Distinguishing fair from unfair compositional generalization tasks

Ahmad Jabbar **Cleo Condoravdi** **Christopher Potts**
Stanford University
{jabbar,cleoc,cgpotts}@stanford.edu

Abstract

Compositional generalization benchmarks seek to assess whether learning agents can successfully combine familiar concepts in novel ways. COGS (Kim and Linzen, 2020) provides a suite of such tasks in the area of interpretive semantics (mapping sentences to logical forms). A noteworthy finding for COGS is that model performance varies widely across tasks. In this paper, we argue that these performance differences reflect deep properties of these tasks. We focus on two COGS tasks: an easy task (models are generally successful) and a hard task (no present-day models get any traction). Using both experiments and conceptual analysis, we argue that the easy task requires only a single distributional generalization that is well-supported by the training data, whereas the hard task involves a learning target that is ambiguous or even contradicted by the training data. We additionally argue that pretraining can disambiguate the hard task without compromising the goal of testing compositional generalization. Overall, our findings offer practical guidance to designers of compositional generalization benchmarks and also yield new insights into the nature of compositionality itself.

1 Introduction

Humans routinely produce and interpret novel sentences. The principle of compositionality (Montague, 1970; Halvorsen and Ladusaw, 1979; Dowty, 2007) seeks to explain this productivity by hypothesizing that the meanings of complex expressions are determined by the meanings of their parts, recursively down to primitive lexical meanings. Thus, even novel combinations of familiar elements have predictable and stable meanings.

Do language models (LMs) process language in a similarly systematic way? Compositional generalization benchmarks (Lake and Baroni, 2018; Kim and Linzen, 2020; Wu et al., 2021) seek to address this question by creating train–test splits in which specific combinations of elements are held-out for testing. One of the most influential benchmarks in this space is COGS (Kim and Linzen, 2020), which tests models on their ability to map simple English sentences to logical forms (LFs). COGS offers a suite of tasks designed to test different kinds of compositional generalization. At a high-level, the tasks all seem conceptually similar. However, the literature suggests that they must in fact differ from each other in significant ways: some of the tasks are easily solved by present-day models, while others lead to effectively 0 accuracy for all models, as documented by Wu et al. (2021, Table 1). What is the underlying cause of these dramatic differences?

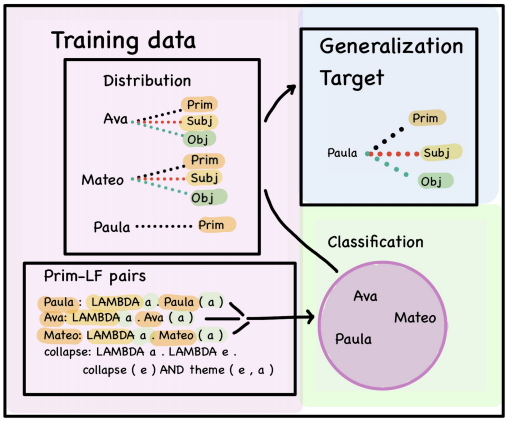

Figure 1: $\text{Gen}_{\text{PROPER}}$: The training data holds out a certain distribution for *Paula*, as depicted by the missing red and green dotted lines for *Paula*. Evaluating on the held-out distribution, as in Generalization Target, leads to failure—unless the lexicon (Prim-LF pairs) contains information that the learner can use for categorizing the distributionally constrained item with other distributionally free items in the training data.

In the present paper, we argue that at least some of these performance differences are the result of fundamental differences in how the tasks themselves are designed. To develop the argument, we focus on two COGS tasks: PRIMITIVE

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 20796–20807
November 4-9, 2025

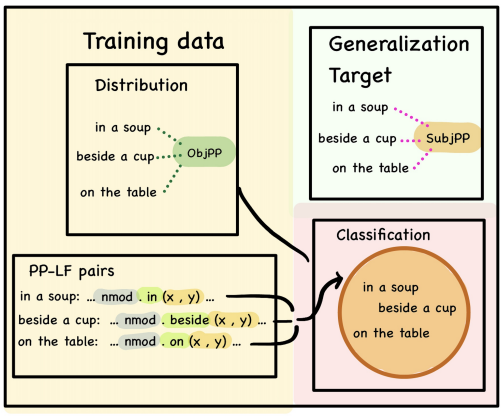

Figure 2: Gen$_{PP}$: The train set holds out the distribution for PPs as modifiers of subjects, depicted by the discrepancy in the training and generalization distribution. PPs can be categorized together by using the information in the lexicon (PP-LF pairs). This is not enough for generalization, for no PP has the distribution which can be used to learn that PPs can occur within subjects.

TO SUBJECT/OBJECT (PROPER NAMES) and OBJECT PP TO SUBJECT PP. In the first (henceforth Gen$_{PROPER}$), a subset of proper names P appear only as lexical primitives during training, while others appear as both subjects and objects. The test-time generalization task is to handle names in P as subjects and objects. This task proves to be straightforward for present-day models that are trained from scratch. In the second (henceforth Gen$_{PP}$), PP modifiers appear only inside direct objects during training. The test-time generalization task is to handle PP modifiers inside subjects. This task is one where all present-day models, trained from scratch, score about 0.

Our core claim is that Gen$_{PROPER}$ is a well-posed generalization task in a way that Gen$_{PP}$ is not. Figures 1 and 2 summarize the key difference. For Gen$_{PROPER}$, the primitive LFs for the held-out names convey to the model that they are members of the broader class of proper names. The train distribution covers all the test-time environments for names, and so the compositional generalization task essentially reduces to the category membership inference, for which LFs for the held-out names serve as evidence. For Gen$_{PP}$, no specific PPs are held out for testing, but the train distribution covers only a subset of the test-time environments for PPs. This is where models fail to generalize in the intended ways.

In our experiments, we reproduce the core findings about performance disparity across tasks. We also substantiate our assessment of the differences with new experiments. In particular, we show that removing the LFs for the held-out names in the Gen$_{PROPER}$ task leads to failures at roughly the same level as for Gen$_{PP}$; the LFs are what support the category membership inference, and without them models cannot make that inferential leap. We also show that the inferential leap is supported by the comprehensive distribution of other proper names, which, if held-out, also independently leads to generalization failure. Our findings also explain why the use of pretrained models results in substantial performance improvement on Gen$_{PP}$, a finding we reproduce for ReCOGS. Pretrained models effectively fill in the distributional gap for PPs, which disambiguates the learning target and provides models with the needed distributional evidence. To further corroborate our assessment, we show that performance on Gen$_{PP}$ can also be improved by minimally introducing the test-time distribution for a subset of PPs (those headed by *beside*), which brings Gen$_{PP}$ on a par with Gen$_{PROPER}$ in terms of the evidence available to the learner.

Kim and Smolensky (2024) (henceforth K&S) is an experimental investigation of a task built in analogy with Gen$_{PP}$. Focusing on human subjects, K&S seek to vindicate Gen$_{PP}$ as a well-posed generalization task for LMs. To create a productive dialectic, we carefully engage with K&S's findings.

2 Background and Related Work

Compositionality. In informal terms. the principle of compositionality says "The meaning of an expression is a function of the meanings of its parts and of the way they are syntactically combined" (Partee, 1984). The principle has been central to linguistic semantics since the work of Montague (1970), who formalized it as a requirement that there exist a homomorphism (structure-preserving mapping) between the syntactic and semantic grammars. While it remains an open question whether the principle can be given in a way that is truly formally restrictive (Zadrozny, 1992, 1994; Kazmi and Pelletier, 1998; Dever, 1999; Dowty, 2007), there is no doubt that it has profoundly shaped linguistic semantics (Partee, 1997, 1984; Kamp and Partee, 1995; Dever, 1999; Werning et al., 2012) as well as the broader study of language and cognition (Fodor et al., 1975; Fodor and Pylyshyn, 1988; Piantadosi et al., 2016; Steinert-Threlkeld, 2020; Nefdt, 2020; Nefdt and Potts, 2024).

Compositional Generalization Benchmarks. Compositional semantic theories are generally expressed as formal semantic grammars. The question naturally arises of what is required to learn such grammars from data. This is the focus of research on compositional generalization. An early exploration of this question is SCAN (Lake and Baroni, 2018), which defines a simple task mapping natural language instructions to action sequences. The train set holds out specific classes of expression for testing, and success or failure is taken as evidence for the model having learned a latent compositional grammar. Since SCAN, a wide variety of such tasks have been proposed to explore more complex language (Geiger et al., 2019, 2020; Kim and Linzen, 2020; Wu et al., 2023; She et al., 2023) and more complex notions of grounding (Ruis et al., 2020; Wu et al., 2021; Lake and Baroni, 2023).

Structural generalization and symbolic models. Kim and Linzen (2020)'s COGS benchmark features subtasks that require generalizing to new structures, as opposed to just new words. It has been noted that the structural subtasks are harder for LMs (Yao and Koller, 2022; Wu et al., 2023). To achieve better performance on structural tasks, researchers have developed symbolic biases within LMs that encourage abstraction and algebraic combination (Liu et al., 2021; Weißenhorn et al., 2022). While such algebraic neural models explore a promising intersection, the hardness of the tasks that they seek to overcome can itself be studied (Yao and Koller, 2022; Wu et al., 2023). Such hard tasks also provide a valuable opportunity to explore normative questions about task fairness.

Assessing Task Fairness. A pressing conceptual question for compositional generalization is whether the tasks they create are *fair* in the sense that the generalization target is unambiguously specified by the training data (Geiger et al., 2019). The distribution of adjectives in English is instructive here. For the most part, if an adjective can appear in a predicational position (*The cat is sleepy*), it can also appear in attributive position (*the sleepy cat*). Thus, one might pose a generalization task in which specific adjectives appear only predicationally during training and in attributive position at test time. Adjectives like *asleep* raise concerns about this. They appear *only* predicationally in English (*the asleep cat* is ill-formed). Thus, a train set containing only predicational uses of an adjective is consistent both with it being allowed in attributive position and with it being restricted to predicational positions. Without additional assumptions, the train set is simply underpecified in this regard. Our central claim (anticipated by Wu et al. 2023) is that some COGS tasks are ambiguous in this sense.

2.1 COGS and ReCOGS

The focus of our work is COGS (Kim and Linzen, 2020). COGS inputs are synthetically generated sentences, and its outputs are logical forms (LFs), inspired by event semantics (Parsons, 1990). The ReCOGS dataset of Wu et al. (2023) seeks to extend and improve COGS by removing confounds in the LFs while maintaining the same evaluation goals. Example (1) illustrates the COGS and ReCOGS formats using a simple example.

(1) a. The lion smiled .
b. COGS LF: `* lion ( x _ 1 ) ; smile . agent ( x _ 2 , x _ 1 )`
c. RECOGS LF: `* lion ( 19 ) ; smile ( 10 ) AND agent ( 10 , 19 )`

Both COGS and ReCOGS provide standard IID train–test splits as well as compositional generalization tasks that, at test time, either place lexical items in novel roles or introduce novel structures. As Wu et al. (2023) document (see their Table 1), performance varies widely across these tasks but is strikingly consistent across different models. For the lexical tasks, present-day models easily achieve about 90% accuracy. For the structural tasks, the same models score at or near 0% accuracy. Wu et al. (2023) show that they can improve scores by removing spurious biases in the COGS data and addressing superficial weaknesses in present-day LMs, but the gap between the lexical and structural tasks remains very large even after these changes. Our overarching goal is to explain this persistent discrepancy. For this, we focus on two representative tasks, one lexical and the other structural. Next, we discuss and analyze the tasks in detail.

3 Generalization Task Analysis

In this section, we analyze our two target tasks, seeking to articulate the core hypotheses that we test experimentally in Section 4.

Gen$_{\text{PROPER}}$. Figure 1 summarizes Gen$_{\text{PROPER}}$. The set of proper names is split into two groups: names in the *distributionally-free* group appear as both subjects and objects in the train set, while

names in the *held-out* group appear only as primitives. The test-time task is to handle names from the held-out group as both subjects and objects.

We hypothesize that this task involves only one compositional generalization step: inferring that the held-out names are in the same category as the distributionally-free ones. The lexical entries for proper names help to convey why this is achievable for models in practice:

(2) a. Ava.
b. COGS LF: `Ava`
c. RECOGS LF: `LAMBDA a . Ava ( a )`

For ReCOGS, all names (and common nouns) have LFs of the form `LAMBDA a .` α `( a )` where α is the name. For COGS, all names are simply single tokens α, and they are the only entries with this format. We know that present-day LMs are highly sensitive to such distributional alignment, and so we hypothesize that models successfully categorize all names together using these LF regularities. In Section 4.2, we support this by showing that models completely fail at the task if we break the distributional alignment by giving held-out names a different LF.

Once a model has made this category inference for the held-out names, the generalization task is essentially an IID one, as no new structural environments are introduced at test time. In Section 4.3, we show that performance drops to near 0 if we modify the task to include a structural generalization component as well (by having proper names appear only as objects during training and testing on subject occurrences).

Gen$_{\text{PP}}$. As Figure 2 indicates, the Gen$_{\text{PP}}$ task is very different from the Gen$_{\text{PROPER}}$ in terms of the generalization challenge it poses. For this task, as in the previous one, models do need to infer that all PPs are distributionally alike. Here, they get evidence from the relevant segments of LFs for sentences featuring PPs.

The much more significant challenge relates to the fact that an entirely new environment for PPs – the subject position – is introduced at test time. In other words, the (Re)COGS training data contains a distributional gap for this task. This gap has no analogue within the Gen$_{\text{PROPER}}$ task. Part of our claim is that the constrained distribution of PPs affects model performance. This can be thought of in terms of syntactic parsing. A learner that doesn't see Subject PPs doesn't know how to parse a sentence with Subject PPs. In Section 4.5, we test this by building on work linking surprisals with syntactic parsing (Hale, 2001; Levy, 2008).

Task	COGS	ReCOGS
Gen$_{\text{PP}}$	0.00	0.01
Gen$_{\text{PROPER}}$	0.98	0.92
IID split	0.99	0.95

Table 1: Baseline experiment results.

4 Experiments

4.1 Baseline Experiments

We begin by reproducing basic results for our main ReCOGS tasks. For these experiments, we use the standard COGS and ReCOGS data splits.

Models We use the encoder–decoder Transformer architecture (Vaswani et al., 2017) from Wu et al. (2023). This architecture has a hidden dimension size of 300, with absolute positional embeddings, two 2-layer Transformer blocks, and 4 attention heads. The model is trained from scratch on COGS and ReCOGS using the HuggingFace `transformers` library. We trained the model for 40 epochs with a learning rate of 3×10^{-4}, using a single NVIDIA A100 GPU. The training time was around 2 and 4 hours for COGS and ReCOGS, respectively.

Results Table 1 summarizes our findings for COGS and ReCOGS, which align with those from the literature: performance on Gen$_{\text{PP}}$ is only slightly higher for ReCOGS compared to COGS. For Gen$_{\text{PROPER}}$, our models perform well on both COGS and ReCOGS. The difference between COGS and ReCOGS here likely traces to the fact that COGS LFs disambiguate between proper names and common nouns (Section 3), aiding the model in classifying all the proper names together. The IID splits show that the tasks themselves are easy where no systematic train–test gaps are involved. The accuracies on Gen$_{\text{PROPER}}$ are comparable to those on the IID splits.

4.2 Proper Name LF Manipulation

In this experiment, we seek to test the hypothesis that the LFs for proper names provide the distributional link for model success at Gen$_{\text{PROPER}}$.

Design As we noted in Section 3, all proper names have identical LF formats/types in both

Sub-Task	Cond 1	Cond 2
PRIM TO SUBJ (PROPER)	0.92	0.03
PRIM TO OBJ (PROPER)	0.81	0.19

Table 2: Proper name LF manipulation results, broken out by Subject and Object. Cond 1 uses the standard proper name LFs for the held-out name. Cond 2 uses the nonstandard LF in (3).

Sub-Task	Subjects Held-Out
PRIM TO SUBJ (PROPER)	0.22
PRIM TO OBJ (PROPER)	0.40

Table 3: Proper name structural manipulation results, broken out by Subject and Object. The manipulation is that we hold out all subject occurrences of names during training. Compare with Cond 1 (the control) in Table 2.

COGS and ReCOGS. For the current experiment, we create two conditions. In Condition 1, *Paula* is associated with its default LFs just like all other proper names. In Condition 2, we associate *Paula* with the following LF, which is structurally like that of an intransitive verb:

(3) `LAMBDA a . LAMBDA e . Paula ( e ) AND theme ( e , a )`

Models We use the same model architecture, hyperparameter settings, and GPUs as in Section 4.1. In Condition 1, the model was trained on the default ReCOGS dataset with the default LF for *Paula*. In Condition 2, the model was trained on the perturbed ReCOGS dataset with the only difference that the LF for *Paula* was set to (3). All other training details were identical.

Results Table 2 summarizes the results, broken out into the subject and object generalization subtasks. We see an 89-point drop in Condition 2 for PRIM TO SUBJ and a 62 percentage point drop for PRIM TO OBJ. This is very strong evidence that the LFs are the distributional link that drives success at this generalization task.

4.3 Proper Name Structural Manipulation (Subject)

In Section 3, we hypothesized that there are two crucial steps to successful generalization. The LFs help with categorization of the distributionally constrained item with the distributionally free ones. The free distribution is used to learn the held-out distribution. We now seek to test this second part of our hypothesis.

Design We modify the standard $\text{Gen}_{\text{PROPER}}$ setup by holding out all subject occurrences of proper names. All proper names, including *Paula*, have their standard LFs.

Models We continue to use the same models and hyperparameter settings as in our previous experiments. However, the distributional change to the train set reduces the train set size. To help address this, we train models for 80 epochs rather than 40.

Results Table 3 reports our findings. For these, we can use the Cond. 1 results from Table 2 as a baseline point of comparison. For this manipulation, PRIM TO SUBJ (PROPER) performance drops from 0.92 to 0.22. For PRIM TO OBJ (PROPER), the decrease is less severe, from 0.81 to 0.40. These findings further support our hypothesis that dismal performance on Gen_{PP} traces to holding out entire distributional environments. As an analogue to the experiment reported here, appendix A contains discussion of the condition with all object occurrences of proper names held out.

4.4 Cosine Similarities for Proper Names

Methods Experiments in Sections 4.1, 4.2, and 4.3 provided us with three different models: (i) the model trained on ReCOGS with the standard LF for the distributionally constrained name *Paula* and the full distribution of other names; (ii) the model trained on ReCOGS with the modified LF for *Paula* and the full distribution of other names; (iii) the model trained on ReCOGS with the standard LF for *Paula* and the constrained distribution of other proper names, achieved by holding out their occurrences as subjects. The three models provide us with the opportunity to test how three different training conditions affect the representation that the LM builds for *Paula*. More specifically, we were interested in finding out how similar this representation was to those for other names in the dataset. To do so, we calculated the mean cosine similarity between *Paula* and all the other proper names.

Results The results for mean cosine similarities for *Paula* with all other proper names, across training conditions, are summarized in Table 4.

4.5 Syntactic Challenges for Held-Out PPs

We turn now to Gen_{PP}. Following Wu et al. (2023), we suspect that many of the challenges of this task

STANDARD LF + FULL	0.068
STANDARD LF + PERTURBED	0.037
PERTURBED LF + FULL	0.009

Table 4: Mean cosine similarities for *Paula* with all other proper names after training the model on different conditions. The similarities drop most in the condition where we associate *Paula* with an LF that differs in form from the LFs for other proper names. This further corroborates our hypothesis that the model learns the similarity between *Paula* and other proper names with the help of the LF for *Paula*.

are syntactic: the train set teaches models that PPs are distributionally restricted to object position and then surprises them with subject cases at test time. The semantic representations across the two conditions are not substantially different, but the parsing task is likely made more challenging by this setting.

Design For human sentence processing, higher surprisal is linked to syntactic parsing difficulties (Hale, 2001; Smith and Levy, 2013), and this hypothesis has frequently been extended to LMs. Our primary question is whether a model trained only on the input ReCOGS strings builds different representations for Object PPs and Subject PPs. For this, we evaluate whether the mean surprisal score assigned to a chosen target site varies between sentences involving PP-modification of subjects and objects. The mean is calculated over 100 sentences, constructed using ChatGPT, for each structure.

We choose and mask the target sites as in (4) and (5). We obtain surprisal scores for the preposition the model considers most likely for (4) and (5) for the mask. Moreover, as we're using a bi-directional encoder model, the context after the [MASK] token is available as well to enable the model to build representations. This accounts for the confound that autoregressive models do not provide as much context to Subject PPs as they do to Object PPs.

(4) Emma ate the cake [MASK] the table.

(5) The cake [MASK] the table burned

Models Our aim here is to use a model that closely resembles the architectures in Kim and Linzen (2020) and Wu et al. (2023). Wu et al. (2023) use an encoder–decoder model, where the encoder is a BERT model with 2 hidden layers. For these reasons, we use TinyBERT (Turc et al., 2019; Bhargava et al., 2021) imported from the HuggingFace library, and randomly initialized. TinyBERT has 4.4M parameters, making it similar in size to the model (4M) used in Wu et al. (2023).

Task	T5-base	Wu et al. (2023)
OBJPP TO SUBJPP	**0.76**	0.00
PRIM TO OBJ (PROPER)	**0.91**	0.50
PRIM TO SUBJ (PROPER)	**0.96**	0.87

Table 5: Pretraining results for selected ReCOGS tasks. The pretrained T5-base model substantially overcomes the challenge posed by both generalization tasks.

As the aim here is to train the re-initialized TinyBERT model on the unsupervised task of predicting the next token in the dataset, we construct our dataset from the input sequences of ReCOGS. This enables the model to build a probability distribution over the input vocabulary of ReCOGS, from which we can then retrieve surprisal scores. The model is trained for a total of 5 epochs on 135,545 examples with a training-rate of 5×10^{-4}.

Results We found a difference in the mean surprisal over 100 strings like (4) that instantiate the ObjPP structure (surprisal: 0.89) and the average surprisal over 100 strings like (5) that instantiate the SubjPP structure (surprisal: 5.35). The surprisal difference here—as a signal for syntactic challenges for the held-out structure—is in tension with a result from Yao and Koller (2022), who use probes to argue that the encoder learns the held-out structure. They conclude that it is the decoder that fails to use the encoded syntactic representation.

4.6 Pretraining Experiments

The previous section suggests that much of the challenge for Gen_{PP} lies in dealing with unexpected syntactic configurations. This suggests that using pretrained models will boost performance, as it is a safe bet that such models have seen sentences in which PPs are in both subject and object position (as well as many other positions).[1] Our prediction is that such models will do substantially better at the structural tasks. Assuming the train set for the model does not contain the COGS LFs, such pretraining doesn't compromise the COGS generalization splits. This assumption is a safe one, as the T5 paper (Raffel et al., 2020) was submitted to arXiv in October 2019, while the COGS (Kim and Linzen, 2020) submission date is in October 2020.

[1]While Yao and Koller (2022) use pretrained models on COGS, we run our pretrained experiments on ReCOGS.

Task	Cond 1	Cond 2
OBJ PP TO SUBJ PP	0.04	0.87

Table 6: Introducing some distributional evidence increases accuracy on Gen_{PP}. Cond 1 is the control condition. In Cond 2, we introduce subject PPs headed by the preposition *beside*.

Methods We imported T5-base (220M; Raffel et al. 2020) from the HuggingFace `transformers` library and kept its pretrained weights. We fine-tuned T5-base on the ReCOGS training set for a total of 4 epochs at a learning rate of 3×10^{-4}.

Results Table 5 compares our T5-base pretrained results with those of Wu et al. (2023). As predicted, pretraining helps substantially for all our tasks. What is most noteworthy for our purposes is a drastic 76 percentage point improvement in performance on OBJ PP TO SUBJ PP. This is in line with our predictions and suggests that the primary challenge posed by Gen_{PP} is syntactic rather than semantic. This is consistent with the conclusions of Wu et al. (2023) that the primary obstacles concern form-based issues rather than interpretive ones.

Next, we try to bring Gen_{PP} on a par with $\text{Gen}_{\text{PROPER}}$ in terms of the available evidence supporting the generalization target.

4.7 Minimally introducing subject PPs

Design We minimally introduce subject PPs into the training data in our next data perturbation. This introduction is minimal in that we only introduce subject PPs where the PP is headed by the preposition *beside*. Other PPs headed by *in* and *on* are still distributionally constrained to object modification position in the training set.

Methods We kept the same hyperparameter settings, models, and GPUs as in the experiments reported in Sections 4.2 and 4.3.

Results Table 6 reports our findings. The model performance on Gen_{PP} increases significantly by showing the model a little more distributional evidence. This evidence is still much more constrained than in $\text{Gen}_{\text{PROPER}}$, as in $\text{Gen}_{\text{PROPER}}$, the model sees multiple names which share their LF forms with *Paula* in a wide distribution.

The sharp increase in accuracy on Gen_{PP} shows that the path to generalization success that we had outlined for $\text{Gen}_{\text{PROPER}}$ is not peculiar to $\text{Gen}_{\text{PROPER}}$. Instead, the same distributional evidence can be leveraged for success on Gen_{PP}. If this information is so crucial across generalizations, then holding it out is manifestly unfair.

Through our analysis and results, we have established an understanding of generalization as inference over distributional evidence. We find that models use particular clues and distribution in the training data to reach the generalization target. A task which by its design holds out these clues and distributional evidence is not well-posed.

5 Discussion

In this section, in light of our analysis and results, we try to answer the following questions: what makes a structural generalization task well-posed, why is pretrained model use important for a better methodology, and can we reconcile our findings for LMs with the human-subjects experiments of Kim and Smolensky (2024)?

5.1 Well-Posed Structural Generalization Tasks

In Section 3, we argued that $\text{Gen}_{\text{PROPER}}$ essentially reduces to a well-studied form of distributional learning: the learner uses lexical regularities to place held-out names into the same category as other names, and it uses compositional regularities to infer from the training data that all names can appear as subjects and objects. These distributional cues seem to determine outcomes; breaking the lexical cues is catastrophic (Section 4.2), as is holding out a distribution (Section 4.3).

For the structural generalization task Gen_{PP}, the situation is quite different. The learner is placed in a training environment in which PPs appear inside objects but not subjects and then confronted with test-time situations that are different in this regard. Distributional learning as usual will likely push the learner to infer that PPs are restricted to object position. Holding out proper names from subject position in that task leads to the same negative outcome (Section 4.3). Moreover, introducing a little distributional evidence for the target results in a sharp increase in accuracy (Section 4.7).

Structural generalization tasks nonetheless seem important to include in compositional generalization benchmarks, and so we should ask: under what circumstances can we fairly hold out entire structures from the training set? While we cannot give a complete answer to this question, our results and

analyses do suggest two productive steps.

First, we advocate for allowing researchers to introduce specific structural learning biases into their experimental procedures. For instance, in the case of Gen_{PP}, it may suffice to tell the model, via some mechanism, that phrases in subject and object position are NPs rather than falling into subcategories like NP_{SUBJ} and NP_{OBJ}. With this bias introduced, we essentially push the model towards the intended generalization target and away from others that are equally supported by the training data. Examples of such systems include Liu et al. (2021)'s LeAR and Weißenhorn et al. (2022)'s AM parser. Such work complements our efforts in this paper to determine which tasks require such learning biases for fairness and which do not.

Second, we advocate for making use of pretrained models in compositional generalization tasks. The community might seek to create models that are exposed to realistic samples of the target language, as was done in the BabyLM challenge (Warstadt et al., 2023). The goal here is to create a realistic simulated learning environment.

We remain cognizant of Kim et al. (2022)'s concerns about using pretrained models on COGS; pretraining potentially exposes the held-out distribution of lexical items. However, as long as we ensure that the pretraining data does not contain (in)direct supervision about the task of mapping sentences to LFs, we feel that the tasks are not compromised; and indeed pretraining can help us isolate semantics as separate from syntax (Section 4.5). In addition, we do see value in training models from scratch on these tasks. Our Proper Name LF experiments (Section 4.2) and Structural Manipulation experiments (Section 4.3) involved training models from scratch, and these experiments helped us to identify the key properties of the task itself.

In light of the above discussion, we propose that it would be productive for benchmark designers to outline routes to solving generalization tasks in advance and make them available to users. Generalization is then the ability to figure out these routes using distributional clues in the training data. If there aren't any routes available, or if different routes lead to different outcomes, then the generalization task is not well-posed.

One might reply that humans possess the ability to generalize systematically and consistently in absence of distributional cues supporting the existence of the held-out structure. Therefore, it is fair to evaluate models on such human-like generalization. Kim and Smolensky (2024) make exactly this argument, with which we engage below.

5.2 Kim and Smolensky (2024)

Kim and Smolensky (2024) (henceforth K&S) is an experimental investigation of the extent to which human participants can solve COGS-like tasks. K&S seek to create experimental situations in which people are given the same kinds of distributional evidence as is available to COGS models trained from scratch. If humans respond systematically in such scenarios, then we can fairly expect models to be able to do the same.

K&S report results of two experiments. Both experiments have a training phase and a testing phase. During training, participants are shown videos depicting scenes along with sentences that describe the scenes. The sentences are constructed from the vocabulary of a nonce VSO language that has five nouns (N), one intransitive verb, one transitive verb (V), and crucially a postnominal adjective (A). The VSO order and postnominal adjectival modification are intended to make the nonce language different from English—the native language of the participants. The rationale behind such differentiation is to preclude participants from extending patterns from English during testing.

During the training phase of Experiment 1, participants saw scenes that are accompanied by sentences of the form V N N-A involving adjectival modification of the object. Successful completion of the test-time task required participants to produce sentences of the order V N-A N. The train-test distributional gap is supposed to be an analogue for Gen_{PP}, where a new structure, PP-modification of the subject, is introduced at test time.

Experiment 2 is motivated by two considerations: the possibility that the participants may be generalizing by extending the subject modification pattern from English in Experiment 1; and the fact that, in English, "a resultative phrase may be predicated of the immediately postverbal NP, but may not be predicated of a subject . . . complement" (Levin and Hovav, 1994, 34). If the production of a resultative construction is elicited and the participants produce a V N-A N sentence of the VSO language, this can be taken to suggest that the production is not an extension of participants' knowledge of English, but an instance of genuine structural generalization. K&S elicit the resultative construction via scenes depicting one shape hitting another such that the hitter gets cracked upon contact.

In both Experiments 1 and 2, the production of V N-A N sentences as the description of test-time scenes is significantly above chance. K&S take this to be evidence for the human bias for generalization to a target in the presence of the same amount of distributional evidence as is available for Gen_{PP} to the COGS models.

First, it is important to note that human participants come to these experiments in a highly pretrained state. Along with equipping the learner with knowledge that can be extended in a wide variety of ways in novel settings, pretraining on natural language data also equips the learner with a rapid in-context learning ability (Chan et al., 2022; Saffran and Kirkham, 2018). Without pretraining, not only do we rid the LM of a vast array of extendable knowledge, but also of this rapid learning ability. These are serious handicaps when comparing LM performance with that of adult human participants.

Second, in taking the production of N-A in the context of a particular scene to unambiguously have a resultative semantics, K&S make an assumption that requires further support. It is hard to pin down what the participants in Experiment 2 take their productions to express semantically, let alone categorically establish that the participants intend to produce the analogue for the English resultative. The production pattern V N-A N can very well correspond to *The circle that broke hit the triangle* or to *The broken circle hit the triangle*. These possibilities highlight that the nonce language does not have the expressive power such that we can reliably associate the V N-A N pattern with a grammatical translation of a resultative construction in English. Therefore, it is not clear that English biases the participants against the V N-A N production pattern for test time scenes in Experiment 2. Successful completion of the test time task in K&S's Experiment 2 cannot be taken to vindicate Gen_{PP}.

In sum, K&S's findings, although interesting, do not establish the claim that Gen_{PP} is a well-posed task for LMs. In fact, we feel that K&S's investigation further reinforces our view: compositional generalization tasks are made more realistic by the use of pretrained models, since such models are likely to embed realistic and useful biases critical for solving such tasks.

6 Conclusion

We focused on two tasks in COGS, $\text{Gen}_{\text{PROPER}}$ and Gen_{PP}. We used our experiments to reveal the distributional cues and gaps that help and preclude generalization for $\text{Gen}_{\text{PROPER}}$ and Gen_{PP}, respectively. Any compositional generalization task requires enough distributional evidence to reach the target generalization. In the absence of such evidence, the task is ill-posed; this, we take to be true for Gen_{PP}. Using our conceptual analysis and results, we offered the methodological proposal of using pretrained models when comparing neural models with humans. A fair comparative study of compositional generalization should lead to a deeper understanding of compositionality and generalization separately as well.

7 Limitations

For our experiments and analysis, we used COGS and ReCOGS datasets. Both COGS and ReCOGS, while highly useful, feature idealized sentences of the English language. Further, the LFs associated with the sentences are also quite complicated, potentially requiring the models to perform well on auxiliary tasks, in addition to semantic parsing of sentences. Future work can aspire to build COGS-like splits for languages other than English, while also building in Wu et al. (2023)'s vein to simplify the LFs. Moreover, while we engage with Kim and Smolensky (2024)'s findings and note their limitations, it is true that we don't yet have a good understanding of what the extent of human-like generalization really is when it comes to structural tasks like Gen_{PP}. This can be attributed to the difficulty of creating experimental conditions where human participants' knowledge w.r.t. the generalization target is made to mimic the knowledge of models trained from scratch. Therefore, it is too early to draw the conclusion that models generalize like humans or that they don't. Further, we think that there is a dearth of literature that explores the normative properties, such as fairness of tasks or legitimacy of the conclusions we draw from model performance. Our paper uses notions like fairness that will eventually get more precise as our methodologies around comparing human cognition with neural models mature. All of these limitations signal great promise for future inquiry.

Acknowledgments

Special thanks to Thomas Icard for his input on this project! We're grateful to three anonymous reviewers for their helpful thoughts and time! Feedback from the following people has also

helped this project immensely: Rhea Kapur, Ayush Chakravarthy, Zhengxuan Wu, Shikhar Murty, Beth Levin, Nathan Roll, Jasper Jian, Susanne Riehemann, Eve Clark, Dan Yamins, and Cory Shain. This research is supported in part by grants from Google and Open Philanthropy.

References

Prajjwal Bhargava, Aleksandr Drozd, and Anna Rogers. 2021. Generalization in nli: Ways (not) to go beyond simple heuristics. *Preprint*, arXiv:2110.01518.

Stephanie Chan, Adam Santoro, Andrew Lampinen, Jane Wang, Aaditya Singh, Pierre Richemond, James McClelland, and Felix Hill. 2022. Data distributional properties drive emergent in-context learning in transformers. *Advances in Neural Information Processing Systems*, 35:18878–18891.

Josh Dever. 1999. Compositionality as methodology. *Linguistics and Philosophy*, 22(3):311–326.

David Dowty. 2007. Compositionality as an empirical problem. In Chris Barker and Pauline Jacobson, editors, *Direct Compositionality*, pages 23–101. Oxford University Press, Oxford.

Jerry A. Fodor and Zenon W. Pylyshyn. 1988. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1):3–71.

Jerry A Fodor et al. 1975. *The language of thought*, volume 5. Harvard university press Cambridge, MA.

Atticus Geiger, Ignacio Cases, Lauri Karttunen, and Christopher Potts. 2019. Posing fair generalization tasks for natural language inference. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4475–4485, Stroudsburg, PA. Association for Computational Linguistics.

Atticus Geiger, Kyle Richardson, and Christopher Potts. 2020. Neural natural language inference models partially embed theories of lexical entailment and negation. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online. Association for Computational Linguistics.

John Hale. 2001. A probabilistic earley parser as a psycholinguistic model. In *Second meeting of the north american chapter of the association for computational linguistics*.

Per-Kristian Halvorsen and William A Ladusaw. 1979. Montague's' universal grammar': An introduction for the linguist. *Linguistics and Philosophy*, pages 185–223.

Hans Kamp and Barbara H. Partee. 1995. Prototype theory and compositionality. *Cognition*, 57(2):129–191.

Ali Kazmi and Francis Jeffry Pelletier. 1998. Is compositionality formally vacuous? *Linguistics and Philosophy*, 21(6):629–633.

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.

Najoung Kim, Tal Linzen, and Paul Smolensky. 2022. Uncontrolled lexical exposure leads to overestimation of compositional generalization in pretrained models. *arXiv preprint arXiv:2212.10769*.

Najoung Kim and Paul Smolensky. 2024. Structural generalization of modification in adult learners of an artificial language. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 46.

Brenden Lake and Marco Baroni. 2018. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International conference on machine learning*, pages 2873–2882. PMLR.

Brenden M Lake and Marco Baroni. 2023. Human-like systematic generalization through a meta-learning neural network. *Nature*, 623(7985):115–121.

Beth Levin and Malka Rappaport Hovav. 1994. *Unaccusativity: At the syntax-lexical semantics interface*, volume 26. MIT press.

Roger Levy. 2008. Expectation-based syntactic comprehension. *Cognition*, 106(3):1126–1177.

Chenyao Liu, Shengnan An, Zeqi Lin, Qian Liu, Bei Chen, Jian-Guang Lou, Lijie Wen, Nanning Zheng, and Dongmei Zhang. 2021. Learning algebraic recombination for compositional generalization. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*, pages 1129–44.

Richard Montague. 1970. Universal grammar. *Theoria*, 36:373–98.

Ryan M. Nefdt. 2020. A puzzle concerning compositionality in machines. *Minds and Machines*, 30(1):47–75.

Ryan M. Nefdt and Christopher Potts. 2024. Compositionality. In Michael C. Frank and Asifa Majid, editors, *Open Encyclopedia of Cognitive Science*. MIT Press.

Terence Parsons. 1990. *Events in the semantics of English: A study in subatomic semantics*. MIT Press.

Barbara H. Partee. 1984. Compositionality. In Fred Landman and Frank Veltman, editors, *Varieties of Formal Semantics*, pages 281–311. Foris, Dordrecht.

Barbara H Partee. 1997. Compositionality. In *Handbook of logic and language*, pages 417–473. Elsevier.

Steven T Piantadosi, Joshua B Tenenbaum, and Noah D Goodman. 2016. The logical primitives of thought: Empirical foundations for compositional cognitive models. *Psychological review*, 123(4):392.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Laura Ruis, Jacob Andreas, Marco Baroni, Diane Bouchacourt, and Brenden M Lake. 2020. A benchmark for systematic generalization in grounded language understanding. *Advances in neural information processing systems*, 33:19861–19872.

Jenny R Saffran and Natasha Z Kirkham. 2018. Infant statistical learning. *Annual review of psychology*, 69(1):181–203.

Jingyuan S. She, Christopher Potts, Samuel R. Bowman, and Atticus Geiger. 2023. ScoNe: Benchmarking negation reasoning in language models with fine-tuning and in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1803–1821, Toronto, Canada. Association for Computational Linguistics.

Nathaniel J Smith and Roger Levy. 2013. The effect of word predictability on reading time is logarithmic. *Cognition*, 128(3):302–319.

Shane Steinert-Threlkeld. 2020. Toward the emergence of nontrivial compositionality. *Philosophy of Science*, 87(5):897–909.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: The impact of student initialization on knowledge distillation. *CoRR*, abs/1908.08962.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjabe, Adina Williams, Tal Linzen, and Ryan Cotterell, editors. 2023. *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*. Association for Computational Linguistics, Singapore.

Pia Weißenhorn, Lucia Donatelli, and Alexander Koller. 2022. Compositional generalization with a broad-coverage semantic parser. In *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, pages 44–54.

Markus Werning, Wolfram Hinzen, and Edouard Machery. 2012. *The Oxford Handbook of Compositionality*. Oxford University Press, Oxford.

Zhengxuan Wu, Elisa Kreiss, Desmond C Ong, and Christopher Potts. 2021. ReaSCAN: Compositional reasoning in language grounding. *arXiv preprint arXiv:2109.08994*.

Zhengxuan Wu, Christopher D Manning, and Christopher Potts. 2023. ReCOGS: How incidental details of a logical form overshadow an evaluation of semantic interpretation. *Transactions of the Association for Computational Linguistics*, 11:1719–1733.

Yuekun Yao and Alexander Koller. 2022. Structural generalization is hard for sequence-to-sequence models. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*, pages 5048–62.

Wlodek Zadrozny. 1992. On compositional semantics. In *Proceedings of COLING 1992*, pages 260–266, Nantes.

Wlodek Zadrozny. 1994. From compositional to systematic semantics. *Linguistics and Philosophy*, 17(4):329–342.

Appendix

A Proper Name Structural Manipulation (Object)

Design We hold out all object occurrences of proper names during training. All proper names, including *Paula*, have their standard LFs.

Models We use the same models and hyperparamater settings as in the experiments in Section 4.

Results The consideration that guided this manipulation was to ensure that it was indeed the holding out of subject occurrences of names in Experiment 4.3, and not just any chunk of the training data, that led to the drop in accuracies as recorded in Table 3. Then, holding out object occurrences of names should not affect accuracies on PRIM TO SUBJ (PROPER). This is indeed what we see, as reported in Table 7.

Sub-Task	Objects Held-Out
PRIM TO SUBJ (PROPER)	0.85
PRIM TO OBJ (PROPER)	0.62

Table 7: Proper name structural manipulation results, broken out by Subject and Object. The manipulation is that we hold out all object occurrences of proper names for testing. These results can be compared with those in Table 2 and Table 3.

Comment It is worth noting that the drop in accuracy on PRIM TO OBJ (PROPER) is higher when we hold out subjects as reported in Table 3, compared to when we hold out objects as recorded in Table 7. Currently, we don't have an explanation for why subjects are more crucial for the more general inference about the free distribution of the held-out name.

License of use. We abide by the licenses of use for both COGS (Kim and Linzen, 2020) and ReCOGS (Wu et al., 2023). We use the models used in Wu et al. (2023), and use and modify the datasets in Kim and Linzen (2020) and Wu et al. (2023), which is in line with the permissions granted in the licenses of use.