Learning Words

Mark Johnson

joint work with many people, including
Ben Börschinger, Eugene Charniak, Katherine Demuth,
Michael Frank, Sharon Goldwater, Tom Griffiths and Bevan Jones

Somewhere over the north Pacific

July 2012

Engineering vs science

- Cool problem and *very interesting people!*
- Differences between engineering and science:
  - some engineering approaches use *far more data than any human ever experiences*

  $\Rightarrow$ *can't be what humans do* (humans must use additional information)
  - engineering can and should exploit "accidental" data sources and "light" supervision
  - engineering applications evaluated on how well they *cover* the phenomena
  - scientific models evaluated on how well they *capture generalisations*

  $\Rightarrow$ *simple models can be scientifically interesting even if they don't cover the data*

Why am I here?

- I'd like to understand how language is used and acquired *as a computational process*
  - it is a computational process because it involves manipulation of meaning-bearing symbols in a meaning-preserving way
  - language acquisition is many other things as well (e.g., a developmental process, a psychological process, etc.)
- Marr's *3 levels* (Marr 1976)
  - *implementational level* (hardware, wetware)
  - *algorithmic level* (data structures, algorithms)
  - *computational or informational level* (information and its interaction)
- "Ideal" Bayesian learners use information optimally (?)
  - use a "Bayes optimial" algorithm to "run" the learner
  - any other Bayes-optimal algorithm should produce same output

  $\Rightarrow$ lets us study the computational-level properties of our models

Why are only incremental algorithms "cognitively realistic"?

- Incrementality is an algorithmic-level property, but we have only the vaguest ideas of algorithms or data structures that neural wetware supports
- Many possible models at informational level; each of which has many algorithms $\Rightarrow$ *space of possible algorithms is much larger than space of possible models*
- popular algorithms (EM, MCMC, etc.) are designed to be *generic* (e.g., applicable to language, vision and much more), but humans only solve *specific* problems
- why isn't accuracy or ability to mimic human developmental patterns at least as important?

What can computational models contribute?

- *Informational sufficiency:* demonstrate certain kinds of information suffice to learn something
- *Identify learning synergies:* learning two things together may be better than learning them separately
  - *joint learning* of two separate phenomena may be *more efficient* than independent learning
- *Trajectory of learning:* predict learners' developmental stages
- The Anna Karenina principle: *"Happy families are all alike; every unhappy family is unhappy in its own way."*
  - compare *characteristic errors made by particular models* with human learner behaviour
- *"Animals don't move on wheels"* (Wasow)
- But many of our models are so bad that non-ideal learners are more accurate than ideal ones (?)

Unsupervised word segmentation

- Input: phoneme sequences with *sentence boundaries* (Brent)
- Task: identify *word boundaries*, and hence *words*

j △ u ▲ w △ ɑ △ n △ t ▲ t △ u ▲ s △ i ▲ ð △ ə ▲ b △ ʊ △ k

ju wɑnt tu si ðə bʊk

"you want to see the book"

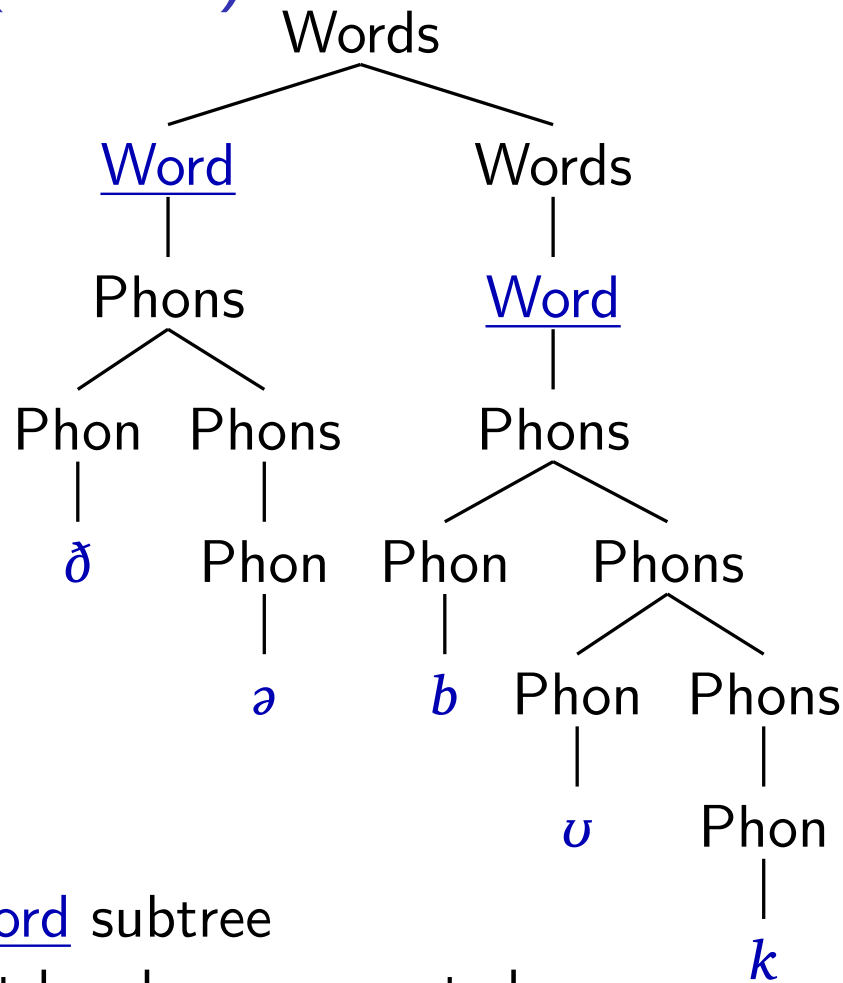
- Ignoring phonology and morphology, this involves learning the pronunciations of the lexical items in the language

Adaptor grammars as non-parametric hierachical Bayesian models

- The trees generated by an adaptor grammar are defined by CFG rules as in a CFG
- A subset of the nonterminals are *adapted*
- *Unadapted nonterminals* expand by picking a rule and recursively expanding its children, as in a PCFG
- *Adapted nonterminals* can expand in two ways:
  - by picking a rule and recursively expanding its children, or
  - by generating a previously generated tree (with probability proportional to the number of times previously generated)

Unigram adaptor grammar (Brent)

Words → Word
Words → Word Words
Word → Phons
Phons → Phon
Phons → Phon Phons



- Word nonterminal is adapted

⇒ To generate a Word:

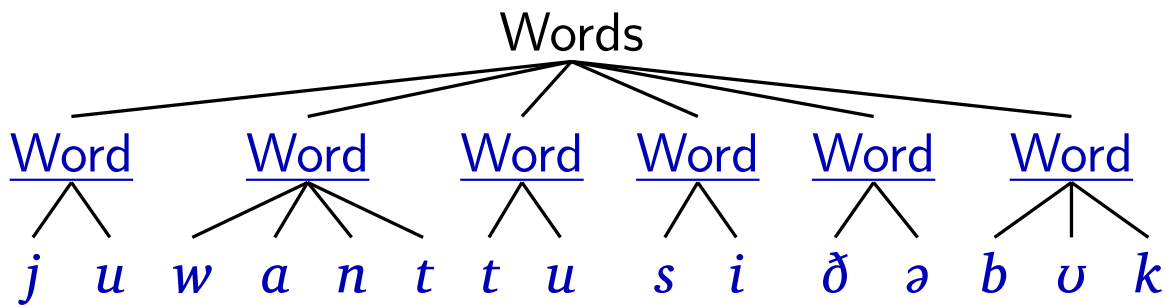
  - select a previously generated Word subtree with prob. $\propto$ number of times it has been generated
  - expand using Word → Phons rule with prob. $\propto \alpha_{\text{Word}}$ and recursively expand Phons

Unigram model of word segmentation

- Unigram "bag of words" model (Brent):
  - generate a *dictionary*, i.e., a set of words, where each word is a random sequence of phonemes
    - Bayesian prior prefers smaller dictionaries
  - generate each utterance by choosing each word at random from dictionary
- Brent's unigram model as an adaptor grammar:

$\text{Words} \rightarrow \text{Word}^+$

$\underline{\text{Word}} \rightarrow \text{Phoneme}^+$



- Accuracy of word segmentation learnt: *56% token f-score* (same as Brent model)
- But we can construct many more word segmentation models using AGs

Adaptor grammar learnt from Brent corpus

- **Initial grammar**

1	Words → Word Words	1	Words → Word
1	Word → Phon		
1	Phons → Phon Phons	1	Phons → Phon
1	Phon → *D*	1	Phon → *G*
1	Phon → *A*	1	Phon → *E*

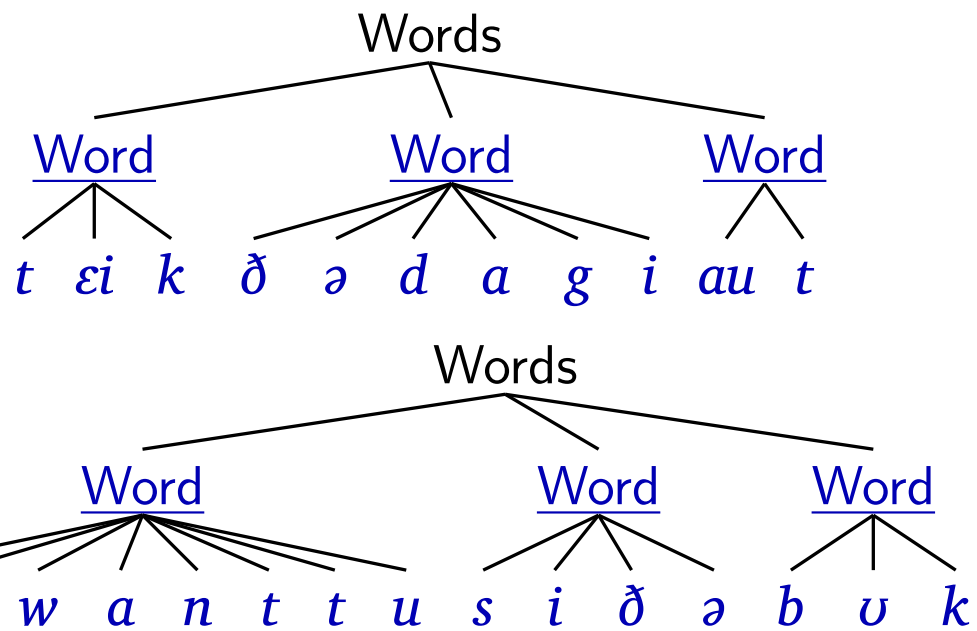
- **A grammar learnt from Brent corpus**

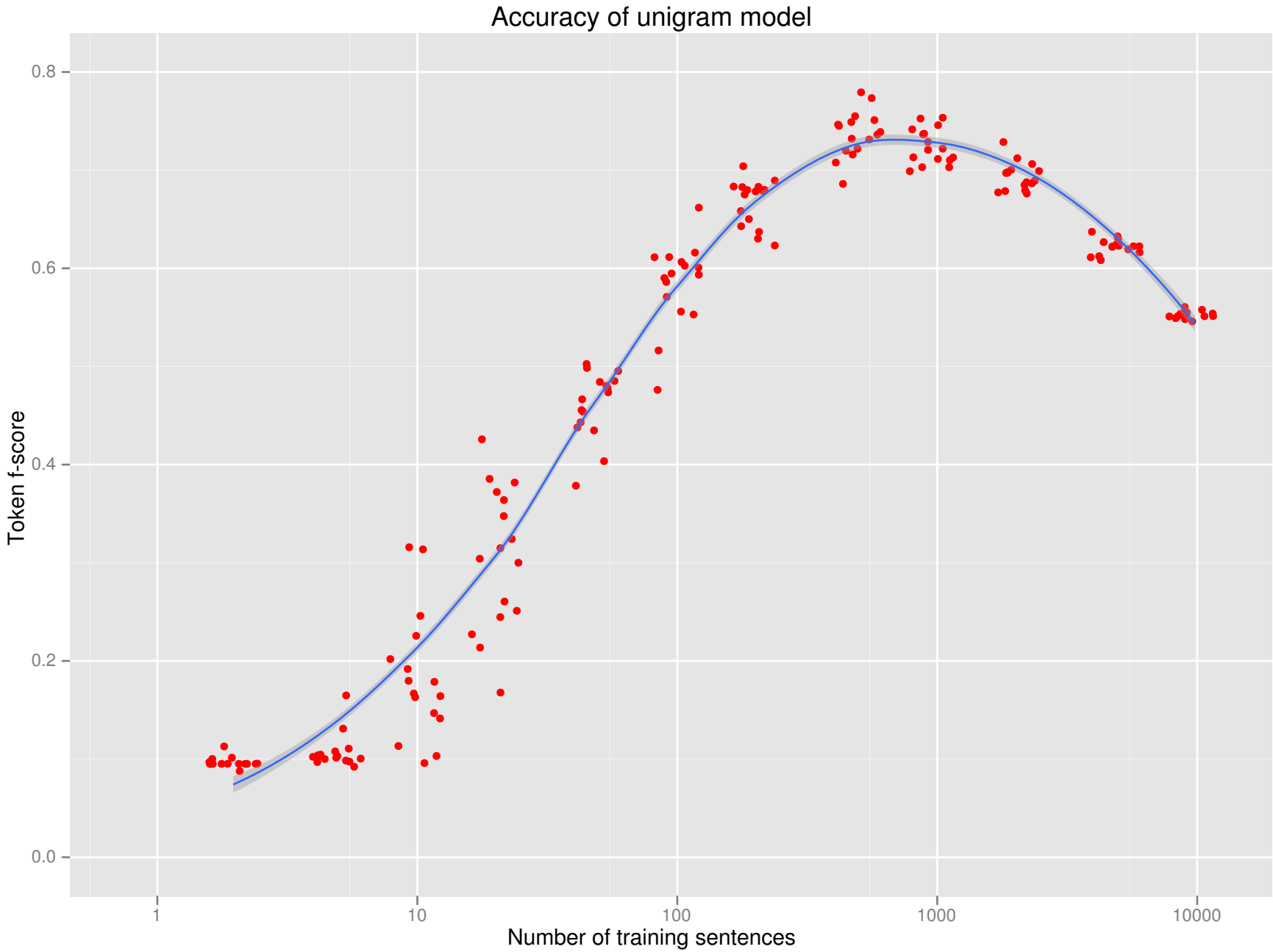
16625	Words → Word Words	9791	Words → Word
1575	Word → Phons		
4962	Phons → Phon Phons	1575	Phons → Phon
134	Phon → *D*	41	Phon → *G*
180	Phon → *A*	152	Phon → *E*
460	Word → (Phons (Phon *y*) (Phons (Phon *u*)))		
446	Word → (Phons (Phon *w*) (Phons (Phon *A*) (Phons (Phon *t*))))		
374	Word → (Phons (Phon *D*) (Phons (Phon *6*)))		
372	Word → (Phons (Phon *&*) (Phons (Phon *n*) (Phons (Phon *d*))))		

Undersegmentation errors with Unigram model

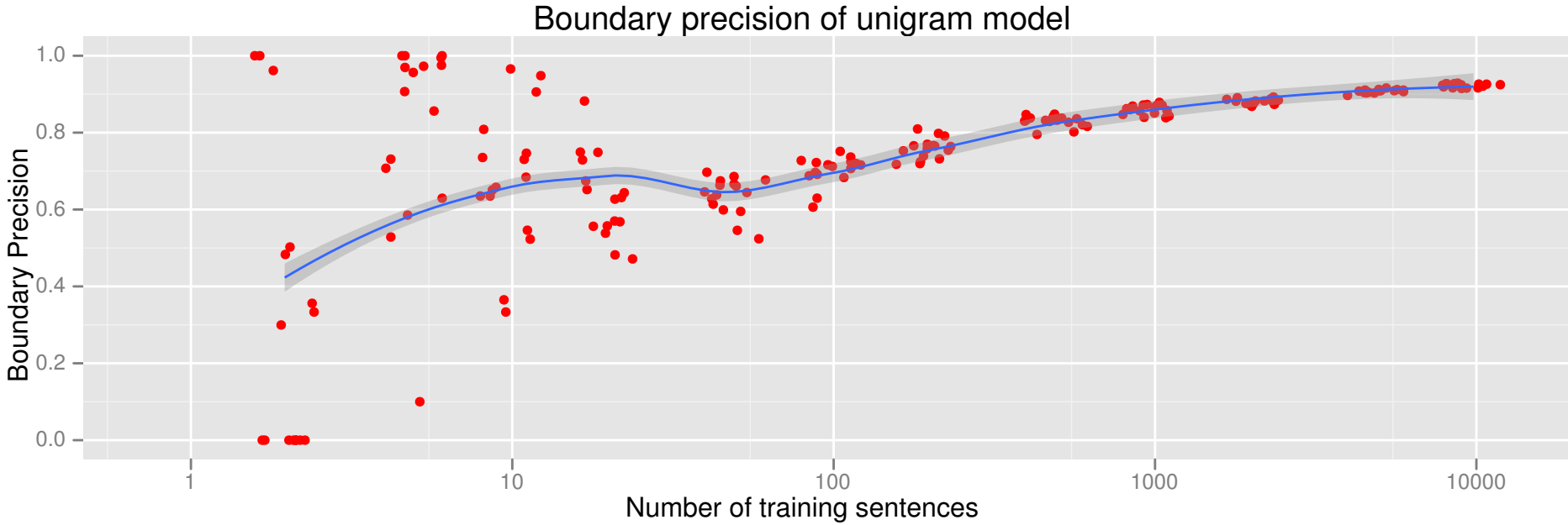
$$\text{Words} \rightarrow \underline{\text{Word}}^{+} \qquad \underline{\text{Word}} \rightarrow \text{Phon}^{+}$$

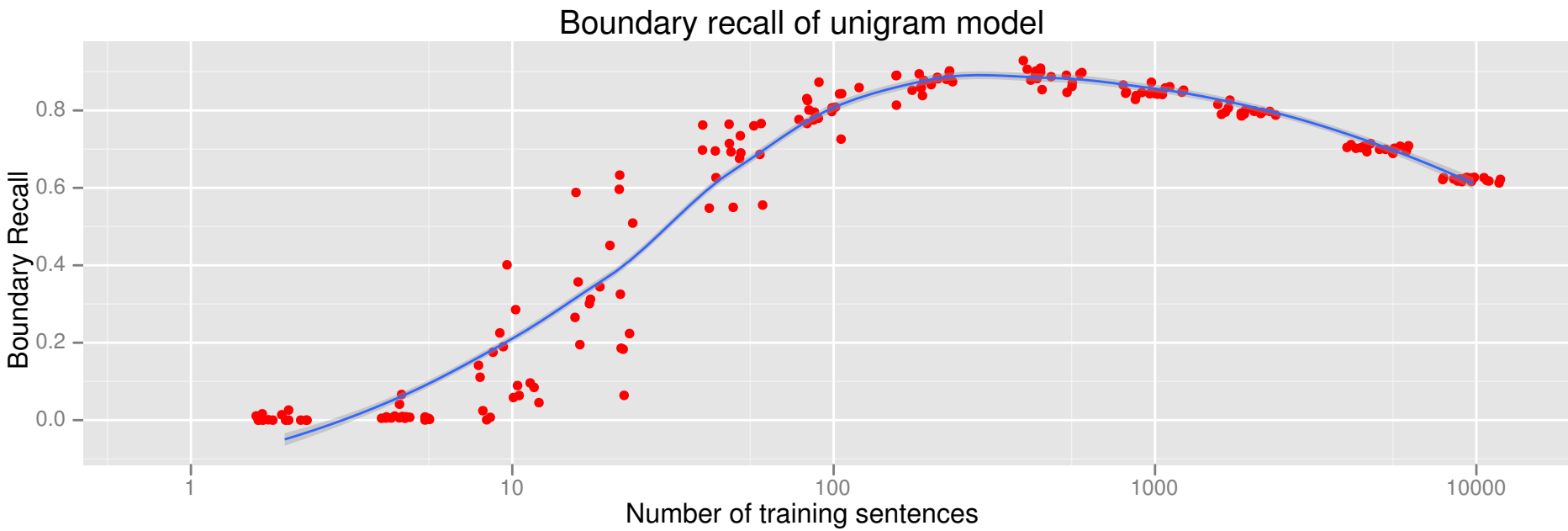
- Unigram word segmentation model assumes each word is generated independently
- But there are strong inter-word dependencies (collocations)
- Unigram model can only capture such dependencies by analyzing collocations as words (Goldwater 2006)





MACQUARIE UNIVERSITY

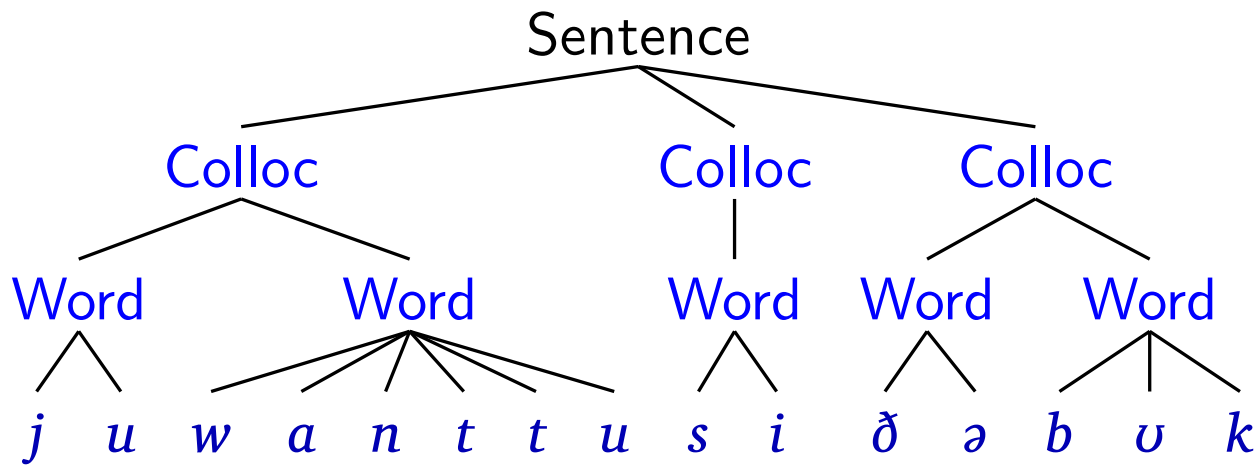




MACQUARIE UNIVERSITY

Collocations ⇒ Words

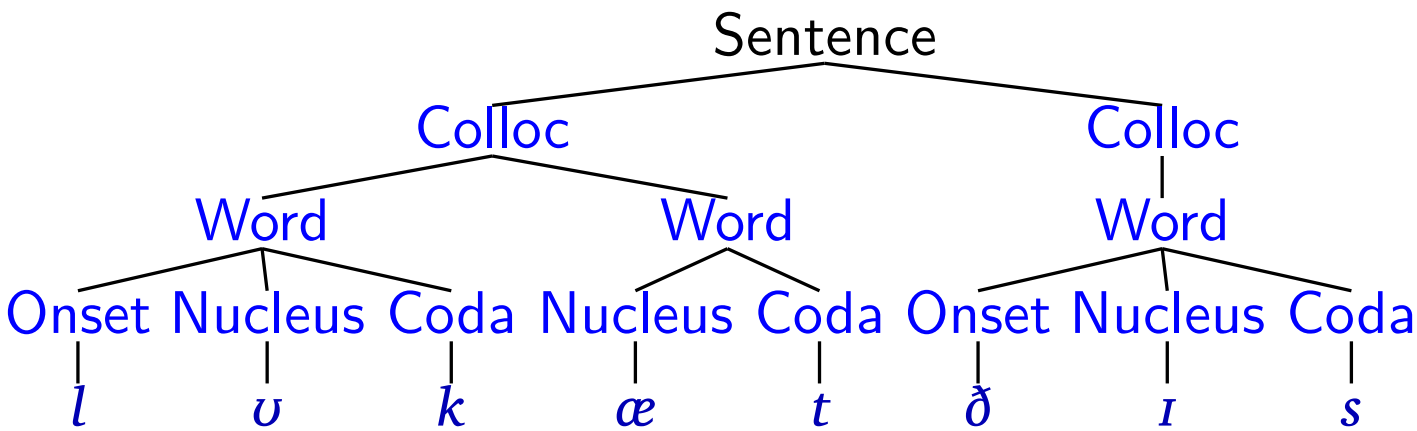
$$\text{Sentence} \rightarrow \text{Colloc}^+$$
$$\underline{\text{Colloc}} \rightarrow \text{Word}^+$$
$$\underline{\text{Word}} \rightarrow \text{Phon}^+$$



- A Colloc(ation) consists of one or more words
- Both Words and Collocs are adapted (learnt)
- Significantly improves word segmentation accuracy over unigram model (76% f-score; ≈ Goldwater's bigram model)

Jointly learning words and syllables

$$\text{Sentence} \rightarrow \text{Colloc}^{+} \qquad \underline{\text{Colloc}} \rightarrow \text{Word}^{+}$$
$$\underline{\text{Word}} \rightarrow \text{Syllable}^{\{1:3\}} \qquad \text{Syllable} \rightarrow (\text{Onset})\ \text{Rhyme}$$
$$\underline{\text{Onset}} \rightarrow \text{Consonant}^{+} \qquad \text{Rhyme} \rightarrow \text{Nucleus}\ (\text{Coda})$$
$$\underline{\text{Nucleus}} \rightarrow \text{Vowel}^{+} \qquad \underline{\text{Coda}} \rightarrow \text{Consonant}^{+}$$

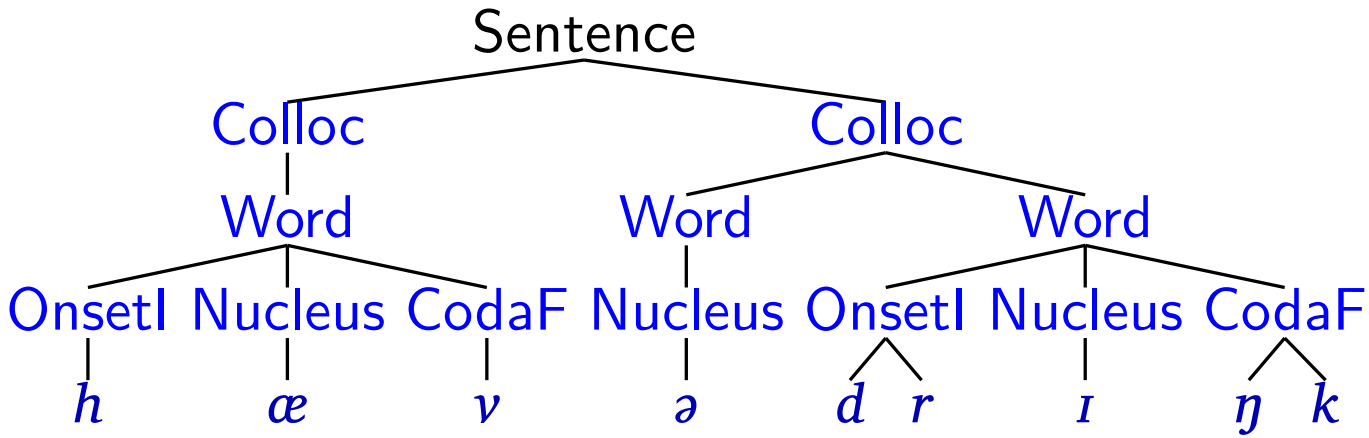


- Rudimentary syllable model (improved model does better)
- With 2 Collocation levels, f-score = 84%

Distinguishing internal onsets/codas helps

Sentence $\rightarrow$ Colloc$^+$
Word $\rightarrow$ SyllableIF
Word $\rightarrow$ SyllableI Syllable SyllableF
OnsetI $\rightarrow$ Consonant$^+$
Nucleus $\rightarrow$ Vowel$^+$

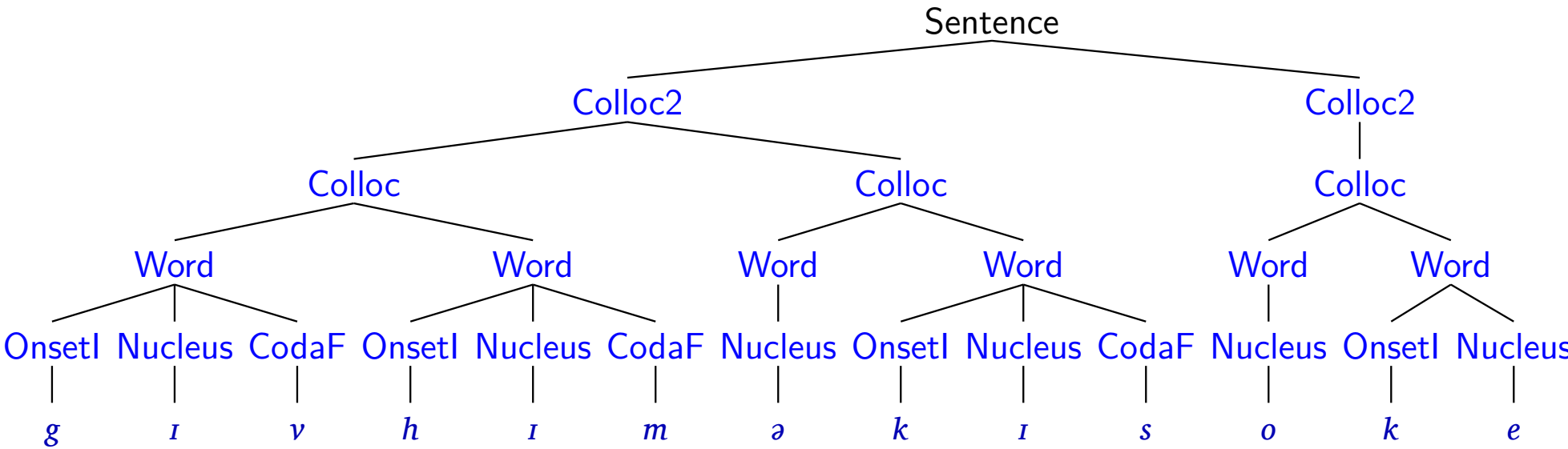
Colloc $\rightarrow$ Word$^+$
Word $\rightarrow$ SyllableI SyllableF
SyllableIF $\rightarrow$ (OnsetI) RhymeF
RhymeF $\rightarrow$ Nucleus (CodaF)
CodaF $\rightarrow$ Consonant$^+$



- With 2 Collocation levels, not distinguishing initial/final clusters, f-score = 84%
- With 3 Collocation levels, distinguishing initial/final clusters, f-score = 87%

MACQUARIE UNIVERSITY

Collocations^2 $\Rightarrow$ Words $\Rightarrow$ Syllables

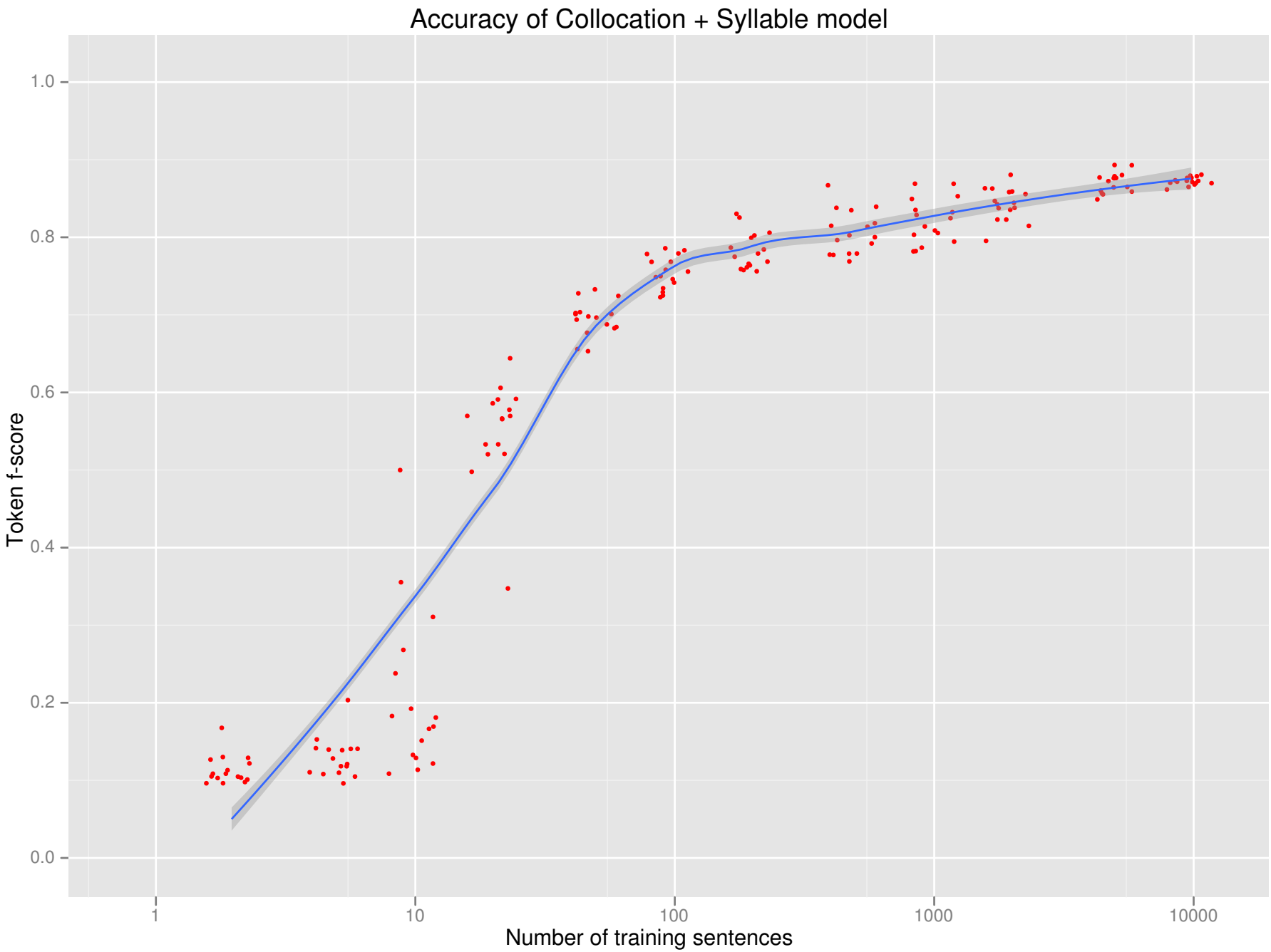


Summary of word segmentation

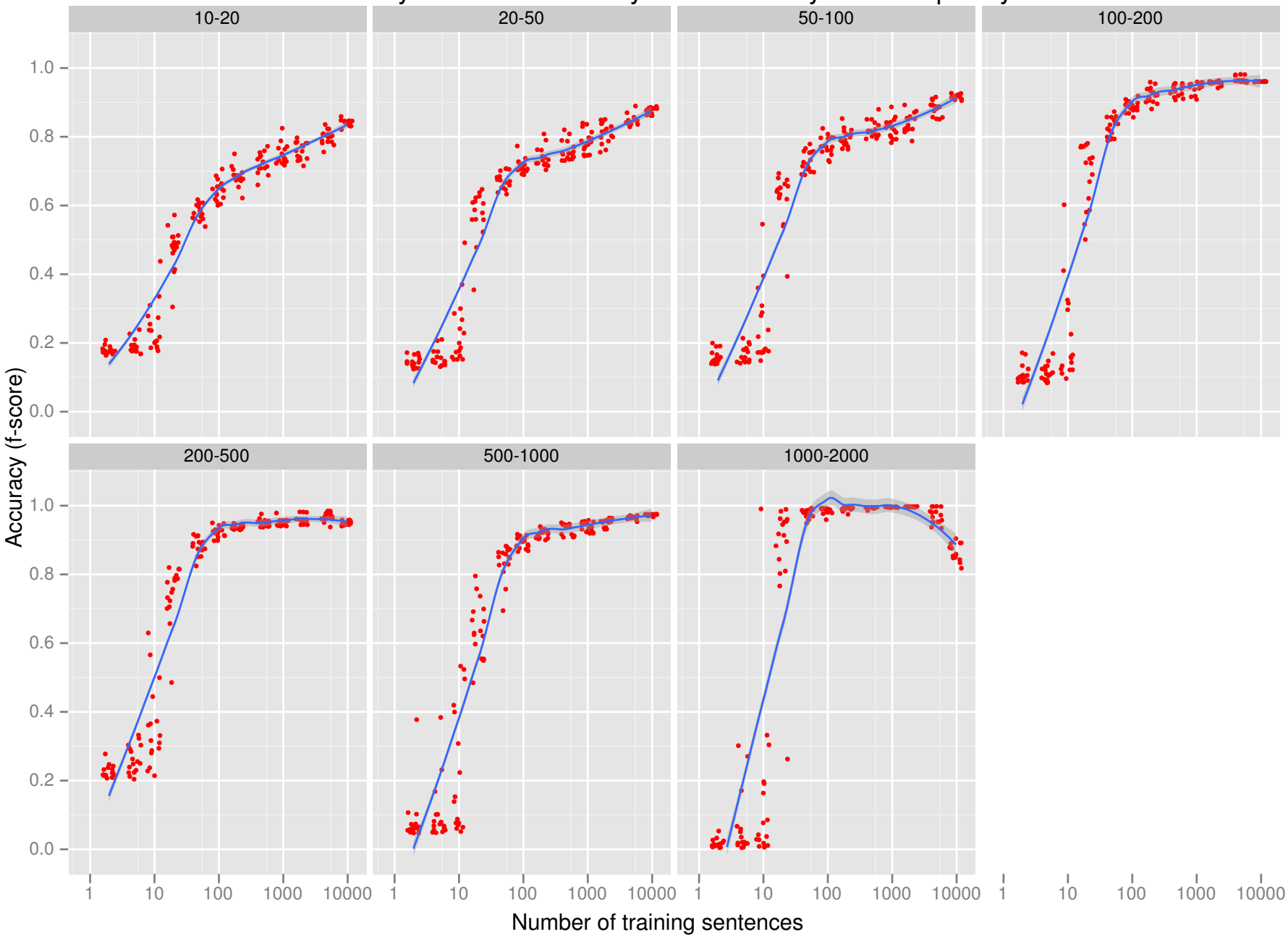
- Word segmentation accuracy depends on the kinds of generalisations learnt.

Generalization	**Accuracy**
words as units (unigram)	56%
+ associations between words (collocations)	76%
+ syllable structure	84%
+ interaction between segmentation and syllable structure	87%

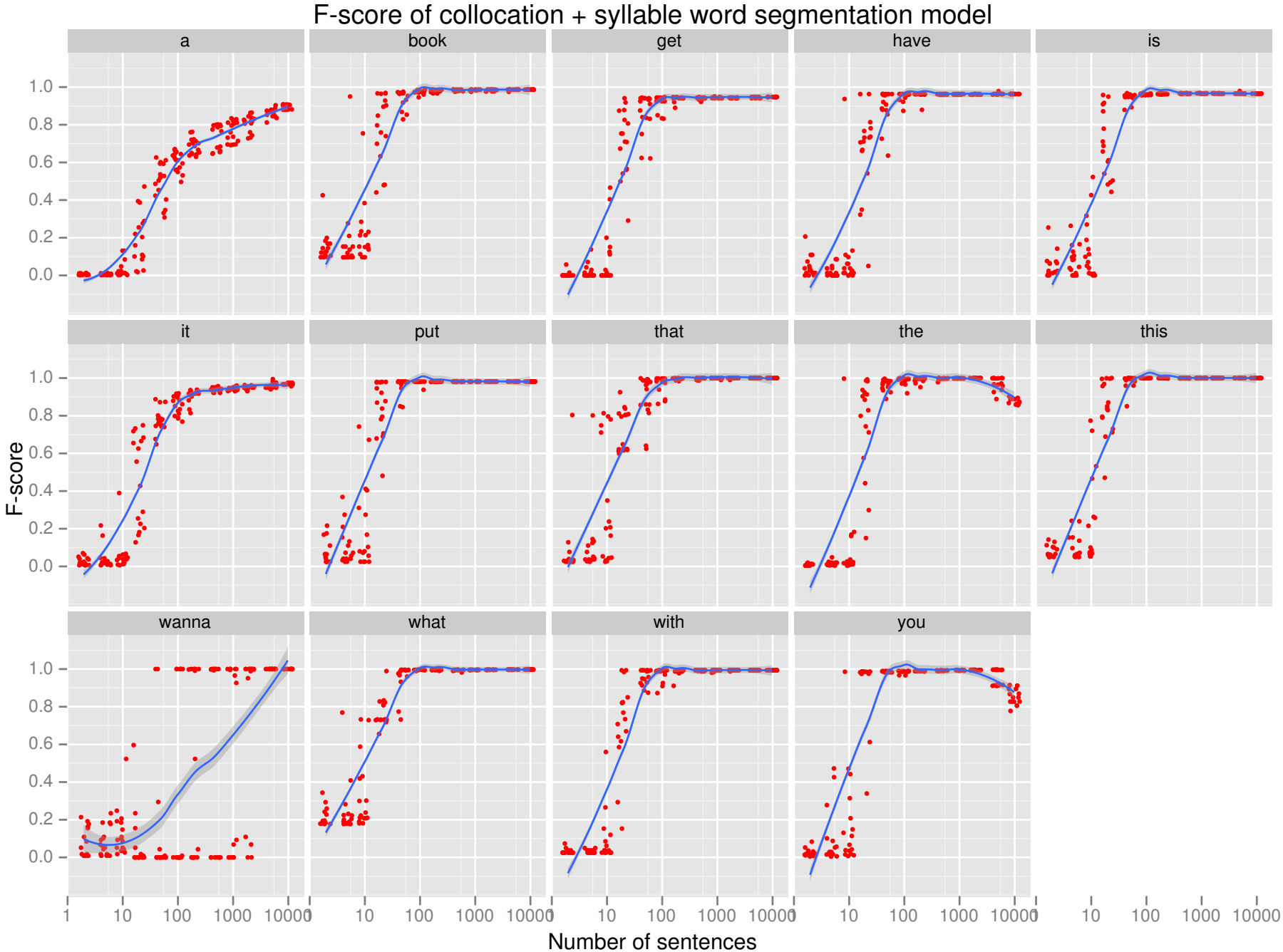
- *Synergies in learning words and syllable structure*
  - joint inference permits the learner to *explain away* potentially misleading generalizations
- We've also modelled word segmentation in *Mandarin* (and showed tone is a useful cue) and in *Sesotho* (where jointly modeling morphology improves accuracy)



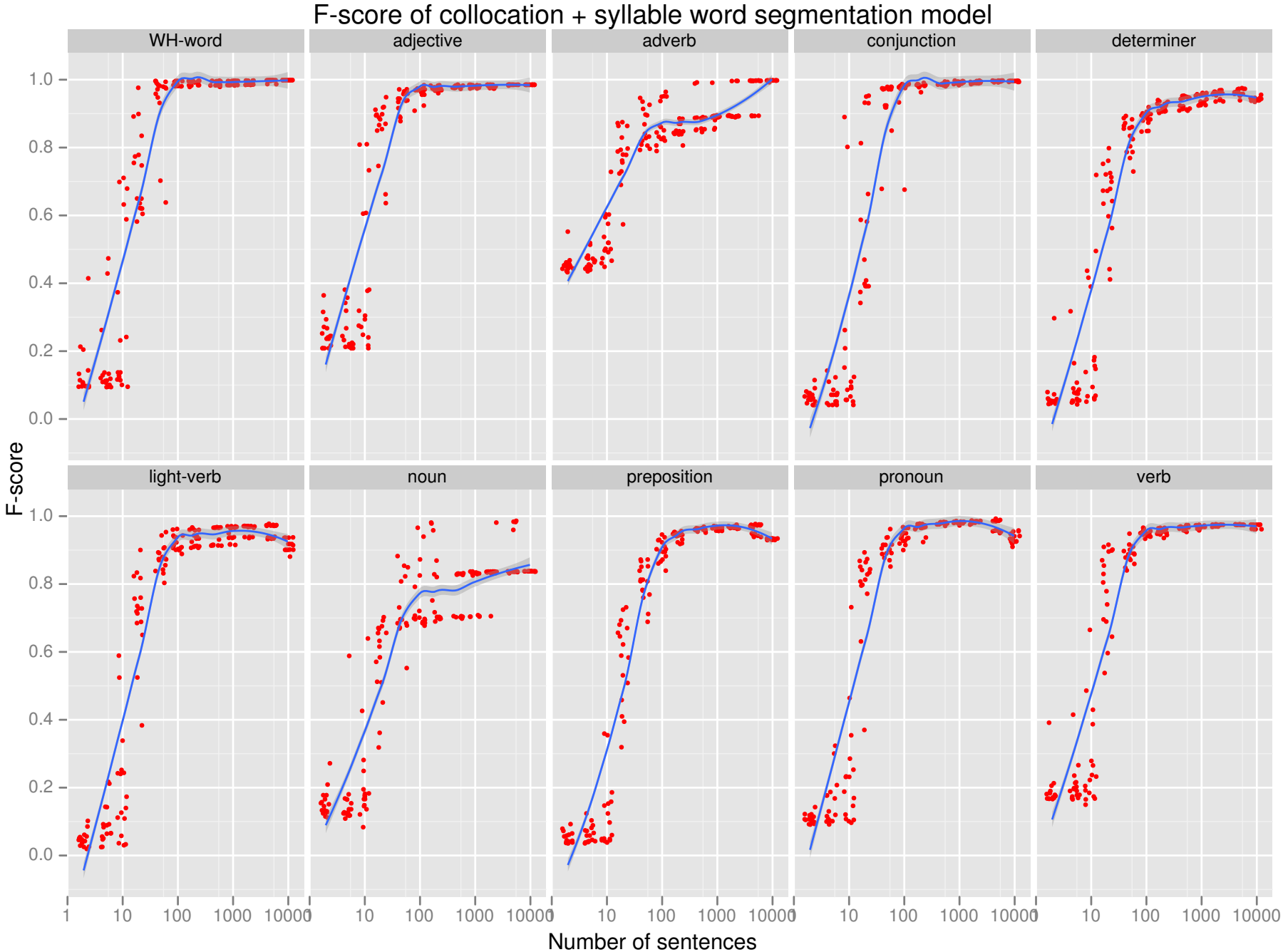
MACQUARIE UNIVERSITY



MACQUARIE UNIVERSITY



MACQUARIE UNIVERSITY



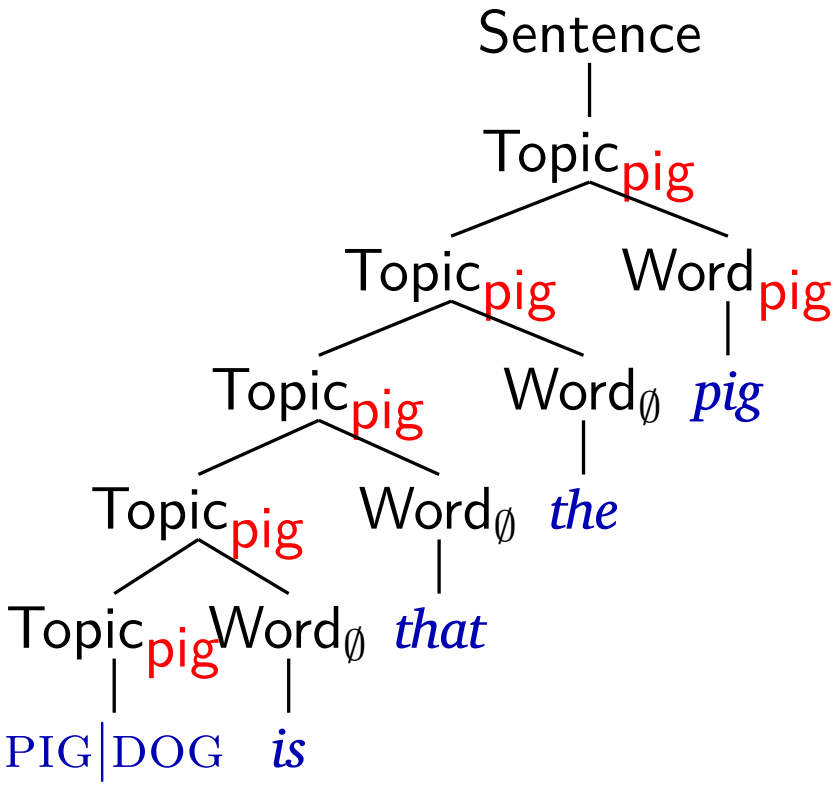
MACQUARIE UNIVERSITY

Mapping words to referents

- Input to learner:
  - word sequence: *Is that the pig?*
  - objects in nonlinguistic context: DOG, PIG
- Learning objectives:
  - identify utterance topic: PIG
  - identify word-topic mapping: *pig* $\rightsquigarrow$ PIG

Frank et al (2009) "topic models" as PCFGs

- Prefix sentences with *possible topic marker*, e.g., PIG|DOG
- PCFG rules *choose a topic* from topic marker and *propagate it through sentence*
- Each word is either generated from sentence topic or null topic ∅

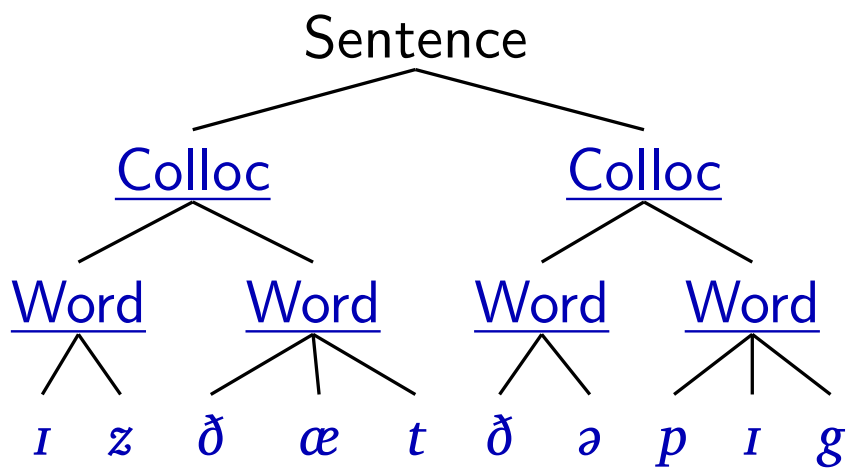


- Grammar can require *at most one topical word per sentence*
- Bayesian inference for PCFG rules and trees corresponds to Bayesian inference for word and sentence topics using topic model (Johnson 2010)

Word segmentation with adaptor grammars

- Adaptor grammars (AGs) can learn the probability of entire subtrees (as well as rules)
- AGs can express several different word segmentation models
- Learning collocations as well as words significantly improves segmentation accuracy

$$\text{Sentence} \rightarrow \underline{\text{Colloc}}^{+}$$
$$\underline{\text{Colloc}} \rightarrow \underline{\text{Word}}^{+}$$
$$\underline{\text{Word}} \rightarrow \text{Phon}^{+}$$

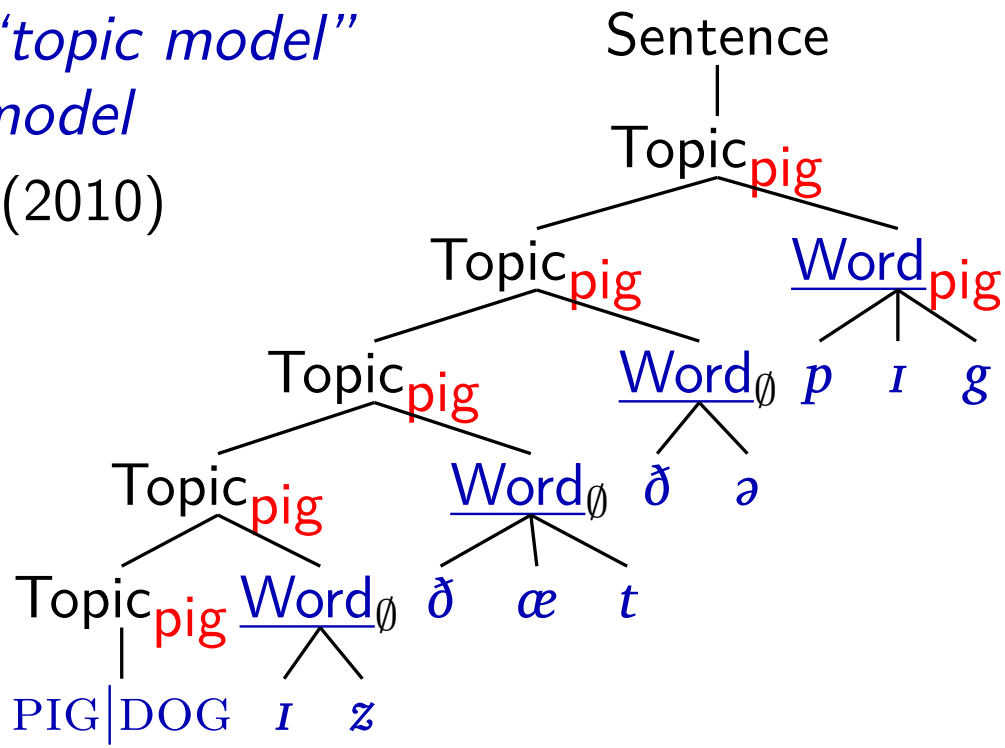


AGs for joint segmentation and referent-mapping

- Combine topic-model PCFG with word segmentation AGs
- Input consists of unsegmented phonemic forms prefixed with possible topics:

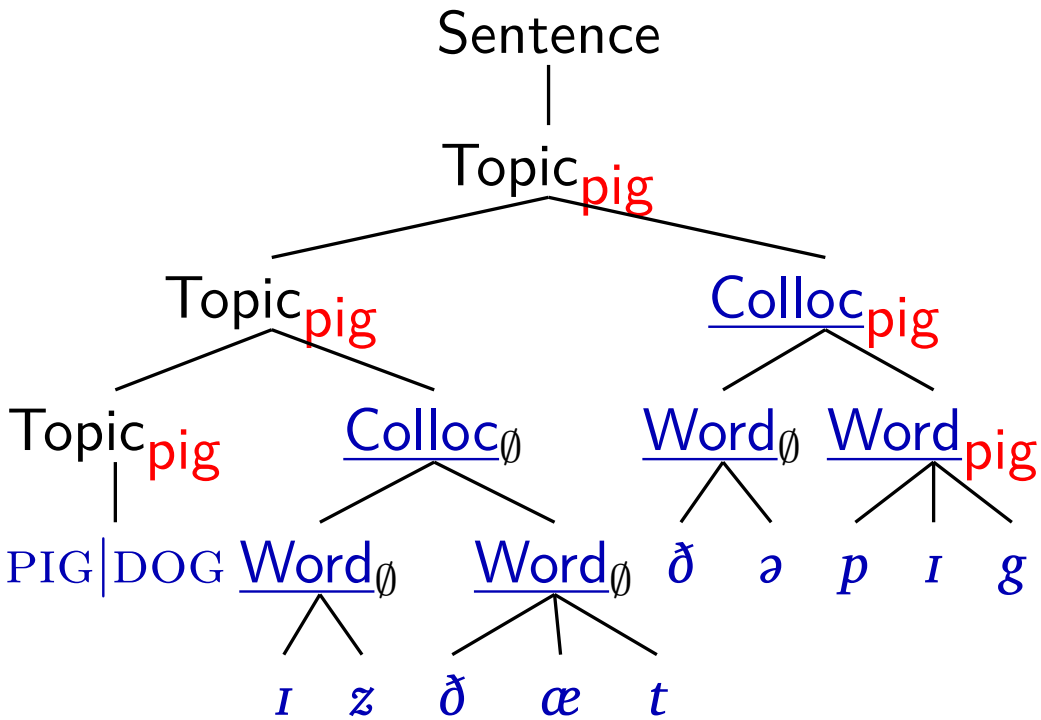
PIG|DOG ɪ z ð æ t ð ə p ɪ g

- E.g., combination of *Frank "topic model"* and *unigram segmentation model*
  - equivalent to Jones et al (2010)



- Easy to define *other combinations of topic models and segmentation models*

Collocation topic model AG



- Collocations are either "topical" or not
- Easy to modify this grammar so
  - at most one topical word per sentence, or
  - at most *one topical word per topical collocation*

Experimental set-up

- Input consists of unsegmented phonemic forms prefixed with possible topics:

$$\text{PIG}|\text{DOG}\ \imath\ z\ \eth\ æ\ t\ \eth\ ə\ p\ \imath\ g$$

  - Child-directed speech corpus collected by Fernald et al (1993)
  - Objects in visual context annotated by Frank et al (2009)
- Bayesian inference for AGs using MCMC (Johnson et al 2009)
  - Uniform prior on PYP *a* parameter
  - "Sparse" Gamma$(100, 0.01)$ on PYP *b* parameter
- For each grammar we ran 8 MCMC chains for 5,000 iterations
  - collected word segmentation and topic assignments at every 10th iteration during last 2,500 iterations
    $\Rightarrow$ 2,000 sample analyses per sentence
  - computed and evaluated the modal (i.e., most frequent) sample analysis of each sentence

Does non-linguistic context help segmentation?

Model		word segmentation
segmentation	**topics**	**token f-score**
unigram	not used	0.533
unigram	any number	0.537
unigram	one per sentence	0.547
collocation	not used	0.695
collocation	any number	0.726
collocation	one per sentence	0.719
collocation	one per collocation	**0.750**

- Not much improvement with unigram model
  - consistent with results from Jones et al (2010)
- Larger improvement with collocation model
  - most gain with *one topical word per topical collocation* (this constraint cannot be imposed on unigram model)

Does better segmentation help topic identification?

- Task: identify object (if any) *this sentence* is about

Model		sentence referent	
segmentation	**topics**	**accuracy**	**f-score**
unigram	not used	0.709	0
unigram	any number	0.702	0.355
unigram	one per sentence	0.503	0.495
collocation	not used	0.709	0
collocation	any number	0.728	0.280
collocation	one per sentence	0.440	0.493
collocation	one per collocation	**0.839**	**0.747**

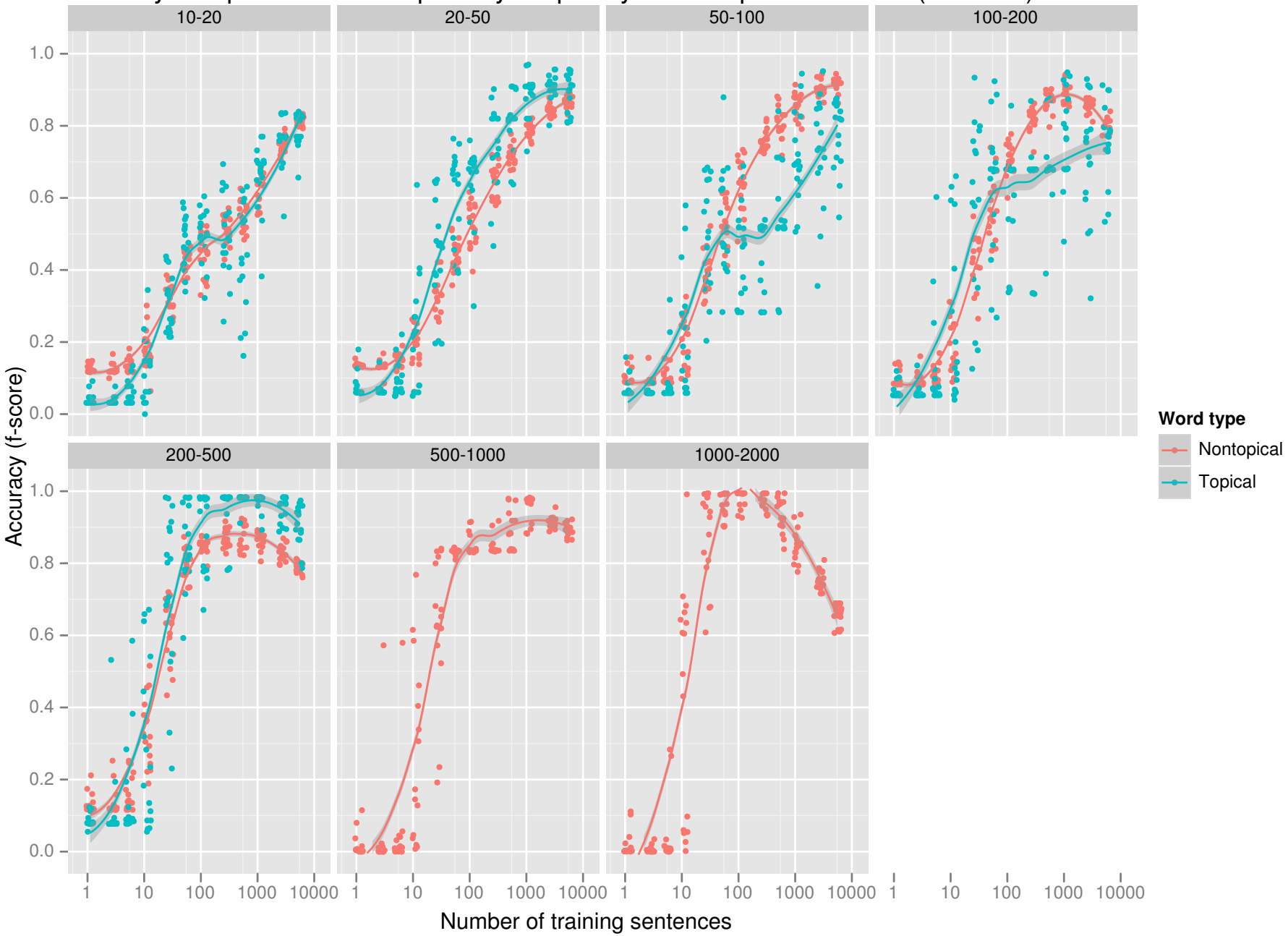
- The collocation grammar with *one topical word per topical collocation* is the only model clearly better than baseline

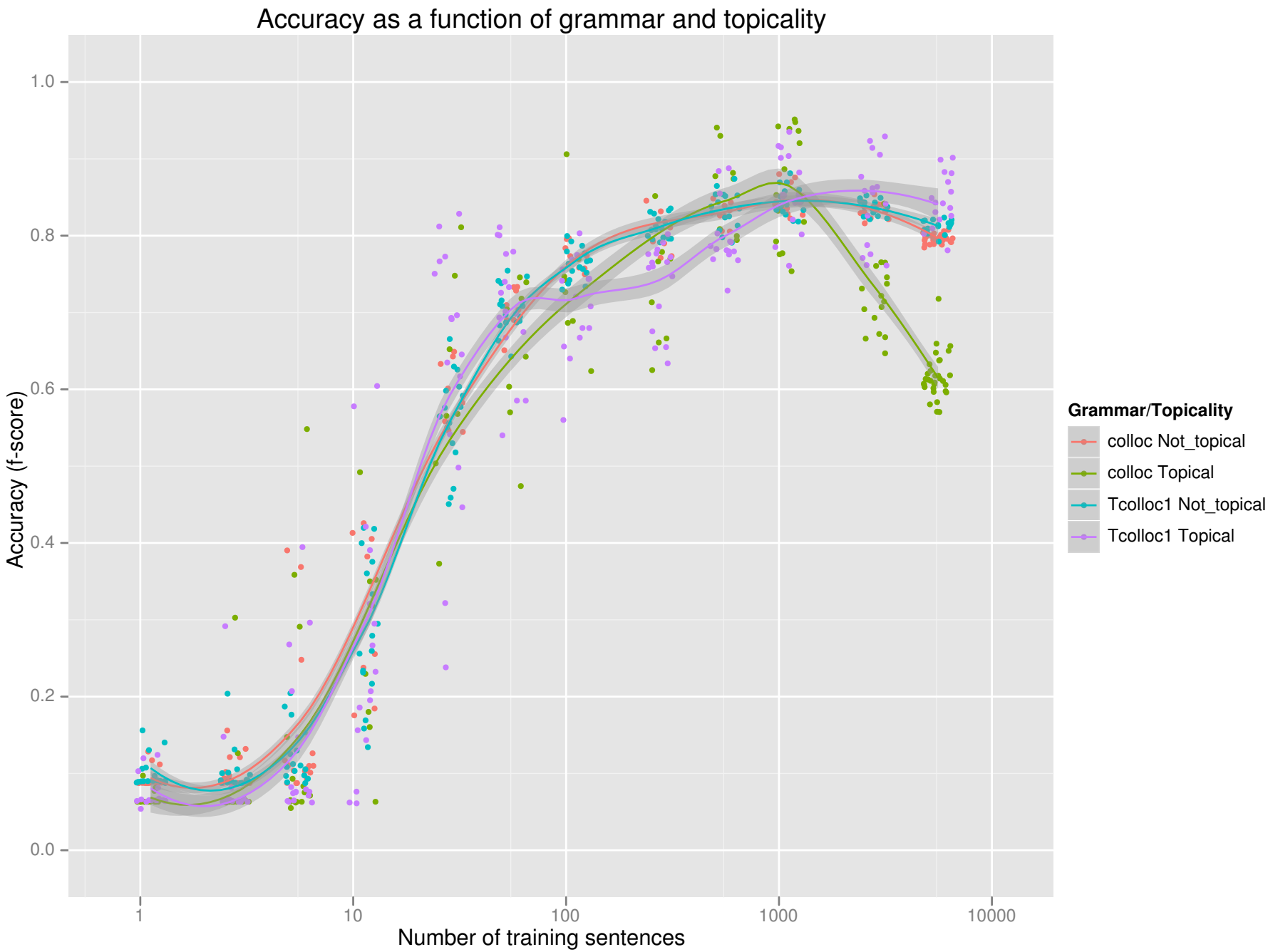
Does better segmentation help learning word-to-referent mappings?

- Task: identify *head nouns* of NPs referring to topical objects (e.g. *pɪg* $\rightsquigarrow$ PIG in input PIG | DOG ɪ z ð æ t ð ə p ɪ g)

Model		topical word
segmentation	**topics**	**f-score**
unigram	not used	0
unigram	any number	0.149
unigram	one per sentence	0.147
collocation	not used	0
collocation	any number	0.220
collocation	one per sentence	0.321
collocation	one per collocation	**0.636**

- The collocation grammar with one topical word per topical collocation is best at identifying head nouns of referring NPs


Accuracy of topical and non-topical by frequency under topic-collocation (Tcolloc1) model
10-20
20-50
50-100
100-200
200-500
500-1000
1000-2000
Accuracy (f-score)
0.0
0.2
0.4
0.6
0.8
1.0
1
10
100
1000
10000
Number of training sentences
Word type
Nontopical
Topical



MACQUARIE UNIVERSITY

Summary of grounded learning and word segmentation

- *Word to object mapping is learnt more accurately when words are segmented more accurately*
  - improving segmentation accuracy improves topic detection and acquisition of topical words
- *Word segmentation accuracy improves when exploiting non-linguistic context information*
  - incorporating word-topic mapping improves segmentation accuracy (at least with collocation grammars)

$\Rightarrow$ *There are synergies a learner can exploit when learning word segmentation and word-object mappings*

- Caveat: results seem to depend on details of model

- Models limited by ability to simulate "feature-passing" in a PCFG

Why study social cues?

- Everyone agrees social interactions are important for children's early language acquisition
  - e.g. children who engage in more joint attention with caregivers (e.g., looking at toys together) learn words faster (Carpenter 1998)
- *Can computational models exploit social cues?*
  - we show this by building models that can exploit social cues, and show they *learns better on data with social cues than on data with social cues removed*
- Many different social cues could be relevant: *can our models learn the importance of different social cues?*
  - our models estimate *probability of each cue occuring with "topical objects"* and *probability of each cue occuring with "non-topical objects"*
  - they do this in an unsupervised way, i.e., they are not told which objects are topical

Exploiting social cues for learning word referents

- Frank et al (2012) corpus of 4,763 utterances with the following information:
  - the orthographic words uttered by the care-giver,
  - a set of *available topics* (i.e., objects in the non-linguistic objects),
  - the values of the social cues, and
  - a set of *intended topics*, which the care-giver refers to.
- Social cues annotated in corpus:

Social cue	**Value**
child.eyes	objects child is looking at
child.hands	objects child is touching
mom.eyes	objects care-giver is looking at
mom.hands	objects care-giver is touching
mom.point	objects care-giver is pointing to

- Frank et al (2012) give extensive information on corpus, including inter-annotator reliability and statistical analyses between the social cues, available topics and intended topics, and instructions obtaining the corpus

Modeling social cue learning as grammatical inference

- Every utterance comes with a list of *available topics*, and the *social cues* associated with each topic
- Idea: encode the available topics and social cues as a *prefix prepended to the utterance*

  .dog # .pig child.eyes mom.eyes mom.hands # ## wheres the pig

- All our grammars associate each utterance with zero or one topics
  - $\Rightarrow$ always misanalyse utterances with more than one topic
    - utterances with no topic are analysed as having the topic None.
- Our grammars parse the prefix and the utterance as separate subtrees, each associated with a topic
- Top-level grammar rules require that prefix and utterance have same topic

Example utterance and its encoding as a string

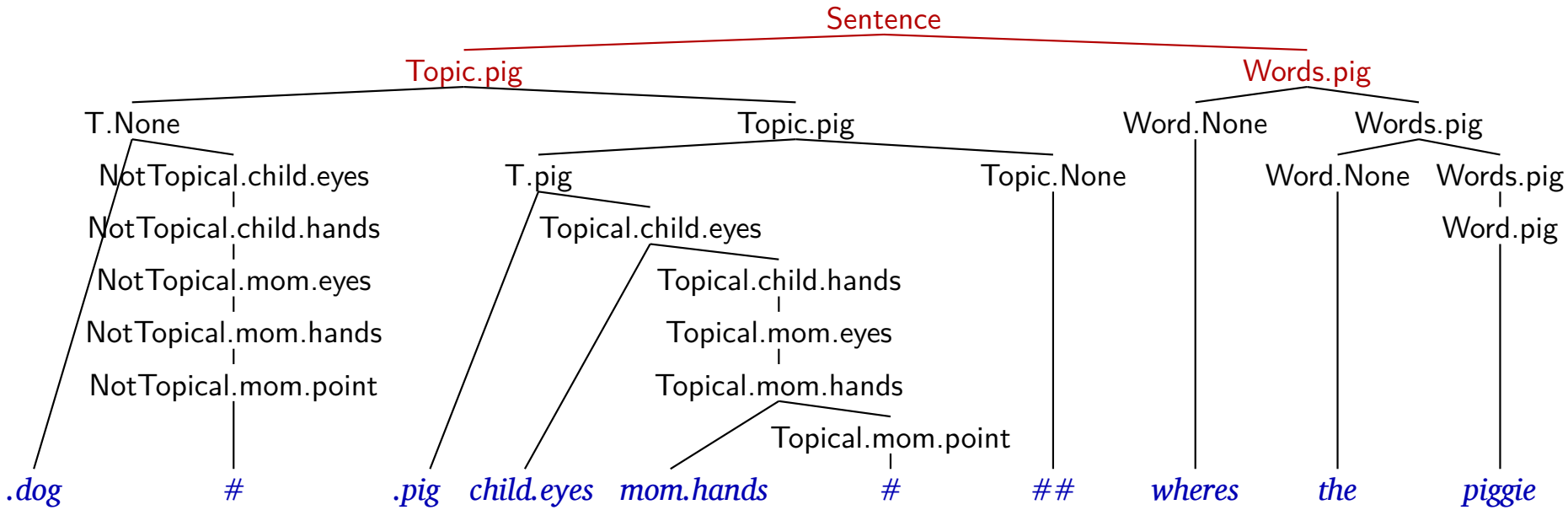
Input to learner:

.dog # .pig child.eyes mom.eyes mom.hands # ## wheres the piggie

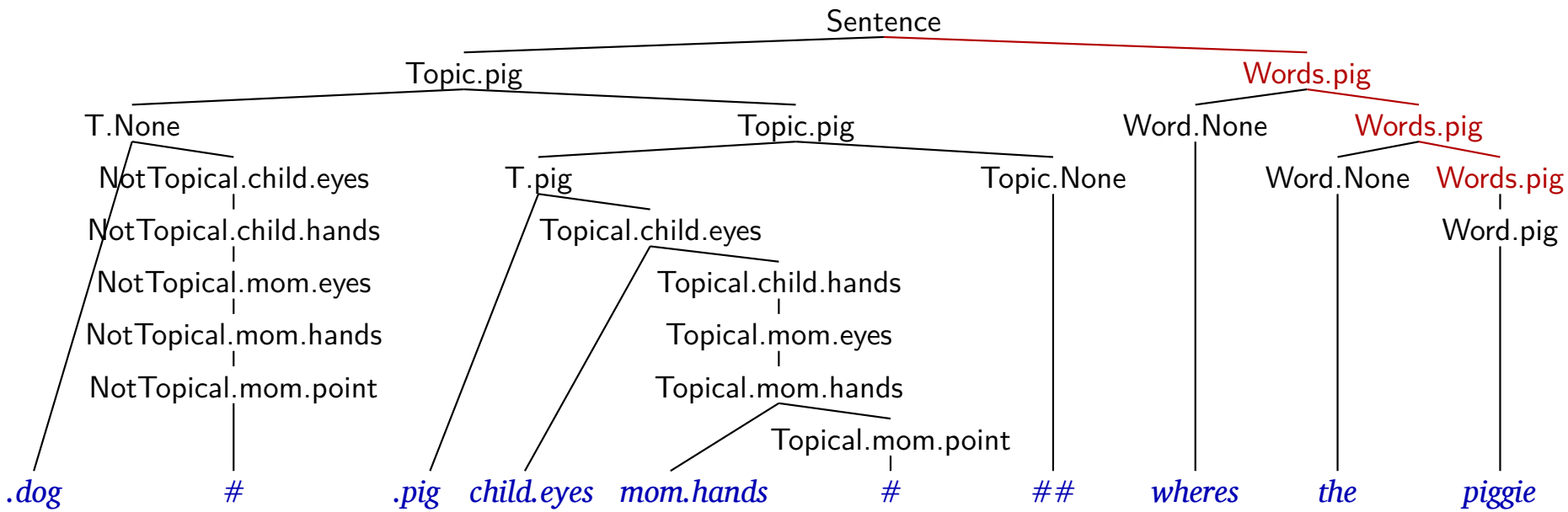
Intended topic: *.pig*

Word-topic associations: *piggie* $\rightsquigarrow$ *.pig*

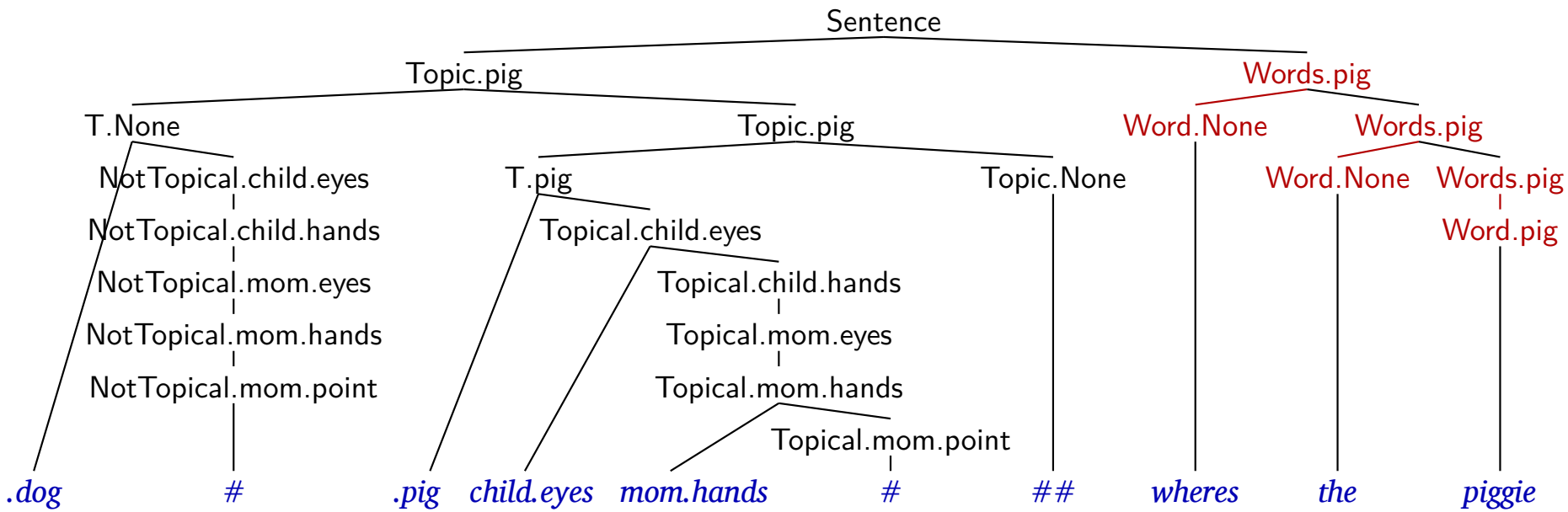
Sharing topic between prefix and utterance



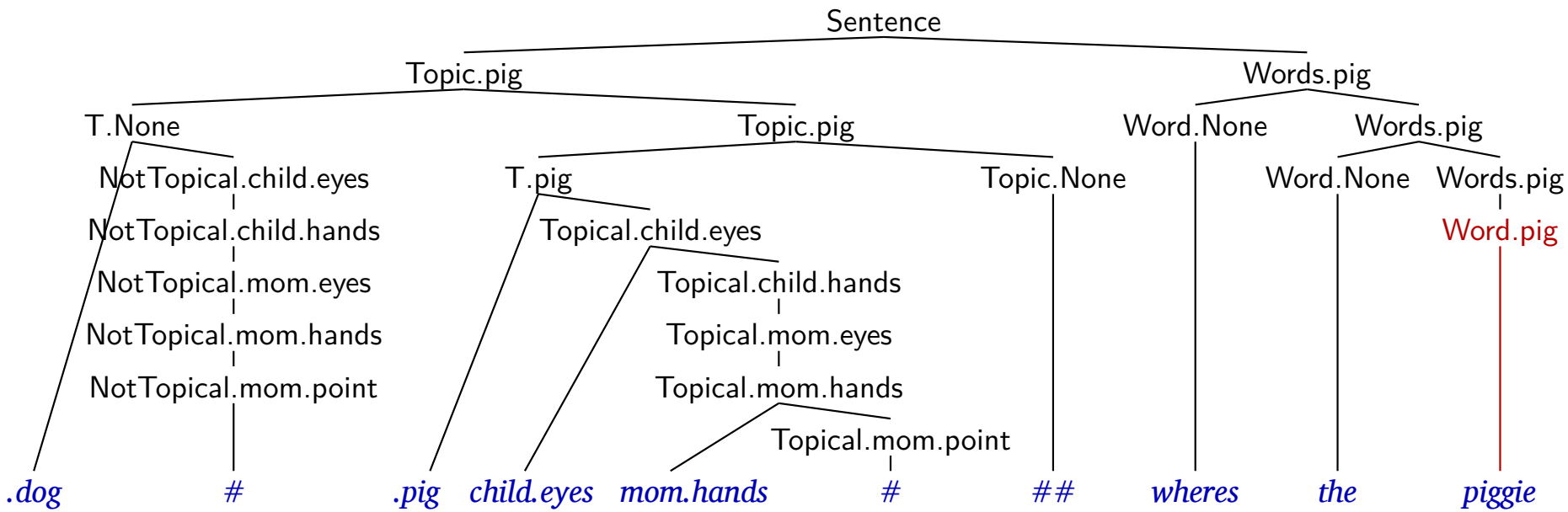
Propagating topic through utterance



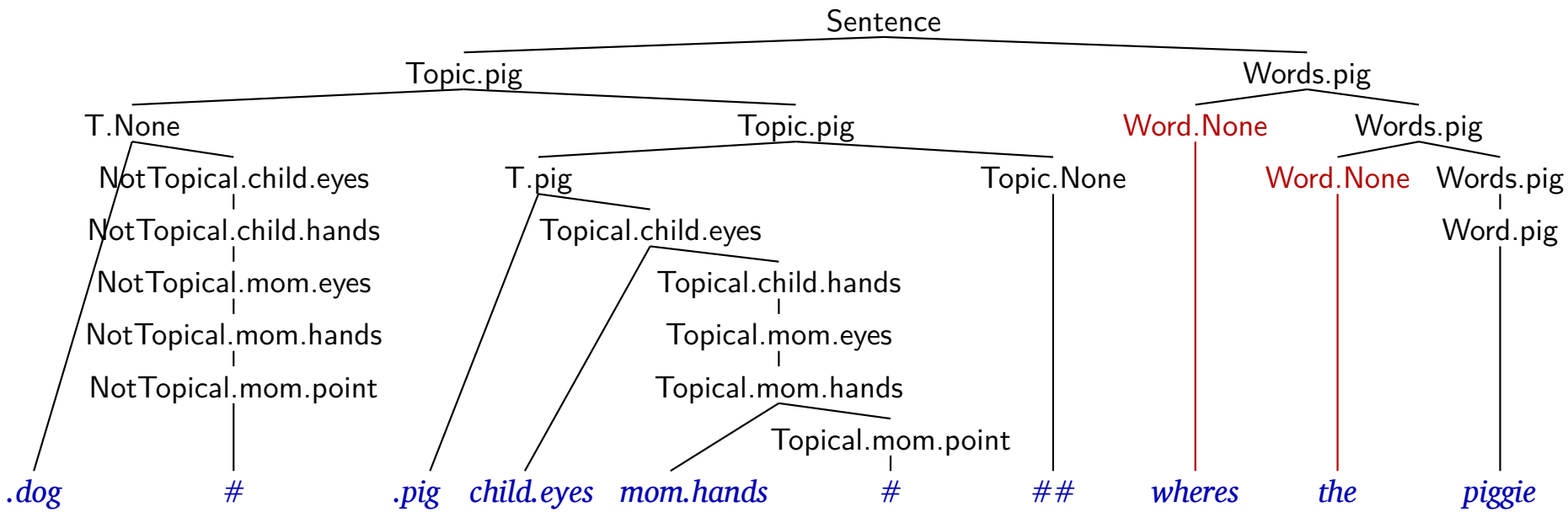
Choosing which words are topical



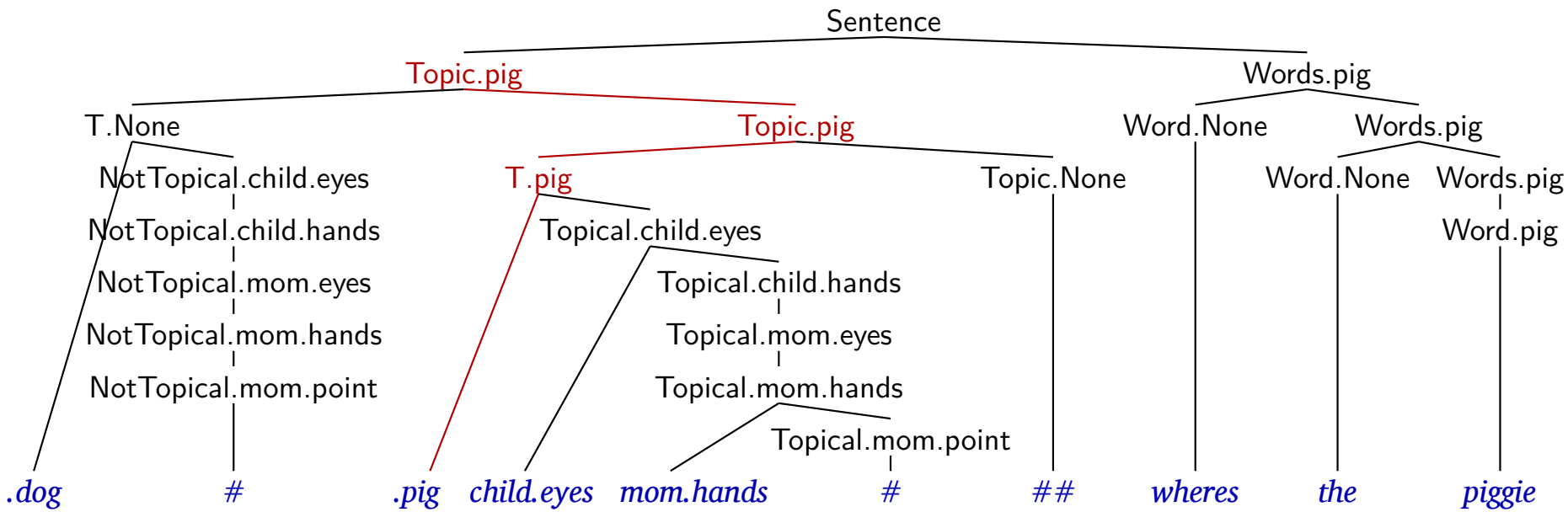
Generating topical words



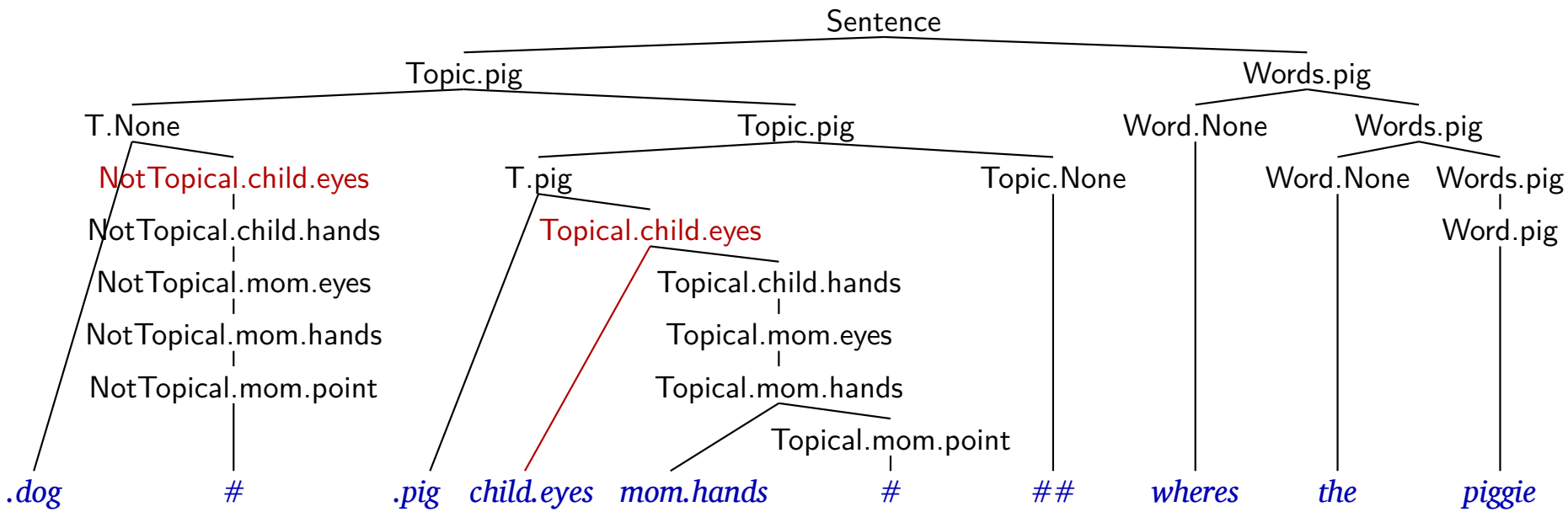
Generating non-topical words



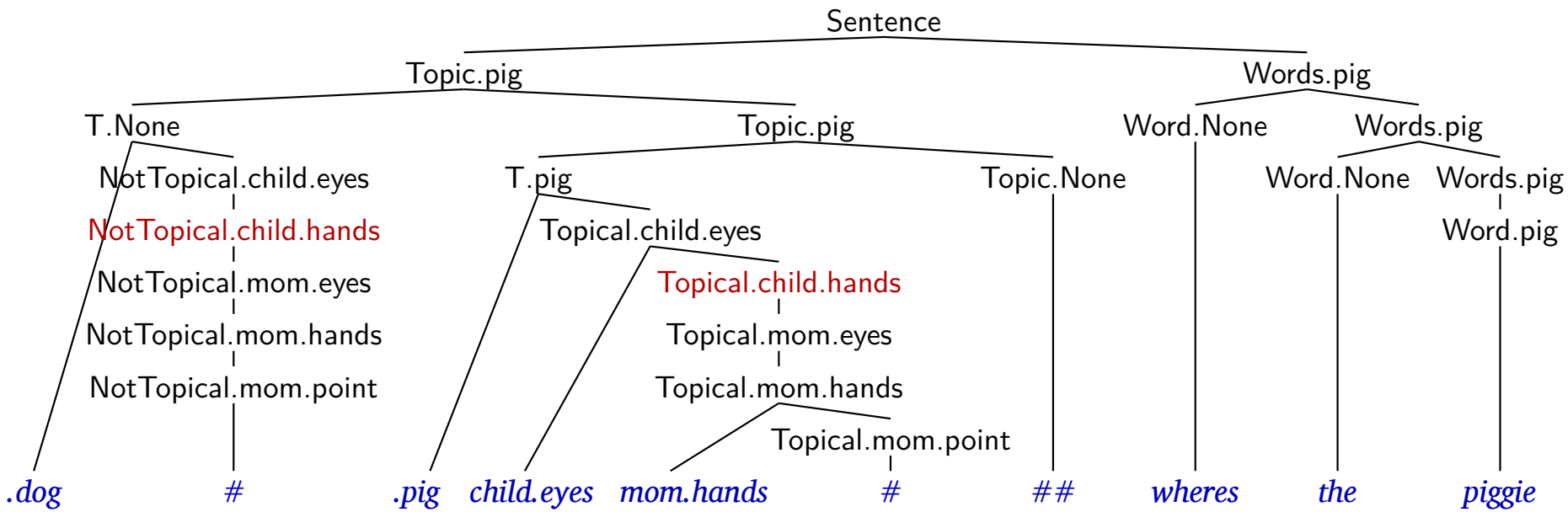
Selecting a topic from available topics



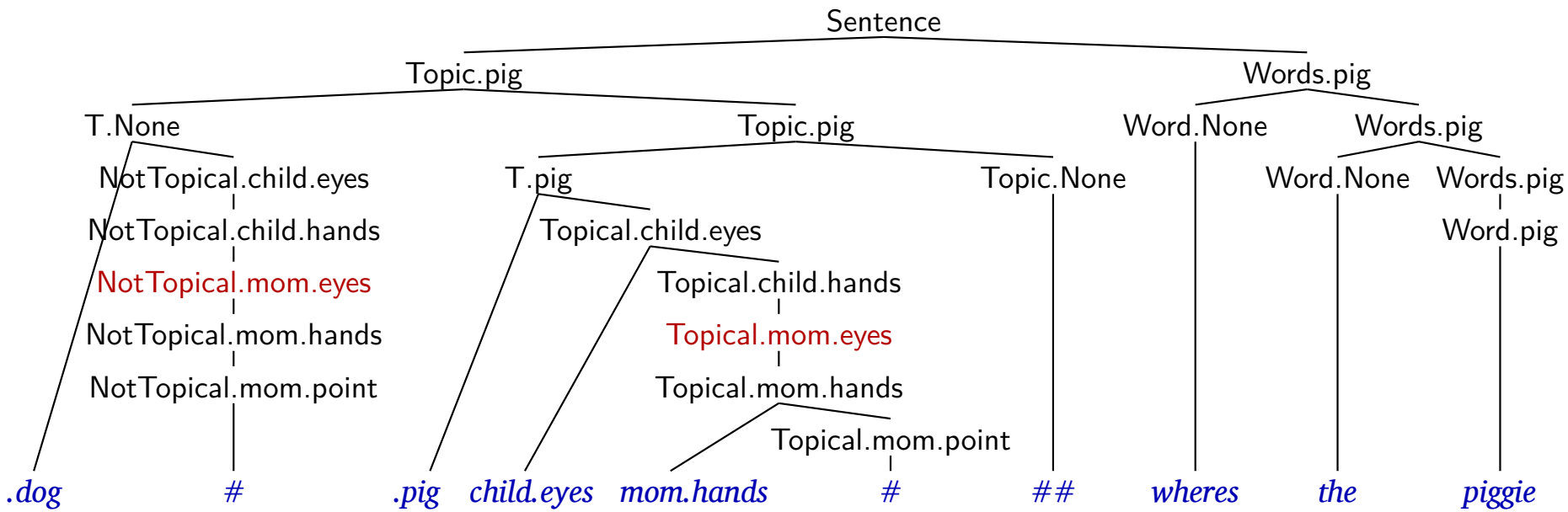
Generating social cues (child.eyes)



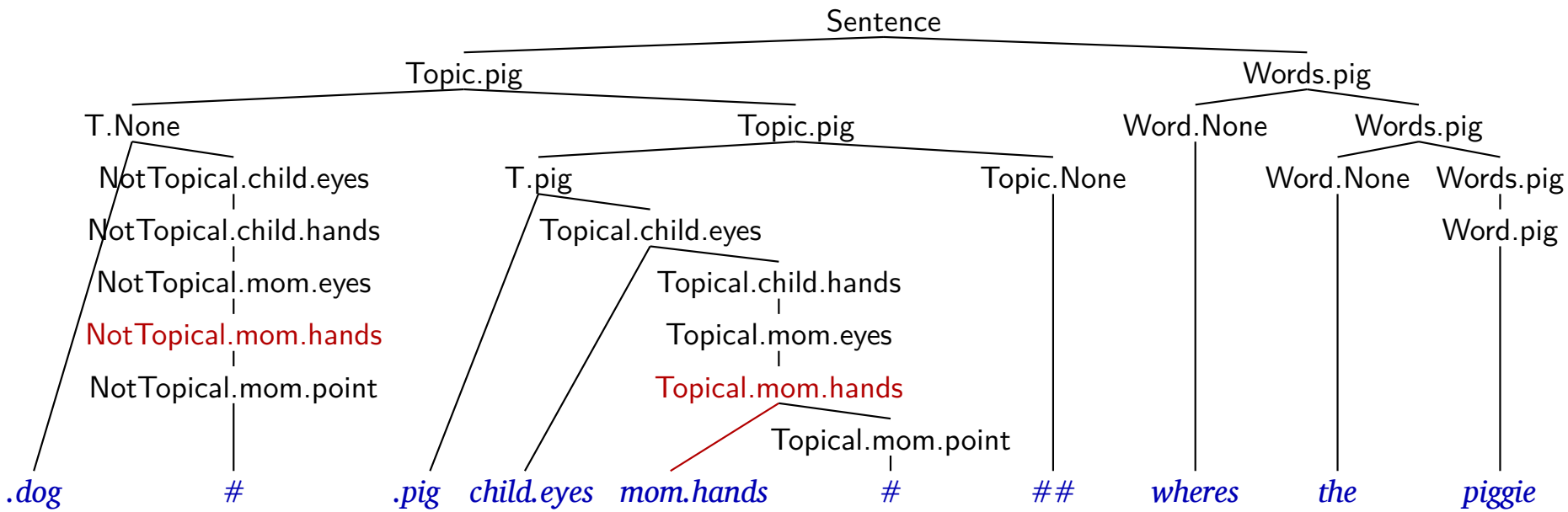
Generating social cues (child.hands)



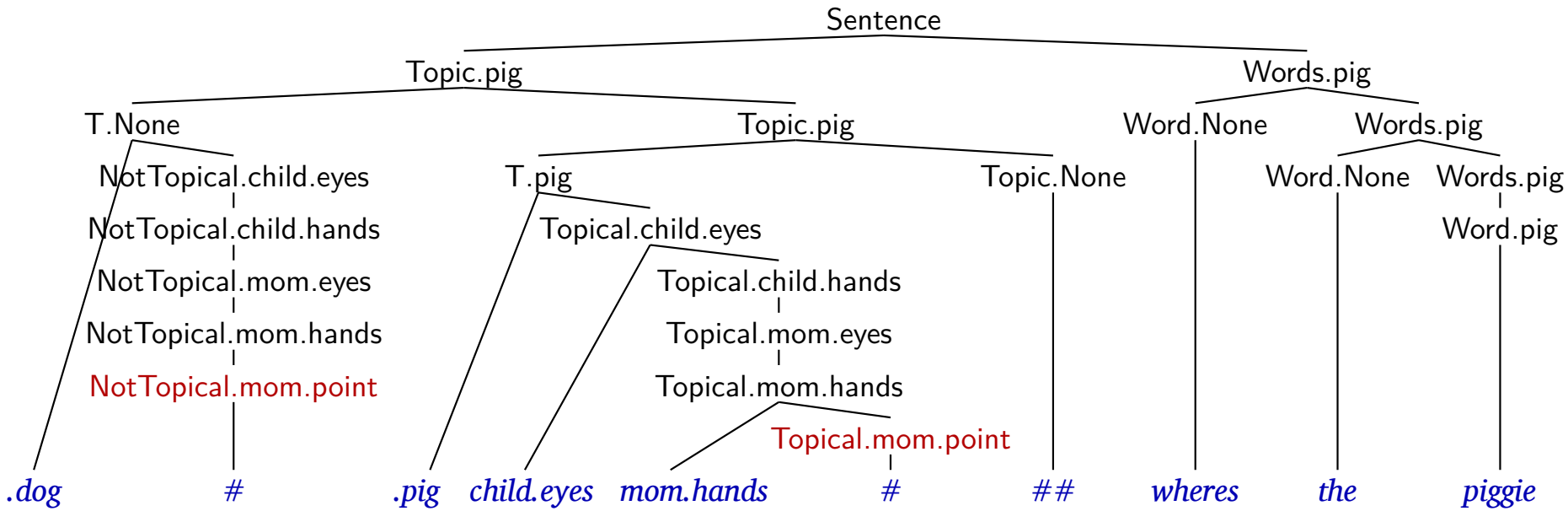
Generating social cues (mom.eyes)



Generating social cues (mom.hands)



Generating social cues (mom.point)



Results for learning social cues

- In the four different models we tried, *social cues* improved the accuracy of:
  - recovering the *utterance topic*
  - identifying the *word(s) referring to the topic*, and
  - *learning a lexicon* (word $\rightsquigarrow$ topic mapping)
- *kideyes* was the most important social cue for each of these tasks in all of the models
- Social cues don't seem to improve word segmentation
  - is this interesting to anyone, or are only positive results publishable?

What have we achieved so far?

- Close to 90% token f-score in word segmentation with models combining:
  - distributional information (including collocations)
  - syllable structure
- Synergies in learning
- Where is the remaining 10%?
- Grounded learning of word $\rightsquigarrow$ topic mapping
  - improves word segmentation
  - another synergy in learning
- Social cues improve grounded learning
  - but not word segmentation (so far)

Where we could go from here?

- Model phonological and morpho-phonological alternations
  - at Brown we forced-aligned the Providence corpus to study /d/ and /t/ deletion
- Replace the phonemic segments with *phonetic feature bundles*
  - adaptor grammar framework would need major extensions
  - Indian Buffet Process (?)
- Develop a *new model that jointly learns mapping from acoustics to words*
  - acoustic signal

  $\Rightarrow$ acoustic/phonetic features

  $\Rightarrow$ phonemic inventory (is this necessary?)

  $\Rightarrow$ (morpho)-phonemic alternations

  $\Rightarrow$ lexical items