Modelling function words improves unsupervised word segmentation

Mark Johnson[1,2], Anne Christophe[3,4], Katherine Demuth[2,6]
Emmanuel Dupoux[3,5]

[1] Department of Computing, Macquarie University, Sydney, Australia
[2] Santa Fe Institute, Santa Fe, New Mexico, USA
[3] Ecole Normale Supérieure, Paris, France
[4] Centre National de la Recherche Scientifique, Paris, France
[5] Ecole des Hautes Etudes en Sciences Sociales, Paris, France
[6] Department of Linguistics, Macquarie University, Sydney, Australia

June 2014

Function vs. content words

- Examples of function words: *the, a, is, are, can, will, in, on . . .*
- Function words:
  1. belong to *closed classes*
  2. have *high token frequency count*
  3. are *morphologically and phonologically simple*
  4. appear in *phrase-peripheral position*
  5. are associated with *specific syntactic categories*
  6. are *semantically more complex* than the corresponding content words
  7. have a *lower rate of innovation* than content words
- Our model captures properties 3 and 4

Outline

Do function words have a special role in language acquisition?

- Some psychologists believe that children pay special attention to function words in early language acquisition (Shi et al 2006, Halle et al 2008)
  - function words typically have high frequency and are phonologically simple ⇒ easy to learn
  - function words typically appear in phrase-peripheral positions ⇒ provide "anchors" for word and phrase segmentation
  - function words can identify syntactic category and syntactic structure (Christophe et al 2008, Demuth et al 2009)
- *Can we use computational models to investigate whether function words are treated specially in language acquisition?*

MACQUARIE UNIVERSITY

Using computational models to study the role of function words

- A "controlled experiment" using computational models
  - construct two computational models that differ only in how they treat function words
  - the model that treats function words specially performs word segmentation 4% more accurately than the model which does not
  - $\Rightarrow$ treating function words specially can improve language learning
  - also sets a *new state-of-the-art in word segmentation* (92.4% token f-score on Bernstein-Ratner corpus)
- Can we identify basic syntactic properties of function words?
  - do they attach to the left or the right periphery?
  - Bayesian model selection correctly identifies left-periphery attachment as overwhelmingly more likely

Outline

Function words and their role in language acquisition

Hierarchical non-parametric models of word learning

Adaptor grammars with and without "function words"

Experiments on a word segmentation task

Left or right attachment of function words

Conclusions and future work

Word segmentation as simplified word learning

- Input: a corpus of *unsegmented utterances*, constructed by:
  - use *pronouncing dictionary* to map each word of *orthographic child-directed speech transcript* to its pronunciation
  - append pronunciation of each word to obtain utterance pronunciation
- Example input:

j △ *u* ▲ *w* △ *a* △ *n* △ *t* ▲ *t* △ *u* ▲ *i* △ *t* ▲ *ð* △ *ə* ▲ *k* △ *ʊ* △ *k* △ *i*

ju want tu it ðə kʊki

"you want to eat the cookie"

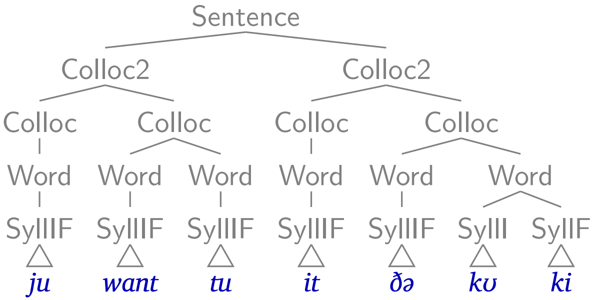
- Evaluation: how accurately the model recovers the original word boundaries

$\Rightarrow$ Identifies the pronunciations of the words of a language

Useful information for word segmentation

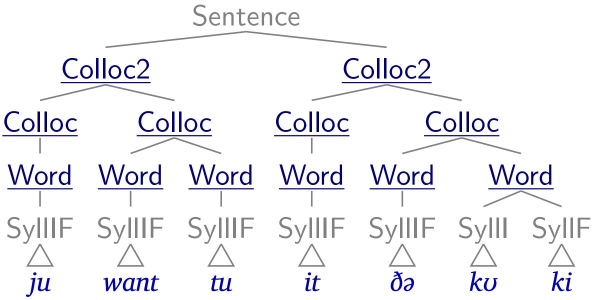
- *Vocabulary of the language*
  - no obvious upper bound ⇒ *non-parametric* learning
- *Exhaustive parsing* (no unparsed speech)
- *Phonotactics* (e.g., syllable structure constraints)
- *Distributional cues* (e.g., collocations)
- Prosodic cues (e.g., stress) (see Börschinger et al, this conference)
- Semantic constraints (e.g., word-topic mappings)
- Social cues (e.g., care-giver's eye-gaze)
- This work: *function words*

Weaknesses of PCFGs for word segmentation



- Trees can *represent* the hierarchical structures required
- But PCFG rules *don't capture the appropriate generalisations*
  - e.g., probability of rule Word → SyllI SyllF encodes how likely 2-syllable words are
  - but the PCFG *doesn't learn* that *kʊki* is a word!

Adaptor grammars memoise entire subtrees



- *Adaptor Grammars* learn the probability of *adapted nonterminals* expanding to *entire subtrees* (as well as probability of CFG rules)
  - e.g. probability of Word $\Rightarrow^{+}$ *kʊki* and Word $\rightarrow$ SyllI SyllF
  - defined as a hierarchy of Pitman-Yor Processes
  - adapted non-terminals are underlined and highlighted

Outline

Function words and their role in language acquisition

Hierarchical non-parametric models of word learning

Adaptor grammars with and without "function words"

Experiments on a word segmentation task

Left or right attachment of function words

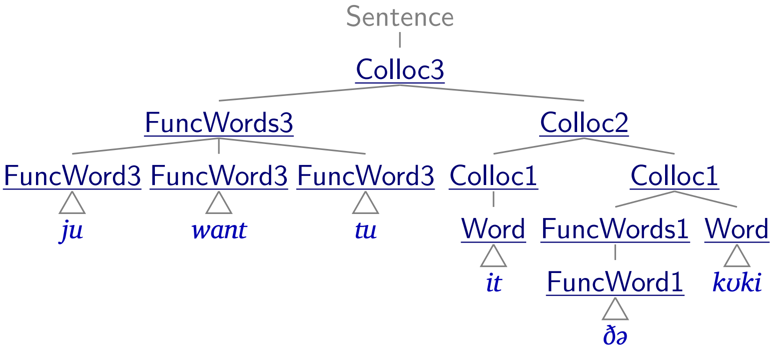
Conclusions and future work

Adaptor grammar with function words on left

$$\begin{array}{rcl} \text{Sentence} & \rightarrow & \text{Colloc3}^+ \\ \underline{\text{Colloc3}} & \rightarrow & (\text{FuncWords3})\ \text{Colloc2}^+ \\ \underline{\text{Colloc2}} & \rightarrow & (\text{FuncWords2})\ \text{Colloc1}^+ \\ \underline{\text{Colloc1}} & \rightarrow & (\text{FuncWords1})\ \text{Word}^+ \end{array} \qquad \begin{array}{rcl} \underline{\text{FuncWords3}} & \rightarrow & \text{FuncWord3}^+ \\ \underline{\text{FuncWord3}} & \rightarrow & \text{SyllableIF} \\ \underline{\text{FuncWords2}} & \rightarrow & \text{FuncWord2}^+ \\ \underline{\text{FuncWord2}} & \rightarrow & \text{SyllableIF} \\ \underline{\text{FuncWords1}} & \rightarrow & \text{FuncWord1}^+ \\ \underline{\text{FuncWord1}} & \rightarrow & \text{SyllableIF} \end{array}$$

- Johnson and Goldwater (2009) grammar plus function words
- 3 levels of collocations
  - collocations often correspond to syntactic phrases
- Each collocational level has its own set of “function words”
- “Function words” are:
  - always *monosyllabic*, and
  - appear on *left periphery of collocations*

Sample parse generated by function word grammar



- 3 levels of collocations (as in Johnson and Goldwater 2009)
- Each collocational level has its own set of "function words"
- "Function words" are:
  - always *monosyllabic*, and
  - appear on *left periphery of collocations*

Outline

Function words and their role in language acquisition

Hierarchical non-parametric models of word learning

Adaptor grammars with and without "function words"

Experiments on a word segmentation task

Left or right attachment of function words

Conclusions and future work

Experimental set-up

- The models we compare:
  1. "Function words" on left periphery (just described)
  2. "Function words" on right periphery (mirror image)
  3. "Function words" on left and right (union of these)
  4. No function words (as in Johnson and Goldwater 2009)
  5. All words are monosyllabic
- Trained on varying-length prefixes of the Bernstein and Ratner (1987) corpus
- Tested on the whole Bernstein and Ratner (1987) corpus

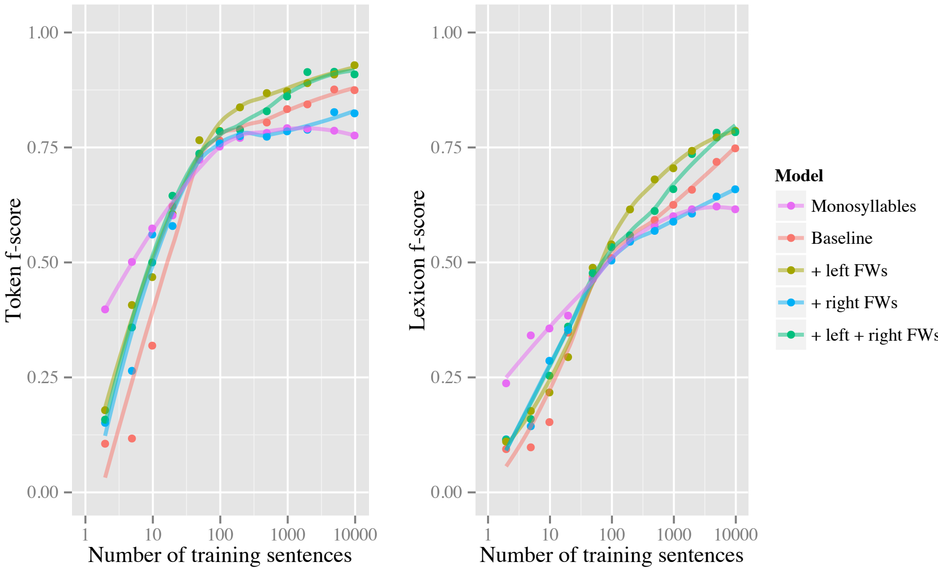
Computational set-up

- All models use the same Adaptor Grammar software with the same hyperparameter settings
  - only the adaptor grammars vary

$\Rightarrow$ Any observed differences are due to differences in the models as encoded in the grammars (not implementation differences)

- Computational details (same as in Johnson and Goldwater 2009):
  - AG software uses a MCMC Metropolis-within-Gibbs algorithm
  - slice sampling for all Pitman-Yor hyperparameters with "vague priors"
  - 8 MCMC runs for each setting, each with 2,000 sweeps of training data
  - collect every 10th sweep of last 1,000 sweeps
  - identify most frequent segmentation for each utterance from these 800 samples

Learning curves for "function word" models



MACQUARIE UNIVERSITY

Learning curve results discussion

- Word-token f-score and Lexicon (i.e., word-type) are very similiar
- Monosyllabic word model does very well on small data
- After several hundred sentences:
  - right "function word" model does worse than no "function word" model
  - left "function word" model initially does better than left+right "function word" model
  - eventually both left and left+right "function word" models do better than no "function word" model
- All models except the monosyllabic word model are improving at 100,000 sentences
  - would probably improve more if given more data

⇒ *Modelling "function words" improves word segmentation*

MACQUARIE UNIVERSITY

Analyses generated by left "function words" model

- 5 most-frequent words in each category over 8 MCMC runs:

  Word : *book, doggy, house, want, I*
  FuncWord1 : *a, the, your, little, in*
  FuncWord2 : *to, in, you, what, put*
  FuncWord3 : *you, a, what, no, can*

- Even though model is designed for word segmentation, it seems to make reasonable content/function word distinction
- Could this be useful for syntactic bootstrapping?

Word segmentation results

Model	**Token f-score**	**Boundary precision**	**Boundary recall**
Baseline	0.872	0.918	0.956
+ left FWs	**0.924**	0.935	**0.990**
+ left + right FWs	0.912	**0.957**	0.953

- Mean token f-scores and boundary precision and recall results averaged over 8 trials
  - each trial consisted of 8 MCMC runs, trained and tested on full Bernstein and Ratner (1987) corpus
  - standard deviations of all values < 0.006
  - means of all token f-scores differ $p < 2\text{e-}4$ (Wilcox sign test)
- New state-of-the-art for token f-score in word segmentation
  - actual score is scientifically uninteresting

Outline

Function words and their role in language acquisition

Hierarchical non-parametric models of word learning

Adaptor grammars with and without "function words"

Experiments on a word segmentation task

Left or right attachment of function words

Conclusions and future work

Do "function words" attach left or right?

- The left "function words" model has much higher segmentation accuracy than the right "function words" model
  - English function words are almost always on the left periphery
  - can an unsupervised learner determine this somehow?
- *Bayesian model selection* uses *Bayes factors* K to identify the more likely model given training data D:

$$K = \frac{\mathrm{P}(D \mid G_1)}{\mathrm{P}(D \mid G_2)}, \text{ where:}$$

$$\mathrm{P}(D \mid G) = \int_{\Delta} \mathrm{P}(D, \boldsymbol{\theta} \mid G)\, d\boldsymbol{\theta}$$

  where Δ is the cross-product of all possible:
  - parses for the utterances in D,
  - Chinese Restaurant Process configurations in the sampler, and
  - values for the hyper-parameters
- This integral is intractable (no surprise)

MACQUARIE UNIVERSITY

Estimating the marginal likelihood of the data

- Bayesian model selection involves computing the *marginal likelihood*

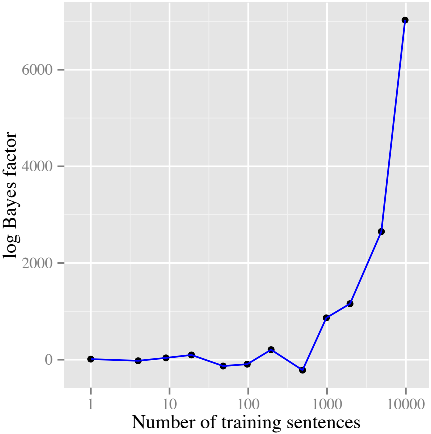
$$\mathrm{P}(D \mid G) \;=\; \int_{\Delta} \mathrm{P}(D, \boldsymbol{\theta} \mid G)\, d\boldsymbol{\theta}$$

- Given a sequence of samples $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n$ from $\mathrm{P}(\boldsymbol{\theta} \mid D, G)$, the *Harmonic Mean Estimator* approximates the marginal likelihood as:

$$\mathrm{P}(D \mid G) \;\approx\; \left(\frac{1}{n} \sum_{i=1}^{n} \frac{1}{\mathrm{P}(D, \boldsymbol{\theta}_i \mid G)} \right)^{-1}$$

  - the MCMC procedure generates samples from $\mathrm{P}(\boldsymbol{\theta} \mid D, G)$
  - $\mathrm{P}(D, \boldsymbol{\theta}_i \mid G)$ is (relatively) easy to calculate
- *Warning: the Harmonic Mean estimator is "the worst MCMC method ever" (Radford Neal)*

Bayes factor in favour of left attachment



- After 1,000 sentences there is *overwhelming evidence* in favour of left-peripheral "function words"
  - but remember warning about Harmonic Mean estimator

Outline

Function words and their role in language acquisition

Hierarchical non-parametric models of word learning

Adaptor grammars with and without “function words”

Experiments on a word segmentation task

Left or right attachment of function words

Conclusions and future work

Conclusions

- A state-of-the-art word segmentation algorithm is significantly improved when it learns generalisations about function words

$\Rightarrow$ This suggests that function words provide useful information for language acquisition in addition to the distributional and phonotactic information the model already exploits

- The left-peripheral "function word" model achieves highest word segmentation accuracy
- Bayes factors can be used to determine that "function words" are left-peripheral
  - instability of Harmonic Mean estimator

    $\Rightarrow$ results may be unreliable

MACQUARIE UNIVERSITY

Future work

- Find a better method for calculating Bayes factors
- Can we use these function word results to "bootstrap" an unsupervised syntactic learner?
  - a joint model with "unsupervised parsing"?
- Are there other ideas from psycholinguists that we should try to incorporate into our models?

MACQUARIE UNIVERSITY