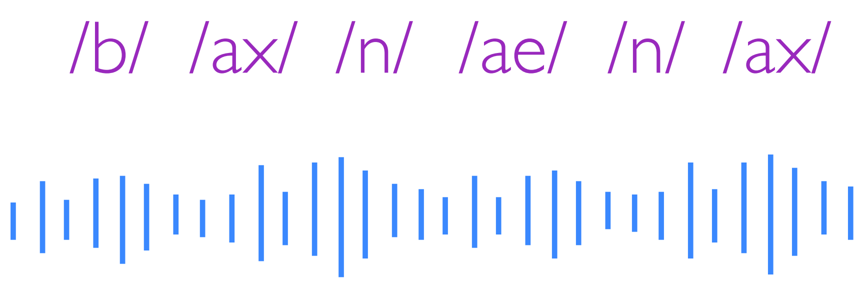
[Lee, O'Donnell, and Glass, TACL 2015]



Models and Applications

Discovering phonetic inventory

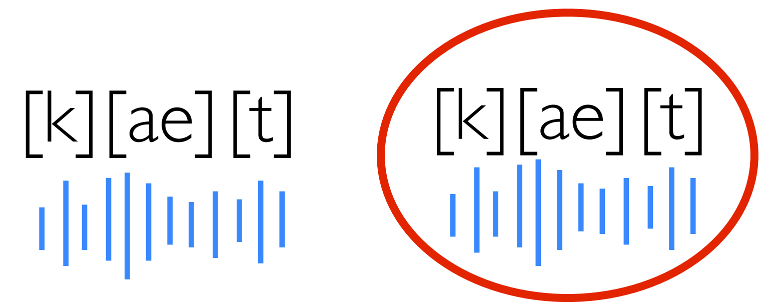
[Lee and Glass, ACL 2012]



One-shot learning spoken words

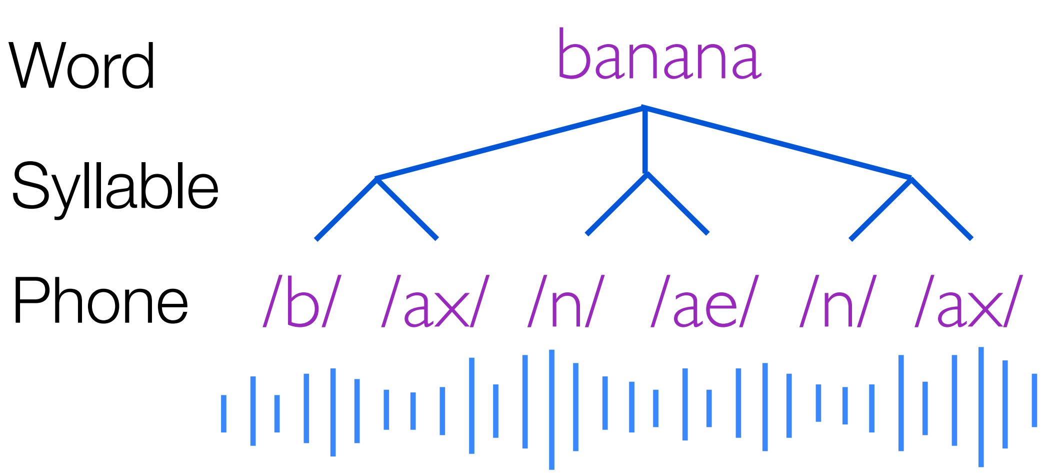
[Lake, Lee*, Glass, Tenenbaum, CogSci 2014]*

** share first authorship*



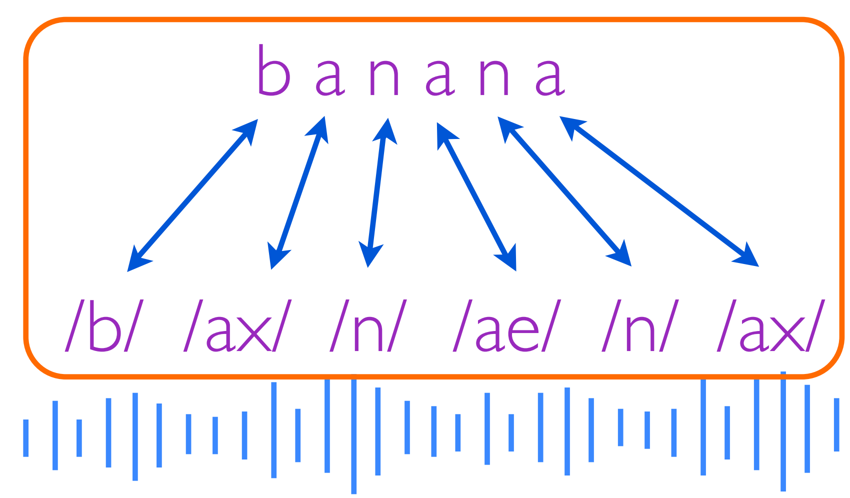
Discovering hierarchical linguistic structures

[Lee, O'Donnell, and Glass, TACL 2015]



Pronunciation lexicon learning

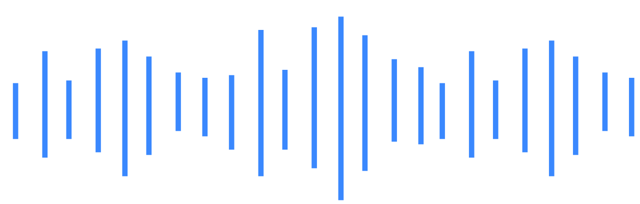
[Lee, Zhang, and Glass, EMNLP 2013]



Future Work

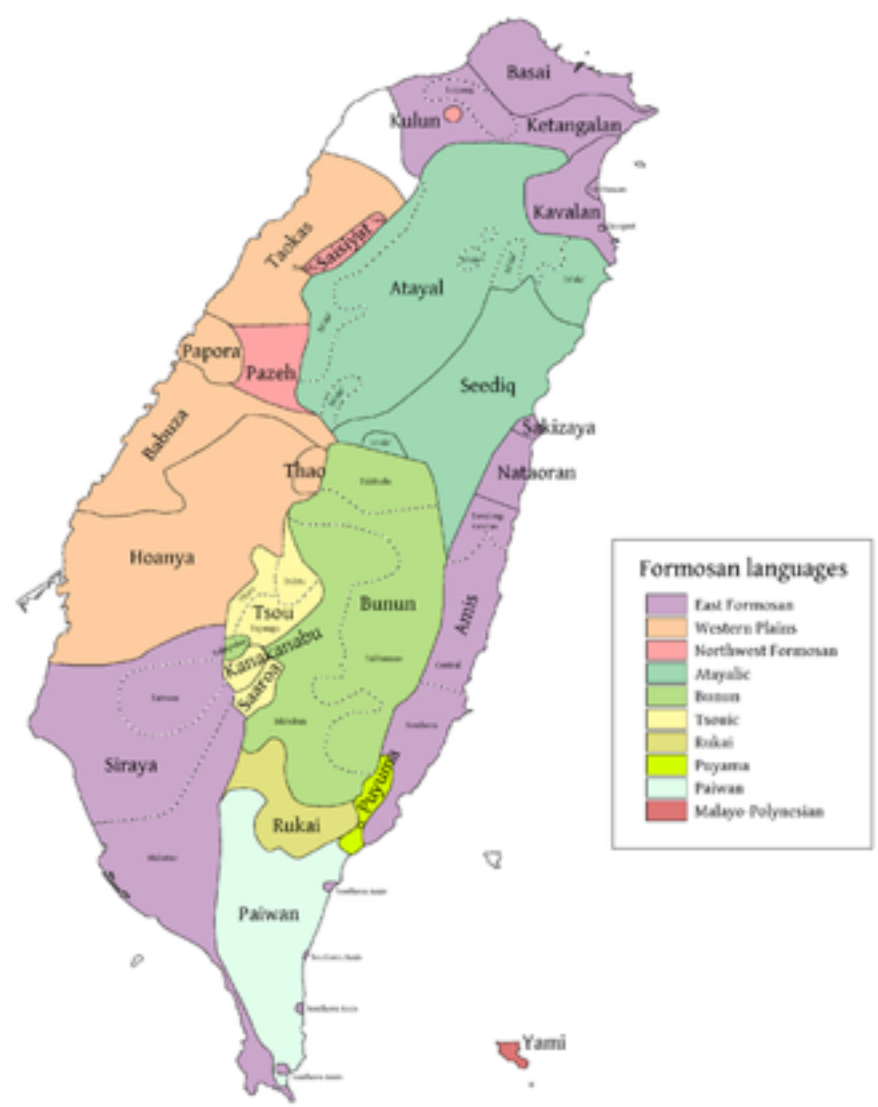
- Learning from more sensory data
  - Speech and visual streams

The doggie is sleeping



Future Work

- **Building spoken language systems based on discovered vocabulary**
  - For low-resource languages or languages without a writing system

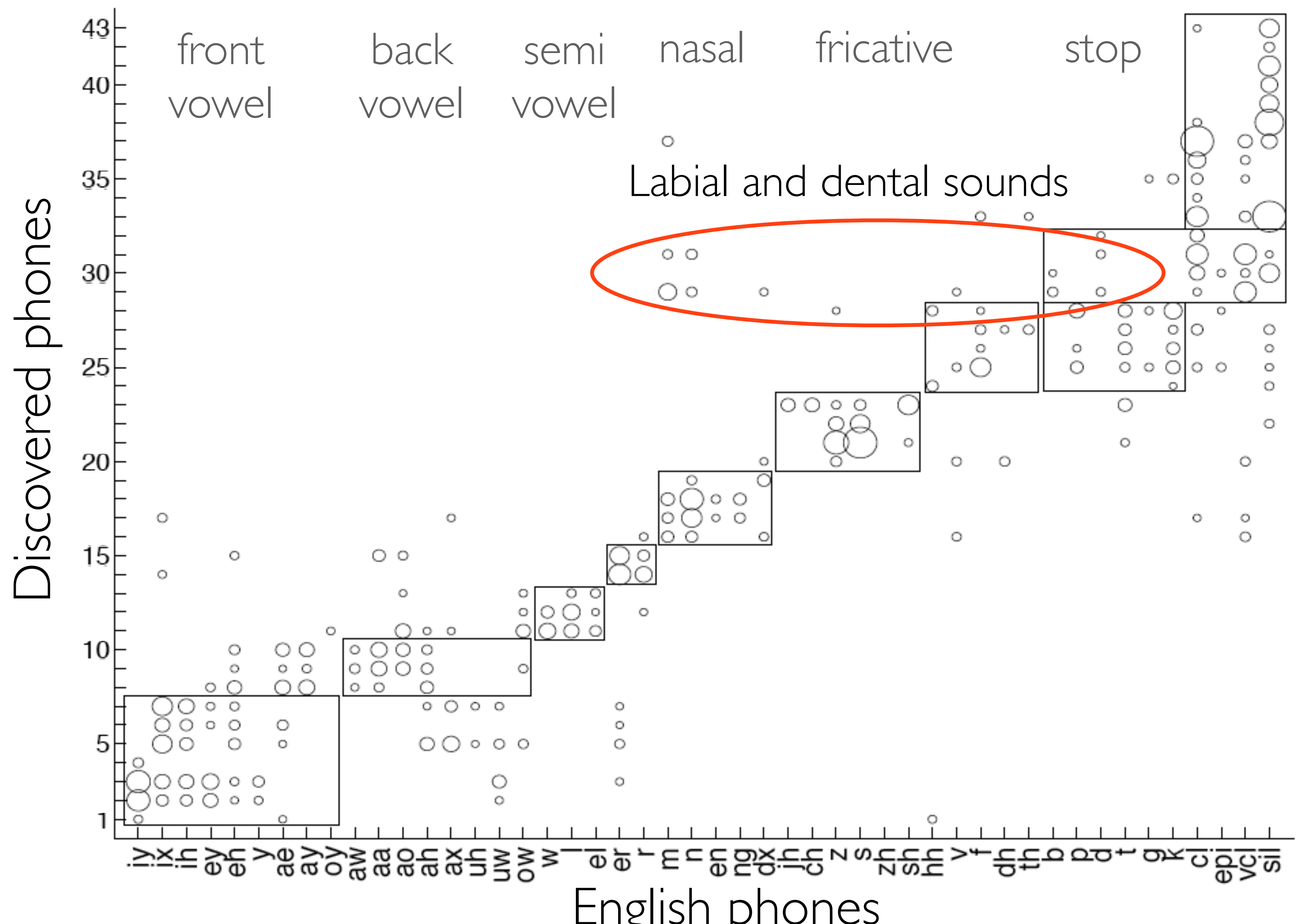


Thank you.

(kite.com)

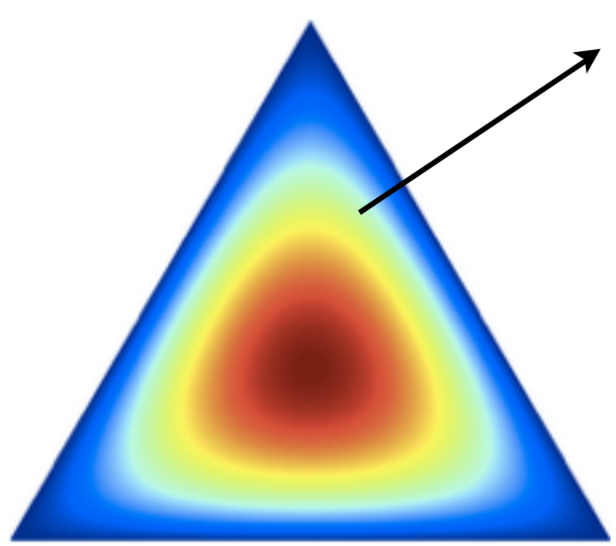
Discovered Phone Units -- 300 utterances

- **43 phone units discovered from 300 TIMIT utterances**
  - Phone units are correlated with English broad phone classes



Dirichlet Process (DP)

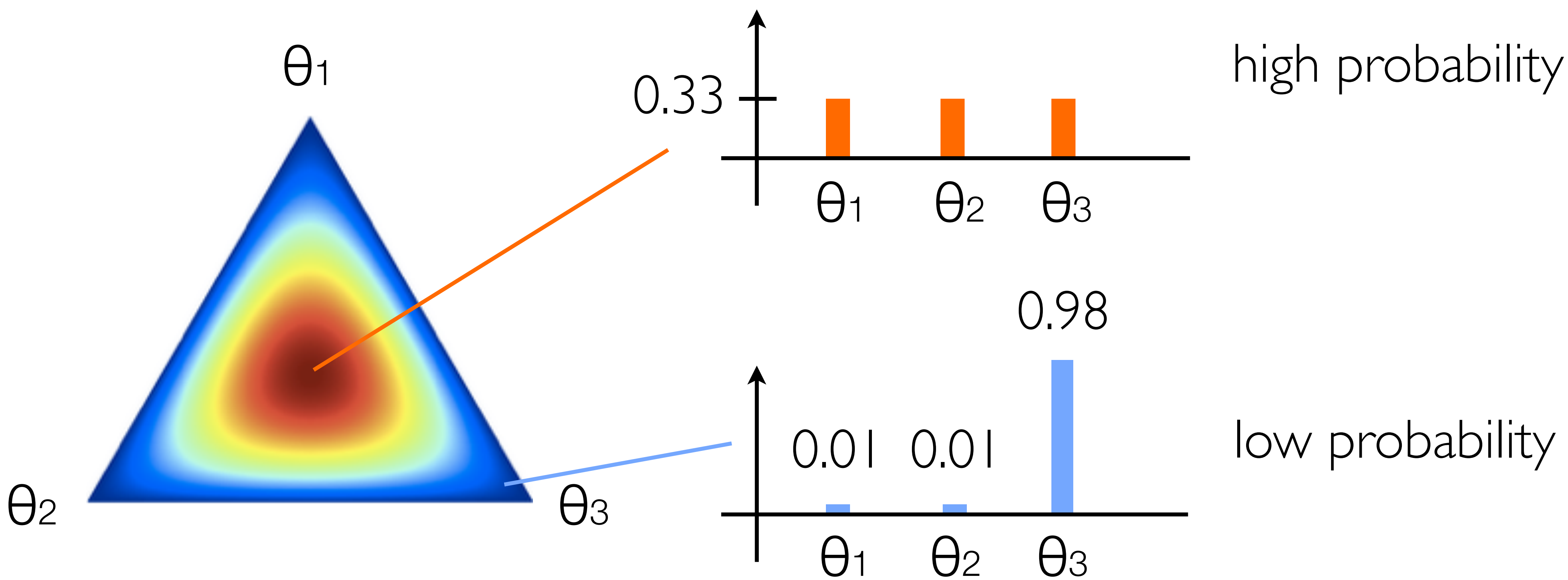
- **Let's start with Dirichlet distribution**
  - Dirichlet distribution is a distribution over the K-dim probability simplex



$$(\alpha_1, \alpha_2, \alpha_3) \quad \sum_{i=1}^{3} \alpha_i = 1$$

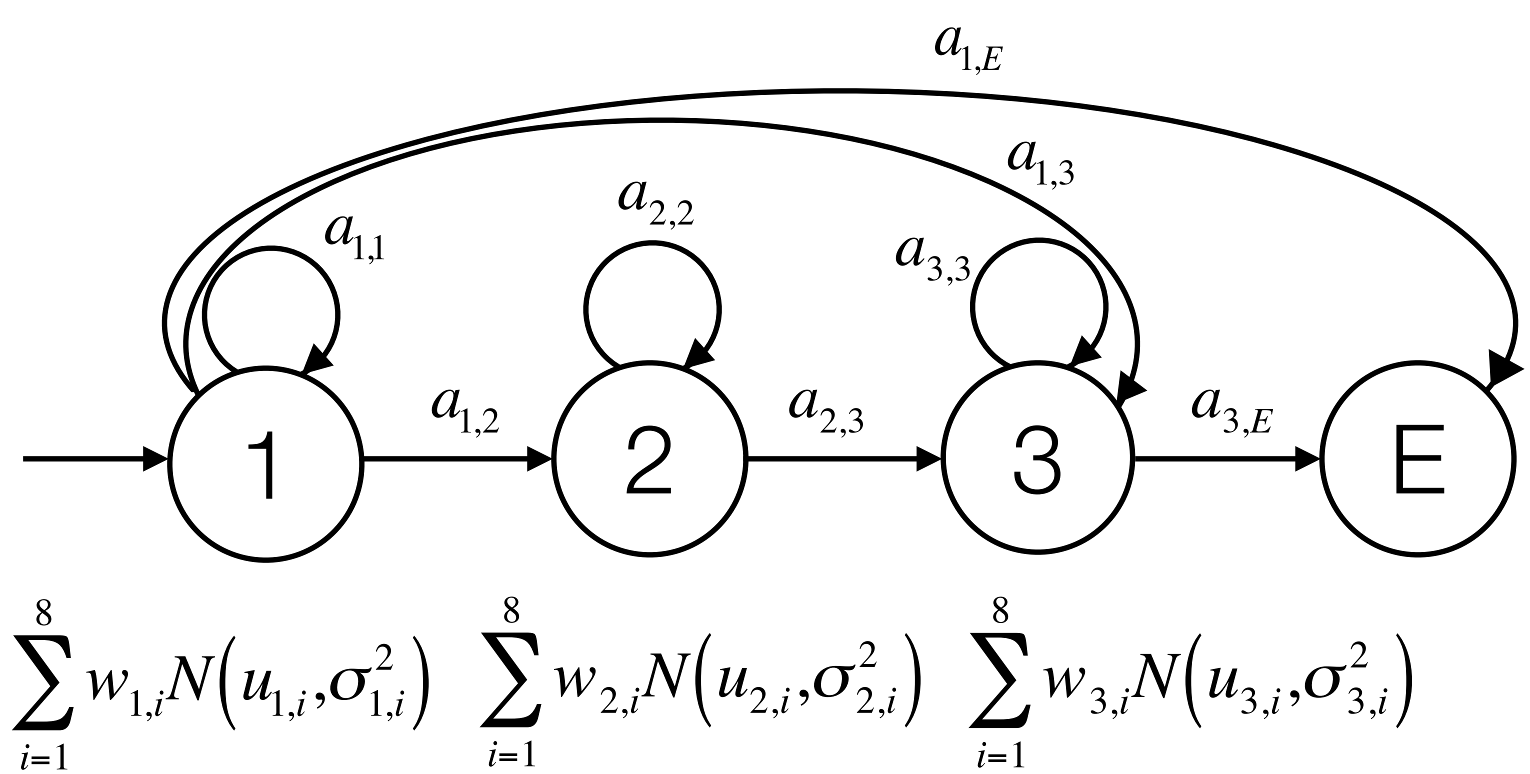
Dirichlet Process (DP)

- **Let's start with Dirichlet distribution**
  - Dirichlet distribution is a distribution over the K-dim probability simplex
  - Assume we have 3 HMMs in the mixture



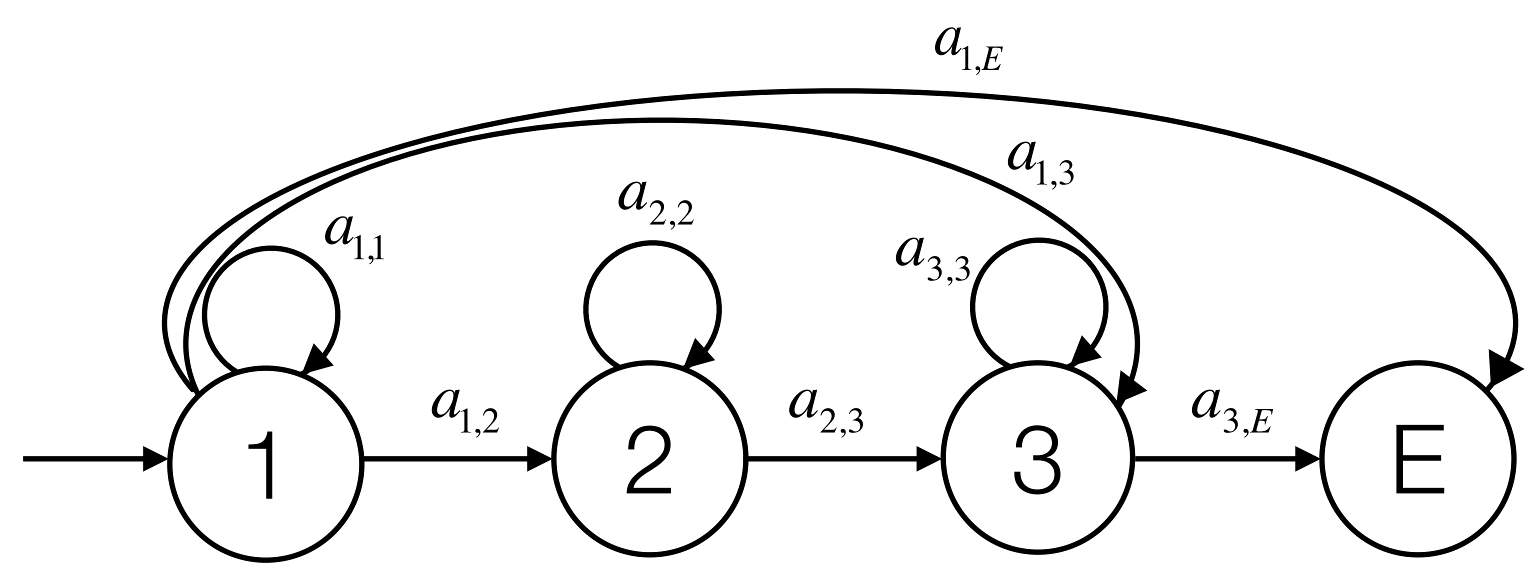
Inference for HMM Parameters (θ)

- **HMM is used to model each phone**
  - Three states with only left-to-right and self transitions
  - Always start from the first state
  - A diagonal GMM is used for the emission distributions



Inference for HMM Parameters (θ)

- **HMM is used to model each phone**
  - Three states with only left-to-right and self transitions
  - Always start from the first state
  - A diagonal GMM is used for the emission distributions
- **Latent variables**
  - Transition probabilities (a)
  - Mixture weights ($\boldsymbol{w}$)
  - Means ($\boldsymbol{\mu}$)
  - Variances ($\boldsymbol{\sigma}^2$)



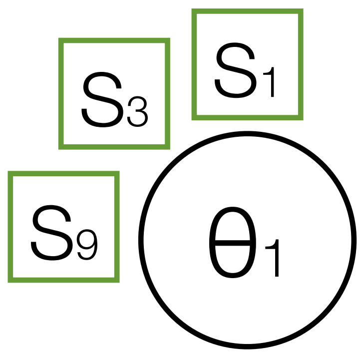
$$\sum_{i=1}^{8} w_{1,i} N\left(u_{1,i}, \sigma_{1,i}^2\right) \quad \sum_{i=1}^{8} w_{2,i} N\left(u_{2,i}, \sigma_{2,i}^2\right) \quad \sum_{i=1}^{8} w_{3,i} N\left(u_{3,i}, \sigma_{3,i}^2\right)$$

Priors and Posteriors for HMM

- Priors
  - Dirichlet distributions for transition probabilities (a) and mixture weights ($\boldsymbol{w}$)
  - Normal-gamma distributions for Gaussian parameters (μ, σ^2)

Priors and Posteriors for HMM

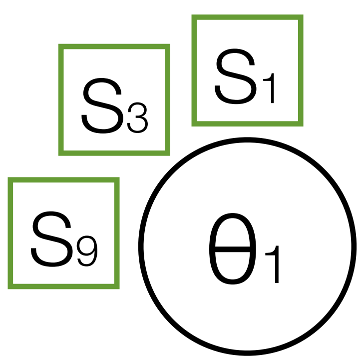
- Priors
  - Dirichlet distributions for transition probabilities (a) and mixture weights ($\boldsymbol{w}$)
  - Normal-gamma distributions for Gaussian parameters ($\boldsymbol{\mu}$, $\boldsymbol{\sigma}^2$)
- Posteriors
  - Gather relevant counts from customer segments



Priors and Posteriors for HMM

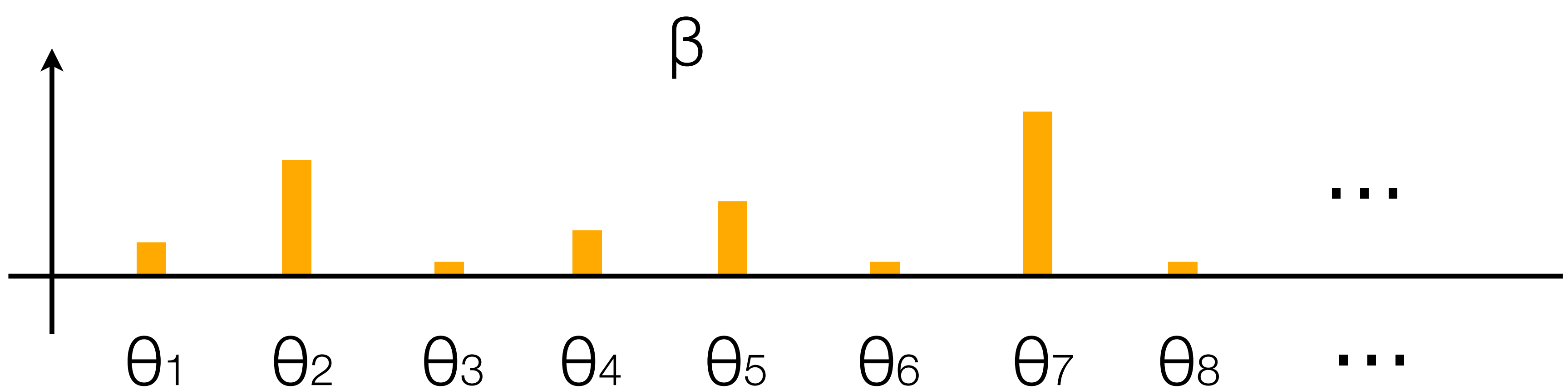
- Priors
  - Dirichlet distributions for transition probabilities (a) and mixture weights (w)
  - Normal-gamma distributions for Gaussian parameters (μ, σ^2)

- Posteriors
  - Gather relevant counts from customer segments
  - Update prior distributions
  - Sample new values for the latent variables



Dirichlet Process (DP)

- Conceptually
  - Dirichlet process can be viewed as an infinite case of Dirichlet distribution



- Unknown # of HMMs
  - Assume there are infinite number of HMMs first
  - Infer the finite number of HMM are needed to explain the finite data
  - By integrating β during inference, DP provides a nice math format to find the #

PCFG Review

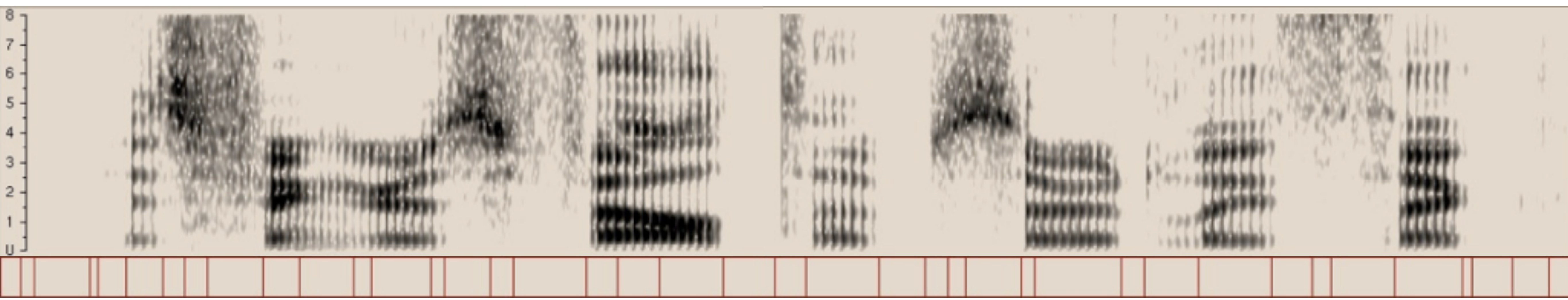
- A PCFG is a quintuple $(N, T, S, R, \{\pi^q\}_{q \in N})$
- N : a finite set of <u>nonterminal</u> symbols
- T : a finite set of <u>terminal</u> symbols
  - $N \cap T = \varnothing$
- S : start symbol
  - $S \in N$
- R : production rules
  - $R = \{N \rightarrow (N \cup T)^*\}$
- π^q : rule probabilities
  - $q \in N$

PCFG

0.5	Sen	$\longrightarrow$	Word Word
0.5	Sen	$\longrightarrow$	Word
0.7	Word	$\longrightarrow$	Syl Syl
0.3	Word	$\longrightarrow$	Syl
0.6	Syl	$\longrightarrow$	Phn Phn
0.4	Syl	$\longrightarrow$	Phn
0.1	Phn	$\longrightarrow$	/ax/
0.05	Phn	$\longrightarrow$	/n/
		...	

Acoustic Landmarks

- Naively, every frame can be a phone boundary
  - In fact, some frames are more likely to be boundaries and some are less likely
  - Compute landmarks [Glass et al. 2003] and only do inference on landmarks
  - A language-independent method



- Disadvantage
  - Put an upper bound on recall rate
- Advantage
  - Reduce inference load

Spoken Term Detection

- **Given a spoken query (w), find all spoken documents that contain w**
  - 3696 utterances for discovering phone units
  - Compute posterior-grams on the HMM states of the discovered phone units

Spoken Term Detection

- **Given a spoken query (w), find all spoken documents that contain w**
  - 3696 utterances for discovering phone units
  - Compute posterior-grams on the HMM states of the discovered phone units

x : a single frame of feature vector

$State_{i,j}$: the j-th state of the i-th HMM

Spoken Term Detection

- **Given a spoken query (w), find all spoken documents that contain w**
  - 3696 utterances for discovering phone units
  - Compute posterior-grams on the HMM states of the discovered phone units

x : a single frame of feature vector

$State_{i,j}$: the j-th state of the i-th HMM

$$\text{posterior-gram}(x) = \left[\frac{p(State_{i,j} \mid x)}{\sum_{i=1}^{K} \sum_{j=1}^{3} p(State_{i,j} \mid x)} \right] \text{ for } 1 \le i \le K \text{ and } 1 \le j \le 3$$

K : the total number of HMMs

Spoken Term Detection

- **Given a spoken query (w), find all spoken documents that contain w**
  - 3696 utterances for discovering phone units
  - Compute posterior-grams on the HMM states of the discovered phone units
  - Apply dynamic time warping to keyword detection [Zhang et al, 2009]

Spoken Term Detection

- **Given a spoken query (w), find all spoken documents that contain w**
  - 3696 utterances for discovering phone units
  - Compute posterior-grams on the HMM states of the discovered phone units
  - Apply dynamic time warping to keyword detection [Zhang et al, 2009]
  - 10 selected keywords

Spoken Term Detection

- **Given a spoken query (w), find all spoken documents that contain w**
  - 3696 utterances for discovering phone units
  - Compute posterior-grams on the HMM states of the discovered phone units
  - Apply dynamic time warping to keyword detection [Zhang et al, 2009]
  - 10 selected keywords

P@N: the average precision of top N hits

P@N	EER

Spoken Term Detection

- **Given a spoken query (w), find all spoken documents that contain w**
  - 3696 utterances for discovering phone units
  - Compute posterior-grams on the HMM states of the discovered phone units
  - Apply dynamic time warping to keyword detection [Zhang et al, 2009]
  - 10 selected keywords

P@N: the average precision of top N hits

	P@N	EER
English Monophone (Supervised)	74	11.8
Thai Monophone Model (Supervised)	56.6	14.9
Our model	63	16.9

Spoken Term Detection

- Given a spoken query (w), find all spoken documents that contain w
  - 3696 utterances for discovering phone units
  - Compute posterior-grams on the HMM states of the discovered phone units
  - Apply dynamic time warping to keyword detection [Zhang et al, 2009]
  - 10 selected keywords

P@N: the average precision of top N hits

	P@N	EER
English Monophone (Supervised)	74	11.8
Thai Monophone Model (Supervised)	56.6	14.9
Our model	63	16.9
Zhang 2009 (GMM) (Unsupervised)	52.5	16.4
Zhang 2012 (DBM) (Unsupervised)	51.1	14.7

Unknown Number of HMMs

- An unknown set of phone units

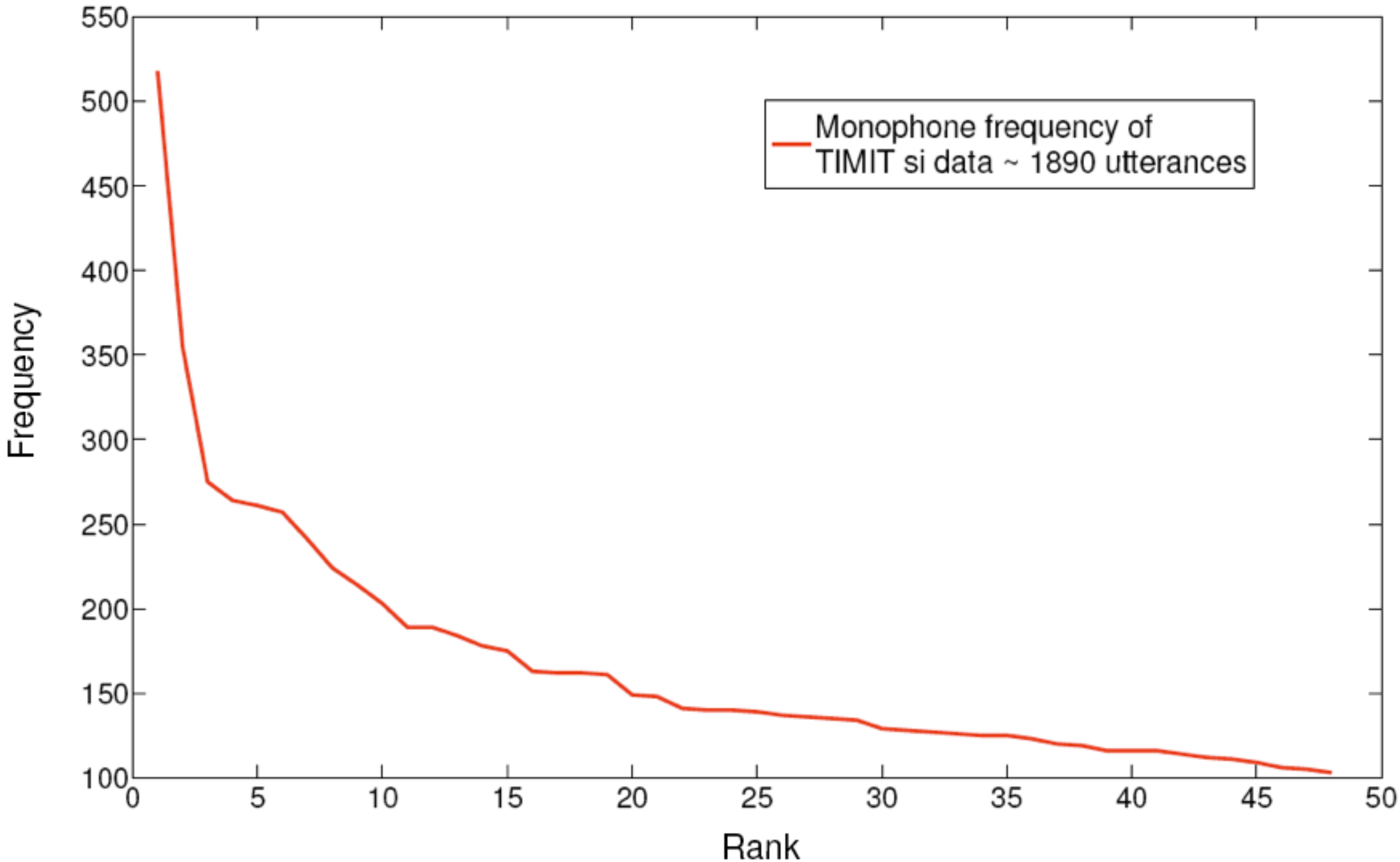
Unknown Number of HMMs

- **An unknown set of phone units**
  - Impose a Dirichlet Process prior to infer the number of phones

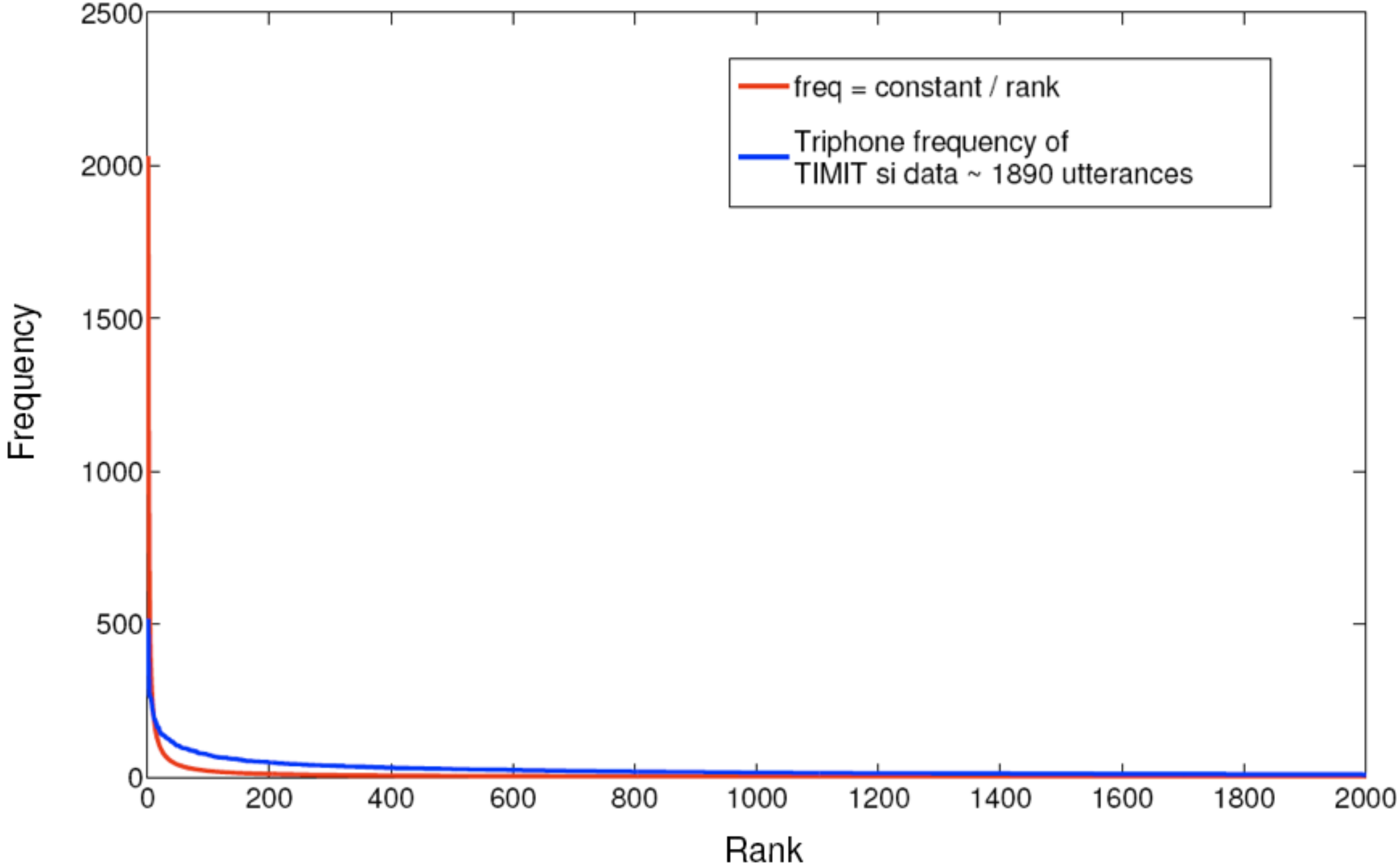
Unknown Number of HMMs

- **An unknown set of phone units**
  - Impose a Dirichlet Process prior to infer the number of phones
- **Is Dirichlet process (DP) a proper prior for this task?**
  - Does phone frequency inherit power law?

Phone Frequency -- Monophone



Phone Frequency -- Triphone



Unknown Number of HMMs

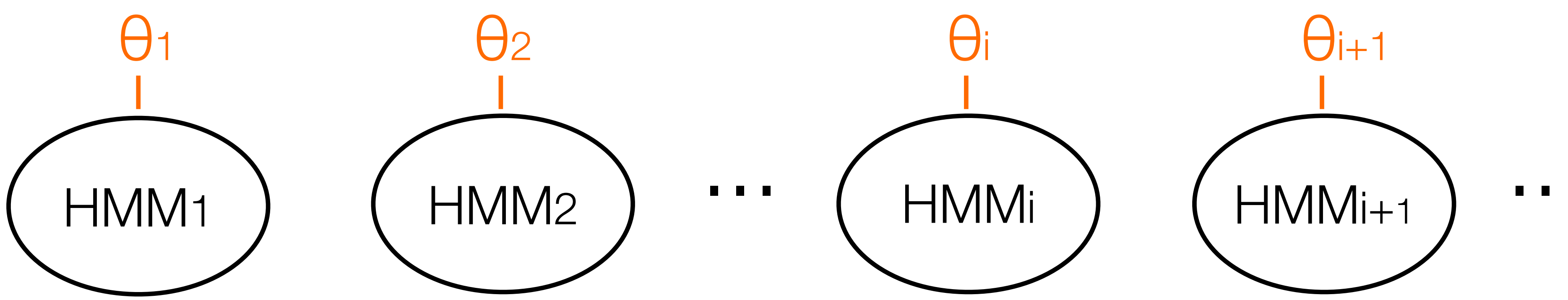
- An unknown set of phone units
  - Impose a Dirichlet Process prior to infer the number of phones
- Is Dirichlet process (DP) a proper prior for this task?
  - Does phone frequency inherit power law?

Unknown Number of HMMs

- **An unknown set of phone units**
  - Impose a Dirichlet Process prior to infer the number of phones
- **Is Dirichlet process (DP) a proper prior for this task?**
  - Does phone frequency inherit power law?
  - DP should be a reasonable prior to start with

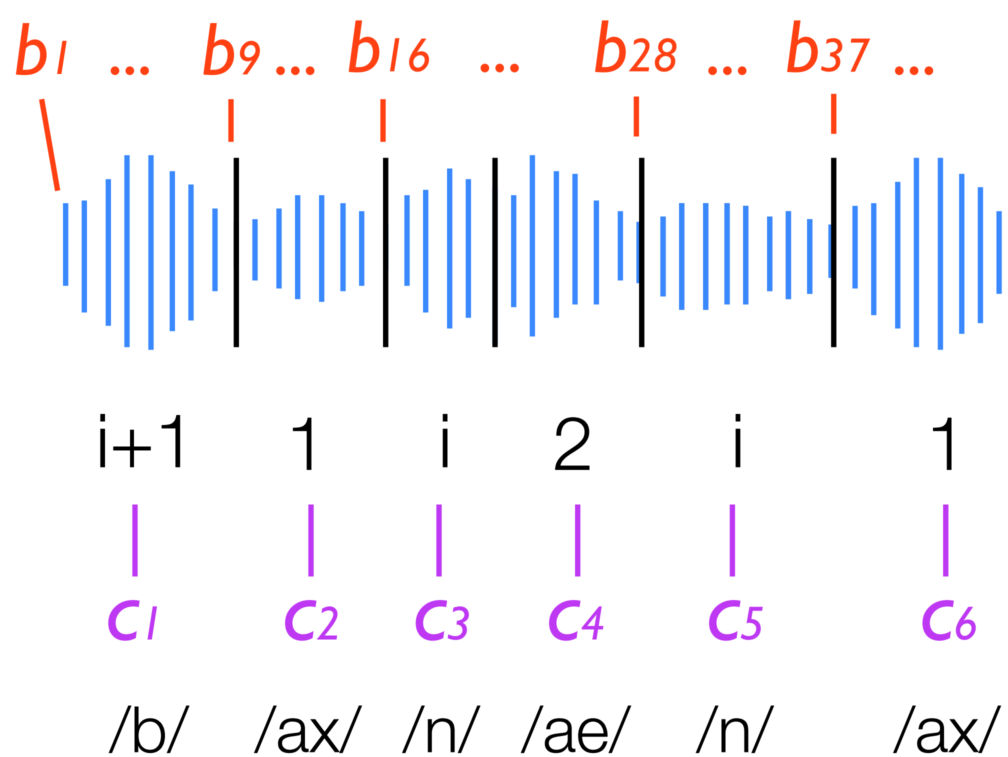
Generative Story

- A simple explanation of how a spoken utterance is generated



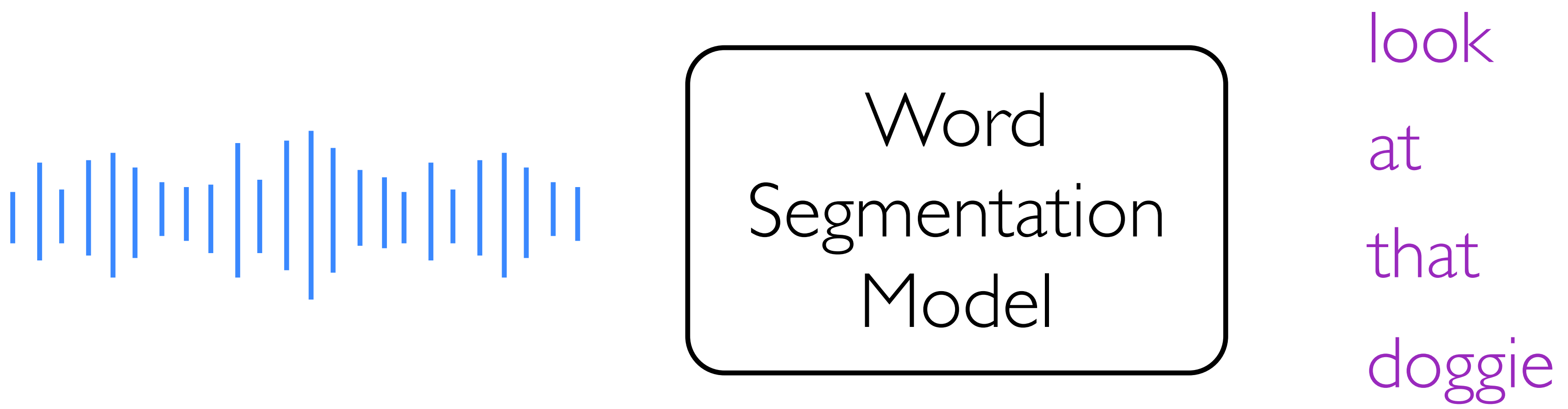
- Main latent variables
  - Phone boundaries (***b***)
  - Phone labels (***c***)
  - HMM parameters (**θ**)
  - # of HMMs (phones)

Dirichlet Process



Language Acquisition Modeling

- Previous work relies on highly pre-processed input data



Language Acquisition Modeling

- Previous work relies on highly pre-processed input data

l uh k ae t dh ae t d ao g iy

Word Segmentation Model

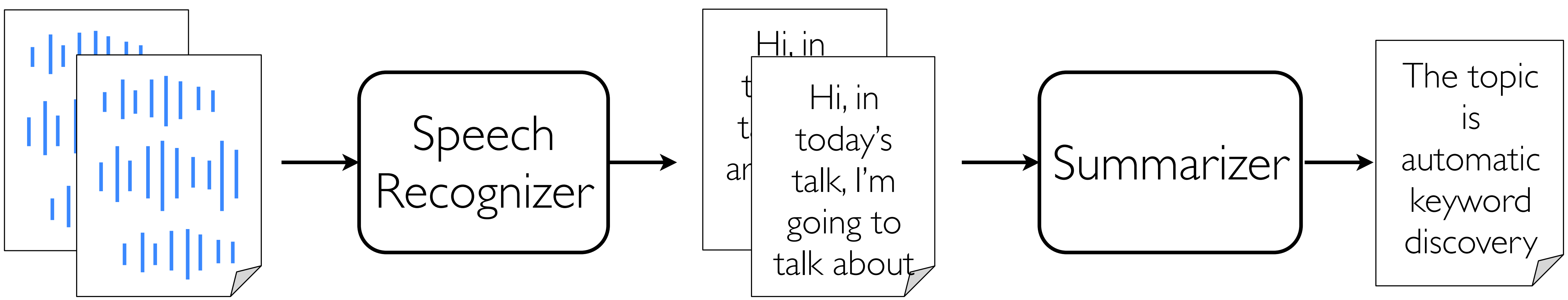
look	(l uh k)
at	(ae t)
that	(dh ae t)
doggie	(d ao g iy)

Other tasks such as phonetic unit learning are ignored

- Ground language acquisition modeling in real sensory data
- Ultimately allow machines to acquire a language like humans

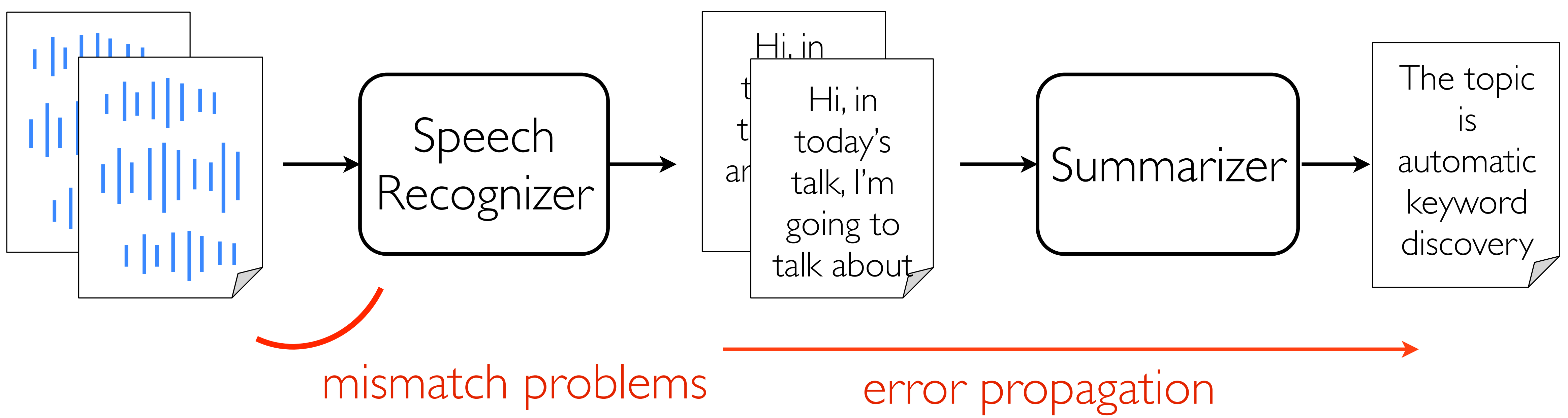
Discovering Structures Beyond Phones

- Useful for representing out-of-vocabulary words
- Spoken document summarization



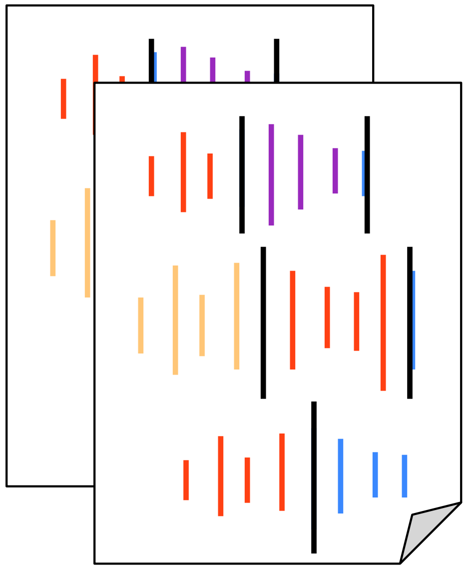
Discovering Structures Beyond Phones

- Useful for representing out-of-vocabulary words
- Spoken document summarization



Discovering Structures Beyond Phones

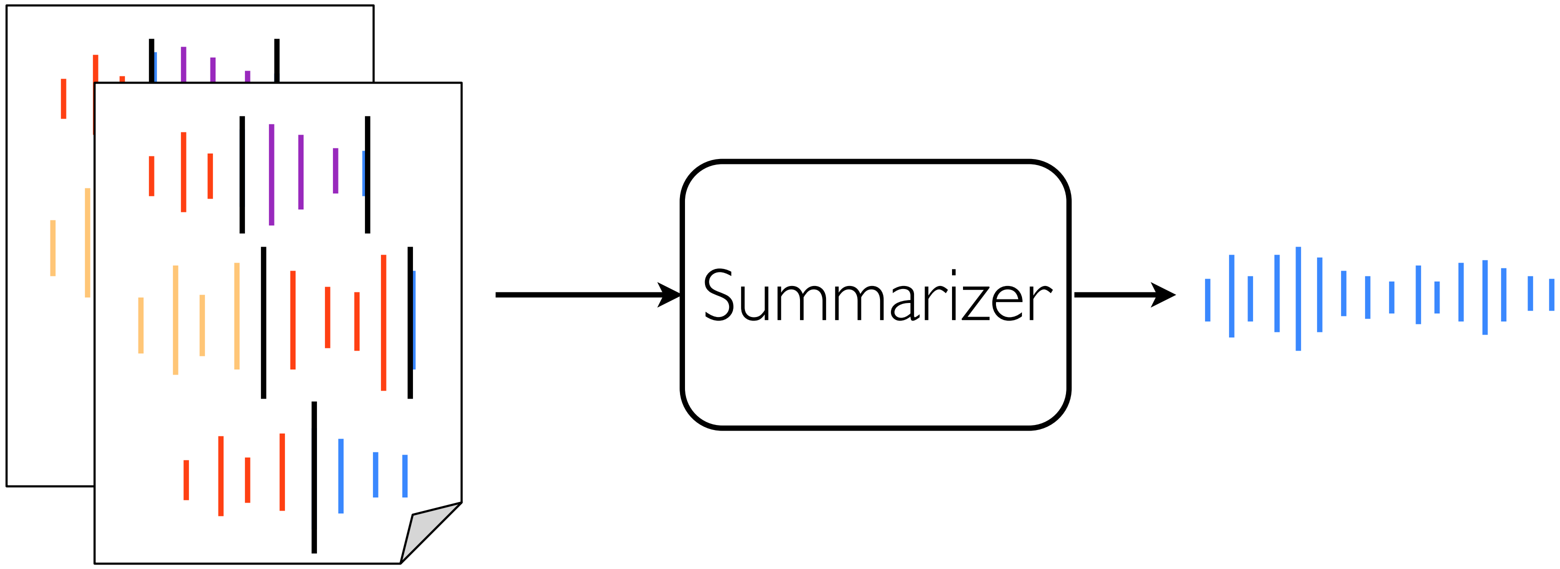
- Sub-word units are useful for representing out-of-vocabulary words
- Unsupervised word discovery
  - Automatic spoken document summarization without speech recognition



latent word structures

Discovering Structures Beyond Phones

- Sub-word units are useful for representing out-of-vocabulary words
- Unsupervised word discovery
  - Automatic spoken document summarization without speech recognition



- Connection to Cognitive Science (CogSci)
  - Computational models for learning from speech are of great interests in CogSci

Model Overview

- **Integrate adaptor grammars and the phone discovery model**
  - To discover rich linguistic structures from speech
- **Three components in the model**

Adaptor grammars

Noisy-channel model

Phonetic discovery model

Discover hierarchical linguistic structures (Words, Syllables etc)

Model Overview

- **Integrate adaptor grammars and the phone discovery model**
  - To discover rich linguistic structures from speech
- **Three components in the model**

Adaptor grammars

Noisy-channel model

Phonetic discovery model — Discover the phonetic units from acoustic data

Model Overview

- Integrate adaptor grammars and the phone discovery model
  - To discover rich linguistic structures from speech
- Three components in the model

Adaptor grammars

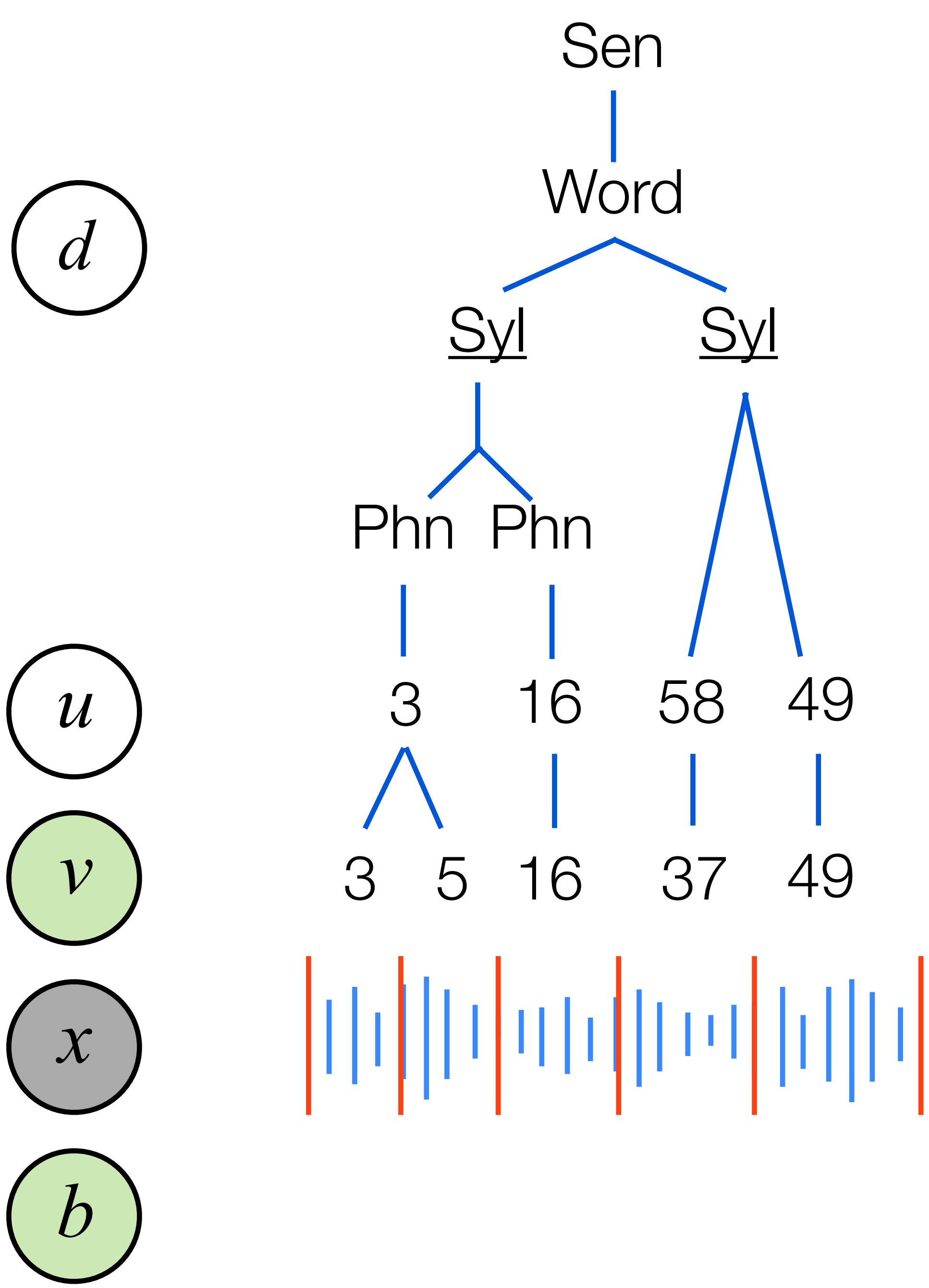
Noisy-channel model

Bridges the other two components

Phonetic discovery model

Inference

- Given d and u



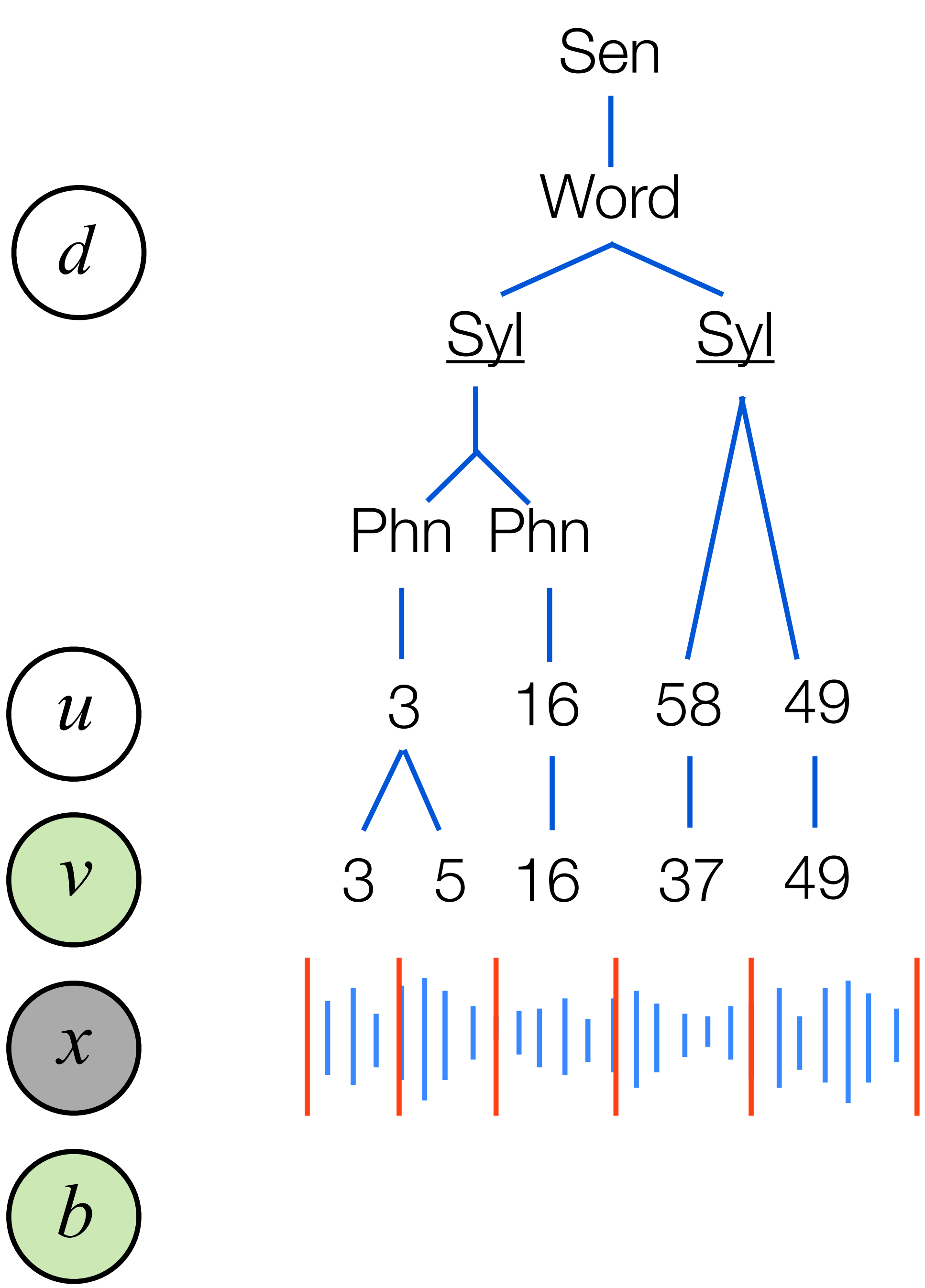
Adaptor grammars

Noisy-channel model

Phonetic discovery model

Inference

- Given d and u resample v and b

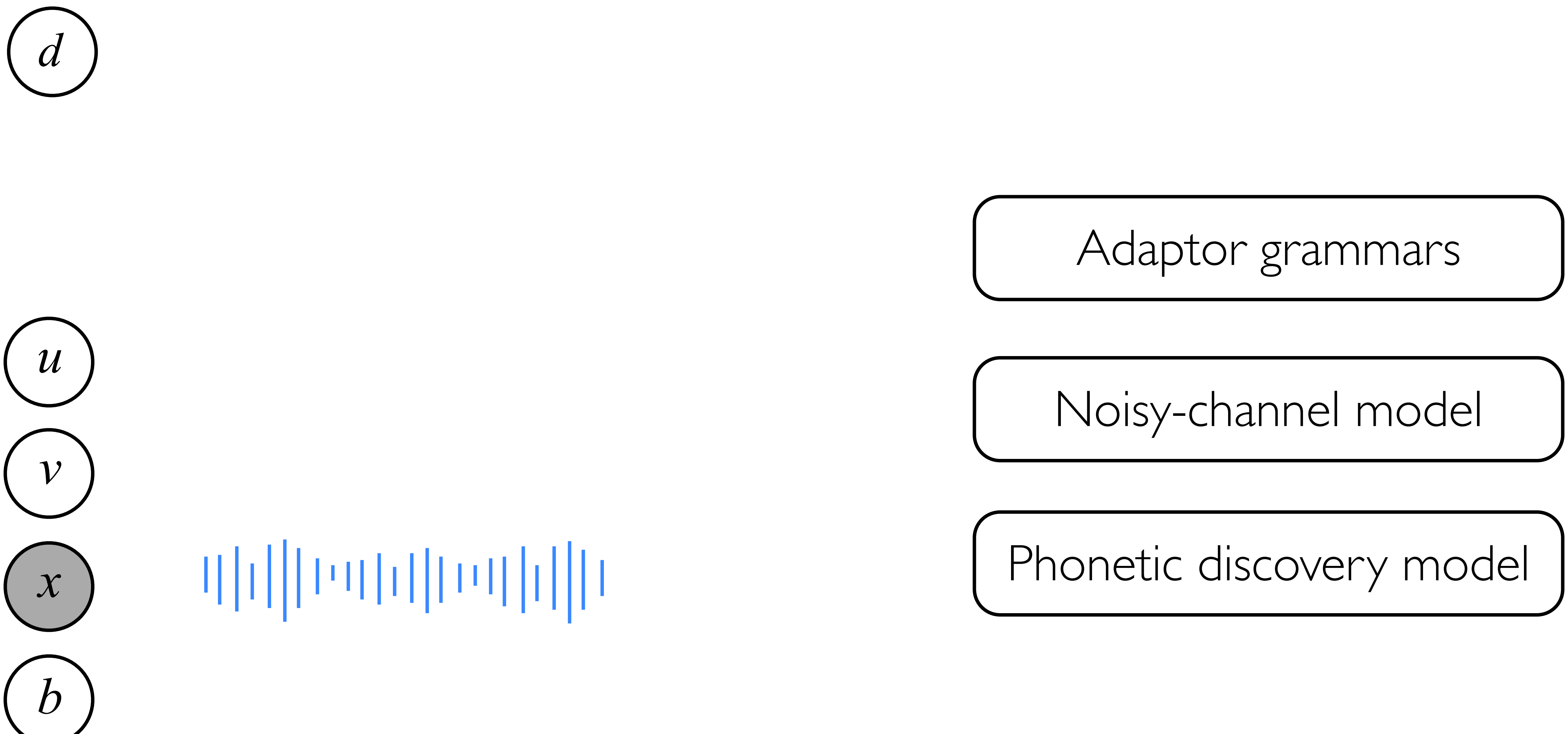


Adaptor grammars

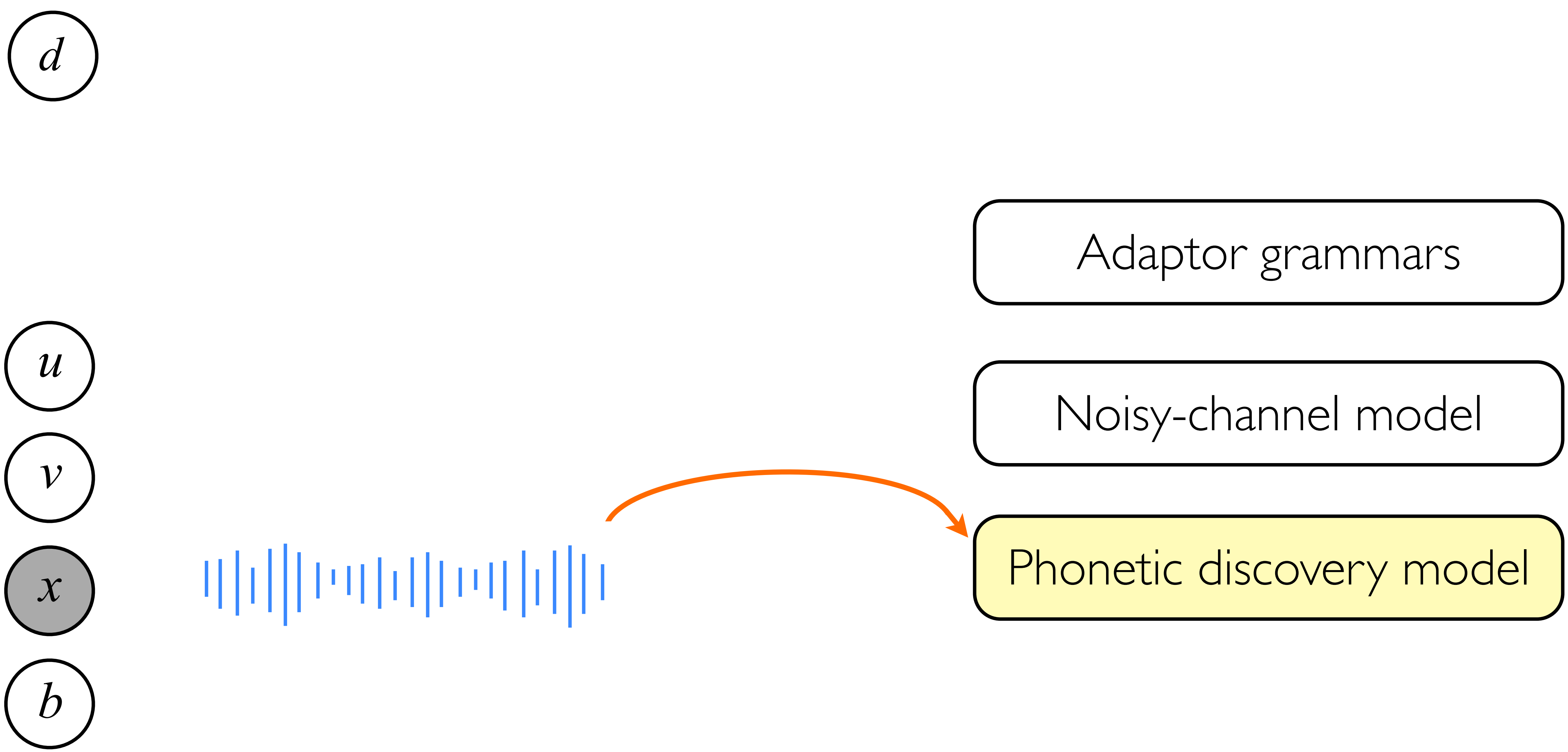
Noisy-channel model

Phonetic discovery model

Initialization

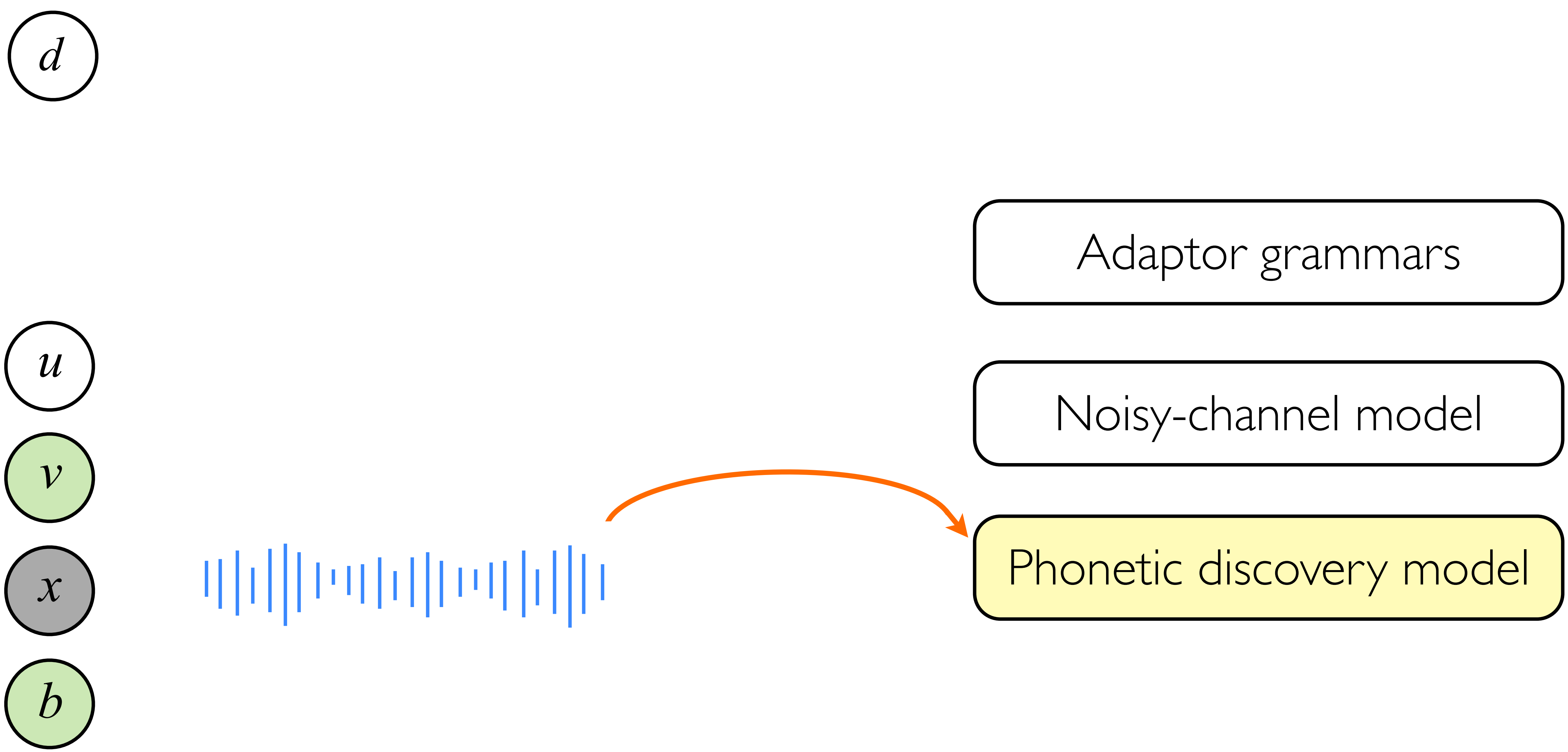


Initialization



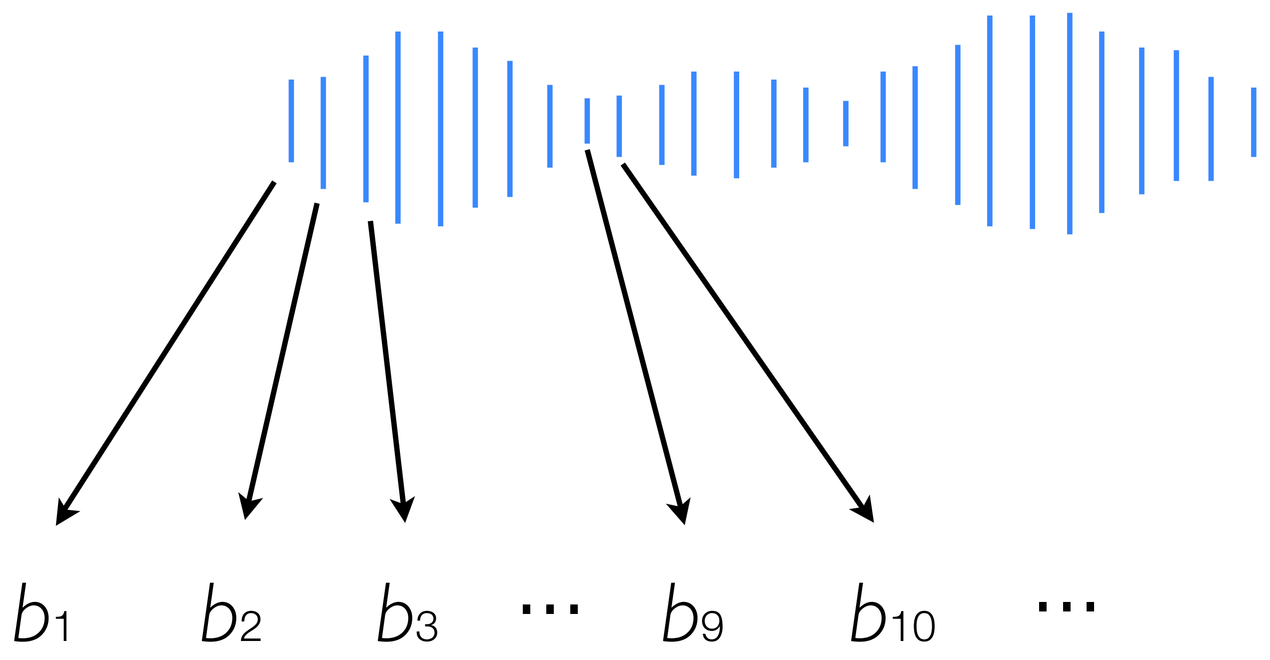
Initialization

- Initialize v and b using the phonetic discovery model



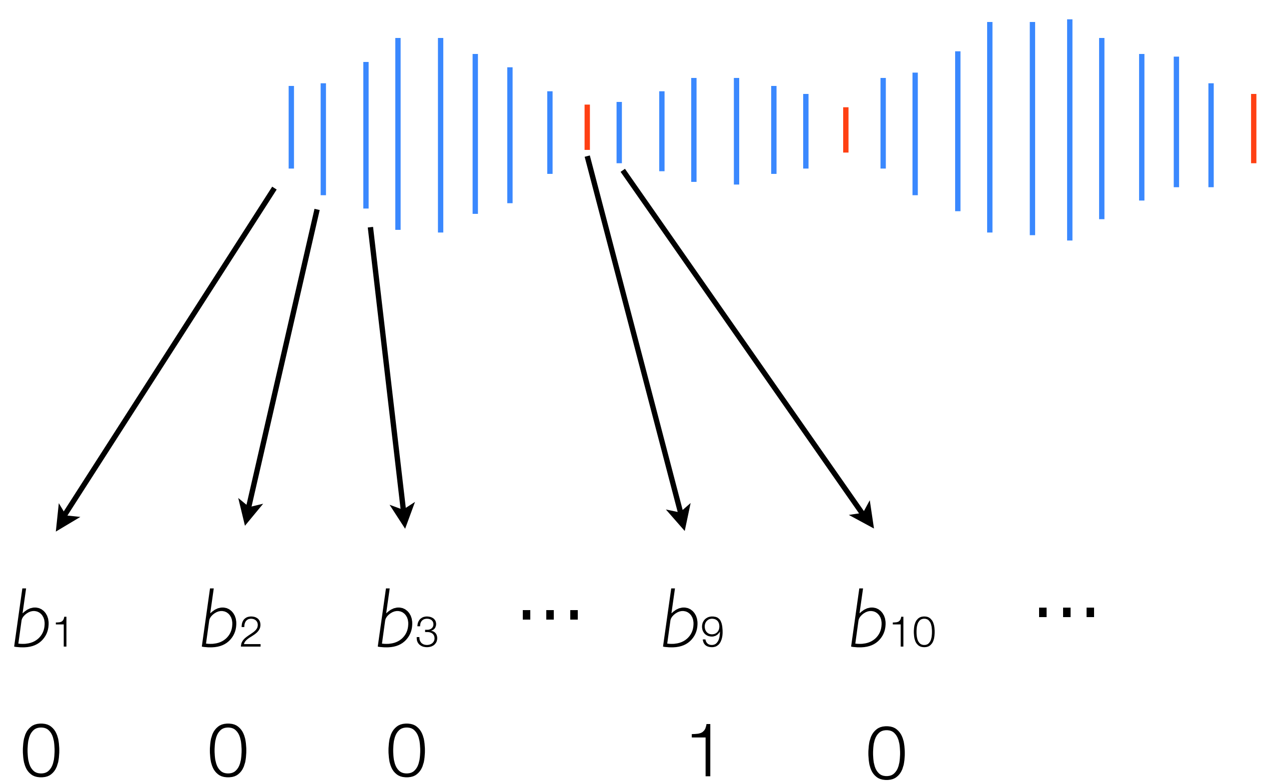
Inference on Phone Boundaries (*b*)

- **Boundary variables**
  - A priori, every frame can be a phone boundary



Inference on Phone Boundaries (b)

- **Boundary variables**
  - A priori, every frame can be a phone boundary
  - Boundary variables take binary values

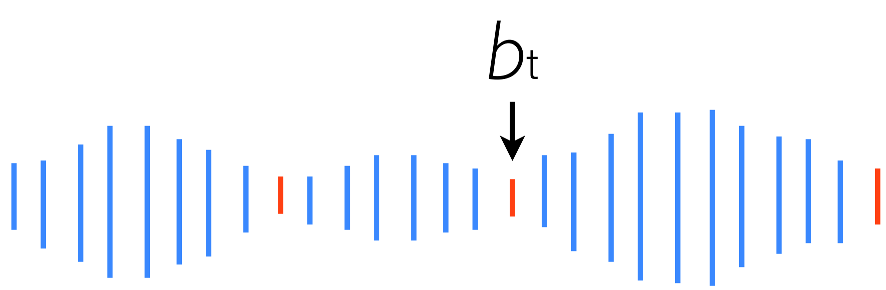


Prior and Posterior for Phone Boundaries

- Prior
  - Fixed prior probabilities $p(b_t = 1) = \alpha_b$ and $p(b_t = 0) = 1 - \alpha_b$

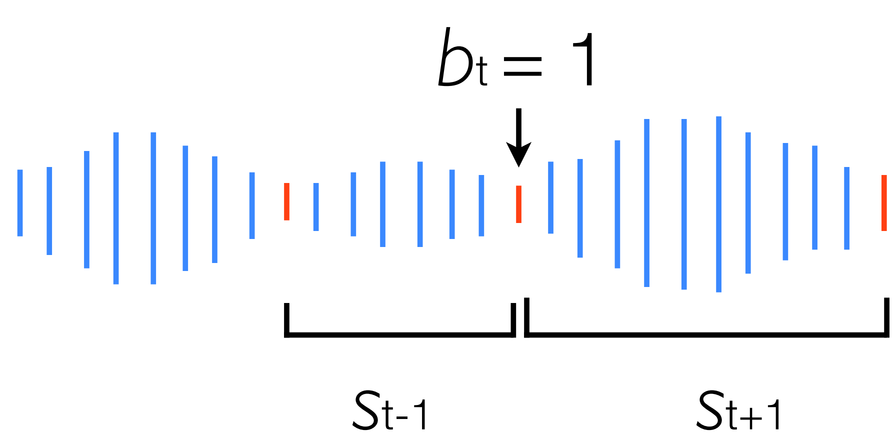
Prior and Posterior for Phone Boundaries

- Prior
  - Fixed prior probabilities $p(b_t = 1) = \alpha_b$ and $p(b_t = 0) = 1 - \alpha_b$
- Posterior: examine one boundary variable (b_t) at a time
  - Fix the current values of other boundary variables
  - Consider both 0 and 1 for b_t and the respective segmentation outcomes



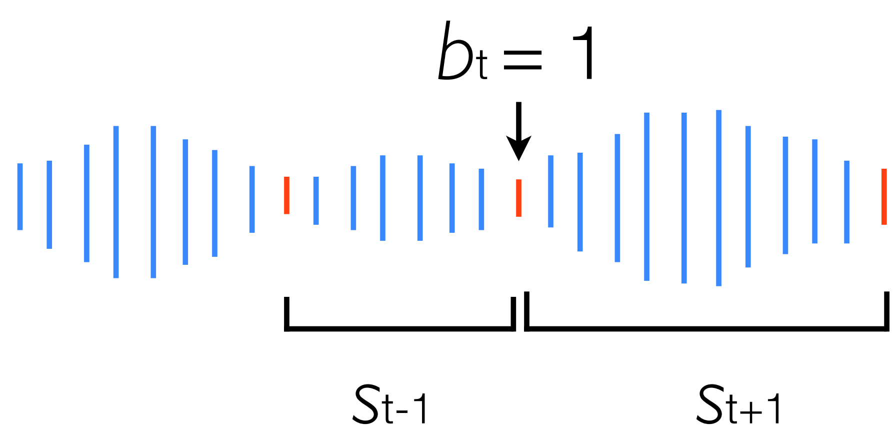
Prior and Posterior for Phone Boundaries

- **Prior**
  - Fixed prior probabilities $p(b_t = 1) = \alpha_b$ and $p(b_t = 0) = 1 - \alpha_b$
- **Posterior: examine one boundary variable (b_t) at a time**
  - Fix the current values of other boundary variables
  - Consider both 0 and 1 for b_t and the respective segmentation outcomes



Prior and Posterior for Phone Boundaries

- Prior
  - Fixed prior probabilities $p(b_t = 1) = \alpha_b$ and $p(b_t = 0) = 1 - \alpha_b$
- Posterior: examine one boundary variable (b_t) at a time
  - Fix the current values of other boundary variables
  - Consider both 0 and 1 for b_t and the respective segmentation outcomes



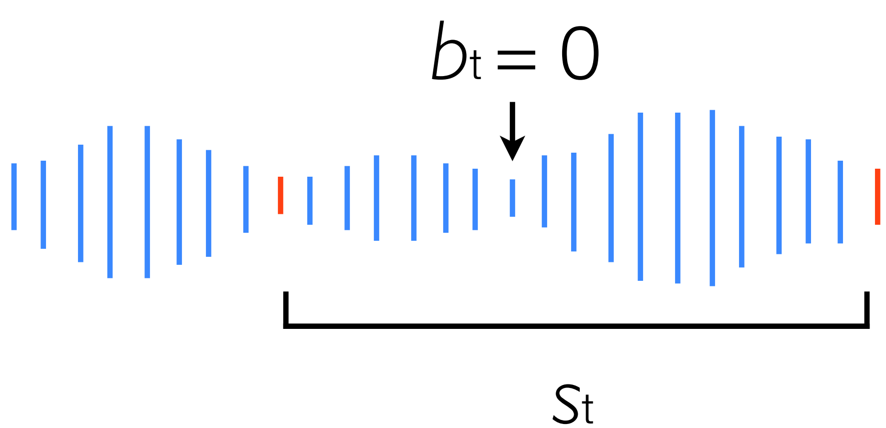
$$p(b_t = 1 \mid \cdots) \propto$$
$$p(b_t = 1)\, p(s_{t-1} \mid c^-, \underline{\theta})\, p(s_{t+1} \mid c^-, \underline{\theta})$$

c^- : cluster labels of all other segments

$\underline{\theta}$: the set of HMMs

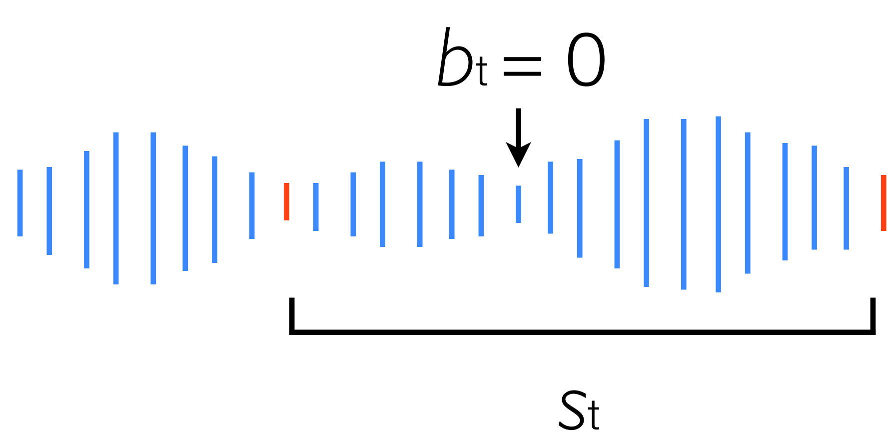
Prior and Posterior for Phone Boundaries

- Prior
  - Fixed prior probabilities $p(b_t = 1) = \alpha_b$ and $p(b_t = 0) = 1 - \alpha_b$
- Posterior: examine one boundary variable (b_t) at a time
  - Fix the current values of other boundary variables
  - Consider both 0 and 1 for b_t and the respective segmentation outcomes



Prior and Posterior for Phone Boundaries

- Prior
  - Fixed prior probabilities $p(b_t = 1) = \alpha_b$ and $p(b_t = 0) = 1 - \alpha_b$
- Posterior: examine one boundary variable (b_t) at a time
  - Fix the current values of other boundary variables
  - Consider both 0 and 1 for b_t and the respective segmentation outcomes



$$p(b_t = 0 \mid \cdots) \propto$$
$$p(b_t = 0)\, p(s_t \mid c^-, \underline{\theta})$$

Prior and Posterior for Phone Boundaries

- **Prior**
  - Fixed prior probabilities $p\left(b_t=1\right)=\alpha_b$ and $p\left(b_t=0\right)=1-\alpha_b$
- **Posterior: examine one boundary variable (b_t) at a time**
  - Fix the current values of other boundary variables
  - Consider both 0 and 1 for b_t and the respective segmentation outcomes

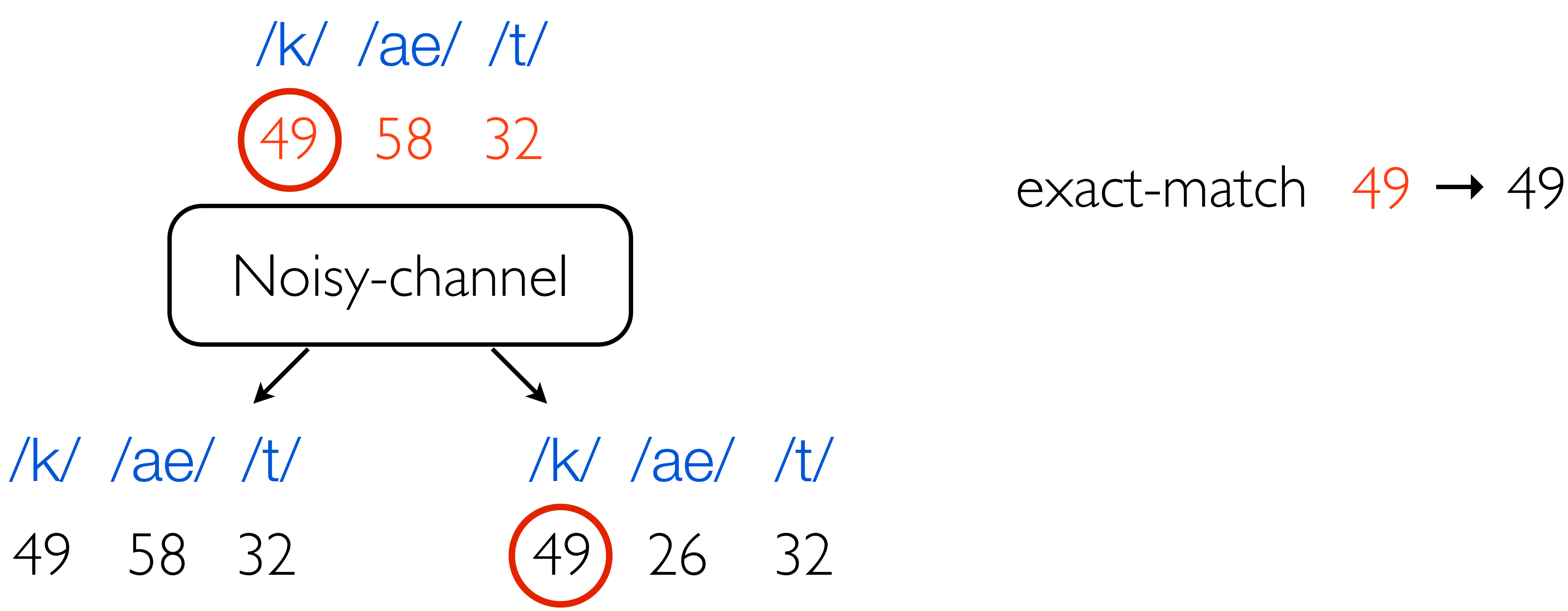
Generate a sample for b_t

$$p\left(b_t=1\mid\cdots\right)\propto p\left(b_t=1\right)p\left(s_{t-1}\mid c^-,\underline{\theta}\right)p\left(s_{t+1}\mid c^-,\underline{\theta}\right)$$

$$p\left(b_t=0\mid\cdots\right)\propto p\left(b_t=0\right)p\left(s_t\mid c^-,\underline{\theta}\right)$$

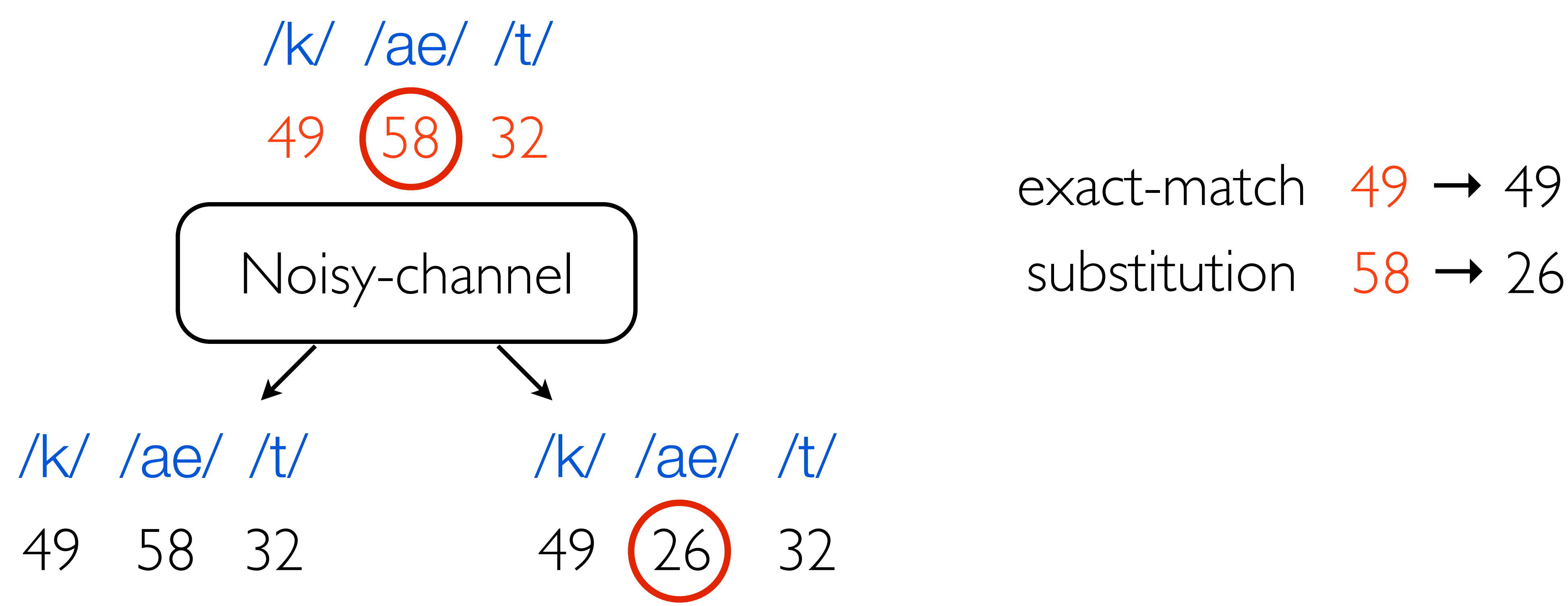
Noisy-channel Model

- Assume the phonetic variations are outcomes of a noisy-channel
- Formulate the noisy-channel model as a set of edit operations
  - Substitution, deletion, insertion, and exact-match



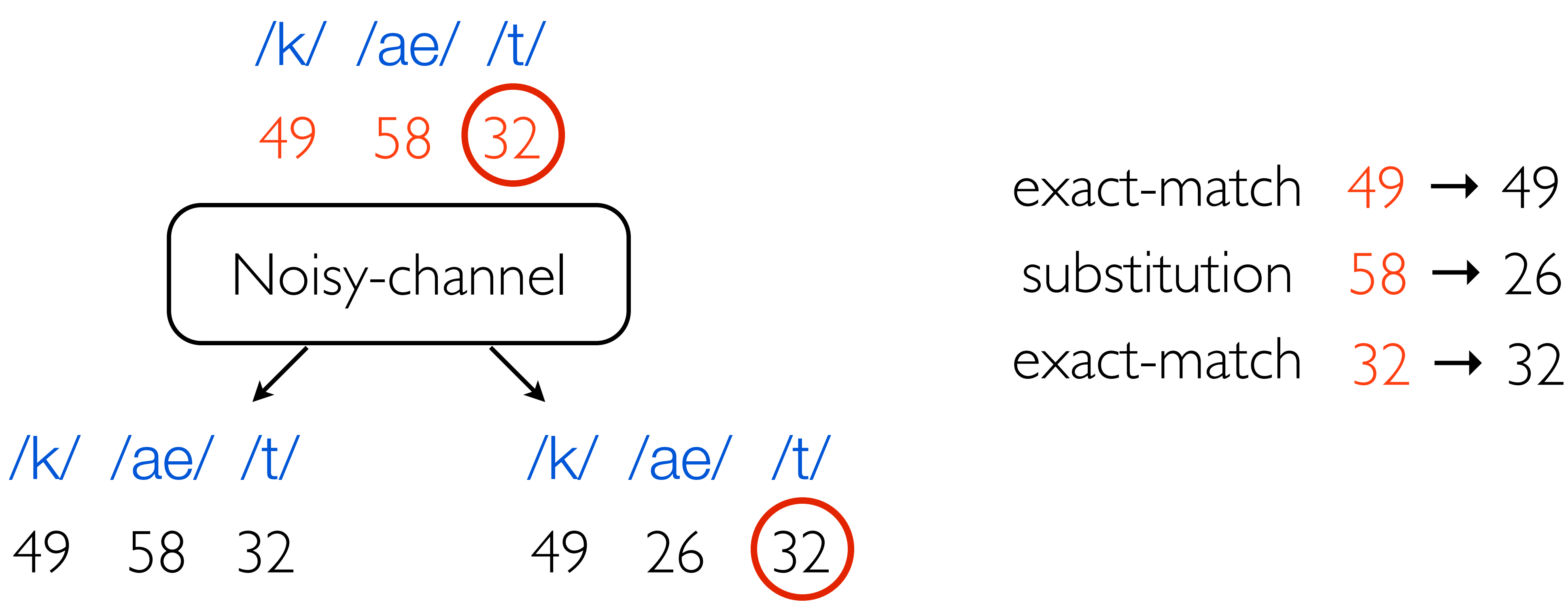
Noisy-channel Model

- Assume the phonetic variations are outcomes of a noisy-channel
- Formulate the noisy-channel model as a set of edit operations
  - Substitution, deletion, insertion, and exact-match



Noisy-channel Model

- Assume the phonetic variations are outcomes of a noisy-channel
- Formulate the noisy-channel model as a set of edit operations
  - Substitution, deletion, insertion, and exact-match



Acknowledgement

- **Thesis advisor**
  - Jim Glass
- **Thesis committee**
  - Regina Barzilay and Victor Zue
- **Collaborators**
  - Tim O'Donnell
  - Brenden Lake
  - Matt Johnson
  - Yu Zhang
  - Lee Hetherington
  - Ian McGraw
  - Stephanie Seneff
  - Oded Ghitza
- **SLS members**
  - Marcia
  - Tuka
  - Xue
  - Daniel
  - Scott
  - Carrie
  - David
  - Michael
  - Najim
  - Ekapol
  - Mandy
  - Stephen
  - Patrick
  - Jennifer
  - Ann
  - Yu
  - Chengjie
- **Previous officemates**
  - Hung-an
  - Paul
  - Yaodong
  - Yuan
- **Friends and family**