Semantic Training Signals Promote Hierarchical Syntactic Generalization in Transformers

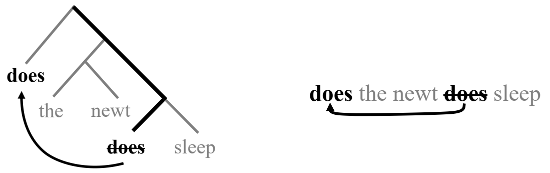
Aditya Yedetore and **Najoung Kim**
Department of Linguistics,
Boston University
yedetore@bu.edu, najoung@bu.edu

Abstract

Neural networks without hierarchical biases often struggle to learn linguistic rules that come naturally to humans. However, neural networks are trained primarily on form alone, while children acquiring language additionally receive data about meaning. Would neural networks generalize more like humans when trained on both form and meaning? We investigate this by examining if Transformers—neural networks without a hierarchical bias—better achieve hierarchical generalization when trained on both form and meaning compared to when trained on form alone. Our results show that Transformers trained on form and meaning do favor the hierarchical generalization more than those trained on form alone, suggesting that statistical learners without hierarchical biases can leverage semantic training signals to bootstrap hierarchical syntactic generalization.

1 Introduction

Language learners encounter sentences through their surface forms: linear sequences of words. However, syntactic rules are sensitive to sentences' underlying hierarchical structure. What evidence lets learners determine that syntactic rules operate on hierarchical structure, rather than linear order? Some contend that the evidence children receive is insufficient for a learner without a hierarchical bias to generalize hierarchically (Chomsky, 1968, 1971, 1980; Berwick et al., 2011). An alternative hypothesis (e.g., Lewis and Elman, 2001) is that learners require no innate hierarchical bias: the input children get includes sufficient cues for hierarchical generalization. Both sides, however, tacitly assume that the data relevant for hierarchical generalization is form alone (i.e., words and their linear order), rather than form and meaning. Since meanings involve hierarchical dependencies which often correspond to syntactic structure (Partee et al., 1984), they may provide additional cues to the hierarchical syntactic generalization.



(a) Hierarchical rule: Move the auxiliary hierarchically closest to the root to the front of the sentence.

(b) Linear rule: Move the auxiliary linearly closest to the left edge to the front of the sentence.

Figure 1: Two possible rules for English yes/no question formation. Modified from McCoy et al. (2020).

The rise of neural networks seems to suggest that the focus on form is warranted: Networks trained on form alone perform well on syntactic evaluations (e.g., Gulordava et al., 2018; Wilcox et al., 2018; Warstadt et al., 2020; Hu et al., 2020; Huebner et al., 2021). However, when networks' input aligns more closely with the sentences children get, models fail to generalize hierarchically, suggesting that attaining hierarchical generalization from form alone requires stronger priors than those of standard neural architectures (Yedetore et al., 2023).

In this work, we test the hypothesis that learners without a hierarchical bias can generalize hierarchically when trained on form and meaning. We train Transformers (Vaswani et al., 2017), an architecture known to prefer linear rules (Petty and Frank, 2021), to translate form to meaning, then test for hierarchical generalization. Following McCoy et al. (2020), our testbed for hierarchical generalization is yes/no question formation, exemplified by the relationship between declarative sentence (1a) and yes/no question (1b). We train models on question formation data like (1) which is consistent with a hierarchical and a linear rule (see Figure 1).

(1) a. The newt ***does*** sleep.
b. ***Does*** the newt sleep?

To test for hierarchical generalization, we evaluate models on examples like (2), where the hierarchical rule predicts (2a), and the linear rule, (2b).

(2) The newt who *does* sleep **doesn't** swim.

a. **Doesn't** the newt who *does* sleep swim?
b. **Does* the newt who sleep **doesn't** swim?

We find that Transformers trained on form and meaning display stronger preferences for the hierarchical rule than Transformers trained on form alone. Our results support the hypothesis that semantic training signals help statistical learners without hierarchical biases learn hierarchical syntactic rules.[1]

2 Background

Meaning representations are not observable in the mind. Forms, however, are observable. This asymmetry makes reasoning about the effect of form on generalization simpler than reasoning about meaning's effect by lessening the need to make assumptions about unobservable representations. Focusing on forms, Chomsky (1971) observes that although English speaking adults acquire the hierarchical rule for yes/no question formation, children's input likely lacks the evidence ruling out the linear rule. Chomsky (1971) conjectures that even a child who never encounters such disambiguating examples (e.g., (2a)) would generalize hierarchically, and argues that a innate hierarchical bias is thus necessary. Empirically, Crain and Nakayama (1987) find that children do behave consistently with the hierarchical rule, and rarely with the linear rule (Ambridge et al., 2008), while disambiguating evidence is very uncommon in children's input (Pullum and Scholz, 2002; Legate and Yang, 2002).

Though such work makes a hierarchical bias seem necessary for child-like generalization, domain general biases may suffice for hierarchical generalization. To explore this possibility, several studies have investigated how artificial learners generalize from form alone. Some argue that their results support an innate hierarchical bias (McCoy et al., 2018; Yedetore et al., 2023), while others argue against this conclusion (Lewis and Elman, 2001; Reali and Christiansen, 2005; Perfors et al., 2011; Bod and Smets, 2012), and still others do not take a strong stance (Frank and Mathis, 2007; Lin et al., 2019; Warstadt and Bowman, 2020).

The direct precursors to our work also vary in their conclusions. McCoy et al. (2020) and Petty and Frank (2021) show that neural networks without hierarchical biases trained on form alone in a sequence-to-sequence setup generalize to the linear rule of question formation. These results support the claim that hierarchical generalization requires a hierarchical bias. However, Murty et al. (2023a) and Ahuja et al. (2024) find that models trained in a language modeling setup on the McCoy et al. (2020) and Petty and Frank's (2021) data generalize linearly early on, but 'grok' the hierarchical generalization after training far beyond saturation on in-domain performance.

Though the role of semantic information in the acquisition of syntax has long been theorized (cf. Chomsky (1965); Pinker (1979)), fewer studies have explored semantic signals' effect on hierarchical generalization. Studying children, Crain and Nakayama (1987) find evidence against Stemmer's (1981) hypothesis about how meanings aid hierarchical generalization, but leave open the possibility that meanings help in other ways. Morgan and Newport (1981) find that visual context aided the acquisition of constituent structure in adult learners of an artificial language, though no more so than adding explicit cues to constituent structure to the forms. This finding with adults, however, does not address how children generalize hierarchically during first language acquisition. Using computational modeling, Fitz and Chang (2017) show that networks with built-in linguistic knowledge generalize hierarchically when trained to map meaning to form, and Abend et al. (2017) explore how semantic training signals help a learner acquire syntactic rules but use statistical modeling techniques that presuppose that syntactic structures must be hierarchical. It is an open question if learners without such built-in knowledge generalize hierarchically when trained on both form and meaning.

The hypothesis we test in this work is in the spirit of the semantic bootstrapping hypothesis: that children leverage sentences paired with structured meaning representations to acquire syntactic rules (Abend et al., 2017). In this work, we generalize semantic bootstrapping to the problem of determining that syntactic rules must be sensitive to hierarchical structure rather than linear order.

3 Experiments

In this work, we train models to form yes/no questions in two ways. In Exp. 1 (Section 4), we train neural networks in a sequence-to-sequence setup on the objective of translating declarative sentences to their yes/no question counterparts. We

[1] GitHub repo with data and code: Will be released soon.

use this setup to enable comparison to Petty and Frank (2021) and McCoy et al. (2020). We then test if models' generalization behavior is more consistent with the linear or the hierarchical rule.

In Exp. 2 (Section 5) we explore grokking: We train models longer and track how training on form and meaning changes models' training dynamics. We additionally train neural networks on the task of language modeling (predicting the next word at every point in a sentence), since grokking may depend on this training objective (Ahuja et al., 2024).

In Exp. 3 (Sections 6 and 7) we vary the representation of meaning and the translation task to investigate several possible causes of the benefit of training models to map form to meaning.

4 Experiment 1: Adding Meaning to McCoy et al. (2020)

4.1 Framing of the Task

In this experiment, we compare the generalization of sequence-to-sequence networks trained on form alone with those additionally trained to translate forms to meanings. Following McCoy et al. (2020), models trained on form alone are tasked with mapping from declarative sentences to themselves as in (3a), or to their yes/no question forms as in (3b), where the input's final token specifies the task.

(3) a. *Input:* the newt does sleep . DECL
Output: the newt does sleep .
b. *Input:* the newt does sleep . QUEST
Output: does the newt sleep ?

Models trained on form and meaning are additionally tasked with translating declarative sentences into logical representations of their meaning as in (4), a task adapted from Kim and Linzen (2020).[2]

(4) *Input:* the newt does sleep . TRANS
Output: Sleep (ι x . Newt (x))

Crucially, all the training instances for the question formation task are consistent with both the hierarchical and the linear generalizations (Figure 1). To evaluate what models learn from the training data, we test models on examples like (2), for which the hierarchical rule produces well-formed questions, like (2a), while the linear rule produces ill-formed questions, like (2b). Table 1 shows the distribution of the training and evaluation data.

4.2 Datasets

We construct two synthetic datasets for this experiment. FORM ALONE consists of data generated using the probabilistic context-free grammar of McCoy et al. (2020), with slight modifications. Specifically, we exclude the quantificational determiner *some* to simplify the semantics, and modify the sampling algorithm to balance the number of DECL and QUEST examples.[3] This dataset consists of 50,000 declarative-declarative pairs like (3a), and 50,000 declarative-question pairs like (3b).

The second dataset, FORM & MEANING, consists of the data in FORM ALONE plus additional input-output pairs generated by translating each input sentence in FORM ALONE's training set into a logical representation of the sentence's meaning, as in (4). The translation is specified by a compositional semantics, listed in Appendix C.3.[4] For the meaning representation, we mostly follow the assumptions of Coppock and Champollion (2022), a textbook which employs notation that is standard in the field of formal semantics, except that we simplify the semantics by treating singular and plural terms equivalently. For instance, the meanings of both *the newt does sleep*, where the subject is singular, and *the newts do sleep*, where the subject is plural, are represented as Sleep(ιx.Newt(x)).

The size of the test set and the generalization set are both 10,000. The test set contains 5,000 declarative-declarative pairs, while the generalization set contains only declarative-question pairs. The test and generalization sets are the same for FORM ALONE and FORM & MEANING.

4.3 Architectures and Training Setup

We use Transformers (Vaswani et al., 2017) in a sequence-to-sequence setup following Petty and Frank's (2021) hyperparameters: 4 heads, embedding size of 128, 3 layers, trained with early stopping. Additionally, we use a batch size of 128, and a patience of 5. We implement models with Open-

[2] We do not claim that the exact structured logical representations of meaning that we use in this work is a part of the input that children explicitly receive. Rather, it is likely that semantic cues available to children derive from language-external modalities, such as visual input. The meaning representations we provide in our experiments correspond to a conservative *upper bound* to what the child could determine about the meaning of the sentence that they heard, possibly leveraging language-external cues. We seek to explore if under this idealized scenario there are benefits to syntactic generalization, which is a precondition to expecting benefits of noisier, more realistic semantic signals.

[3] See Appendix C for our grammar's syntax & vocabulary.

[4] We implement the semantics to generate the translations using the Lambda Notebook (BSD 3-Clause License): https://rawlins.io/research/lambdanotebook/

	DECL/QUEST	TRANS
No RC	The newts do see the yak by the zebra. → The newts do see the yak by the zebra. The newts do see the yak by the zebra. → Do the newts see the yak by the zebra?	The newts do see the yak by the zebra. $\rightarrow \text{See}(\iota x.\text{Newt}(x), \iota y.\text{Yak}(y)\wedge \text{By}(y, \iota z.\text{Zebra}(z)))$
RC on object	The newts do see the yak who doesn't fly. → The newts do see the yak who doesn't fly. The newts do see the yak who doesn't fly. → Do the newts see the yak who doesn't fly?	The newts do see the yak who doesn't fly. $\rightarrow \text{See}(\iota x.\text{Newt}(x), \iota y.\text{Yak}(y) \wedge \neg\text{Fly}(y))$
RC on subject	The newts who don't fly do see the yak. → The newts who don't fly do see the yak. The newts who don't fly do see the yak. → Do the newts who don't fly see the yak?	The newts who don't fly do see the yak. $\rightarrow \text{See}(\iota x.\text{Newt}(x) \wedge \neg\text{Fly}(x), \iota y.\text{Yak}(y))$

Table 1: White cells (□) indicate the type of data in the Form Alone training set and the in-distribution test set. Light gray cells (▨) indicate the additional data in the Form & Meaning training set. Dark gray cells (■) indicate the generalization data. RC stands for "relative clause." To save space, this table uses some words not present in the vocabulary used to generate the training instances. For instance, *fly* is not in the vocabulary, though *see* is.

NMT (Klein et al., 2017).[5] See Appendix A for more hyperparameter details. We tokenize declaratives and questions by splitting at whitespaces, and tokenize meaning representations as in (4).

4.4 Evaluation

Following McCoy et al. (2020), we use two evaluation metrics: full sentence accuracy on the test set (consisting of held out examples similar to those seen in training), and first word accuracy on the generalization set. Full sentence accuracy measures if the model's output is exactly correct given the input. The first word metric evaluates whether the first word of the question is correct, abstracting away from extraneous errors irrelevant to the choice between hierarchical and linear generalizations (e.g., the model may incorrectly replace the verb *sleep* with *giggle*). Crucially, the first word of the question is sufficient to disambiguate the linear and hierarchical rules. For instance, when given (2) as an input, a model that has learned the hierarchical rule would choose *doesn't* as the first word of the output, as in (2a), while a model that has learned the linear rule would choose *does* as the first word of the output, as in (2b).

4.5 Results

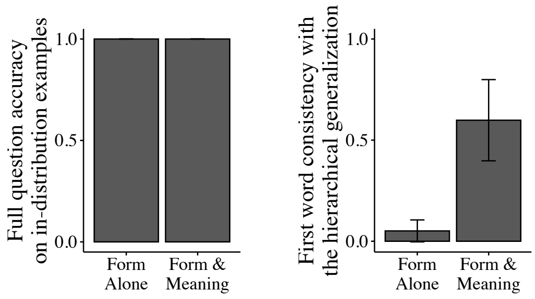
Across ten random reruns, Transformers trained on Form Alone and those trained on Form & Meaning achieve perfect performance on the test set: they always produce the full question correctly. This indicates that models successfully learned to handle questions like those seen during training. On the generalization set, models seldom produce the full sentence correctly, replicating prior findings (Petty and Frank, 2021). Turning to the more lenient measure of first word accuracy, Transformers trained on Form Alone generalize linearly (choosing the linear option for 95% of the generalization sentences, and the hierarchical option for 5%), again replicating prior findings of Petty and Frank (2021). On the other hand, Transformers trained additionally on meaning prefer the hierarchical generalization (60% hierarchical, 40% linear).[6] See Figure 2 for a summary.

5 Experiment 2: Grokking

Recent work suggests that Transformer models display *structural grokking*: when trained past saturation on in-domain accuracy, out-of-domain generalization continues to improve, and eventually hierarchical generalization is achieved (Murty et al., 2023a). This raises the possibility that early stopping caused the lack of hierarchical generalization in Experiment 1. To explore the interaction between grokking and semantic training signals, we train models on the datasets from Experiment 1 (namely Form Alone and Form & Meaning),

[5] https://github.com/OpenNMT/OpenNMT-py

[6] We also train models on the original (form alone) data from McCoy et al. (2020), which includes *some* and has 8% more DECL than QUEST examples, to ensure that our findings are not due to our minor modifications to the sampling process. We find similar results to those for Form Alone in Experiment 1. See Appendix D.



(a) Test Set

(b) Generalization Set

Figure 2: Plot (a) shows the full question accuracy on the test set. Plot (b) shows the first word accuracy on the generalization set. Results are averages across 10 random reruns. Error bars are single standard deviations.

but far past the point of perfect in-domain accuracy.

5.1 Architecture and Training Setup

Since prior work (Ahuja et al., 2024) has found that grokking varies according to the choice of a sequence-to-sequence or a language modeling setup, we use both in this experiment. For both setups, we use hyperparameters following those in Ahuja et al. (2024): 8 heads, embedding size of 512, 6 layers, word-level tokenization, and batch size 8. We use the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.0001, and train models for 300k steps, without early stopping.[7] See Appendix A for more hyperparameters details.

5.2 Results

Our results are reported in Figure 3. With language modeling, we find that models trained on FORM ALONE do display grokking (i.e., they at first generalize linearly, but then shift to hierarchical generalization after a long period of training), consistent with the results in Murty et al. (2023a). However, we observe large variability in generalization across random seeds: some models display perfect hierarchical generalization after 300k training steps, while others only show a slight preference. In contrast to the FORM ALONE result, the Transformers trained on FORM & MEANING exhibit on average much stronger hierarchical generalization much faster, with much less variability.

In the sequence-to-sequence setup, consistent with the results in Ahuja et al. (2024), models trained on FORM ALONE do not show grokking—they stay below 50% accuracy past 100k training steps. On the other hand, when trained on FORM & MEANING, these models quickly generalize hierarchically, predominantly staying above 75% first word accuracy after 10k training steps.

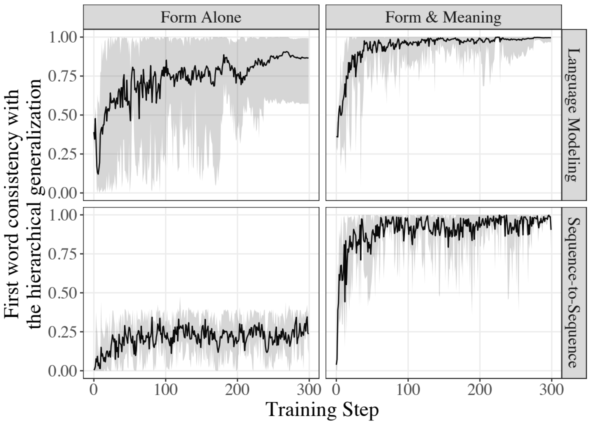


Figure 3: Left: first word accuracy across 300k training steps for models trained on FORM ALONE (left) or FORM AND MEANING (right), in a language modeling (top) or sequence-to-sequence (bottom) setup. The black line indicates the average across 10 random reruns. Light gray shaded areas are the minimum and maximum values for any model at each training step across the 10 random reruns. All models reach in-domain saturation (>99% full question accuracy) by 10k training steps.

Overall, our results suggest that models trained on form and meaning generalize more consistently to the hierarchical rule than models trained on form alone, and that this preference emerges much earlier compared to models that exhibit grokking.

6 Experiment 3: Why Do the Meaning Representations Help?

We found in Experiments 1 and 2 that adding meaning to models' training data resulted in a stronger preference for hierarchical generalization. In this experiment, we explore the following questions:

(5) a. How dependent is the increased preference for hierarchical generalization on specific aspects of the meaning representation?
  b. Which aspects of the form-to-meaning translation task aid hierarchical generalization?

We build seven new datasets: three designed to address Question (5a) (Section 6.1), and four to address Question (5b) (Sections 6.2 and 6.3).

[7] We use Murty et al.'s (2023a) code for the language modeling setup: `https://github.com/MurtyShikhar/structural-grokking`

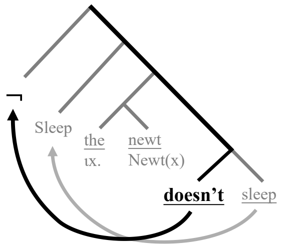


Figure 4: The correlation between the hierarchical rule and the relation between the main auxiliary (*doesn't*) and the sentential negation (¬). Compare to the hierarchical rule in Figure 1.

6.1 Specifics of the Meaning Representation?

One reason that the data in Experiments 1 and 2 may lead to hierarchical generalization is that the relationship between the main auxiliary and the first word of the question strongly parallels the relationship between the negation's location in the declarative and in the meaning representation. For instance, though models do not receive questions like (6a), they do receive meanings like (6b), where the relationship between the main auxiliary (*don't* in (6)) and the negation (¬ in (6b)) closely corresponds to the relationship between that main auxiliary in the declarative and the first word in the question in (6a). See Figure 4 for an illustration.

(6) the newts who do fly don't see the yak .
  a. don't the newts that do fly see the yak ?
  b. $\underline{\neg}\text{See}(\iota x.\text{Newt}(x) \land \text{Fly}(x), \iota y.\text{Yak}(y))$

To test whether this close parallel is responsible for the models' preference for hierarchical generalization, we remove the negation from the dataset, transition to an event semantic representation where the element of meaning corresponding to the auxiliary is no longer directly at the front of the meaning representation, and introduce the necessary variability in the auxiliary using tense.[8] For example, (7a) translates to (7b):

(7) a. the newt did fly .
  b. $\exists e : \underline{\text{Past}}(e) \land \text{Fly}(e, \iota x.\text{Newt}(x))$

Now the element of meaning corresponding to the auxiliary is no longer directly at the front of the meaning representation, due to '$\exists e$:'. However, the relationship between (7a) and (7b) still bears a close correspondence to the hierarchical question formation rule. As (8) shows, the tense predicate (Past) corresponding to the main auxiliary (did) appears near the front of the meaning representation.

(8) a. the newts who do fly did see the yak .
  b. $\exists e : (\underline{\text{Past}}(e) \land \text{See}(e, \iota x.\text{Newt}(x) \land \exists e' : \text{Pres}(e') \land \text{Fly}(e', x), \iota y.\text{Yak}(y)))$

For this reason, we explore an equivalent semantics in which the tense predicate corresponding to the main auxiliary is located at the end of the meaning representation, as demonstrated in (9) and (10), which are translations of (7a) and (8a), respectively. If we see an equivalent boost in preference for hierarchical generalization using this representation scheme, this suggests that the similarity of the hierarchical question formation rule and the placement of negation at the front of the meaning representation is not the source of hierarchical generalization in Experiments 1 and 2.

(9) $\exists e : \text{Fly}(e, \iota x.\text{Newt}(x)) \land \underline{\text{Past}}(e)$

(10) $\exists e : (\text{See}(e, \iota x.\text{Newt}(x) \land \exists e' : \text{Fly}(e', x) \land \text{Pres}(e'), \iota y.\text{Yak}(y)) \land \underline{\text{Past}}(e))$

6.1.1 Datasets

We rename the datasets from Experiments 1 and 2 FORM[+neg] and MEANING[+neg] to distinguish them from the datasets introduced here. We introduce FORM[+tense], which is like FORM[+neg] but differentiates auxiliaries with tense. MEANING[+tense$_{\text{first}}$] includes the examples in FORM[+tense] plus translations into a meaning representation like (8b). MEANING[+tense$_{\text{last}}$] is like MEANING[+tense$_{\text{first}}$] but with representations as in (9) and (10).

6.1.2 Results

Figure 5 shows the results with the alternative meaning representations. Here, models trained on FORM[+tense] in a language modeling setup vary in their generalization patterns, four of ten showing a preference for the hierarchical generalization (after grokking), and six of ten showing a preference for the linear generalization. Models trained in the sequence-to-sequence setup on FORM[+tense] display no hierarchical generalization.

Now we compare these results to those from Experiment 2 (Figure 3). Though models trained on FORM[+tense] in the the sequence-to-sequence setup behave similarly to models trained on FORM[+neg], the corresponding results in the language modeling setup show differences. Specifically, there is more variability in the generalization of the models trained on FORM[+tense] than the models on FORM[+neg], with one choosing the linear generalization even after 300k training steps.

[8] To use the first word evaluation for hierarchical generalization, we require at least two distinct auxiliaries that share the same number agreement marking (e.g., *does* and *doesn't*). Thus, we cannot remove negation without adding additional auxiliaries. Here we add *did*.

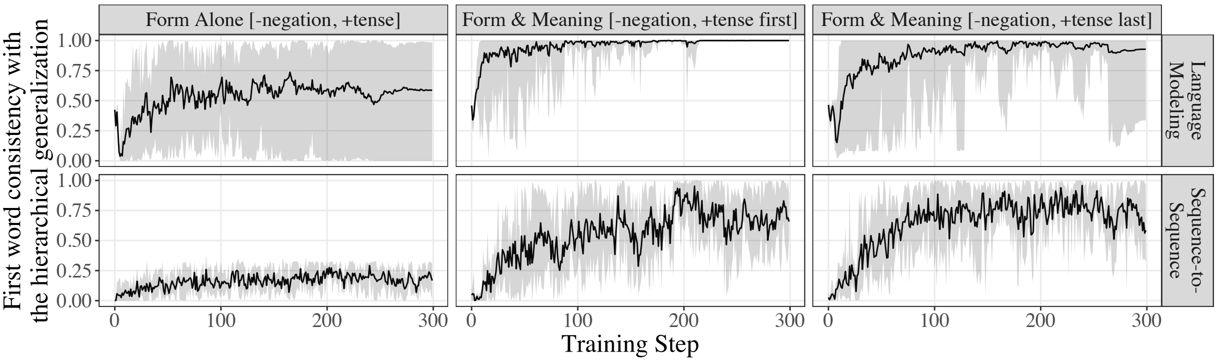


Figure 5: First word consistency with the hierarchical generalization for models trained on the [+tense] datasets. Top: models trained in a language modeling setup. Bottom: models trained in a sequence-to-sequence setup. The black line indicates the average across 10 random reruns. Light gray shaded areas are the minimum and maximum values for any model at each training step across the 10 random reruns.

The relative difficulty models have with generalizing hierarchically when trained on FORM[+tense] makes the models' generalization to the hierarchical rule when trained on MEANING[+tense$_{\text{first}}$] and MEANING[+tense$_{\text{last}}$] more striking. Models in both the language modeling and sequence-to-sequence setups trained on either MEANING[+tense$_{\text{first}}$] or MEANING[+tense$_{\text{last}}$] display stronger hierarchical generalization than when trained FORM[+tense]. These results suggest that the similarity between the hierarchical question formation rule and negation's location in the forms versus meanings does not account for Transformers' behavior in Experiments 1 and 2.

6.2 Relation Between Form and Meaning?

Turning to Question (5b): although we train models to translate form to meaning, the meanings themselves may suffice for hierarchical generalization, rendering the translation from forms to meanings unnecessary. Here, we explore this possibility.

6.2.1 Dataset

The set of examples for this experiment (MEANING TO MEANING) is the same as in Experiments 1 and 2, except the translation task now maps meanings to themselves, rather than mapping declarative sentences to their meanings. For instance, rather than the tasks like in (4), in the new translation task the inputs and outputs are meanings, as in (11).

(11) *Input:* Sleep ($\iota\, x$. Newt (x)) . TRANS
Output: Sleep ($\iota\, x$. Newt (x))

This manipulation ablates training signals about the relationship between form and meaning, leaving only the structures of the meanings themselves, and lets us test if the relationship between the declarative forms and the meaning representations is critical for the benefit to hierarchical generalization.

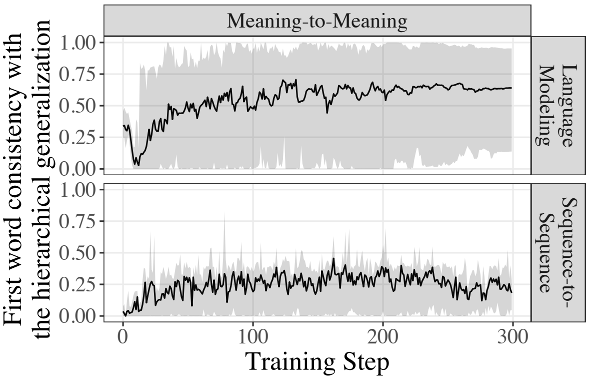


Figure 6: First word consistency with the hierarchical generalization for models trained on the MEANING TO MEANING dataset. The black line indicates the average across 10 random reruns. Light gray shaded areas are the minimum and maximum values for any model at each training step across the 10 random reruns.

6.2.2 Results

See Figure 6. Here, we see no benefit of training on meaning beyond training on form alone: compare the results in Figure 6 with the results in Experiment 2 (Figure 3). These results suggest that mere training on structured meaning representations is not sufficient to induce hierarchical generalization. More specifically, the benefit of training Transformers to translate from form to meaning is not solely due to the mere presence of hierarchical structure in the meaning representations or to the increase in the variety of data fed into the model. Rather, the relation between form and meaning is critical.

6.3 Cues to Syntactic Structure in the Meanings?

In these experiments, continuing to explore (5b), we introduce syntactic translation tasks to determine the cause of the hierarchical generalization.

6.3.1 Datasets

For these datasets, we use the same grammar as in Experiment 1 and 2. Our first dataset, Identify Main Auxiliary, as shown in (12a), explores the possibility that a translation task in which models need to identify the main auxiliary aids hierarchical generalization. Our second dataset, Identify Main Verb, is exemplified in (12b).

Another possible reason for the benefit of the meanings is that the meaning representation contains hierarchical structure that is similar to the syntactic structure underlying the sentence. Translating declarative to such hierarchical structures might be the source of the hierarchical generalization. To explore this possibility, we introduce Constituency Parsing, shown in (12c).

(12) *Input:* the newt does sleep . trans

a. *Output* (Identify Main Auxiliary):
the newt (does) sleep .

b. *Output* (Identify Main Verb):
the newt does (sleep) .

c. *Output* (Constituency Parsing):
[[[the newt] [does sleep]] .]

6.3.2 Results

See Figure 7. In the language modeling setup, models trained on Identify Main Auxiliary quickly generalize to the hierarchical rule with little variation between random seeds. Models trained on Identify Main Verb and on Constituency Parsing in the language modeling setup often generalize to the hierarchical rule, though with large variability. Models trained in the sequence-to-sequence setup do not generalize to the hierarchical rule, but instead prefer the linear rule.

Although training on Identify Main Auxiliary causes stronger hierarchical generalization in the language modeling setup, Identify Main Verb provides no benefit compared to Form Alone in Experiment 2 (see Figure 3). These results support the conclusion that the task of translating sentences to hierarchical representations is not the source of hierarchical generalization. Rather, the benefit seems to be due to the requirement that the neural network identify the main auxiliary.

This result suggests that, though other reasons are possible, identifying the main auxiliary may be key to hierarchical generalization. If so, the reason for grokking on Form Alone could be that cues in the training data make models identify the main auxiliary as distinct from the first auxiliary. One such cue is the presence of subject-auxiliary agreement. We explore this possibility in the next experiment.

7 Experiment 3.5: Ablating Agreement

In McCoy et al.'s (2020) grammar, the subject agrees in number with the hierarchically determined main auxiliary, as shown in examples (13), in which the number of the subject (*newt* vs. *newts*) determines the form of the auxiliary (*does* vs. *do*).

(13) the $\left[\begin{smallmatrix}\text{newt}\\ \text{newts}\end{smallmatrix}\right]$ $\left[\begin{smallmatrix}\text{does}\\ \text{do}\end{smallmatrix}\right]$ sleep .

We hypothesize that the presence of subject-auxiliary agreement drives the grokking of the hierarchical generalization in models trained on Form[+neg]. We test this by removing the subject-auxiliary agreement from the models' training data. Importantly, this ablation of subject-auxiliary agreement does not fundamentally change the generalization problem the models face: the training data is still ambiguous between the linear and hierarchical rules shown in Figure 1.

7.1 Datasets

The datasets in this experiment are similar to those in Experiments 1 and 2 (Form[+neg] and Meaning[+neg]), except the grammar is modified to exclude plural nouns and auxiliaries. This means that the auxiliary *does* and *doesn't* are present, but *do* and *don't* are excluded. These datasets, which we call Form[−agr] and Meaning[−agr] allow us to test the extent to which structural grokking is due the subject-auxiliary agreement.

7.2 Results

See Figure 8. When trained on Form[−agr], eight of the ten transformers generalize linearly from step 50k on (100% linear, 0% hierarchical), while two models generalize partially to the hierarchical rule (43% and 67% hierarchical after 300k training steps). This suggests that the source of structural grokking in the language modeling setup is the subject-auxiliary agreement, serving as a cue to identify the main auxiliary.

On Meaning[−agr], models generalize hierarchically. These results are similar to the results

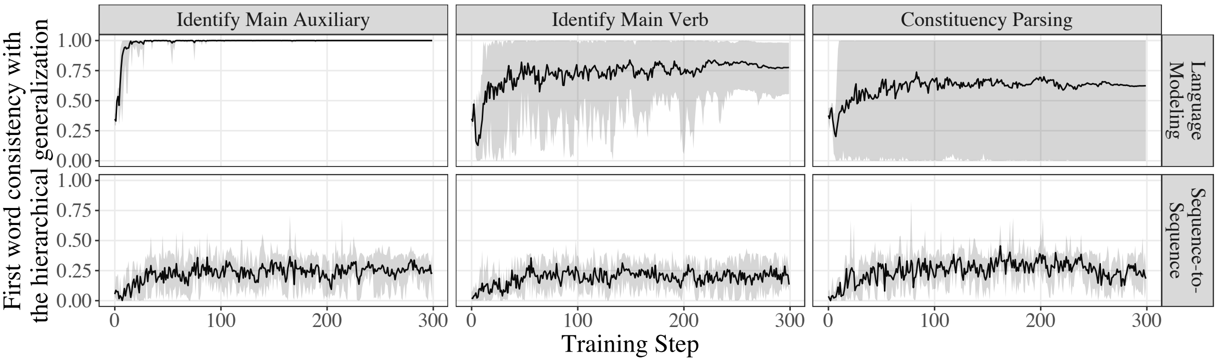


Figure 7: First word consistency with the hierarchical generalization for models trained on the datasets containing auxiliary tasks from Experiment 3. Top: models trained in a language modeling setup. Bottom: models trained in a sequence-to-sequence setup. The black line indicates the average across 10 random reruns. Light gray shaded areas are the minimum and maximum values for any model at each training step across the 10 random reruns.

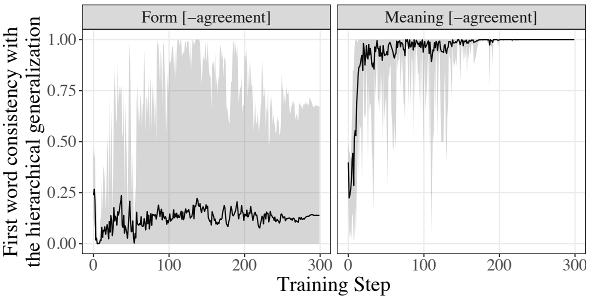


Figure 8: First word consistency with the hierarchical generalization for models trained on the [−agr] datasets. The black line indicates the average across 10 random reruns. Light gray shaded areas are the minimum and maximum values for any model at each training step across the 10 random reruns.

on MEANING[+neg] in Experiment 2, and suggest that the form to meaning translation allows the models to generalize hierarchically even in the absence of subject-auxiliary agreement.

8 Discussion

Across three experiments and several training setups, we find that training models to translate from form to meaning enables stronger hierarchical generalization than training on form alone. These results suggest that one avenue to hierarchical generalization in learners without a hierarchical bias is leveraging semantic signals in the training data.

For Transformers specifically, our results show that Transformers generalize more like humans when trained to translate forms to meanings than when trained on form alone. In practice, this takeaway must be tempered by the possibility that the large quantities of form that large language models receive make the benefits of training to translate form to meaning inconsequential. However, recent results suggest that translations from forms to meaning-like representations may provide benefits even to language models trained at scale: Kim et al. (2024) find that large language models trained additionally on code performed better on a entity tracking task. This benefit may be due to the the presence of translations from natural language sentences to code in the training data. Further work is necessary to disentangle this possibility from other possible contributing factors.

For child acquisition, our results suggest that one possible source of hierarchical generalization is the relationship between forms and meanings, so long as children can construct logical meaning representations either innately or develop this capacity sometime before making hierarchical generalizations. The early development of such logical capabilities is consistent with recent work on the logic in infants (e.g., Cesana-Arlotti et al. 2018), though this line of research is still in its early stages.

Though in this work we focus on meaning, it may be that other language-external cues also facilitate hierarchical generalization. For instance, prosody (Morgan and Demuth, 2014) may also provide a hierarchical signal of a similar nature to meanings, and visual information (Shi et al., 2019; Wang et al., 2023) may provide information about lexical semantics that is useful to determine how meanings must combine. Future work should also better align the input with what children get, perhaps following the lead of Yedetore et al. (2023) and using a corpus of child-directed speech as model training data, to strengthen the inferences about the innate biases necessary for children to acquire language.

Limitations

We view our behavioral analysis in this work as a strong starting point for understanding how semantic training signals affect generalization to hierarchical syntactic rules in Transformers. However, we see it as critical that future work look into the internal mechanisms of these models to determine the computations underlying model behavior. This will allow the determination of whether hierarchical generalization corresponds to hierarchical representation, or if neural networks that generalize hierarchically employ shortcut mechanisms that do not involve hierarchical representation.

With respect to child acquisition, the connection between our work and the acquisition problem children face hinges on our assumption that the child can recover a structured representation of the meaning of a sentence from utterances and their contexts. If children cannot construct structured representations of meaning given a sentence and its context, our work may not bear on the language acquisition problem. Future work is needed to determine the nature of the meaning representations children can recover from context.

Acknowledgments

We are grateful to Hayley Ross, Liz Coppock, the audience at the 2024 New England Natural Language Processing (NENLP) meetup, the MIT Computational Psycholinguistics Lab, tinlab at BU, and the audience at Harvard's Language and Cognition, for helpful comments and feedback. Any errors are our own. We are also grateful to the Boston University Shared Computing Cluster (SCC) for providing the computing resources used in our experiments.

References

Omri Abend, Tom Kwiatkowski, Nathaniel J. Smith, Sharon Goldwater, and Mark Steedman. 2017. Bootstrapping language acquisition. *Cognition*, 164:116–143.

Kabir Ahuja, Vidhisha Balachandran, Madhur Panwar, Tianxing He, Noah A. Smith, Navin Goyal, and Yulia Tsvetkov. 2024. Learning syntax without planting trees: Understanding when and why transformers generalize hierarchically.

Ben Ambridge, Caroline F Rowland, and Julian M Pine. 2008. Is structure dependence an innate constraint? New experimental evidence from children's complex-question production. *Cognitive Science*, 32(1):222–255.

Robert C. Berwick, Paul Pietroski, Beracah Yankama, and Noam Chomsky. 2011. Poverty of the stimulus revisited. *Cognitive Science*, 35(7):1207–1242.

Rens Bod and Margaux Smets. 2012. Empiricist solutions to nativist puzzles by means of unsupervised TSG. In *Proceedings of the Workshop on Computational Models of Language Acquisition and Loss*, pages 10–18, Avignon, France. Association for Computational Linguistics.

Nicoló Cesana-Arlotti, Ana Martín, Ernő Téglás, Liza Vorobyova, Ryszard Cetnarski, and Luca L. Bonatti. 2018. Precursors of logical reasoning in preverbal human infants. *Science*, 359(6381):1263–1266.

Noam Chomsky. 1965. *Aspects of the Theory of Syntax*. The MIT Press.

Noam Chomsky. 1968. *Language and mind*. Harcourt, Brace & World, New York.

Noam Chomsky. 1971. *Problems of Knowledge and Freedom: The Russell Lectures*. Pantheon Books, New York.

Noam Chomsky. 1980. On cognitive structures and their development: A reply to piaget. *Language and learning: the debate between Jean Piaget and Noam Chomsky*, pages 35–54.

Elizabeth Coppock and Lucas Champollion. 2022. *Invitation to formal semantics*. Manuscript in Preparation.

Stephen Crain and Mineharu Nakayama. 1987. Structure dependence in grammar formation. *Language*, pages 522–543.

Hartmut Fitz and Franklin Chang. 2017. Meaningful questions: The acquisition of auxiliary inversion in a connectionist model of sentence production. *Cognition*, 166:225–250.

Robert Frank and Donald Mathis. 2007. Transformational networks. *Models of Human Language Acquisition*, pages 22–27.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. 2018. Colorless green recurrent networks dream hierarchically. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1195–1205, New Orleans, Louisiana. Association for Computational Linguistics.

Jennifer Hu, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger Levy. 2020. A systematic assessment of syntactic generalization in neural language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1725–1744, Online. Association for Computational Linguistics.

Philip A. Huebner, Elior Sulem, Cynthia Fisher, and Dan Roth. 2021. BabyBERTa: Learning more grammar with small-scale child-directed language. In *Proceedings of CoNLL*.

Najoung Kim and Tal Linzen. 2020. COGS: A compositional generalization challenge based on semantic interpretation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9087–9105, Online. Association for Computational Linguistics.

Najoung Kim, Sebastian Schuster, and Shubham Toshniwal. 2024. Code pretraining improves entity tracking abilities of language models.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

Julie Anne Legate and Charles D Yang. 2002. Empirical re-assessment of stimulus poverty arguments. *The Linguistic Review*, 19(1-2):151–162.

John Lewis and Jeffrey Elman. 2001. Learnability and the statistical structure of language: Poverty of stimulus arguments revisited. *Proceedings of the 26th Annual Boston University Conference on Language Development*, 1.

Yongjie Lin, Yi Chern Tan, and Robert Frank. 2019. Open sesame: Getting inside BERT's linguistic knowledge. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 241–253, Florence, Italy. Association for Computational Linguistics.

R. Thomas McCoy, Robert Frank, and Tal Linzen. 2018. Revisiting the poverty of the stimulus: hierarchical generalization without a hierarchical bias in recurrent neural networks. In *Proceedings of the 40th Annual Meeting of the Cognitive Science Society*, Madison, WI.

R. Thomas McCoy, Robert Frank, and Tal Linzen. 2020. Does syntax need to grow on trees? sources of hierarchical inductive bias in sequence-to-sequence networks. *Transactions of the Association for Computational Linguistics*, 8:125–140.

James L Morgan and Katherine Demuth. 2014. *Signal to syntax: Bootstrapping from speech to grammar in early acquisition*. Psychology Press.

James L Morgan and Elissa L Newport. 1981. The role of constituent structure in the induction of an artificial language. *Journal of verbal learning and verbal behavior*, 20(1):67–85.

Shikhar Murty, Pratyusha Sharma, Jacob Andreas, and Christopher Manning. 2023a. Grokking of hierarchical structure in vanilla transformers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 439–448, Toronto, Canada. Association for Computational Linguistics.

Shikhar Murty, Pratyusha Sharma, Jacob Andreas, and Christopher D Manning. 2023b. Characterizing intrinsic compositionality in transformers with tree projections. In *The Eleventh International Conference on Learning Representations*.

Barbara Partee et al. 1984. Compositionality. *Varieties of formal semantics*, 3:281–311.

Andrew Perfors, Josh Tenenbaum, and Terry Regier. 2011. The learnability of abstract syntactic principles. *Cognition*, 118:306–338.

Jackson Petty and Robert Frank. 2021. Transformers generalize linearly.

Steven Pinker. 1979. Formal models of language learning. *Cognition*, 7(3):217–283.

Geoffrey K. Pullum and Barbara C. Scholz. 2002. Empirical assessment of stimulus poverty arguments. *The Linguistic Review*, 18(1-2):9–50.

Florencia Reali and Morten H. Christiansen. 2005. Uncovering the richness of the stimulus: Structure dependence and indirect statistical evidence. *Cognitive Science*, 29(6):1007–1028.

Haoyue Shi, Jiayuan Mao, Kevin Gimpel, and Karen Livescu. 2019. Visually grounded neural syntax acquisition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1842–1861, Florence, Italy. Association for Computational Linguistics.

Nathan Stemmer. 1981. A note on empiricism and structure-dependence. *Journal of Child Language*, 8(3):649–656.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Wentao Wang, Wai Keen Vong, Najoung Kim, and Brenden M Lake. 2023. Finding structure in one child's linguistic experience. *Cognitive science*, 47(6):e13305.

Alex Warstadt and Samuel R Bowman. 2020. Can neural networks acquire a structural bias from raw linguistic data? *Proceedings of the 42nd Annual Conference of the Cognitive Science Society.*

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R.

Bowman. 2020. BLiMP: A benchmark of linguistic minimal pairs for English. In *Proceedings of the Society for Computation in Linguistics 2020*, pages 409–410, New York, New York. Association for Computational Linguistics.

Ethan Wilcox, Roger Levy, Takashi Morita, and Richard Futrell. 2018. What do RNN language models learn about filler–gap dependencies? In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 211–221, Brussels, Belgium. Association for Computational Linguistics.

Aditya Yedetore, Tal Linzen, Robert Frank, and R. Thomas McCoy. 2023. How poor is the stimulus? evaluating hierarchical generalization in neural networks trained on child-directed speech. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9370–9393, Toronto, Canada. Association for Computational Linguistics.

A Hyperparameters

For our experiments in both the language modeling and sequence-to-sequence setups, we additionally use the following hyperparameters: AdamW ($\beta 1$: 0.9, $\beta 2$: 0.999, ϵ: 1e-7), and linear warmup scheduling for 10k steps. We clip gradients to have a max L2 norm of 10, and tie input and output embeddings for the encoder and decoder in the sequence-to-sequence setup.

B Model implementation

OpenNMT (Klein et al., 2017), used in this work for the sequence-to-sequence setup, has a MIT License. Though the codebase developed in Murty et al. (2023a), used in this work for the language modeling setup, does not specify a license, that code is built upon the codebase of Murty et al. (2023b), which has a MIT License.

Models with trained using these codebases with the hyperparameters from Appendix A have 18M trainable parameters in the language modeling case, and 25M in the sequence-to-sequence case, and take 3 hours to train on Nvidia k80 GPUs. Including all models trained, our experiments take approximately 1000 GPU hours.

C Grammars

C.1 Syntax

This grammar is similar to that of McCoy et al. (2020) (See `https://github.com/tommccoy1/rnn-hierarchical-biases/blob/master/cfgs/question.gr`), but with the quantificational determiner *some* removed in the vocabulary.

$$\text{ROOT} \rightarrow \text{S}\,.$$

$$\begin{aligned}
&\text{S} \rightarrow \text{NP}_{[m,s]}\ \text{VP}_{[m,s]}\\
&\text{S} \rightarrow \text{NP}_{[m,p]}\ \text{VP}_{[m,p]}
\end{aligned}$$

$$\begin{aligned}
&\text{NP}_{[m,s]} \rightarrow \text{Det}\ \text{N}_{[s]}\\
&\text{NP}_{[m,s]} \rightarrow \text{Det}\ \text{N}_{[s]}\ \text{RC}_{[s]}\\
&\text{NP}_{[m,s]} \rightarrow \text{Det}\ \text{N}_{[s]}\ \text{Prep}\ \text{Det}\ \text{N}_{[s]}\\
&\text{NP}_{[m,s]} \rightarrow \text{Det}\ \text{N}_{[s]}\ \text{Prep}\ \text{Det}\ \text{N}_{[p]}
\end{aligned}$$

$$\begin{aligned}
&\text{NP}_{[m,p]} \rightarrow \text{Det}\ \text{N}_{[p]}\\
&\text{NP}_{[m,p]} \rightarrow \text{Det}\ \text{N}_{[p]}\ \text{RC}_{[p]}\\
&\text{NP}_{[m,p]} \rightarrow \text{Det}\ \text{N}_{[p]}\ \text{Prep}\ \text{Det}\ \text{N}_{[s]}
\end{aligned}$$

$$\text{NP}_{[m,p]} \rightarrow \text{Det}\ \text{N}_{[p]}\ \text{Prep}\ \text{Det}\ \text{N}_{[p]}$$

$$\begin{aligned}
&\text{NP}_{[m,o]} \rightarrow \text{Det}\ \text{N}_{[s]}\\
&\text{NP}_{[m,o]} \rightarrow \text{Det}\ \text{N}_{[p]}\\
&\text{NP}_{[m,o]} \rightarrow \text{Det}\ \text{N}_{[s]}\ \text{Prep}\ \text{Det}\ \text{N}_{[s]}\\
&\text{NP}_{[m,o]} \rightarrow \text{Det}\ \text{N}_{[s]}\ \text{Prep}\ \text{Det}\ \text{N}_{[p]}\\
&\text{NP}_{[m,o]} \rightarrow \text{Det}\ \text{N}_{[p]}\ \text{Prep}\ \text{Det}\ \text{N}_{[s]}\\
&\text{NP}_{[m,o]} \rightarrow \text{Det}\ \text{N}_{[p]}\ \text{Prep}\ \text{Det}\ \text{N}_{[p]}\\
&\text{NP}_{[m,o]} \rightarrow \text{Det}\ \text{N}_{[s]}\ \text{RC}_{[s]}\\
&\text{NP}_{[m,o]} \rightarrow \text{Det}\ \text{N}_{[p]}\ \text{RC}_{[p]}
\end{aligned}$$

$$\begin{aligned}
&\text{VP}_{[m,s]} \rightarrow \text{Aux}_{[s]}\ \text{V}_{\text{intrans}}\\
&\text{VP}_{[m,s]} \rightarrow \text{Aux}_{[s]}\ \text{V}_{\text{trans}}\ \text{NP}_{[m,o]}\\
&\text{VP}_{[m,p]} \rightarrow \text{Aux}_{[p]}\ \text{V}_{\text{intrans}}\\
&\text{VP}_{[m,p]} \rightarrow \text{Aux}_{[p]}\ \text{V}_{\text{trans}}\ \text{NP}_{[m,o]}
\end{aligned}$$

$$\begin{aligned}
&\text{RC}_{[s]} \rightarrow \text{Rel}\ \text{Trace}\ \text{Aux}_{[s]}\ \text{V}_{\text{intrans}}\\
&\text{RC}_{[s]} \rightarrow \text{Rel}\ \text{NP}_{[e,s]}\ \text{Aux}_{[s]}\ \text{V}_{\text{trans}}\ \text{Trace}\\
&\text{RC}_{[s]} \rightarrow \text{Rel}\ \text{NP}_{[e,p]}\ \text{Aux}_{[p]}\ \text{V}_{\text{trans}}\ \text{Trace}\\
&\text{RC}_{[s]} \rightarrow \text{Rel}\ \text{Trace}\ \text{Aux}_{[s]}\ \text{V}_{\text{trans}}\ \text{Det}\ \text{N}_{[s]}\\
&\text{RC}_{[s]} \rightarrow \text{Rel}\ \text{Trace}\ \text{Aux}_{[s]}\ \text{V}_{\text{trans}}\ \text{Det}\ \text{N}_{[p]}\\
&\text{RC}_{[p]} \rightarrow \text{Rel}\ \text{Trace}\ \text{Aux}_{[p]}\ \text{V}_{\text{intrans}}\\
&\text{RC}_{[p]} \rightarrow \text{Rel}\ \text{NP}_{[e,s]}\ \text{Aux}_{[s]}\ \text{V}_{\text{trans}}\ \text{Trace}\\
&\text{RC}_{[p]} \rightarrow \text{Rel}\ \text{NP}_{[e,p]}\ \text{Aux}_{[p]}\ \text{V}_{\text{trans}}\ \text{Trace}\\
&\text{RC}_{[p]} \rightarrow \text{Rel}\ \text{Trace}\ \text{Aux}_{[p]}\ \text{V}_{\text{trans}}\ \text{Det}\ \text{N}_{[s]}\\
&\text{RC}_{[p]} \rightarrow \text{Rel}\ \text{Trace}\ \text{Aux}_{[p]}\ \text{V}_{\text{trans}}\ \text{Det}\ \text{N}_{[p]}
\end{aligned}$$

$$\begin{aligned}
&\text{NP}_{[e,s]} \rightarrow \text{Det}\ \text{N}_{[s]}\\
&\text{NP}_{[e,p]} \rightarrow \text{Det}\ \text{N}_{[p]}
\end{aligned}$$

C.2 Vocabulary

As can be determined from inspecting our vocabulary, our data does not contain any information that names or uniquely identifies individual people or any offensive content.

$\mathrm{N}_{[s]} \rightarrow$ *newt* | *orangutan* | *peacock* | *quail* | *raven* | *salamander* | *vulture* | *walrus* | *yak* | *zebra* | *xylophone* | *unicorn* | *tyrannosaurus*

$\mathrm{N}_{[p]} \rightarrow$ *newts* | *orangutans* | *peacocks* | *quails* | *ravens* | *salamanders* | *vultures* | *walruses* | *yaks* | *zebras* | *xylophones* | *unicorns* | *tyrannosauruses*

$\mathrm{V}_{\text{intrans}} \rightarrow$ *giggle* | *smile* | *sleep* | *swim* | *wait* | *move* | *change* | *read* | *eat*

$\mathrm{V}_{\text{trans}} \rightarrow$ *entertain* | *amuse* | *highfive* | *applaud* | *confuse* | *admire* | *accept* | *remember* | *comfort*

Prep $\rightarrow$ *around* | *near* | *beside* | *upon* | *by* | *above* | *behind* | *below*

Det $\rightarrow$ *the* | *my* | *your* | *her* | *our*

Rel $\rightarrow$ *that* | *who*

Trace $\rightarrow$ *t*

C.3 Semantics

We do not handle plurals in our semantics (e.g., $[\![\text{newt}]\!] = [\![\text{newts}]\!]$ for all of the relevant cases)

We exclude *some* from our grammar due to added syntactic/semantic complexities. The quantificational determiner *some* introduces an existential quantifier $\exists$. An example translation of a sentence with *some* is shown in (14).

(14) The newt doesn't see some yak.

a. $\exists x.\neg\text{See}(\iota y.\text{Newt}(y), x) \wedge \text{Yak}(x)$

Since *some* scopes above negation, either the semantics requires an additional rule of type lifting, or the syntax needs to be complicated to include quantifier raising (Coppock and Champollion, 2022). We keep the grammar simple and of similar complexity to prior work by excluding *some*.

$[\![\text{the}]\!] := \lambda f.\iota x.f(x)$

$[\![\text{my}]\!] := \lambda f.\iota x.(f(x) \wedge \text{Poss}(\text{Speaker}, x))$

$[\![\text{your}]\!] := \lambda f.\iota x.(f(x) \wedge \text{Poss}(\text{Addressee}, x))$

$[\![\text{her}]\!] := \lambda f.\iota x.(f(x) \wedge \text{Poss}(y, x) \wedge \text{Female}(y))$

$[\![\text{our}]\!] := \lambda f.\iota x.(f(x) \wedge \text{Poss}(\text{Speaker}, x) \wedge \text{Poss}(\text{Addressee}, x))$

$[\![\text{newt}]\!] := \lambda x.\text{Newt}(x)$

$[\![\text{orangutan}]\!] := \lambda x.\text{Orangutan}(x)$

$[\![\text{peacock}]\!] := \lambda x.\text{Peacock}(x)$

$[\![\text{quail}]\!] := \lambda x.\text{Quail}(x)$

$[\![\text{raven}]\!] := \lambda x.\text{Raven}(x)$

$[\![\text{salamander}]\!] := \lambda x.\text{Salamander}(x)$

$[\![\text{vulture}]\!] := \lambda x.\text{Vulture}(x)$

$[\![\text{walrus}]\!] := \lambda x.\text{Walrus}(x)$

$[\![\text{yak}]\!] := \lambda x.\text{Yak}(x)$

$[\![\text{zebra}]\!] := \lambda x.\text{Zebra}(x)$

$[\![\text{xylophone}]\!] := \lambda x.\text{Xylophone}(x)$

$[\![\text{unicorn}]\!] := \lambda x.\text{Unicorn}(x)$

$[\![\text{tyrannosaurus}]\!] := \lambda x.\text{Tyrannosaurus}(x)$

$[\![\text{around}]\!] := \lambda x.\lambda y.\text{Around}(y, x)$

$[\![\text{near}]\!] := \lambda x.\lambda y.\text{Near}(y, x)$

$[\![\text{beside}]\!] := \lambda x.\lambda y.\text{Beside}(y, x)$

$[\![\text{upon}]\!] := \lambda x.\lambda y.\text{Upon}(y, x)$

$[\![\text{by}]\!] := \lambda x.\lambda y.\text{By}(y, x)$

$[\![\text{above}]\!] := \lambda x.\lambda y.\text{Above}(y, x)$

$[\![\text{behind}]\!] := \lambda x.\lambda y.\text{Behind}(y, x)$

$[\![\text{below}]\!] := \lambda x.\lambda y.\text{Below}(y, x)$

$[\![\text{giggle}]\!] := \lambda x.\text{Giggle}(x)$

$[\![\text{smile}]\!] := \lambda x.\text{Smile}(x)$

$[\![\text{sleep}]\!] := \lambda x.\text{Sleep}(x)$

$[\![\text{swim}]\!] := \lambda x.\text{Swim}(x)$

$[\![\text{wait}]\!] := \lambda x.\text{Wait}(x)$

$[\![\text{move}]\!] := \lambda x.\text{Move}(x)$

$[\![\text{change}]\!] := \lambda x.\text{Change}(x)$

$[\![\text{read}]\!] := \lambda x.\text{Read}(x)$

$[\![\text{eat}]\!] := \lambda x.\text{Eat}(x)$

$[\![\text{entertain}]\!] := \lambda x.\lambda y.\text{Entertain}(y, x)$

$[\![\text{amuse}]\!] := \lambda x.\lambda y.\text{Amuse}(y, x)$

$[\![\text{highfive}]\!] := \lambda x.\lambda y.\text{Highfive}(y, x)$

$[\![\text{applaud}]\!] := \lambda x.\lambda y.\text{Applaud}(y, x)$

$[\![\text{confuse}]\!] := \lambda x.\lambda y.\text{Confuse}(y, x)$

$[\![\text{admire}]\!] := \lambda x.\lambda y.\text{Admire}(y, x)$

$[\![\text{accept}]\!] := \lambda x.\lambda y.\text{Accept}(y, x)$

$[\![\text{remember}]\!] := \lambda x.\lambda y.\text{Remember}(y, x)$

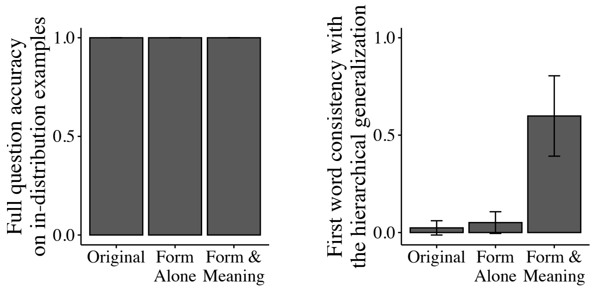
$[\![\text{comfort}]\!] := \lambda x.\lambda y.\text{Comfort}(y, x)$

$[\![\text{does}]\!] := \lambda P.\lambda x.P(x)$

$[\![\text{do}]\!] := \lambda P.\lambda x.P(x)$

$[\![\text{doesn't}]\!] := \lambda P.\lambda x.\neg P(x)$

$[\![\text{don't}]\!] := \lambda P.\lambda x.\neg P(x)$



(a) Test Set

(b) Generalization Set

Figure 9: The plot in (a) shows the full question accuracy on the test set. The plot in (b) shows the first word accuracy on the generalization set. Model results across 10 random reruns. Error bars are single standard deviations.

D Results on McCoy et al.'s (2020) Data

Here we compare the results when using the data in McCoy et al. (2020), which we label ORIGINAL, with the results reported in Experiment 1 and experiment 2. Figure 9 (corresponding to Figure 2 in the main text) compares the results for the data generated for Experiment 1 with the results using the data in McCoy et al. (2020). Overall, models trained on ORIGINAL and FORM ALONE generalized similarly on the in distribution test set and on the generalization set.

Figure 10 (corresponding to Figure 3 in the main text) compares the results on ORIGINAL, on FORM ALONE, and on FORM & MEANING using the evaluation setup reported for Experiment 2. Our results here differ from those reported in Murty et al. (2023a). This difference is due to choice of random seeds. The 10 random seeds chosen in Murty et al. (2023a) display hierarchical grokking (namely, generalization to the hierarchical rule after many training steps), but a few seeds excluded from the results display systematic generalization to the linear rule (see, e.g., seed 222 in `https://github.com/MurtyShikhar/structural-grokking/blob/main/all_test_scores_lm.csv`).

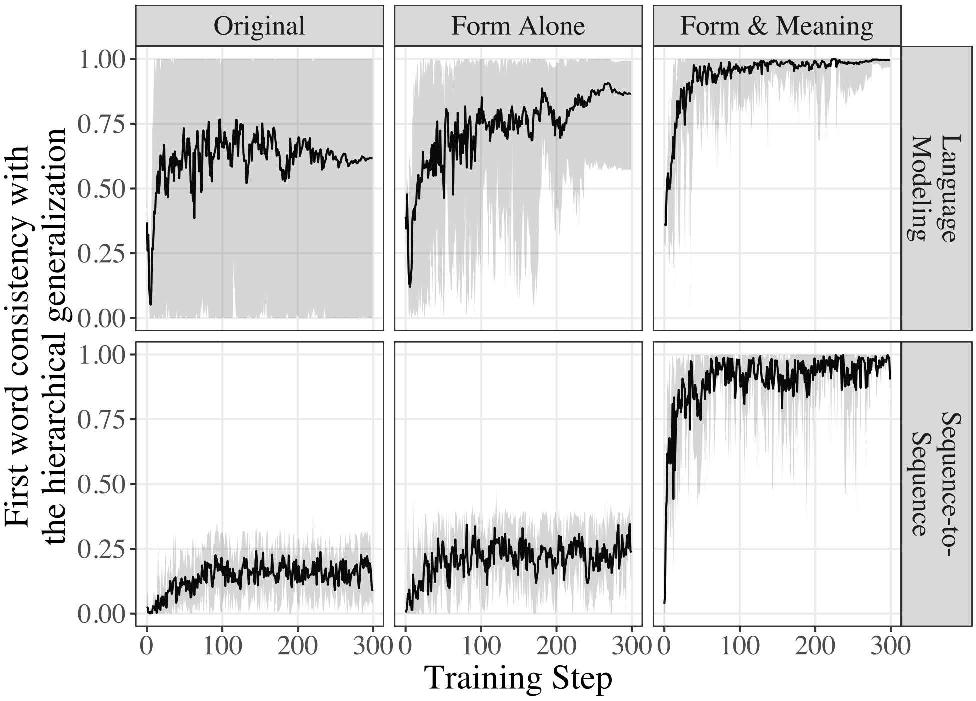


Figure 10: First word consistency with the hierarchical generalization. Top: models trained in a language modeling setup. Bottom: models trained in a sequence to sequence setup. The black line indicates the average across 10 random reruns. Light gray shaded areas are the minimum and maximum values for any model at each training step across the 10 random reruns.