Using adaptor grammars to identify synergies in the unsupervised acquisition of linguistic structure

Mark Johnson
Brown University

June, 2008

Summary

- *Adaptor grammars* are an extension of PCFGs
  - set of possible trees defined just as in a PCFG
  - but learns probabilities of entire subtrees (not just rules)
  - designed to generalize Goldwater's word segmentation and morphology models
- Subtrees (and their probabilities) learnt depend apon previously generated sentences $\Rightarrow$ grammar "adapts" to data
- Used to learn words in *unsupervised word segmentation*

  Example: $y_{\vartriangle}u_{\blacktriangle}w_{\vartriangle}a_{\vartriangle}n_{\vartriangle}t_{\blacktriangle}t_{\vartriangle}u_{\blacktriangle}s_{\vartriangle}i_{\blacktriangle}D_{\vartriangle}6_{\blacktriangle}b_{\vartriangle}U_{\vartriangle}k$
- By changing base grammar, we can simultaneously learn:
  - collocations
  - stem-suffix morphology
  - syllable structure
- Simultaneously learning collocations and syllable structure *significantly improves word segmentation accuracy*

Language acquisition as Bayesian inference

$$\underbrace{\mathrm{P}(\mathrm{Grammar} \mid \mathrm{Data})}_{\text{Posterior}} \propto \underbrace{\mathrm{P}(\mathrm{Data} \mid \mathrm{Grammar})}_{\text{Likelihood}} \underbrace{\mathrm{P}(\mathrm{Grammar})}_{\text{Prior}}$$

- Likelihood measures how well grammar describes data
- Prior expresses knowledge of grammar before data is seen
  - can be very specific (e.g., Universal Grammar)
  - can be very general (e.g., prefer shorter grammars)
- Posterior is a *distribution* over grammars
  - expresses uncertainty about which grammar is correct

Using Bayesian posterior for parsing

- Usually *infinitely many* grammars G with non-zero probability in posterior $\mathrm{P}(G \mid D)$ given data D
  - pick one grammar somehow (e.g., MAP), or
  - *use full posterior distribution for parsing*
- "Integrate out" grammar G to obtain posterior distribution over parse trees T given data D

$$\mathrm{P}(T \mid D) \;=\; \int \mathrm{P}(T \mid D, G)\, \mathrm{P}(G \mid D)\, dG$$

$\Rightarrow$ Grammatical inference need not produce an explicit grammar

- We use *Markov Chain Monte Carlo* to sample directly from $\mathrm{P}(T \mid D)$

Informal description of Adaptor Grammars

- An Adaptor Grammar is a PCFG where a subset of nonterminals are specified as *adapted*
- Each adapted nonterminal A has a user-specified concentration parameter α_A
  - SIGMORPH workshop paper describes how to learn α_A
- An *unadapted nonterminal* U expands just as in a PCFG
  - to children $V_1 \ldots V_m$ with probability $\theta_{U \to V_1 \ldots V_m}$
- An *adapted nonterminal* A expands:
  - to a previously generated subtree t rooted in A with probability $\propto$ number of times t was previously selected
  - to children $B_1 \ldots B_m$ with probability $\propto \alpha_A \, \theta_{U \to V_1 \ldots V_m}$

$\Rightarrow$ "Rich get richer" power-law distribution over subtrees

$\Rightarrow$ A tree can be more probable than the subtrees it contains

Word segmentation task

- Brent corpus of 9,790 transcribed child-directed utterances of 33,399 words in Bernstein-Ratner corpus
- Phonemic representation from pronouncing dictionary
- Given utterance boundaries but not word boundaries
  Example: *l U k D * z 6 b 7 w I T h I z h & t*
- Evaluate f-score of recovered words (Goldwater et al, 2006)
- Used MCMC inference procedure from Johnson et al (2007)
  - Metropolis-within-Gibbs sampler integrating out grammar
  - samples parses from PCFG approximation (one rule for each previously generated subtree)
  - clamped concentration parameters α_A to 1, 10, 100 or 1,000
  - uniform Dirichlet prior on rule probabilities $\theta_{U \to V_1 \dots V_m}$
  - results averaged over 8 runs of 10,000 epochs each
  - software available from http://cog.brown.edu/~mj

Unigram adaptor grammar

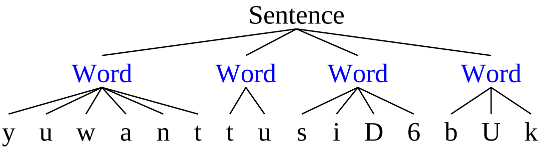
- Adaptor grammar (adapted nonterminals highlighted):

Sentence $\rightarrow$ Words
Words $\rightarrow$ Word
Words $\rightarrow$ Word Words
<u>Word</u> $\rightarrow$ Phonemes
Phonemes $\rightarrow$ Phoneme
Phonemes $\rightarrow$ Phoneme Phonemes

or in abbreviated format:

Sentence $\rightarrow$ Word$^+$
<u>Word</u> $\rightarrow$ Phoneme$^+$

- Sample parse (only showing root and adapted nonterminals):



- Word segmentation f-score = 0.55 (same as Goldwater et al)
- Can't capture dependencies between words
  $\Rightarrow$ tends to undersegment

Unigram word grammar as a Dirichlet Process

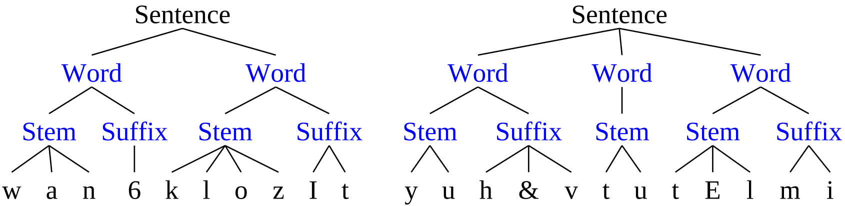
- Unigram word grammar implements unigram word segmentation model of Goldwater et al (2006)
- Generative process:
  - expand Sentence into a sequence of Words using PCFG rules
  - expand each Word into:
    - a sequence of Phonemes with prob. $\propto$ number of times Word expanded to this sequence before
    - a sequence of phonemes generated by PCFG rules with prob. $\propto \alpha_{\text{Word}}$
- This is a *Dirichlet Process* where the PCFG rules expanding Word define the *base distribution*

Unigram morphology adaptor grammar

- Adaptor grammar memorizes Word, Stem and Suffix:

$$
\begin{aligned}
\text{Sentence} &\rightarrow \text{Word}^{+} \\
\underline{\text{Word}} &\rightarrow \text{Stem (Suffix)} \\
\underline{\text{Stem}} &\rightarrow \text{Phoneme}^{+} \\
\underline{\text{Suffix}} &\rightarrow \text{Phoneme}^{+}
\end{aligned}
$$

- Sample parse:



- Combines Goldwater's morphology and unigram model
- Word segmentation f-score = 0.46 (worse than unigram)
- Tends to misanalyse words as Stems or Suffixes

Morphology grammar as a Hierarchical Dirichlet Process

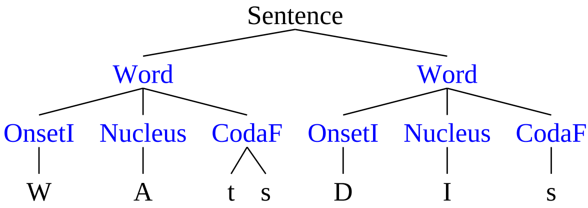
- Expand Sentence into a sequence of Word
- Expand each Word into:
  - a sequence of Phonemes with prob. $\propto$ number of times sequence was generated before
  - a Stem and optional Suffix with prob. $\propto \alpha_{\text{Word}}$
- Expand Stem into:
  - a sequence of Phoneme with prob. $\propto$ number of times Stem expanded to this sequence before
  - a sequence of Phoneme generated by PCFG rules with prob. $\propto \alpha_{\text{Stem}}$
- Suffix expands in same way as Stem
- This is a *Hierarchical Dirichlet Process* where Stem and Suffix distributions define the base distribution for Word DP

Unigram syllable adaptor grammar

- Adaptor grammar distinguishes initial and final syllables

Sentence → Word$^+$
Word → SyllableI SyllableF
Syllable → (Onset) Rhyme
SyllableF → (Onset) RhymeF
Rhyme → Nucleus (Coda)
Onset → Consonant$^+$
Coda → Consonant$^+$
Nucleus → Vowel$^+$

Word → SyllableIF
Word → SyllableI Syllable SyllableF
SyllableI → (OnsetI) Rhyme
SyllableIF → (OnsetI) RhymeF
RhymeF → Nucleus (CodaF)
OnsetI → Consonant$^+$
CodaF → Consonant$^+$

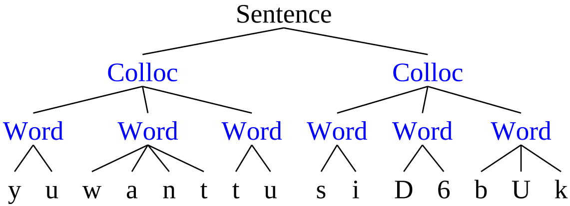


- Word segmentation f-score = 0.52 (also worse than Unigram)

Collocation adaptor grammar

- Adaptor grammar memorizes collocations (sequences of words) as well as words

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

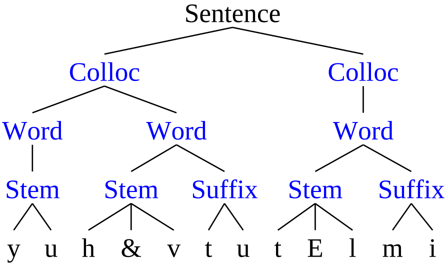


- Word segmentation f-score = 0.76 (approx same as Goldwater's bigram model)

Collocation + morphology adaptor grammar

- Adaptor grammar memorizes collocations, words, stems and suffixes

$$\text{Sentence} \rightarrow \text{Colloc}^{+}$$
$$\underline{\text{Colloc}} \rightarrow \text{Word}^{+}$$
$$\underline{\text{Word}} \rightarrow \text{Stem (Suffix)}$$
$$\underline{\text{Stem}} \rightarrow \text{Phoneme}^{+}$$
$$\underline{\text{Suffix}} \rightarrow \text{Phoneme}^{+}$$

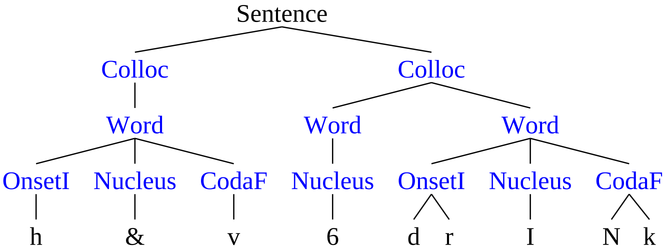


- Word segmentation f-score = 0.73 (worse than Collocation)

Collocation + syllable adaptor grammar

- Adaptor grammar is combination of collocation and syllable grammars

$$\text{Sentence} \rightarrow \text{Colloc}^{+} \qquad \underline{\text{Colloc}} \rightarrow \text{Word}^{+}$$
$$\underline{\text{Word}} \rightarrow \text{(as in syllable grammar)}$$



- Word segmentation f-score = 0.78
- Significantly better ($p = 0.006$) than Collocation on its own

Word segmentation f-score summary

		Concentration parameter α			
		1	10	100	1000
unigram	word	**0.55**	**0.55**	**0.55**	0.53
unigram	morph	**0.46**	**0.46**	0.42	0.36
unigram	syll	**0.52**	0.51	0.49	0.46
collocation	word	0.53	0.64	0.74	**0.76**
collocation	morph	0.56	0.63	**0.73**	0.63
collocation	syll	0.77	0.77	**0.78**	0.74

- Concentration parameter α tied for all adapted non-terminals

Conclusion and future work

- Adaptor grammars are a flexible framework for expressing non-parametric Bayesian models
- Probability of a parse depends on how often its subtrees were generated before $\Rightarrow$ grammar *adapts* to corpus as it parses
- This paper used Adaptor Grammars to develop several models of unsupervised word segmentation
- Confirmed Goldwater's result about importance of modeling intra-word dependencies
- No improvement found in modeling morphology
- Learning collocations and syllable structure in conjunction with word segmentation significantly improves f-score
  $\Rightarrow$ synergies in language learning
- In this work concentration parameters α are fixed, but in further work they are learned $\Rightarrow$ improves f-score to 0.84

PCFGs as recursive mixtures

- A PCFG defines distributions G_A over trees for each $A \in N \cup T$
  - if $w \in T$ then $G_w = \delta_w$ (puts all mass on singleton tree w)
  - if $A \in N$ then

$$G_A = \sum_{A \to B_1 \ldots B_n \in R_A} \theta_{A \to B_1 \ldots B_n} \mathrm{TD}_A(G_{B_1}, \ldots, G_{B_n})$$

where $\mathrm{TD}_A(G_{B_1}, \ldots, G_{B_n})$ is the distribution over trees with root label A satisfying:

$$\mathrm{TD}_A(G_1, \ldots, G_n) \left(\begin{array}{c} A \\ \overline{t_1 \quad \ldots \quad t_n} \end{array} \right) = \prod_{i=1}^{n} G_i(t_i).$$

$\mathrm{TD}_A(G_1, \ldots, G_n)$ is the distribution over trees wit root node A and each subtree t_i is generated *independently* from G_i.

Adaptor grammars

- An adaptor grammar is just like a PCFG, except that each adapted nonterminal's distribution is passed through a Dirichlet Process

$$
\begin{array}{rcll}
H_A & = & \displaystyle\sum_{A \to B_1 \ldots B_n \in R_A} \theta_{A \to B_1 \ldots B_n} \mathrm{TD}_A(G_{B_1}, \ldots, G_{B_n}) & \\
G_A & \sim & \mathrm{DP}(\alpha_A, H_A) & \text{if } A \text{ is adapted} \\
G_A & = & H_A & \text{if } A \text{ is not adapted}
\end{array}
$$

- The Dirichlet Process concentrates mass on frequently used subtrees
- Implemented using Chinese Restaurant Processes