Categorial Grammar Induction from Raw Data

Christian Clark and **William Schuler**
Department of Linguistics
The Ohio State University
{clark.3664,schuler.77}@osu.edu

Abstract

Grammar induction, the task of learning a set of grammatical rules from raw or minimally labeled text data, can provide clues about what kinds of syntactic structures are learnable without prior knowledge. Recent work (e.g., Kim et al., 2019; Zhu et al., 2020; Jin et al., 2021a) has achieved advances in unsupervised induction of probabilistic context-free grammars (PCFGs). However, categorial grammar induction has received less recent attention, despite allowing inducers to support a larger set of syntactic categories—due to restrictions on how categories can combine—and providing a transparent interface with compositional semantics, opening up possibilities for models that jointly learn form and meaning. Motivated by this, we propose a new model for inducing a basic (Ajdukiewicz, 1935; Bar-Hillel, 1953) categorial grammar. In contrast to earlier categorial grammar induction systems (e.g., Bisk and Hockenmaier, 2012), our model learns from raw data without any part-of-speech information. Experiments on child-directed speech show that our model attains a recall-homogeneity of 0.33 on average, which dramatically increases to 0.59 when a bias toward forward function application is added to the model.

1 Introduction

One of the core motivating questions of modern linguistics relates to language acquisition: How can a child pick up complex linguistic rules from limited exposure to language? Chomsky (e.g., 1965) introduced the well-known argument from the poverty of the stimulus, which claims that the linguistic input received by children is insufficiently rich to account for the knowledge they acquire—and therefore that humans must be born with prior knowledge about language. In contrast, empiricist accounts of language acquisition argue that statistical cues (Saffran et al., 1996) or other factors such as social interaction (Tomasello, 2005) may provide enough information on their own to support language acquisition.

Computational modeling provides one useful tool for judging between these competing accounts. Questions about the learnability of linguistic structures can be tested empirically by seeing if a model with minimal prior knowledge can learn these structures from corpus data (Pullum and Scholz, 2002).

Along these lines, a range of studies over several decades have tested whether induction models can acquire probabilistic context-free grammars (PCFGs) from text data (Lari and Young, 1990; Klein and Manning, 2002). Although PCFG induction is considered a difficult problem (Carroll and Charniak, 1992), recent systems have achieved performance improvements thanks to new types of Bayesian and neural network models. Recent systems have been able to induce grammars with accuracy levels (measured by recall-homogeneity) approaching fifty percent on corpora of child-directed speech (Jin et al., 2018, 2021a,b).

Although PCFGs are a convenient formalism for computational modeling, they are not the only viable option. A second line of research—albeit one less currently active than PCFG modeling—has examined the learnability of categorial grammar formalisms (Bisk and Hockenmaier, 2012; Bisk et al., 2015), particularly Combinatory Categorial Grammar (CCG; Steedman, 2000). A notable advantage of categorial grammars over PCFGs is their clean mapping between syntactic and semantic composition, which allows them to be used as a tool for predicting lambda calculus encodings of meaning (Zettlemoyer and Collins, 2005). Categorial grammars also impose constraints regarding which syntactic categories can combine, providing a practical advantage for designing induction systems that support a large set of categories.

Motivated by these advantages, this work presents a neural network–based system that adapts a state-of-the-art PCFG induction model (Jin et al.,

Findings of the Association for Computational Linguistics: ACL 2023, pages 2368–2379
July 9-14, 2023 ©2023 Association for Computational Linguistics

2021a) to instead learn a basic categorial grammar.[1] Unlike the previously mentioned categorial grammar induction systems, our model learns from entirely unlabeled data.

An initial version of the model attains an average recall-homogeneity (RH) score of 0.33 on an English corpus of child-directed speech (Experiment 1). A high variance across randomly initialized runs is observed, with a cluster of runs achieving RH on par with state-of-the-art PCFG inducers and another cluster achieving poor RH. In Experiment 2, we test a modified version of the model with a bias term encouraging forward function application, which appeared more often in the better-performing runs in Experiment 1. The modified model reaches an average RH of 0.59, surpassing results reported from Jin et al. (2021a) and other PCFG inducers.

2 Related Work

PCFG induction is a longstanding area of interest in computational linguistics (Lari and Young, 1990; Carroll and Charniak, 1992; Klein and Manning, 2002). As neural modeling has made unsupervised induction more feasible, recent work has experimented with learning compound PCFGs (Kim et al., 2019), simultaneously inducing phrase structure grammars and lexical dependencies (Zhu et al., 2020), and boosting model performance by grounding on multimodal data (Zhao and Titov, 2020; Zhang et al., 2021, 2022), among other innovations.

A somewhat earlier line of research established the potential for learning an alternative type of grammar, a CCG, from data with a small set of broadly defined part-of-speech categories (noun, verb, etc.) (Bisk and Hockenmaier, 2012, 2013; Bisk et al., 2015). Bisk et al. (2015) showed that only a small number of labeled data points with POS tags are needed to induce a CCG. However, induction of CCGs (or other categorial grammars) has received less recent attention, with CCG research more focused on tasks such as supertagging (Bhargava and Penn, 2020; Prange et al., 2021) or incremental parsing (Stanojević et al., 2021).

A third relevant area of research is work focused on mapping a sentence to its logical form via CCG parsing (Zettlemoyer and Collins, 2005; Kwiatkowski et al., 2010; Kwiatkowski et al., 2013). These studies reveal that categorial grammar induction may be useful not only as a method of testing the learnability of syntactic rules, but also as a tool for semantic parsing.

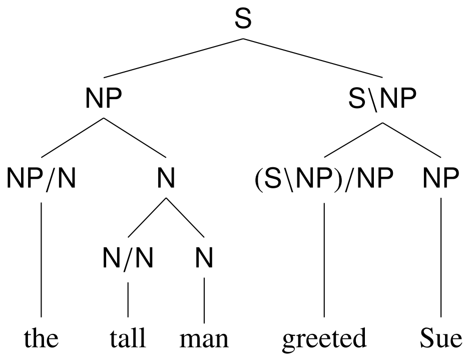


Figure 1: Example derivation using a basic categorial grammar with the primitives S, N, and NP.

3 Background

3.1 Basic Categorial Grammar

The induction models in this paper learn a basic categorial grammar, also known as an AB grammar (Ajdukiewicz, 1935; Bar-Hillel, 1953). This type of grammar was chosen for its simplicity and its suitability for extension through additional composition operations. A basic categorial grammar uses a set of primitive categories (e.g., S or N for sentences or nouns) as well as the type-combining operators \ and /, which indicate compatibility with an argument preceding or following the category, respectively. These type-combining operators can be used to define complex categories (e.g., N\N or (S/N)/N).

The models use two composition operations: backward function application and forward function application. Backward function application occurs when a phrase of category $X\backslash Y$ combines with a phrase of category Y on the left to yield a larger phrase of category X. Forward function application occurs when a phrase of category X/Y combines with a phrase of category Y on the right to yield a larger phrase of category X. In such cases, $X\backslash Y$ and X/Y are called the *functor* categories, Y is called the *argument* category, and X is called the *result* category. See Figure 1 for an example parse using a basic categorial grammar.

[1] The code for the categorial grammar inducer is available at `https://github.com/christian-clark/cgInduction`.

3.2 The Jin et al. (2021a) PCFG induction model

Our induction model uses a formulation for sentence probabilities based on the word-level PCFG model from Jin et al. (2021a), which we summarize in this section.

In unsupervised training, the objective function that the model maximizes is the marginal probability of the sentences in the dataset. For a single sentence σ, each possible parse tree (assumed to be in Chomsky Normal Form) can be divided into a set of of nodes τ undergoing nonterminal expansions $c_\eta \to c_{\eta 1}\ c_{\eta 2}$ and a set of nodes τ' undergoing terminal expansions $c_\eta \to w_\eta$. Here, $\eta \in \{1,2\}^*$ is a Gorn address specifying a path of left and right branches from the root node of the parse tree, c_η is the nonterminal category at node η, and w_η is the word located at node η. C is the set of nonterminal categories. The marginal probability of σ is calculated by summing over all possible parse trees:

$$\mathsf{P}(\sigma) = \sum_{\tau,\tau'} \prod_{\eta\in\tau} \mathsf{P}(c_\eta \to c_{\eta 1}\ c_{\eta 2}) \cdot \prod_{\eta\in\tau'} \mathsf{P}(c_\eta \to w_\eta) \tag{1}$$

A set of Bernoulli distributions are defined to separate the nonterminal and terminal expansion rules:

$$\mathsf{P}(\text{Term} \mid c_\eta) = \underset{\{0,1\}}{\text{softmax}}(\text{N}_{\text{Term}}(\mathbf{E}\,\delta_{c_\eta})) \tag{2}$$

Here, c_η is a nonterminal category, δ_{c_η} is a vector representing a Kronecker delta function with 1 at index c_η and 0 elsewhere, and $\mathbf{E} \in \mathbb{R}^{d\times|C|}$ is a matrix of nonterminal category embeddings of size d. N_{Term} is a residual network with 2 identical blocks. Given the input $\mathbf{x}_{b-1,c_\eta}$, each residual block computes its output as follows:

$$\begin{aligned}\mathbf{x}_{b,c_\eta} = \text{ReLU}(\mathbf{W}'_b\ \text{ReLU}(\mathbf{W}_b\,\mathbf{x}_{b-1,c_\eta} + \mathbf{b}_b) \\ + \mathbf{b}'_b) + \mathbf{x}_{b-1,c_\eta}\end{aligned} \tag{3}$$

Fully connected layers are used before and after the residual blocks:

$$\mathbf{x}_{0,c_\eta} = \text{ReLU}(\mathbf{W}_0\,\mathbf{E}\,\delta_{c_\eta} + \mathbf{b}_0), \tag{4}$$

$$s_{c_\eta} = \text{ReLU}(\mathbf{W}_{\text{soft}}\,\mathbf{x}_{B,c_\eta} + \mathbf{b}_{\text{soft}}) \tag{5}$$

All $\mathbf{W}$'s and $\mathbf{b}$'s are weight and bias parameters respectively.

Binary-branching nonterminal expansion probabilities are computed as follows:

$$\begin{aligned}\mathsf{P}(c_\eta \to c_{\eta 1}\ c_{\eta 2}) = \mathsf{P}(\text{Term=0} \mid c_\eta) \cdot \\ \mathsf{P}(c_\eta \to c_{\eta 1}\ c_{\eta 2} \mid c_\eta, \text{Term=0}),\end{aligned} \tag{6}$$

which in turn uses the following distribution over expansion rules:

$$\begin{aligned}\mathsf{P}(c_\eta \to c_{\eta 1}\ c_{\eta 2} \mid c_\eta, \text{Term=0}) = \\ \underset{c_{\eta 1},c_{\eta 2}}{\text{softmax}}(\mathbf{W}_{\text{nont}}\,\mathbf{E}\,\delta_{c_\eta} + \mathbf{b}_{\text{nont}}),\end{aligned} \tag{7}$$

where $\mathbf{W}_{\text{nont}}$ and $\mathbf{b}_{\text{nont}}$ are additional model parameters. The categorial grammar induction model presented in this work modifies Equation (7); see Section 4.2.

Finally, lexical unary-expansion rule probabilities are computed as follows:

$$\begin{aligned}\mathsf{P}(c_\eta \to w_\eta) = \mathsf{P}(\text{Term=1} \mid c_\eta) \cdot \\ \mathsf{P}(c_\eta \to w_\eta \mid c_\eta, \text{Term=1})\end{aligned} \tag{8}$$

A softmax is taken over words in the vocabulary:

$$\mathsf{P}(c_\eta \to w_\eta \mid c_\eta, \text{Term=1}) = \underset{w_\eta}{\text{softmax}}(\text{N}'(\mathbf{E}\,\delta_{c_\eta})), \tag{9}$$

where N′ is another residual network, similar to N_{Term} except that the output layer's dimension is the size of the vocabulary.

Jin et al. (2021a) also introduce a character-level expansion model as an alternative to Equation (9). However, they report that the word-level model performs slightly better on English data from the CHILDES corpus. Because the current study works with the same English data, we only test the word-level model.

4 Induction Model

The model introduced in this paper extends the Jin et al. (2021a) model from Section 3.2 to induce a categorial grammar. This section details how categories and expansion rule probabilities are defined in the new model.

4.1 Categories

We define the set of categories C according to a number of primitives P and a maximum category depth D. Primitives in the induction model are labeled as integers $\mathsf{0}, \mathsf{1}, \mathsf{2}, \ldots$. A category's depth is defined according to its tree representation (see Figure 3(a) of Prange et al. (2021) for an example).[2] For instance, the primitive category $\mathsf{1}$ has depth 0, and the category $\mathsf{2/(1\backslash 0)}$ has depth 2.

The number of possible categories $|C_{P,D}|$ with P primitives and maximum depth D can be computed

[2] The tree representation of a syntactic category should not be confused with the parse tree for an entire sentence.

with the following recurrence relation:

$$|C_{P,0}| = P \tag{10}$$

$$|C_{P,i}| = 2|C_{P,i-1}|^2 + P \tag{11}$$

Our experiments below use the category set $C = C_{3,2}$, the entire set of 885 possible categories with $P = 3$ and $D = 2$. This is nearly 10 times the number of categories (90) used by Jin et al. (2021a).

4.2 Binary expansion rule probabilities

The categorial grammar induction model modifies Equation (7) from Jin et al. (2021a) to take advantage of constraints imposed by categorial grammar categories. In a PCFG, a parent category c_η may expand into any two child categories $c_{\eta 1}$ and $c_{\eta 2}$ However, in a basic CG, this expansion is only possible if one child (the functor) can combine with the other child (the argument) to produce the parent (the result). There are two possibilities:

- The argument is the left child $c_{\eta 1}$. Then the right child must be the functor, and its category must be $c_{\eta 2} = c_\eta \backslash c_{\eta 1}$.
- The argument is the right child $c_{\eta 2}$. Then the left child must be the functor, and its category must be $c_{\eta 1} = c_\eta / c_{\eta 2}$.

If $c_{\eta 2} \neq c_\eta \backslash c_{\eta 1}$ and $c_{\eta 1} \neq c_\eta / c_{\eta 2}$, then it is impossible for c_η to expand to $c_{\eta 1}$ and $c_{\eta 2}$, and so $\mathsf{P}(c_\eta \rightarrow c_{\eta 1}\ c_{\eta 2} \mid c_\eta, \text{Term=0}) = 0$. For all other cases, where the binary expansion is possible, the probabilities are calculated as follows:

$$\mathsf{P}(c_\eta \rightarrow c_{\eta 1}\ c_{\eta 2} \mid c_\eta, \text{Term=0}) = \underset{(c',o)\in C_{\text{arg}}\times\{\text{L,R}\}}{\text{softmax}} \left(\begin{bmatrix} \mathbf{W}_\text{L} \\ \mathbf{W}_\text{R} \end{bmatrix} \delta_{c_\eta} + \begin{bmatrix} \mathbf{b}_\text{L} \\ \mathbf{b}_\text{R} \end{bmatrix} \right) \tag{12}$$

The model parameters $\mathbf{W}_\text{L}, \mathbf{W}_\text{R} \in \mathbb{R}^{|C_{\text{arg}}|\times|C_{\text{res}}|}$ are weights associating each parent category with each possible left-child and right-child argument category; $\mathbf{b}_\text{L}, \mathbf{b}_\text{R} \in \mathbb{R}^{|C_{\text{arg}}|}$ are the corresponding bias vectors. $C_{\text{arg}}, C_{\text{res}} \subset C$ are the sets of possible argument and result categories respectively, both of which comprise all categories of depth up to $D - 1$:

$$C_{\text{arg}} = C_{\text{res}} = \{c \in C \mid \text{depth}(c) \leq D - 1\} \tag{13}$$

Argument and result categories cannot have depth D because this would require functor categories to have depth greater than D.

The variable $o \in \{\text{L}, \text{R}\}$ expresses the location of the argument child relative to the functor child. If $o = \text{L}$, then the argument is to the left of the functor, and so $c_{\eta 1} = c'$ and $c_{\eta 2} = c_\eta \backslash c'$. If $o = \text{R}$, then the argument is to the right of the functor, and so $c_{\eta 1} = c_\eta / c'$ and $c_{\eta 2} = c'$.

Equation (12) results in a considerable space complexity improvement compared to (7). For an induction model using category set C, Equation (7) requires taking a softmax over $|C|^2$ possible pairs of children. Equation (12) only requires a softmax over $2|C_{\text{arg}}| = O(\sqrt{|C|})$ categories.

The experiment presented in Section 6 uses a modified bias term in Equation (12) in order to encourage the model to prefer forward function application over backward function application. The bias term for left-child arguments $\mathbf{b}_\text{L}$ is replaced with a new bias $\mathbf{b}'_\text{L} = \mathbf{b}_\text{L} - \mathbf{k}$, where $\mathbf{k} \in \mathbb{R}^{|C_{\text{arg}}|}$ has the same constant value in each dimension. As will be explained below, a bias toward forward function application results in a preference for right-branching structures, which improves the performance of the induction model.

5 Experiment 1: Basic Induction Model

5.1 Corpora

The induction model was evaluated on child-directed speech in English from CHILDES (MacWhinney, 2000), specifically the Adam and Eve sections of the Brown corpus (Brown, 1973). The Adam section, which was used for hyperparameter optimization, contains interactions between a child and his caretakers, with the child's age ranging from 2 years and 3 months to 5 years and 2 months and a total of 28,780 sentences. The Eve section was used for held-out testing; it contains similar interactions from a child whose age ranges from 1 year and 6 months to 2 years and 3 months, with a total of 14,251 sentences. Syntactic annotations for the Adam and Eve sections came from Pearl and Sprouse (2013).

5.2 Procedures

The induction model used the Adam optimizer with a learning rate of 0.0001, a category embedding size of 64, and a hidden layer size of 64. Hyperparameters were selected based on a grid search on the Adam corpus. For evaluation on Eve, ten randomly initialized models were run for 20 epochs each with a batch size of 2 sentences.

The evaluation metrics we considered were unlabeled F1 score and recall-homogeneity (RH; Jin et al., 2021b). Recall measures what proportion of (unlabeled) constituents in the annotated trees are present in the predicted trees. Homogeneity—a commonly used metric in part-of-speech tagging evaluations—measures to what degree a single induced category maps to a single category in the annotations. Specifically, it measures the relative increase in the log of the expected probability of a gold category, given the predicted category that covers the same span. RH is simply the product of unlabeled recall and homogeneity. The RH metric is motivated by assumptions that (a) induced grammars should not be penalized for predicting extra constituents, since flatter trees in the annotations may have been chosen for convenience rather than any theoretical motivation; and (b) induced grammars should not be penalized for making finer-grained distinctions between categories (e.g., noun cases) than are present in the annotations, since less granular categories similarly may have been chosen for convenience.

In keeping with Seginer (2007) and Jin et al. (2021a), punctuation was retained in the input data during training but removed during evaluation. Unary chains were removed from parse trees, with only the top category used for evaluation.

5.3 Results

Figure 2 presents the main results. The mean RH and F1 score across the ten runs were 0.33 and 0.52 respectively. The mean RH value is well below the average of 0.49 reported for the word-level PCFG inducer from Jin et al. (2021a). However, the means alone do not best describe Figure 2, as RH and F1 both seem to show bimodal distributions. Six runs produced poor RH and F1 (averaging 0.22 and 0.37 respectively), while the other four runs produced much better values (averaging 0.50 and 0.74 respectively).[3]

Figure 3 offers another vantage point into the pattern of results by separating recall and homogeneity, the two metrics combined in RH. (Recall also influences F1.) Again, a sharp division is observed between runs with low versus high recall. However, homogeneity appears to vary somewhat independently from recall.

One possible explanation for this trend would be that the poorly performing runs get caught in local maxima of the objective function. If this were the case, we would expect to see higher log likelihood in the well-performing runs, which should reach a better maximum. However, Figure 4 shows that this does not occur: The well-performing and poorly performing runs are associated with similar ranges of log likelihood values.

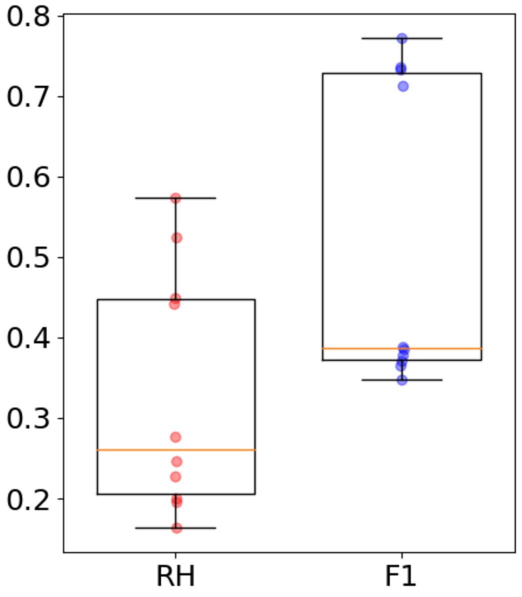


Figure 2: Box plots of the labeled recall-homogeneity (RH) and unlabeled F1 scores of 10 runs of the induction system on the Eve corpus (Experiment 1). Scattered points show results from the 10 individual runs. The mean RH was 0.33 (median 0.26) and the mean F1 was 0.52 (median 0.39).

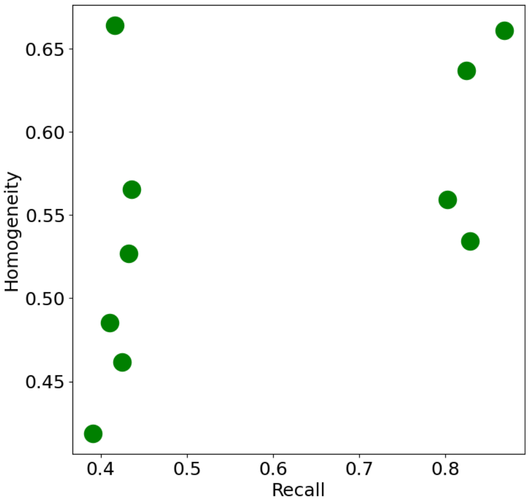


Figure 3: Recall and homogeneity of the 10 runs of the induction system from Experiment 1.

[3] We verified that these were the same six and four runs, i.e., no run produced good RH and poor F1 or vice versa.

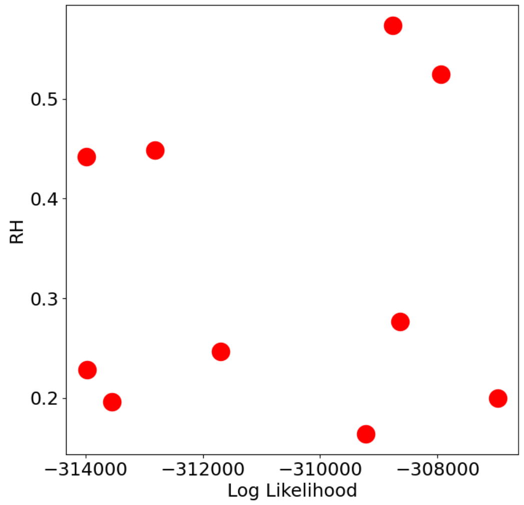
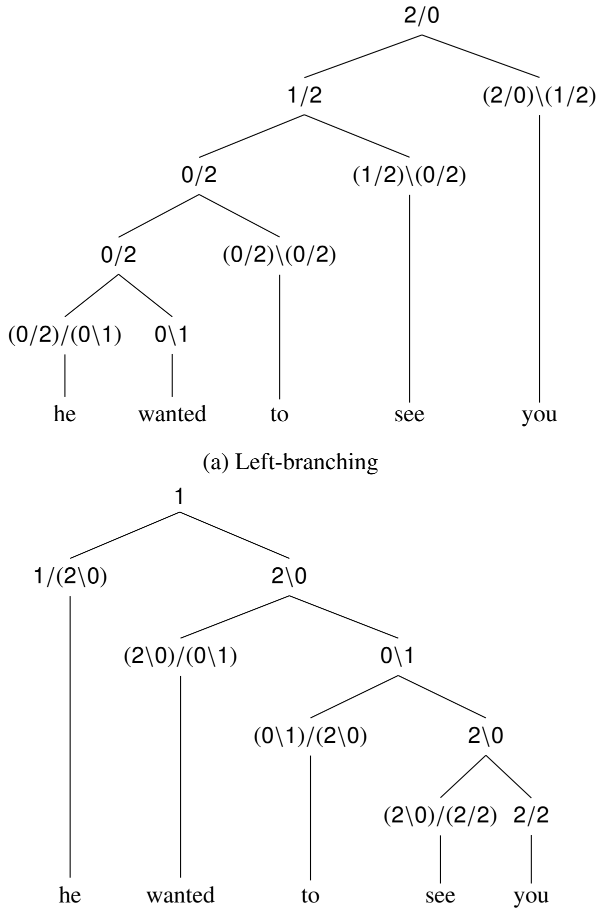


Figure 4: Log likelihood and recall-homogeneity of the 10 runs of the induction system from Experiment 1.

If log likelihood fails to distinguish the clusters of good and poor runs, what else might? One pattern immediately stood out during qualitative inspection of the models' predicted trees: Models with high RH and F1 scores tend to predict trees with frequent forward function application and right branching, while poorly performing models predict trees with backward function application and left branching. This pattern was quantitatively confirmed by counting the proportion of right- and left-branching nodes in the induced trees. The six runs with worse performance use left branching 82% of the time, while the four runs with better performance use right branching 65% of the time. These values were computed by counting the proportion of branching nodes who appear as left versus right children.

As an illustration, Figure 5 compares predictions from two models on the same sentence from the Eve corpus. Figure 5a shows the prediction from a model that performed poorly overall, containing a left-branching pattern combined with the use of backward function application. Figure 5b shows the prediction from a model that performed well, which has opposite patterns. In both cases, the dispreferred type-combining operator (e.g., / in 5a) appears in complex categories but is rarely applied, so that a category such as 1/2 in 5a is treated like a primitive.



(a) Left-branching

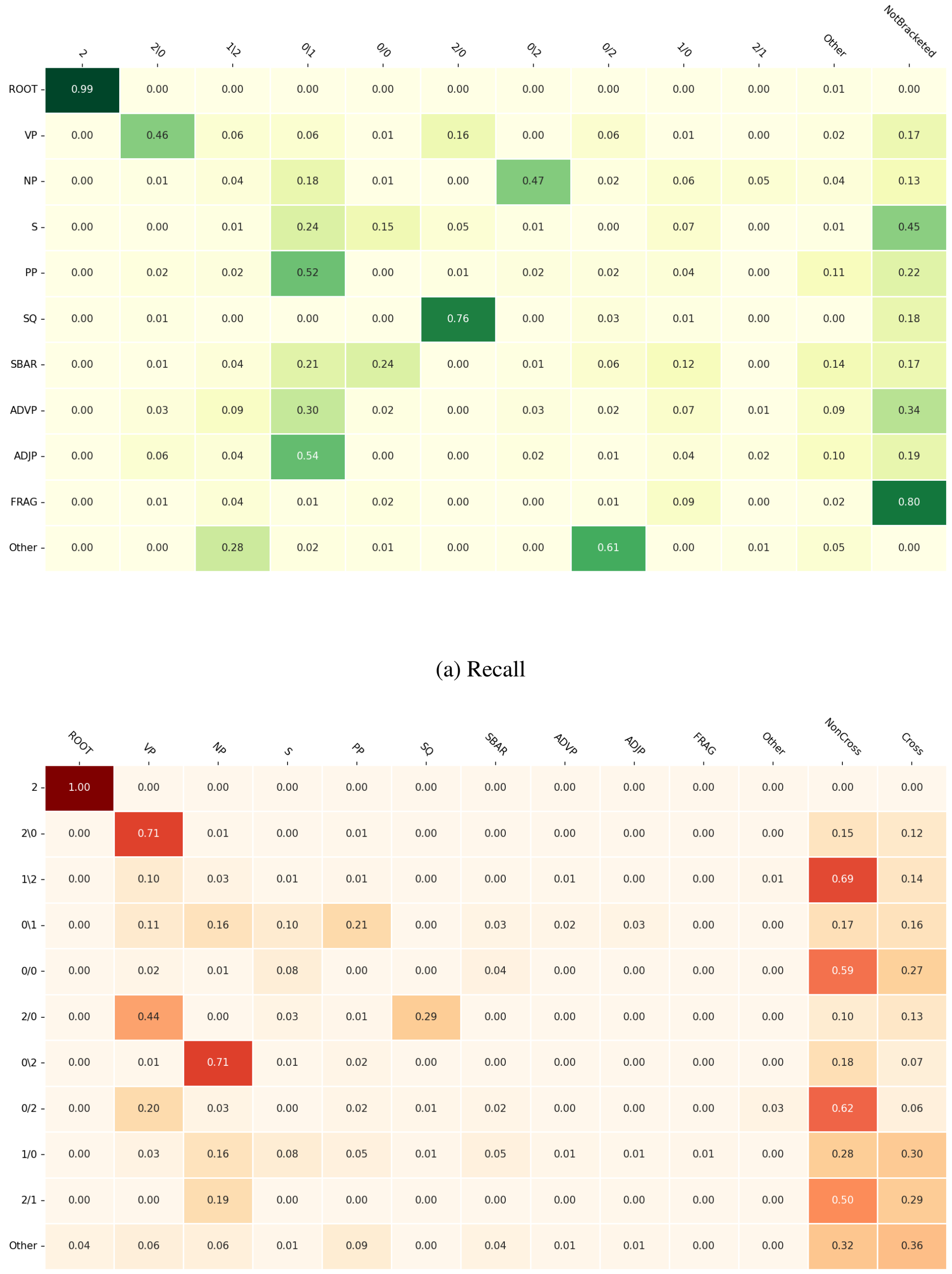
(b) Right-branching

Figure 5: Examples of left- and right-branching structures predicted for the same sentence in the Eve corpus, using induction models from two different runs from Experiment 1. The model that predicted (a) had an RH of 0.16, F1 score of 0.35, and log likelihood of -309,217 on the full Eve corpus, compared with an RH of 0.57, F1 score of 0.77, and log likelihood of -308,764 for the model that predicted (b).

Figure 6 contains confusion matrices relating the induced categories with the human-annotated categories for the run that produced the highest RH and F1 score in Experiment 1. The recall table (a) shows that most of the annotated categories are only represented by one or two different induced categories, and the precision table (b) shows that induced categories are seldom crossing brackets.

6 Experiment 2: Induction Model with Forward Function Application Bias

To try to produce more consistent induction results with English-like branching behavior, our second experiment biased the induction model toward using forward function applications (i.e., the / operator). While in principle it is possible for right-branching structures to use backward function ap-



(a) Recall

(b) Precision

Figure 6: Comparison of frequent induced and annotated categories in the Eve corpus, according to the Experiment 1 run with the best RH and F1 score. The "NotBracketed" column in (a) tells the proportion of phrases of a particular category that were not bracketed together in the predicted parse. In (b), the "NonCross" column tells the proportion of phrases belonging to an induced category that did not appear as constituent in the annotated parse but did not cross constituent boundaries in the annotation. "Cross" tells what proportion did cross annotated constituent boundaries.

plication and left-branching structures to use forward function application, this did not often occur in the results from Experiment 1 and seems less likely in general given the available categories in C.

6.1 Procedures

This experiment used the same corpora and procedures as Experiment 1, with the exception of the modified bias term mentioned at the end of Sec-

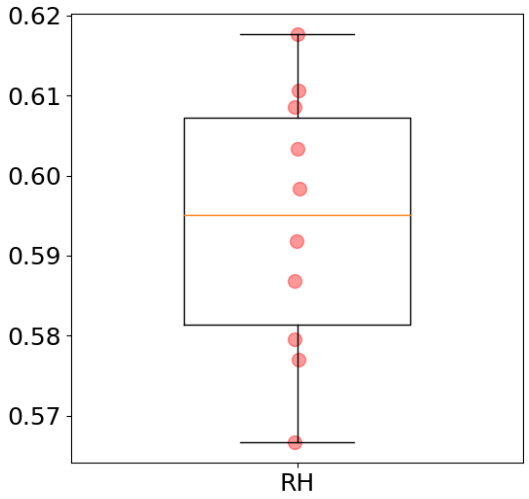


Figure 7: Box plot of the labeled recall-homogeneity of 10 runs of the induction model with a bias for forward function application (Experiment 2). The mean and median RH were 0.59 and 0.60 respectively. Unlabeled F1 is not shown because all runs reached an identical score of F1=0.76.

tion 4.2. Each dimension in the vector **k** was set to 100; this value was large enough to ensure that forward function application and right-branching tree structures were exclusively used. Smaller values for **k** were also tested on the Adam corpus but achieved slightly worse log likelihoods.

6.2 Results

Figure 7 shows the RH across the 10 runs in this experiment. Because the models invariably predicted right-branching structures, all had the same F1 score of 0.76. Compared to Experiment 1, RH scores showed much more consistency, with a relatively uniform spread of values within the narrow range of 0.57 to 0.62. (Since recall did not vary, all variation in RH was due to differences in homogeneity between models.) Despite the models' inflexibility in assigning tree structures, these RH scores surpassed those reported by Jin et al. (2021a,b).

7 Discussion and Conclusion

We introduce an induction model that learns a basic categorial grammar from unlabeled data. The original version of the model, tested in Experiment 1, shows promising results in several runs, but inconsistent performance in general. A modified version of the model that consistently uses forward function application far outperforms the original model. In general, the experimental results appear to support the empiricist claim that syntactic structure is learnable with relatively simple prior knowledge.

While results from the biased model achieve an impressive RH compared to Jin et al. (2021a,b), they leave open several questions. One obvious question is whether it is possible to consistently achieve comparable results to the biased runs without removing the model's ability to do backward function application, since this operation is a core ingredient of basic categorial grammars and is regularly used in hand-labeled parses of English sentences, e.g., to combine the NP and N\NP in Figure 1. Although the log likelihood objective on its own appears to be insufficient to ensure stable behavior (similar to behavior reported in earlier PCFG studies such as Johnson et al. 2007), it may be possible to find a middle ground between Experiments 1 and 2 with a modified objective function or a weaker form of bias.

Another question is whether the induction model can support more complex operations, such as the forward and backward composition operations defined by CCG. This seems possible in principle; additional weight matrices could be added to Equation (12), so that probabilities for additional operations could be learned. We are excited to explore this possibility in future work.

Limitations

More work is needed to uncover the causes of the inconsistent performance across randomly initialized models in Experiment 1. Although the bias toward forward function application implemented in Experiment 2 was effective in our experiments, it is unlikely to work as a general-purpose method, since languages vary in their branching characteristics and in the contexts in which they apply forward and backward function application.

Ethics Statement

We foresee no ethical issues arising from this research. The reader may refer to Brown (1973) for information on how the Adam and Eve child-directed speech corpora were collected.

Acknowledgements

We thank the anonymous reviewers for their helpful comments. This work was supported by the National Science Foundation grant #1816891. All views expressed are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Kazimierz Ajdukiewicz. 1935. Die syntaktische konnexitat. In S. McCall, editor, *Polish Logic 1920-1939*, pages 207–231. Oxford University Press. Translated from Studia Philosophica 1: 1–27.

Yehoshua Bar-Hillel. 1953. A quasi-arithmetical notation for syntactic description. *Language*, 29:47–58.

Aditya Bhargava and Gerald Penn. 2020. Supertagging with ccg primitives. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 194–204.

Yonatan Bisk, Christos Christodoulopoulos, and Julia Hockenmaier. 2015. Labeled grammar induction with minimal supervision. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 870–876.

Yonatan Bisk and Julia Hockenmaier. 2012. Simple robust grammar induction with combinatory categorial grammars. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*.

Yonatan Bisk and Julia Hockenmaier. 2013. An hdp model for inducing combinatory categorial grammars. *Transactions of the Association for Computational Linguistics*, 1:75–88.

R. Brown. 1973. *A First Language*. Harvard University Press, Cambridge, MA.

Glenn Carroll and Eugene Charniak. 1992. Two Experiments on Learning Probabilistic Dependency Grammars from Corpora. *Working Notes of the Workshop on Statistically-Based {NLP} Techniques*, (March):1–13.

Noam Chomsky. 1965. *Aspects of the Theory of Syntax*. MIT Press, Cambridge, Mass.

Lifeng Jin, Finale Doshi-Velez, Timothy Miller, William Schuler, and Lane Schwartz. 2018. Depth-bounding is effective: Improvements and evaluation of unsupervised PCFG induction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2721–2731.

Lifeng Jin, Byung-Doh Oh, and William Schuler. 2021a. Character-based PCFG induction for modeling the syntactic acquisition of morphologically rich languages. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4367–4378, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lifeng Jin, Lane Schwartz, Finale Doshi-Velez, Timothy Miller, and William Schuler. 2021b. Depth-Bounded Statistical PCFG Induction as a Model of Human Grammar Acquisition. *Computational Linguistics*, 47(1):181–216.

Mark Johnson, Thomas Griffiths, and Sharon Goldwater. 2007. Bayesian inference for PCFGs via Markov chain Monte Carlo. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 139–146, Rochester, New York. Association for Computational Linguistics.

Yoon Kim, Chris Dyer, and Alexander Rush. 2019. Compound probabilistic context-free grammars for grammar induction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2369–2385.

Dan Klein and Christopher D. Manning. 2002. A generative constituent-context model for improved grammar induction. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

Tom Kwiatkowski, Eunsol Choi, Yoav Artzi, and Luke Zettlemoyer. 2013. Scaling semantic parsers with on-the-fly ontology matching. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1545–1556.

Tom Kwiatkowski, Luke S. Zettlemoyer, Sharon Goldwater, and Mark Steedman. 2010. Inducing probabilistic CCG grammars from logical form with higher-order unification. In *EMNLP*, pages 1223–1233.

Karim Lari and Steve J Young. 1990. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer speech & language*, 4(1):35–56.

Brian MacWhinney. 2000. *The CHILDES project: Tools for analyzing talk*, third edition. Lawrence Elrbaum Associates, Mahwah, NJ.

Lisa Pearl and Jon Sprouse. 2013. Syntactic islands and learning biases: Combining experimental syntax and computational modeling to investigate the language acquisition problem. *Language Acquisition*, 20:23–68.

Jakob Prange, Nathan Schneider, and Vivek Srikumar. 2021. Supertagging the long tail with tree-structured decoding of complex categories. *Transactions of the Association for Computational Linguistics*, 9:243–260.

Geoffrey K Pullum and Barbara C Scholz. 2002. Empirical assessment of stimulus poverty arguments. *The linguistic review*, 19(1-2):9–50.

Jenny R Saffran, Richard N Aslin, and Elissa L Newport. 1996. Statistical learning by 8-month-old infants. *Science*, 274(5294):1926–1928.

Yoav Seginer. 2007. Fast unsupervised incremental parsing. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 384–391.

Miloš Stanojević, Shohini Bhattasali, Donald Dunagan, Luca Campanelli, Mark Steedman, Jonathan Brennan, and John Hale. 2021. Modeling incremental language comprehension in the brain with combinatory categorial grammar. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 23–38.

Mark Steedman. 2000. *The syntactic process*. MIT Press/Bradford Books, Cambridge, MA.

Michael Tomasello. 2005. *Constructing a language: A usage-based theory of language acquisition*. Harvard university press.

Luke Zettlemoyer and Michael Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the Proceedings of the Twenty-First Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pages 658–666, Arlington, Virginia. AUAI Press.

Songyang Zhang, Linfeng Song, Lifeng Jin, Haitao Mi, Kun Xu, Dong Yu, and Jiebo Luo. 2022. Learning a grammar inducer from massive uncurated instructional videos. *arXiv preprint arXiv:2210.12309*.

Songyang Zhang, Linfeng Song, Lifeng Jin, Kun Xu, Dong Yu, and Jiebo Luo. 2021. Video-aided unsupervised grammar induction. *arXiv preprint arXiv:2104.04369*.

Yanpeng Zhao and Ivan Titov. 2020. Visually grounded compound pcfgs. *arXiv preprint arXiv:2009.12404*.

Hao Zhu, Yonatan Bisk, and Graham Neubig. 2020. The return of lexical dependencies: Neural lexicalized PCFGs. *Transactions of the Association for Computational Linguistics*, 8:647–661.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
In and following Section 7.

☒ A2. Did you discuss any potential risks of your work?
We see no apparent risks.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Section 1.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 4

☑ B1. Did you cite the creators of artifacts you used?
Section 3.2

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
We are not currently sharing any new artifacts for this project. Should we share them in the future, we will discuss the terms of use.

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
It seems very clear that our extension of the existing Jin et al. 2021 model is being used for the intended purpose, to model grammar induction.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 5.1

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☑ Did you run computational experiments?

Sections 5 and 6

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
These are relatively small-scale models, so there are few concerns about others having the resources to run them

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 5.2

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Sections 5.3 and 6.2

□ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.