Strong learning of some Probabilistic Multiple Context-Free Grammars

Alexander Clark

CLASP,
Department of Philosophy, Linguistics and Theory of Science,
University of Gothenburg
alexsclark@gmail.com

MOL 2021

Outline

- What is the problem?
- Motivation
- Why is it hard?
- PCFG setting
- Problem with extension to MCFG
- Elementary solution using Dyck languages
- Discussion

The Strong Probabilistic Learning Problem

Horning [1969]

- We have a sequence of *strings* drawn i.i.d. from a distribution defined by a probabilistic grammar/automaton
  - PDFA [Clark and Thollard, 2004]
  - HMM [Stratos et al., 2016]
  - PCFG [Clark and Fijalkow, 2020]
  - Probabilistic Multiple Context-Free Grammars (this paper)
- We want to learn the grammar and the parameters to arbitrary accuracy.
  - Input: only the sample of strings
  - Output: converges to a grammar *isomorphic* to the original grammar, and with *same parameters*

Why?

Motivation

First language acquisition:

Key question:

- Do the surface strings contain enough information to infer syntactic structure?
- Or must the learner rely on other sources of information (semantic, prosodic, innate . . .)?

Why?

Motivation

First language acquisition:
Key question:

- Do the surface strings contain enough information to infer syntactic structure?
- Or must the learner rely on other sources of information (semantic, prosodic, innate . . .)?

Caveat

Some tension in this paper between validity of modeling assumptions and the desire for mathematical cleanliness.

Mildly context-sensitive languages

Grammar class

Well-nested MCFGs [Seki et al., 1991] of dimension 2

- TAG, LIG, HG, CCG (depending on the version)[Joshi et al., 1990]
- Assume standard restrictions on the rule format: non-deleting, non-permuting, epsilon-free,. . .

Smallest class which is not definitely descriptively inadequate for natural language syntax.

- Weakly and strongly more powerful than CFGs.
- Only a few cases where the additional power is definitely necessary weakly [Shieber, 1985] . . .
- but lots of cases where we need the additional structural power.

Contributions of this paper

It's about understanding the problems of moving strong learning from CFGs to MCFGs:

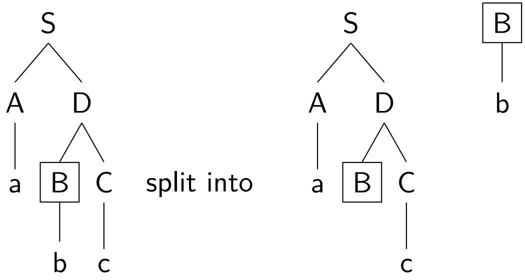
Negative There is a serious technical problem about identifying discontinuous constituents.

Positive We can overcome this quite naturally under some unreasonably strong restrictions on the class of grammars. This gives a strong learning algorithm for a small class of probabilistic MCFGs.
Simplest and most direct way of solving this problem

Context Free Grammars

CFG in Chomsky Normal Form:

Set of productions P of the form $A \to BC$ or $A \to a$
S only occurs on the left hand side of productions.



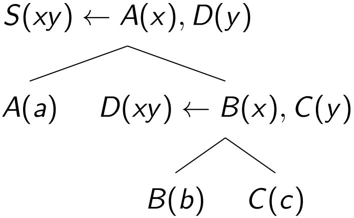
Tree

Context
$a\Box c$

Yield
b

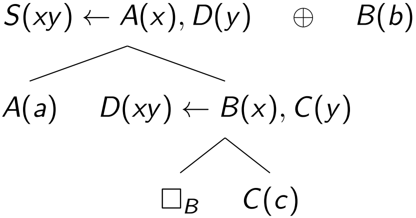
Context Free Grammars

- Write productions in Horn clause notation
- Label derivation tree with productions



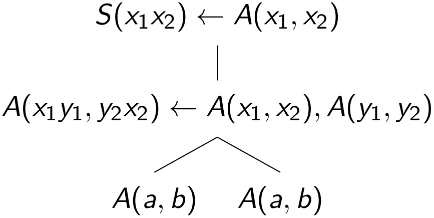
Context Free Grammars

- Write productions in Horn clause notation
- Label derivation tree with productions



Multiple Context Free Grammars, dimension 2

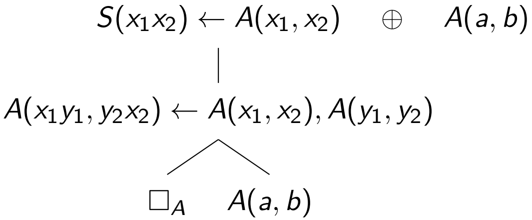
- Have some nonterminals that generate pairs of strings, rather than strings:
  A generates the set of pairs $\{(a^n, b^n) \mid n > 0\}$
- This grammar generates $\{a^n b^n \mid n > 0\}$, tree has yield *aabb*.



Multiple Context Free Grammars, dimension 2

Nonterminal of dimension 2:

- Context is a string with two gaps $\square ab \square$
- Yield is a pair of strings (a, b)
- Combine (with $\oplus$) to get the string *aabb*.



Distributional learning

CFG

Look at distribution of a string:

- The words "that cat" and "the kitten" occur in similar contexts:
- □ is so cute!

Distributional learning

CFG

Look at distribution of a string:

- The words "that cat" and "the kitten" occur in similar contexts:
- □ is so cute!

MCFG [Yoshinaka, 2009]

Look at distribution of pairs of string:

- The tuples "which book, read" and "which cake, eat" occur in similar contexts:
- □ did you □ yesterday?

Notation

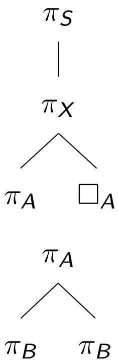
For a nonterminal A,

Contexts

$\Xi(A)$ is a set of contexts with one $\square_A$, and S at the root.
$\Xi(A, l\square r)$ subset with yield $l\square r$

Yields

$\Omega(A)$ is a set of trees with A at the root.
$\Omega(A, w)$ subset with yield w



Weighted Context Free Grammars

Smith and Johnson [2007]

Weighted (M)CFG

Parameter θ for each production in $\mathbb{R}^+$, defines the weight of a tree as

$$w(\tau) = \prod_{\pi} \theta(\pi)^{n(\pi;\tau)}$$

For each nonterminal A define:

$$I(A) = w(\Omega(A)) \text{ (sum over yields)}$$

$$O(A) = w(\Xi(A)) \text{ (sum over contexts)}$$

Stipulate that $I(S) = 1$ and define $\mathbb{P}(u) = w(\Omega(S, u))$

$$I(A)O(A) = \mathbb{E}(A)$$

Probabilistic Context Free Grammars

Stipulate that $I(A) = 1$, and so $O(A) = \mathbb{E}(A)$. Each nonterminal defines a probability distribution over its yields.
Parameters are in $[0, 1]$ and satisfy:

$$\theta(A \leftarrow BC) = \frac{\mathbb{E}(A \leftarrow BC)}{\mathbb{E}(A)}$$

$$\theta(A(a)) = \frac{\mathbb{E}(A(a))}{\mathbb{E}(A)}$$

Parameters have interpretation as conditional probabilities in a top down generative process starting with S.

Bottom up parameterization of Weighted CFGs

Stipulate that $O(A) = 1$, and $I(A) = \mathbb{E}(A)$: each nonterminal defines a probability distribution over its contexts.
Parameters are no longer in $[0, 1]$ but satisfy:

$$\theta(A \leftarrow BC) = \frac{\mathbb{E}(A \leftarrow BC)}{\mathbb{E}(B)\mathbb{E}(C)}$$

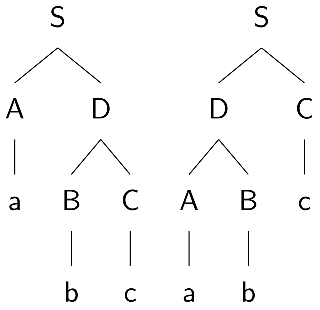
$$\theta(A(a)) = \mathbb{E}(A(a))$$

The major problem:

Non identifiability of PCFGs and CFGs from strings [Hsu et al., 2013]

Given distribution over strings

$\mathbb{P}(abc) = 1$

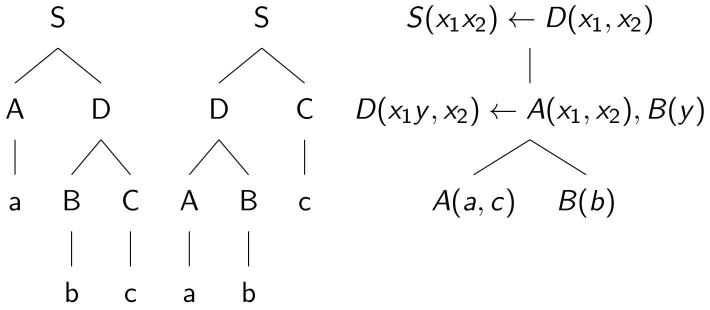


The major problem:

Non identifiability of PCFGs and CFGs from strings [Hsu et al., 2013]

Given distribution over strings

$\mathbb{P}(abc) = 1$



Anchored Context Free Grammars

Stratos et al. [2016]

Assume that for every nonterminal A there is a terminal a which occurs only in the production $A(a)$.
Reasonable assumption if number of words is much greater than number of nonterminals.

Example in English

- she (NP)
- the (Det)
- kitten (N)

Bottom up and anchored

Key property of anchoring

So for all contexts $l\Box r$

$$\Omega(S, lar) = \Xi(A, l\Box r) \oplus A(a)$$

Bottom up and anchored

Key property of anchoring

So for all contexts $l\Box r$

$$\Omega(S, lar) = \Xi(A, l\Box r) \oplus A(a)$$

$$w(\Omega(S, lar)) = w(\Xi(A, l\Box r))\theta(A(a))$$

So, sum over all contexts:

$$\theta(A(a)) = \mathbb{E}(a)$$

and

$$w(\Xi(A, l\Box r)) = \frac{\mathbb{P}(lar)}{\mathbb{E}(a)}$$

The strings

she and *the kitten*

The production

$NP \rightarrow Det\ N$

The strings

she and *the kitten*

The production

$NP \rightarrow Det\ N$

Two old ideas [Harris, 1955]:

1. There should be high MI between *the* and *kitten*
2. *she* and *the kitten* should occur in the same contexts

Distributional similarity

A string u defines a distribution over its contexts:

$$l, r \text{ has probability } \frac{\mathbb{P}(lur)}{\mathbb{E}(u)}$$

Divergence between context distributions

Rényi divergence, $\alpha = \infty$, between discrete distributions P and Q:

$$\mathcal{R}_\infty(P \| Q) = \log \sup_x \frac{P(x)}{Q(x)}$$

- Asymmetric
- Satisfies triangle inequality
- In $[0, \infty]$

Define for strings u and v

$$\mathcal{R}_\infty(u \| v) = \log \sup_{l,r} \frac{P(lur)/\mathbb{E}(u)}{P(lvr)/\mathbb{E}(v)}$$

Binary rule

Given nonterminals A, B, C anchored by a, b, c resp.:

$$\underbrace{\log \theta(A \leftarrow BC)}_{\text{bottom-up parameter}}$$

Binary rule

Given nonterminals A, B, C anchored by a, b, c resp.:

$$\underbrace{\log \theta(A \leftarrow BC)}_{\text{bottom-up parameter}} = \underbrace{\log \frac{\mathbb{E}(bc)}{\mathbb{E}(b)\mathbb{E}(c)}}_{\text{PMI of rhs}}$$

Binary rule

Given nonterminals A, B, C anchored by a, b, c resp.:

$$\underbrace{\log \theta(A \leftarrow BC)}_{\text{bottom-up parameter}} = \underbrace{\log \frac{\mathbb{E}(bc)}{\mathbb{E}(b)\mathbb{E}(c)}}_{\text{PMI of rhs}} - \underbrace{\mathcal{R}_\infty(a \| bc)}_{\text{divergence of lhs from rhs}}$$

Right hand side depends only on the distribution over strings.

Lexical rule

Given nonterminal A anchored by a, and a terminal d:

$$\underbrace{\log \theta(A(d))}_{\text{bottom-up parameter}} = \underbrace{\log \mathbb{E}(d)}_{\text{lexical frequency}} - \underbrace{\mathcal{R}_\infty(a \| d)}_{\text{divergence of lhs from rhs}}$$

Further conditions

Local Unambiguity

A weak condition limiting how ambiguous the grammar is:
For every production $A \to \alpha$, there is a string which always uses that production "in the same place".

For every production $\pi = A \leftarrow B, C$
there is a string $w = luvr$ such that

$$\Omega(G, w) = \Xi(A, l\Box r) \oplus \pi(\Omega(B, u), \Omega(C, v))$$

Completeness

All productions of rank at most k, that don't overgenerate are either

- in the grammar
- Or can be derived in the grammar.
  For example: for CFG productions $A \leftarrow BC$ and $C \leftarrow DE$, we can derive $A \leftarrow BDE$.

Proof for lexical production

$A(a)$ is an anchor, $A(d)$ some other production.

$$\Omega(S, ldr) \supseteq \Xi(A, l\Box r) \oplus A(d)$$

$$w(\Omega(S, ldr)) \geq w(\Xi(A, l\Box r))w(A(d))$$

Proof for lexical production

$A(a)$ is an anchor, $A(d)$ some other production.

$$\Omega(S, ldr) \supseteq \Xi(A, l\Box r) \oplus A(d)$$
$$w(\Omega(S, ldr)) \geq w(\Xi(A, l\Box r))w(A(d))$$

$$\mathbb{P}(ldr) \geq \frac{\mathbb{P}(lar)}{\mathbb{E}(a)}\theta(A(d))$$

Rearranging

$$\theta(A(d)) \leq \frac{\mathbb{P}(ldr)\mathbb{E}(a)}{\mathbb{P}(lar)}$$

Proof for lexical production

$A(a)$ is an anchor, $A(d)$ some other production.

$$\Omega(S, ldr) \supseteq \Xi(A, l\Box r) \oplus A(d)$$
$$w(\Omega(S, ldr)) \geq w(\Xi(A, l\Box r))w(A(d))$$

$$\mathbb{P}(ldr) \geq \frac{\mathbb{P}(lar)}{\mathbb{E}(a)}\theta(A(d))$$

Rearranging

$$\theta(A(d)) \leq \frac{\mathbb{P}(ldr)\mathbb{E}(a)}{\mathbb{P}(lar)}$$

Minimizing over the contexts:

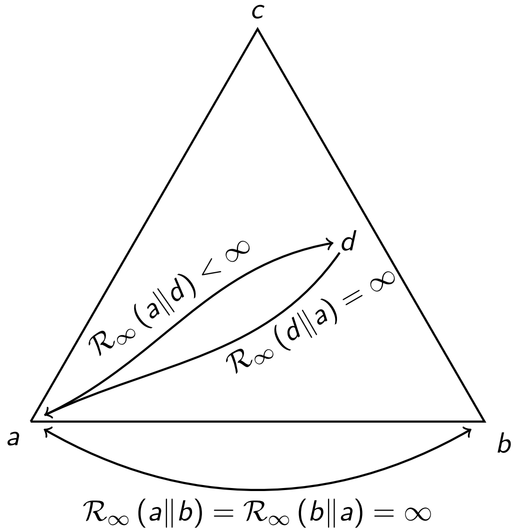
$$\theta(A(d)) \leq \mathbb{E}(d) \inf_{l\Box r} \frac{\mathbb{P}(ldr)\mathbb{E}(a)}{\mathbb{P}(lar)\mathbb{E}(d)}$$

Then by local unambiguity:

$$\theta(A(d)) = \mathbb{E}(d) \inf_{l\Box r} \frac{\mathbb{P}(ldr)\mathbb{E}(a)}{\mathbb{P}(lar)\mathbb{E}(d)}$$

Identifying terminals as anchors

Context distributions of all terminals will lie in the convex hull of the anchors:



Result of Clark and Fijalkow [2020], Clark [2021]

There is computationally efficient consistent estimator from strings, for all PCFGs whose underlying CFG is

1. Anchored
2. Locally Unambiguous
3. Complete

Using naive plug-in estimators that are slow to converge.

Extending to MCFGs

Straightforward

- Completeness
- Local unambiguity

Anchoring for MCFGs

For every nonterminal A of dimension 2, there are distinct terminals a and b such that

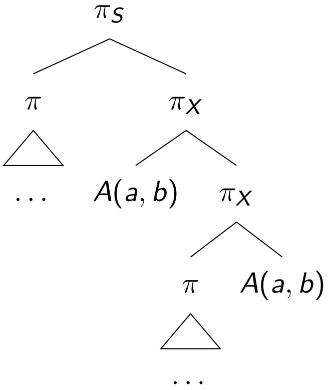
$$A(a, b)$$

is the only production in the grammar using a or b.

Key property is in general false!

(Not true that) For all contexts $l\Box m\Box r$

$$\Omega(S, lambr) = \Xi(A, l\Box m\Box r) \oplus A(a, b)$$



Running Example

Because we might have more than one occurrence of $A(a, b)$ and we don't know which ones match up.

$$\Omega(G, aabb) = \Xi(A, \square ab \square) \oplus \Omega(A, (a, b))$$

Running Example

Because we might have more than one occurrence of $A(a, b)$ and we don't know which ones match up.

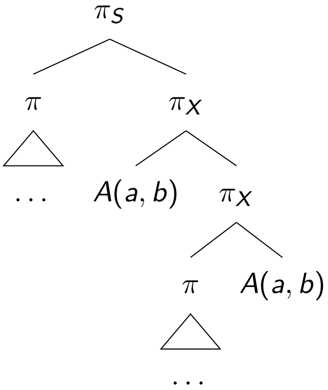
$$\Omega(G, aabb) = \Xi(A, \Box ab \Box) \oplus \Omega(A, (a, b))$$

But

$$\Omega(G, aabb) \neq \underbrace{\Xi(A, a\Box b\Box)}_{empty} \oplus \Omega(A, (a, b))$$

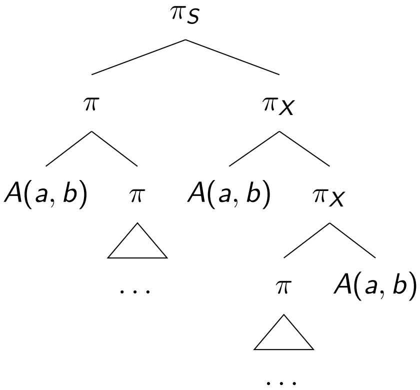
There are 4 contexts that combine with (a, b) to give *aabb* but only 2 of them correspond to contexts of A.

Pattern I



Ignoring all other terminals, this can only be *abab* or *aabb*.

Pattern II



Ignoring all other terminals, this can only be *ababab*, *aabbab*, *abaabb* or *aaabbb*.

Dyck language

There is a pattern, and it's the Dyck language, the language of matching brackets: where a is open bracket and b is close bracket.

Well-nested

If the grammar is well-nested then occurrences of a, b generated by same anchoring production will match as brackets.

If $w = \ldots a \ldots a \ldots b \ldots b \ldots,$
then

$$\Omega(G, w) = \Xi(A, \ldots \Box \ldots a \ldots b \ldots \Box \ldots) \oplus A(a, b)$$

and

$$\Omega(G, w) = \Xi(A, \ldots a \ldots \Box \ldots \Box \ldots b \ldots) \oplus A(a, b)$$

$$\Xi(A, \ldots a \ldots \Box \ldots b \ldots \Box) = \emptyset$$

Well-nestedness

Well-nested:

$$A(x_1\mathbf{y_1}, \mathbf{y_2}x_2) \leftarrow A(x_1, x_2), A(\mathbf{y_1}, \mathbf{y_2})$$

Non well-nested:

$$A(x_1\mathbf{y_1}, x_2\mathbf{y_2}) \leftarrow A(x_1, x_2), A(\mathbf{y_1}, \mathbf{y_2})$$

Using this Dyck idea

Identifying nonterminals of dimension 2

Find pairs of distinct terminals which only occur in these Dyck patterns: (Dyck pairs)

- Handle ambiguity in the same way that it is handled with anchors for dimension 1 nonterminals.
- $A(c, d)$ and $B(c, d)$

Identifying parameters

Restrict contexts to those that are compatible with the Dyck bracketing.

Similar decomposition for a production (careful with defn.):

$$\pi = A(x_1\mathbf{y_1}, \mathbf{y_2}x_2) \leftarrow B(x_1, x_2), C(\mathbf{y_1}, \mathbf{y_2})$$

$$\log(\theta(\pi)) = \log \frac{\mathbb{E}(bc, c'b')}{\mathbb{E}(b, b')\mathbb{E}(c, c')} - \mathcal{R}_\infty\left((a, a')\|(bc, c'b')\right)$$

Result

Grammar class

Well-nested MCFGs of dimension 2 in a restricted normal form up to rank k.

- Doubly anchored:

  Nonterminal of dimension 2 anchor is $A(a, a')$

  Nonterminal of dimension 1 two anchors $A(a)$ and $A(a')$

- Technical conditions:

  Locally unambiguous

  Complete All possible productions that don't overgenerate can be derived.

Theorem

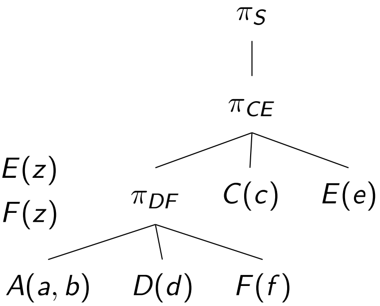
A consistent learning algorithm for all probabilistic grammars where the grammar is in this class.

Example generating a non-context-free language

Cross-serial dependencies

$$\pi_S = S(x_1 x_2) \leftarrow A(x_1, x_2)$$
$$\pi_{CE} = A(x_1 y, x_2 z) \leftarrow A(x_1, x_2), C(y), E(z)$$
$$\pi_{DF} = A(x_1 y, x_2 z) \leftarrow A(x_1, x_2), D(y), F(z)$$

$A(a, b)$, $C(c)$, $D(d)$,
$E(e)$, $F(f)$, $C(c')$, $D(d')$, $E(e')$, $F(f')$



Yield is *adcbfe*

Discussion

Solution is not as interesting as the problem:

- Key technical obstacle is identifying the discontinuous constituents: same problem as for CFGs with anchors of length greater than 1.
- Dimension 2 nonterminals will be used extensively even for the CFG modelable components of the language.
- It seems like some additional information would be helpful to help identify discontinuous constituents. But probably not necessary.
- Anchoring assumption is unreasonable, at least if the terminal symbols are words.
- Well-nestedness [Kanazawa et al., 2011] seems important (again).

Conclusion

- A first algorithm for strong probabilistic learning of a standard mildly context-sensitive formalism from strings.

Take-home point

It is in principle possible to efficiently learn derivation trees of mildly context-sensitive grammars just from strings.

Open questions

- Can the anchoring assumption be weakened?
  (Yes)
- Can we do this with Minimalist grammars or CCG?

Bibliography

Alexander Clark. Beyond Chomsky normal form: Extending strong learning algorithms for PCFGs. In Jane Chandlee, Rémi Eyraud, Jeff Heinz, Adam Jardine, and Menno van Zaanen, editors, *Proceedings of the Fifteenth International Conference on Grammatical Inference*, volume 153 of *Proceedings of Machine Learning Research*, pages 4–17. PMLR, 23–27 Aug 2021. URL https://proceedings.mlr.press/v153/clark21a.html.

Alexander Clark and Nathanaël Fijalkow. Consistent unsupervised estimators for anchored PCFGs. *Transactions of the Association for Computational Linguistics*, 8:409–422, 2020. doi: 10.1162/tacl_a_00323. URL https://doi.org/10.1162/tacl_a_00323.

Alexander Clark and Franck Thollard. PAC-learnability of probabilistic deterministic finite state automata. *Journal of Machine Learning Research*, 5:473–497, May 2004.

Zellig Harris. From phonemes to morphemes. *Language*, 31:190–222, 1955.

James Jay Horning. *A study of grammatical inference*. PhD thesis, Computer Science Department, Stanford University, 1969.

D. Hsu, S. M. Kakade, and P. Liang. Identifiability and unmixing of latent parse trees. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1520–1528, 2013.

A.K. Joshi, K. Vijay-Shanker, and D.J. Weir. The convergence of mildly context-sensitive grammar formalisms. Technical Report MS-CIS-90-01, University of Pennsylvania, Dept. of Computer and Information Science, 1990.

Makoto Kanazawa, Jens Michaelis, Sylvain Salvati, and Ryo Yoshinaka. Well-nestedness properly subsumes strict derivational minimalism. In Sylvain Pogodalla and Jean-Philippe Prost, editors, *Logical Aspects of Computational Linguistics*, volume 6736 of *Lecture Notes in Computer Science*, pages 112–128. Springer Berlin Heidelberg, 2011. ISBN 978-3-642-22220-7. doi: 10.1007/978-3-642-22221-4_8. URL http://dx.doi.org/10.1007/978-3-642-22221-4_8.

Hiroyuki Seki, Takashi Matsumura, Mamoru Fujii, and Tadao Kasami. On multiple context-free grammars. *Theoretical Computer Science*, 88(2):229, 1991.

Stuart M. Shieber. Evidence against the context-freeness of natural language. *Linguistics and Philosophy*, 8:333–343, 1985.

Noah A Smith and Mark Johnson. Weighted and probabilistic context-free grammars are equally expressive. *Computational Linguistics*, 33(4):477–491, 2007.

Karl Stratos, Michael Collins, and Daniel Hsu. Unsupervised part-of-speech tagging with anchor hidden markov models. *Transactions of the Association for Computational Linguistics*, 4:245–257, 2016.

Ryo Yoshinaka. Learning mildly context-sensitive languages with multidimensional substitutability from positive data. In *International Conference on Algorithmic Learning Theory*, pages 278–292, 2009.