Flexible Generation from Fragmentary Linguistic Input

Peng Qian **Roger P. Levy**
Department of Brain and Cognitive Sciences
Massachusetts Institute of Technology
{pqian,rplevy}@mit.edu

Abstract

The dominant paradigm for high-performance models in novel NLP tasks today is direct specialization for the task via training from scratch or fine-tuning large pre-trained models. But does direct specialization capture how humans approach novel language tasks? We hypothesize that human performance is better characterized by flexible inference through composition of basic computational motifs available to the human language user. To test this hypothesis, we formulate a set of novel *fragmentary text completion* tasks, and compare the behavior of three direct-specialization models against a new model we introduce, GibbsComplete, which composes two basic computational motifs central to contemporary models: masked and autoregressive word prediction. We conduct three types of evaluation: human judgments of completion quality, satisfaction of syntactic constraints imposed by the input fragment, and similarity to human behavior in the structural statistics of the completions. With no task-specific parameter tuning, GibbsComplete performs comparably to direct-specialization models in the first two evaluations, and outperforms all direct-specialization models in the third evaluation. These results support our hypothesis that human behavior in novel language tasks and environments may be better characterized by flexible composition of basic computational motifs rather than by direct specialization.

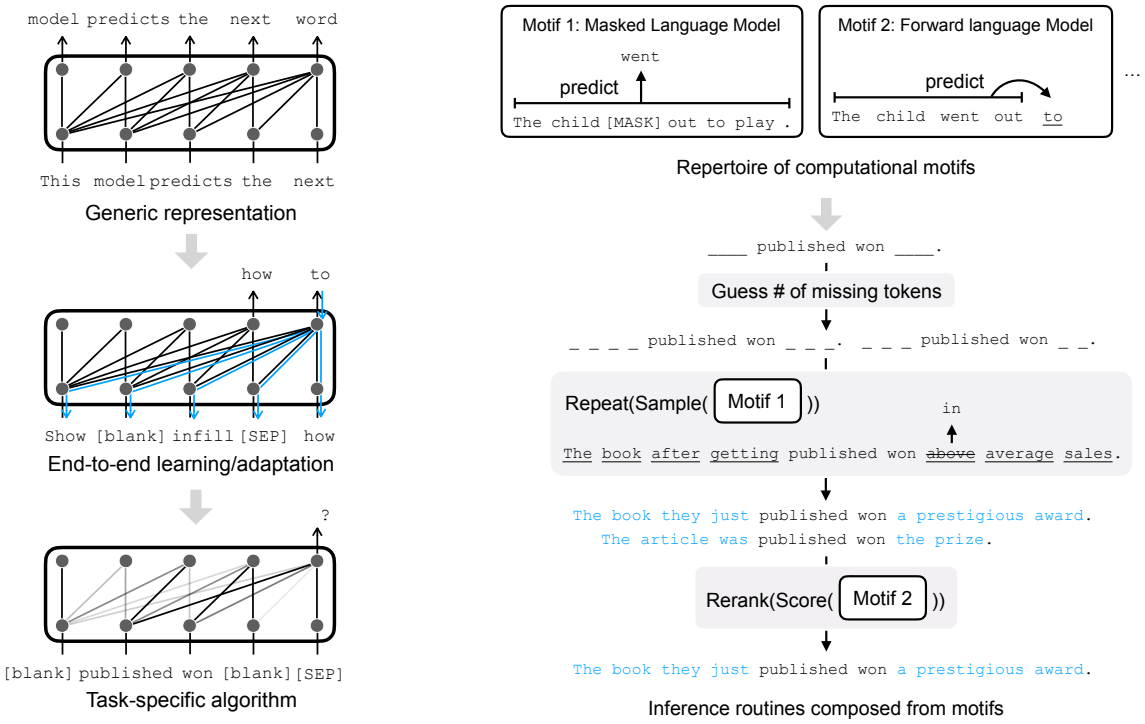
1 Introduction

Representation learning has tremendously benefited engineering for language comprehension and generation systems. General frameworks such as encoder-decoder or auto-regressive model can be flexibly applied to a diverse range of problems. With scaled models, enormous data, and well-chosen training objective functions, generative pre-training extracts useful and transferable information from unlabeled text data (Howard and Ruder, 2018; Liu et al., 2019b). These generic representations can then be finetuned to quickly yield performant models directly specialized for downstream tasks (Howard and Ruder, 2018; Radford et al., 2019; Devlin et al., 2019, *inter alia*). While the broad idea of exploiting rich statistical information in general-purpose representations has proven practically effective for deep learning models since a decade ago (Erhan et al., 2010; Collobert et al., 2011), it remains an open question how well this "direct specialization" approach yields models that behave similarly to humans in flexible linguistic behavior in novel situations.

Here we take this "direct specialization" approach as a scientific hypothesis regarding the nature of linguistic knowledge and its flexible deployment in the human mind: that the human capacity of processing linguistic information in novel tasks arises from directly specializing and reshaping a generic yet versatile representation. We contrast it with an alternative hypothesis: that flexible knowledge deployment reflects algorithmic composition of a constrained repertoire of simple, reusable inference motifs. We depict these competing hypotheses in Figure 1. In the "compositional inference" hypothesis, the basic motifs arise from learning processes and could potentially involve distinct internal representations, but can be recombined into new inference routines guided by the principles of approximate probabilistic inference. In this hypothesis, the computational-level specifications (Marr, 1982) of motif functional forms play a crucial role. As a theory of how the human mind approaches new and potentially complex problems, the "direct specialization" hypothesis entails generic starting representations, task-specific supervision, and specialization. The "compositional inference" hypothesis, in contrast, entails novel combinations of solutions to old problems.

Directly testing these two hypotheses would require a comprehensive set of experimental stud-

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 8176 - 8196



(a) Direct specialization

(b) Compositional inference

Figure 1: Sketches of (a) "direct specialization" and (b) "compositional inference" paradigms. Blue arrows in the left panel (a) represent the gradient flow of back-propagation.

ies of human behaviors involving various tasks, which goes beyond the scope of this work. However, as an initial step, we consider these two hypotheses as computational paradigms that inform and inspire the architecture of models that flexibly perform novel language tasks, and ask which paradigm gives rise to more human-like behaviors. Here we ground the general question in a specific setting: a novel set of fragmentary input completion challenges, inspired by the classic cloze task (Ebbinghaus, 1897) and its contemporary variants. Taking a reverse-engineering approach, we instantiate the "direct specialization" and "compositional inference" hypotheses as explicit computational models and compared the models' behaviors to that of humans in a behavioral task of fragmentary input completion. Our fragmentary input designs highlight various aspects of abstract reasoning involving subtle features of grammar and semantics. We find that the model of the compositional inference approach generates high-quality completions without direct training on the target task, achieves comparable performance to the models from the direct specialization approach in reasoning about constrained syntactic contexts, and better matches the fine-grained structural statistics of completions written by human subjects.

2 The Fragment Completion Challenge

For purposes of this paper, we define a *fragmentary linguistic input*, or simply a *fragment*, as a sequence of word strings. *Completing* a fragment involves adding a word string of any length between adjacent strings in the input to yield a single overall well-formed sentence. The fragment completion problem is formally equivalent to the text infilling problem studied in a number of recent papers (Fedus et al., 2018; Zhu et al., 2019; Liu et al., 2019a; Donahue et al., 2020; Shen et al., 2020; Huang et al., 2020), but here we study considerably more open-ended completion problems from potentially much briefer input than has been studied before. For example, many native English speakers find the simple fragmentary input

_____ published won _____ .

initially challenging yet solvable.[1] We find that carefully chosen simple fragmentary inputs can

[1]The input requires building a nested center-embedding context such as "*The most recent book she* published *won a prize*".

offer insight into the abilities of "direct specialization" and "compositional inference" models, and the similarity of model and human behavior.

2.1 Computational Models

Formally, fragment completion involves generating, given input comprised of a sequence of k word strings $C = \{C_1, \ldots, C_k\}$, a sequence of $k-1$ word strings $B = \{B_1, \ldots, B_{k-1}\}$, such that the resulting completion is $C_1 \circ B_1 \circ \cdots \circ C_{k-1} \circ B_{k-1} \circ C_k$.[2] In general, exact inference over the full conditional distribution $P(B|C)$ will be intractable. The "direct specialization" and "compositional inference" paradigms offer differing model specifications and algorithmic options.

2.1.1 Direct Specialization

With direct specialization, we learn or fine-tune representations under supervision for the task. We take the learning objective to be maximizing the likelihood $p(B|C)$ for some sampled (B, C) pairs generated from a training corpus (see Section 2.2):

$$p(B|C) = \prod_{i=1}^{|B|} p(B_i \mid B_{<i}, C)$$

We ground this approach into two learning procedures: (a) fine-tuning pretrained language model to solve the target task, and (b) training on the target task from scratch. The fine-tuning procedure takes advantage of transferring knowledge from large pretrained models, while the training from scratch procedure allows us to further control the effect of pretrained representation. For both learning procedures, we use three existing models trained or fine-tuned for infilling: T5 (Raffel et al., 2019; InfillT5), BART (Lewis et al., 2020; InfillBART), and GPT-2 (Radford et al., 2019) fine-tuned for text infilling, which was previously explored in Donahue et al. (2020) and which we will follow them in calling the Infilling Language Model (ILM) for short. Implementation details can be found in Section 2.2.

2.1.2 Compositional Inference Approach

For the compositional inference approach, we propose **GibbsComplete**, a neurally-guided approximate inference algorithm that combines two canonical computational motifs: masked word prediction and autoregressive word prediction. Consider a candidate for the i-th completion string B_i to consist of $b_{i,<j} \circ b_{i,j} \circ b_{i,j<}$. GibbsComplete takes a masked language modelling motif as a proposal distribution $p(b_{i,j}|B_{\setminus i}, b_{i,<j}, b_{i,j<}, C)$ and composes it with a global scoring function $\phi(B, C)$ given by a unidirectional language modelling motif, in line with previous work on applying sampling-based methods to generative neural sequence models (Berglund et al., 2015; Su et al., 2018; Miao et al., 2019; Wang and Cho, 2019; He and Li, 2021). GibbsComplete can also be broadly viewed as an example of unsupervised language generation, among many other alternatives (Liu et al., 2019a; Qin et al., 2020; West et al., 2021).

Algorithm 1: GibbsComplete

```
Data: An incomplete sentence with blanks.
Result: M completions
initialize N; // Number of candidates
initialize T; // Number of iterations
completions ← [];
for n = 0; n < N; n = n + 1 do
    propose a blank configuration;
    for i = 0; i < T; i = i + 1 do
        randomly choose a token in the blanks;
        sample a replacement from MASKED
          LANGUAGE MODEL;
    add the final sample to completions;
rerank N completions to top-M based on average
  word surprisal estimated by FORWARD LANGUAGE
  MODEL;
```

Our GibbsComplete algorithm first proposes a random uniform guess on $[1, 10]$ about the length of each $\{B_i\}$, and initializes them as "`[MASK]`" sequences of the guessed lengths. It then proposes an edit to a randomly chosen position $b_{i,j}$ within the blanks by sampling from the sorted list of likely replacements according to the conditional probability $p(b_{i,j}|B_{\setminus i}, b_{i,<j}, b_{i,j<}, C)$ given by a masked language model. We take 500 stochastic editing steps: for the first 250 steps as a burn-in period, the replacement is sampled from the top 50 likely words; for the remaining iterations, the most likely word is picked as the replacement, following the common practice of annealing temperature for better generation quality (Wang and Cho, 2019). The final output of the editing process is a candidate completion. We sample 1000 such candidates, and then rerank them with an autoregressive forward language model, using mean per-word conditional log-probability as a scoring function to promote fluency.

Note that neither the masked language model nor the autoregressive language model is fine-tuned

[2] We represent cases where material can be added at the beginning or the end of the input as c_1 or c_k being the empty string, ϵ.

or retrained on the target infilling task: we take them as basic computational motifs that are immediately available to a language-using agent for use in sampling-based probabilistic inference to facilitate novel behaviors. Completing fragmentary inputs of the type studied here is not a major form of everyday language use, yet as our experiments show, native speakers can perform even challenging fragment completions fairly well with little practice.

Like the two learning settings considered in the "direct specialization" approach, we implement instances of GibbsComplete algorithm with (a) pretrained language models as the computational motifs, and (b) computational motifs of the same architecture as the pretrained counterparts but learned from the same corpus as those trained-from-scratch models from the "direct optimization" approach.

2.2 Model Implementations and Training Datasets

We implement instances of the models under two learning settings: (a) transferring knowledge from pretrained models, and (b) training from scratch. Implementations are based on the Huggingface `transformers` package (Wolf et al., 2020).

In the case of knowledge transfer with pretrained models, no fine-tuning or learning is needed for GibbsComplete. We simply use the small version of pretrained GPT-2 (Radford et al., 2019) and the base cased version of pretrained BERT (Devlin et al., 2019) as the corresponding computational motifs. For direct specialization models, we fine-tune pretrained GPT-2 small, T5 base, and BART large[3] to get ILM, InfillT5 and InfillBART respectively. The total number of parameters of these model architectures is listed in Table 1. All models were fine-tuned on a 10 million token subset of New York Times Corpus 2007 portion (Sandhaus, 2008), with a batch size of 32 and learning rate of 10^{-5}. The supervision signal is generated by randomly cropping some spans of words in a sentence to get the fragmentary context C and a plausible completion B (see Appendix C.2 for details). We stopped fine-tuning when the validation loss increases for two epochs in a row. To generate completions from ILM, InfillT5, and InfillBART, we apply ancestral sampling from a list of top 50 mostly likely tokens at each time step. In experimenting with the fine-tuned models, we noticed lower diversity in InfillBART samples compared to the other models. Hence we set the sampling temperature as 1 for other models but 1.8 for the fine-tuned InfillBART to ensure that all models generate a good variety of completions.

When training from scratch, we train all components to be learned in all models on two separate 42-million-token datasets: (1) part of the 2006 year portion of New York Times Corpus (Sandhaus, 2008; `NYT`), and (2) part of the `BLLIP` corpus (Charniak et al., 2000) previously prepared by Hu et al. (2020). For GibbsComplete, we train an auto-regressive Transformer decoder language model of the same size as pretrained GPT-2 small and a masked language model of the same size as pretrained base cased BERT. For ILM, InfillT5, and InfillBART, we initialize the same architecture and tokenizer as their fine-tuned counterparts and train each model from scratch. We use a batch size of 16 for the masked language model in GibbsComplete and 32 for all the other models. The learning rate is set as 10^{-5} across all the models. Training is early stopped if either the validation loss increases for two epochs in a row or the total number of training epochs exceeds 100.

2.3 Collecting Human Completions

We also collect human completions of our fragments on Mechanical Turk, to evaluate performance and for fine-grained comparison of human and model behavior. The visual layout of these experiments is shown in Appendix A. Participants were instructed to use as many or little words as they see appropriate to fill in the blanks in the fragmentary input, so that each completed sentence is coherent, grammatical, and meaningful. The interface required each blank to be filled in with at least one word. We imposed no time constraints.

3 Experiments

To address the question of whether the "direct specialization" or "compositional inference" paradigm gives rise to more humanlike behaviors, we focus on designing a series of linguistically-motivated experiments[4].

[3] Although the pretraining tasks of T5 and BART do include modified versions of the sentence infilling problem or related text denoising tasks, our initial experimentation suggested that pretrained T5 and BART could not fully support the flexible generalization required in our studies, hence we fine-tuned them as above.

[4] Code and stimuli available at `https://github.com/pqian11/fragment-completion`

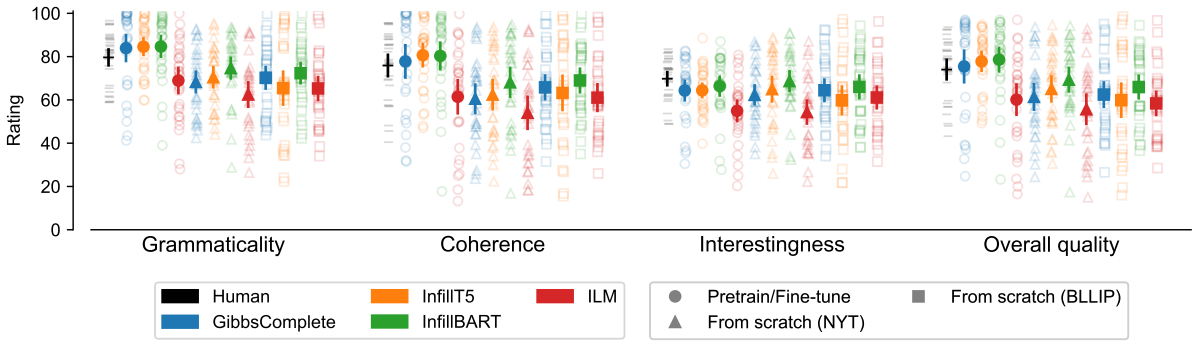


Figure 2: Human evaluation of completion quality for human participants and models in Experiment I. Semi-transparent hollow markers indicate the average rating for a particular item. Solid markers indicate the mean score, with error bar representing asymptotic 95% confidence interval of the population mean score.

3.1 Evaluation I: Completion Quality Given Bi-directional Context

Our first experiment qualitatively confirms that each model respects basic bidirectional constraints, and quantitatively evaluates the fluency of each model's completions using human judgments. We design 30 two-fragment stimuli of the form

$$\alpha \_\_\_\_ \beta$$

by adapting sentences from the Brown Corpus (Francis and Kucera, 1979) and British National Corpus (2001), choosing spans to crop out such that successful completion does not require outside-of-sentence information or much factual world knowledge, but does require non-trivial respect of grammatical constraints: we require that both α and β are multi-word fragments that cross conventional constituent boundaries.

Table 4 in Appendix G.1 lists one randomly-generated completion from different models to a subset of the stimuli. Qualitatively, all models generate high-quality completions that fit the context and sound fluent, although coherence is sometimes lacking. For quantitative evaluation, we recruit human raters on Prolific to evaluate the quality of the completed sentences written by models as well as human writers previously recruited on Mechanical Turk. Human raters were presented with the fragmentary input together with a completion and asked to judge the grammaticality, coherence, interestingness, and overall quality of the presented completion. Ratings range from 1 to 100, with 1 the lowest score and 100 the highest. Each human rater judged 150 completions in total, with 30 completions from a human writer or each of the models. Raters did not know whether a completion comes from a human or a model. The results of these ratings are shown in Figure 2: GibbsComplete, InfillT5, and InfillBART achieve similar performance on grammaticality judgment on this set of stimuli. The same human evaluation procedure is also applied to the set of models trained from scratch on the `NYT` and `BLLIP` data; results are given in Figure 2. Overall performance is worse than with the pre-trained models, but the relative patterns from model to model are similar. Overall, the results of Evaluation 1 suggest that when extensive bidirectional context is given, *all* models are able to generate structurally well-formed completions. This success motivates our next experiment, which involves briefer input in syntactically constrained configurations that more strongly challenge models' grammatical abilities.

3.2 Evaluation II: Satisfying Syntactic Constraints

To illustrate our syntactic constraint satisfaction tests, consider this example:

The paintings that the artist gave to _____ is _____ .

The first fragment consists of a plural noun phrase with an incomplete relative clause postmodifier; the second fragment is a singular verb, "is". The plural noun in the first fragment cannot be the subject of "is" due to number agreement in English, which forces a syntactically complex completion, such as "The paintings that the artist gave to *the museum are gorgeous and one of them* is *absolutely a masterpiece*.".

We design 26 novel fragment configurations to test models' syntactic behavior, ranging broadly across subject-predicate agreement, clausal structure, coordination, and filler-gap dependencies. For

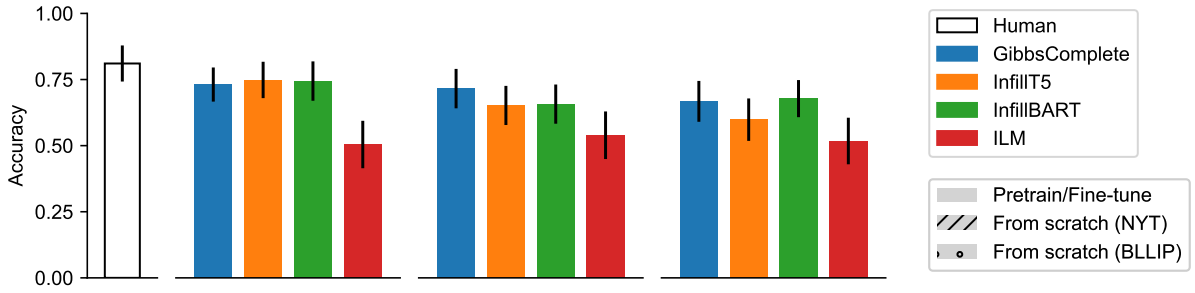


Figure 3: Aggregated results on syntactic reasoning tests. Error bars represent 95% confidence intervals.

each configuration we construct a set of semantically diverse fragmentary inputs that share the critical high-level syntactic structure for inducing similar grammatical constraints. The semantic diversity of the items facilitates more reliable estimation of the models' syntactic reasoning abilities. We provide descriptions and examples of each syntactic reasoning test in Appendix E.

With each model we generated 35 completions for each stimulus in each of the 26 tests. To collect human judgments for every single completion would be laborious and difficult to scale, in particular because evaluating whether the syntactic constraint is satisfied often requires some linguistic expertise. Instead, we represent the key syntactic constraints imposed by the fragments as tree patterns to be expected in the constituency parse of a completed stimulus. For example, given a stimulus " ____ published won ____ ." from the test (6) as shown in Appendix E, the linguistic intuition is that "published" should be part of a Verb Phrase embedded in a relative clause that modifies the subject of the predicate "won", despite other possible structural variations. We express these tree patterns using the Tregex tool (Levy and Andrew, 2006), compute the average rate of hitting the desired syntactic patterns out of the 35 completions which are annotated with syntactic parses by an off-the-shelf neural constituency parser `benepar` (Kitaev and Klein, 2018), and average across all the stimuli in a test as the final accuracy score for that test.[5]

Figure 3 shows the performance of humans and each model; Figure 8 in Appendix G.2 breaks down accuracy scores by each test separately. Human performance[6] is as good as or superior to all models ($p < 0.05$ for all models except fine-tuned InfillT5 and InfillBART, two-sided paired t-test). For the pretrained models, ILM performs the worst; the other three models' performance is comparable. When training from scratch, GibbsComplete outperforms all other models except on `BLLIP`, where its performance is matched by InfillBART. To address a potential concern about the ensembling effect of the reranking process in GibbsComplete, we also examine the performance obtained when composing the outputs of the directly-specialized models with the reranking process of GibbsComplete, generating 1000 candidate completions per fragment and selecting the top-ranked 35 completions using the same reranker as in GibbsComplete. Figure 10 in Appendix G.2 shows the results: InfillT5 and InfillBART now perform the best and even match human performance, further underscoring the value of a compositional approach even when dedicated training for direct specialization is available.

3.3 Evaluation III: Structural Similarity to Human Behavior

Evaluation II showed that models can perform very well even on more grammatically challenging fragment completion tasks, in some cases matching human performance. But do these models complete fragments *in a similar way* as humans do? Evaluation III turns to this question, using still more open-ended fragments and fine-grained analysis of the structural similarity of human and model completions.

To evaluate fine-grained similarity of human and model behavior, we define summary statistics of features of completions and assess the similarity of the summary statistics seen in human and model completions. We designed 120 stimuli in the form

[5] We also conducted manual evaluation of a sample of the pattern-matching results, which confirmed high accuracy of this automated evaluation procedure; see Appendix F for details.

[6] We collected human completions from Mechanical Turk for two stimulus items of each test, with 5–6 responses for each item. Human completions were evaluated with the same tree pattern-based method as model-generated completions.

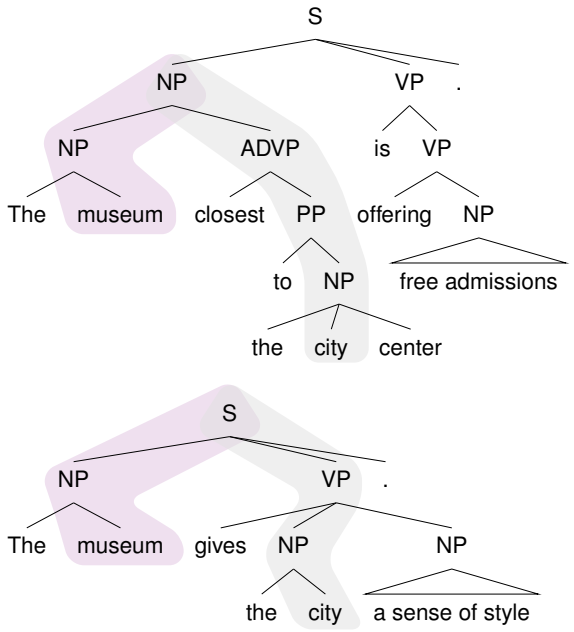


Figure 4: Lowest common ancestor of the fragments "museum" and "city" in constituency parses.

of " ____ w_1 ____ w_2 ____ .", where w_1 and w_2 are single-word fragments, allowing for a diverse range of plausible syntactic choices of the global context. For example,

____ museum ____ city ____ .

may be completed using a variety of different structural configurations, as shown in Figure 4. We parse each completion with `benepar` (Kitaev and Klein, 2018) and use the syntactic category of the lowest common ancestor (LCA) of w_1 and w_2 in the parse tree as a feature of the completion. We choose 40 Noun-Noun, 40 Adjective-Adjective, and 40 Adjective-Noun combination as w_1 and w_2 respectively, once again with diverse semantic content. We recruited human subjects from Mechanical Turk, with 18 subjects for Noun-Noun condition, 18 subjects for Adjective-Adjective, and 18 subjects for Adjective-Noun. Each subject wrote one completion for every item in the assigned condition. For an item, the completions from the subjects provide the human data from which we estimate the summary statistics of interest. We estimate the LCA frequency distribution across five syntactic category types: S, NP, VP, ADJP, and Other (everything else). For models, we sample and parse 35 completions for each stimulus to estimate the LCA frequency distribution.

We evaluate the performance of a model by its mean squared error (MSE) against human relative frequencies of the five syntactic category types for each item. Statistical significance of difference between model performances is tested with two-sided paired t-test. Figure 5 shows quantitative results for each model+training condition. Overall, GibbsComplete is the best-performing model. With pretrained models, GibbsComplete has significantly lower aggregated MSE than fine-tuned InfillT5 ($p = 0.012$) and fine-tuned ILM ($p = 0.010$), and is numerically lower than that of fine-tuned InfillBART ($p = 0.108$). When training from scratch, GibbsComplete is not significantly better. The MSE of GibbsComplete with motifs trained from scratch is not significantly better than other models trained from scratch on `NYT`, but when training on `BLLIP` it significantly outperforms InfillT5 ($p = 0.017$) and InfillBART ($p = 0.048$) trained from scratch on `BLLIP`. Looking at specific LCA categories, we find that GibbsComplete outperforms all the other models ($p < 0.005$) in matching the frequency of NP in human completions. Except when comparing ILM trained on `NYT` to GibbsComplete on VP ($p = 0.041$), no other directly-specialized models significantly performs better than GibbsComplete for category S, VP, and ADJP. Overall, these results suggest that the statistics of LCA categories in the parsed completions by GibbsComplete with pretrained models better match those of humans than those fine-tuned models, and that the advantage of GibbsComplete may also extend to low-resource setting.

4 Discussion

In this paper we have considered the direct-specialization approach to novel tasks dominant in contemporary NLP today as a cognitive hypothesis: that the knowledge structures and processes humans deploy for novel language tasks are best captured by the end result of fine-tuning or from-scratch training on the novel task itself. We have contrasted this with a competing hypothesis, namely that human behavior in novel tasks is better captured by inferences resulting from flexibly composing existing basic computational motifs, which relates to the idea explored in compositional use of neural modules (Andreas et al., 2016). To compare these hypotheses, we have developed and tested new, more challenging, and more open-ended versions of the text infilling task, which we term fragmentary input completion, and evaluated the performance of different models instantiating the two

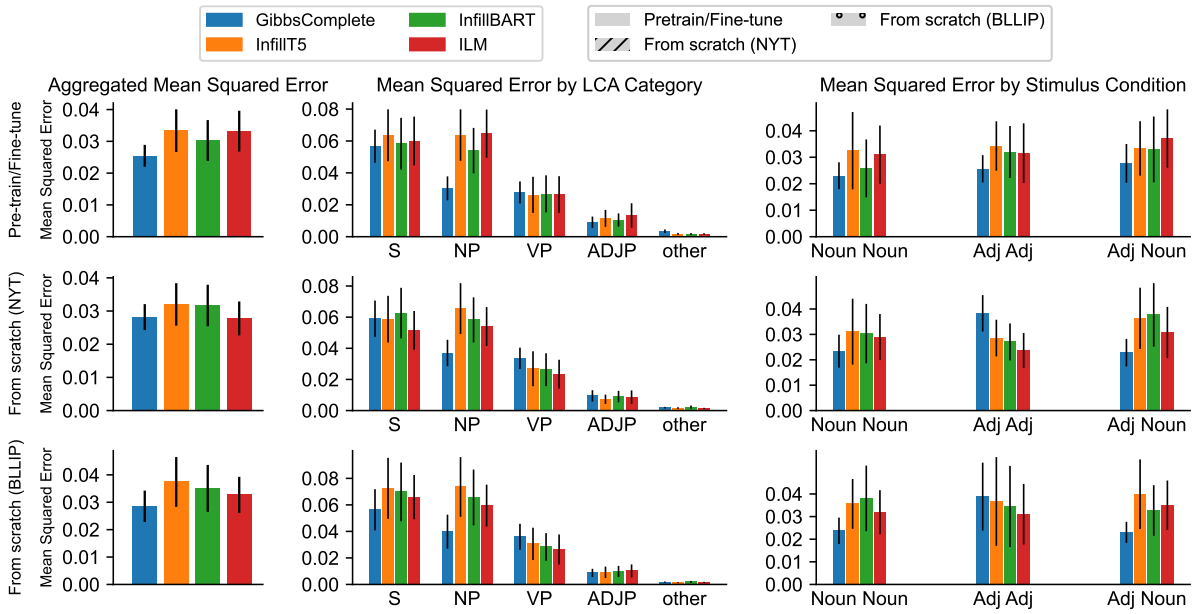


Figure 5: Comparing structural statistics in model's completions with that of the human-written completions. Error bars represent asymptotic 95% confidence intervals.

hypotheses against subjective human judgments, fixed success criteria, and fine-grained comparisons with human task performance. In the future this approach could be extended further for more comprehensive evaluation of conditioned language generation systems.

Our results are generally favorable for the compositional inference hypothesis as exemplified by our novel GibbsComplete algorithm, which composes the two fundamental language modelling motifs central to today's language models, masked word prediction and autoregressive modeling. The motifs themselves need to be learned in the first place, but there is a strong case that these motifs reflect tasks fundamental to everyday language use: identifying an uncertain word using bidirectional context (Connine et al., 1991; Dilley and Pitt, 2010; Levy, 2008b) and predicting upcoming input (Hale, 2001; Levy, 2008a; Kutas et al., 2011; Kuperberg and Jaeger, 2016). The idea we advance here, that these fundamental motifs are pre-existing and flexibly deployed for novel tasks, echoes a long-standing perspective in cognitive science well-summarized by Bruner et al. (1986), that "*Thinking is not the acquisition of knowledge, but the use of knowledge in the interest of solving problems*".

Of course, we do not interpret these results as suggesting that the human mind is literally a Gibbs sampler. But there are broader arguments that sampling-based approaches may capture important general features of human inferential patterns (Vul et al., 2014). Furthermore, the compositional inference approach to modelling flexible language generation by no means diminishes the value of learning or fine-tuning—indeed, fine-tuned models can themselves be composed (see also our exploratory work in Appendix G.2). Rather, we hope that this work may help widen the perspective on the relationship between learning and inference in novel language tasks and contexts.

5 Conclusion

Scaled learning and quick adaptation of linguistic representation have enabled huge progress in the engineering of high-performance NLP systems, but our results suggest that flexible redeployment of basic computational motifs may have advantages for capturing how humans flexibly use language in novel circumstances where they do not have extensive experience. Our studies offer systematic model comparisons with materials designed to highlight subtle features of grammatical knowledge and featural statistics of human completion preferences, and point to the need for longer-term efforts in understanding and modeling human cognitive flexibility in computational terms. As an initial step, we explored the "compositional inference" hypothesis by sketching out an inference algorithm based on the principles of approximate Bayesian

inference. The results suggest certain advantages of an inference-oriented view of human language generation, and an alternative path towards building models that process linguistic information as flexibly as humans do.

Acknowledgments

We thank the anonymous reviewers and members of the MIT Computational Psycholinguistics Lab for their helpful comments, and members of the Goals, Problems and Stories working group for discussions on early idea of the project. This work was supported by the MIT-IBM Watson AI Lab and by NSF award BCS-2121074.

References

Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. 2016. Neural module networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 39–48.

Mathias Berglund, Tapani Raiko, Mikko Honkala, Leo Kärkkäinen, Akos Vetek, and Juha T Karhunen. 2015. Bidirectional recurrent neural networks as generative models. *Advances in Neural Information Processing Systems*, 28:856–864.

British National Corpus. 2001.

Jerome Seymour Bruner, Jacqueline J. Goodnow, and George Allen Austin. 1986. *A Study of Thinking*. Transaction publishers.

Eugene Charniak, Don Blaheta, Niyu Ge, Keith Hall, John Hale, and Mark Johnson. 2000. Bllip 1987-89 wsj corpus release 1. *Linguistic Data Consortium, Philadelphia*, 36.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(ARTICLE):2493–2537.

Cynthia M. Connine, Dawn G. Blasko, and Michael Hall. 1991. Effects of subsequent sentence context in auditory word recognition: Temporal and linguistic constraints. 30(2):234–250.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Laura C Dilley and Mark A Pitt. 2010. Altering context speech rate can cause words to appear or disappear. *Psychological Science*, 21(11):1664–1670.

Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2492–2501, Online. Association for Computational Linguistics.

Hermann Ebbinghaus. 1897. *Über eine neue Methode zur Prüfung geistiger Fähigkeiten und ihre Anwendung bei Schulkindern*. Leop. Voss.

Dumitru Erhan, Aaron Courville, Yoshua Bengio, and Pascal Vincent. 2010. Why does unsupervised pretraining help deep learning? In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 201–208. JMLR Workshop and Conference Proceedings.

William Fedus, Ian Goodfellow, and Andrew M Dai. 2018. Maskgan: Better text generation via filling in the _. In *International Conference on Learning Representations*.

W Nelson Francis and Henry Kucera. 1979. Brown corpus manual. *Letters to the Editor*, 5(2):7.

John Hale. 2001. A probabilistic Earley parser as a psycholinguistic model. pages 159–166, Pittsburgh, Pennsylvania.

Xingwei He and Victor OK Li. 2021. Show me how to revise: Improving lexically constrained sentence generation with xlnet. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12989–12997.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. 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.

Yichen Huang, Yizhe Zhang, Oussama Elachqar, and Yu Cheng. 2020. INSET: Sentence infilling with INter-SEntential transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2502–2515, Online. Association for Computational Linguistics.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*,

pages 2676–2686, Melbourne, Australia. Association for Computational Linguistics.

Gina R Kuperberg and T Florian Jaeger. 2016. What do we mean by prediction in language comprehension? *Language, cognition and neuroscience*, 31(1):32–59.

Marta Kutas, Katherine A DeLong, and Nathaniel J Smith. 2011. A look around at what lies ahead: Prediction and predictability in language processing. In M. Bar, editor, *Predictions in the brain: Using our past to generate a future*, pages 190–207. Oxford University Press.

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

Roger Levy. 2008b. A noisy-channel model of rational human sentence comprehension under uncertain input. pages 234–243, Waikiki, Honolulu.

Roger Levy and Galen Andrew. 2006. Tregex and tsurgeon: tools for querying and manipulating tree data structures. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA).

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Dayiheng Liu, Jie Fu, Pengfei Liu, and Jiancheng Lv. 2019a. TIGS: An inference algorithm for text infilling with gradient search. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4146–4156, Florence, Italy. Association for Computational Linguistics.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019b. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

David Marr. 1982. *Vision: A computational investigation into the human representation and processing of visual information*. W.H. Freeman & Company.

Ning Miao, Hao Zhou, Lili Mou, Rui Yan, and Lei Li. 2019. Cgmh: Constrained sentence generation by metropolis-hastings sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6834–6842.

Lianhui Qin, Vered Shwartz, Peter West, Chandra Bhagavatula, Jena D. Hwang, Ronan Le Bras, Antoine Bosselut, and Yejin Choi. 2020. Back to the future: Unsupervised backprop-based decoding for counterfactual and abductive commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 794–805, Online. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Evan Sandhaus. 2008. The New York Times Annotated Corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

Tianxiao Shen, Victor Quach, Regina Barzilay, and Tommi Jaakkola. 2020. Blank language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5186–5198, Online. Association for Computational Linguistics.

Jinyue Su, Jiacheng Xu, Xipeng Qiu, and Xuanjing Huang. 2018. Incorporating discriminator in sentence generation: a Gibbs sampling method. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Edward Vul, Noah Goodman, Thomas L Griffiths, and Joshua B Tenenbaum. 2014. One and done? optimal decisions from very few samples. *Cognitive science*, 38(4):599–637.

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 30–36, Minneapolis, Minnesota. Association for Computational Linguistics.

Peter West, Ximing Lu, Ari Holtzman, Chandra Bhagavatula, Jena D. Hwang, and Yejin Choi. 2021. Reflective decoding: Beyond unidirectional generation with off-the-shelf language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1435–1450, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu,

Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Wanrong Zhu, Zhiting Hu, and Eric Xing. 2019. Text infilling. *arXiv preprint arXiv:1901.00158*.

A Behavioral Experiment

Figure 6 shows the screenshot of one trial in the behavioral experiment, through which we collected human-written completions to the fragmentary linguistic stimuli. In our implementation, we depict the blanks in the fragmentary input as a short underline placeholder. As soon as one starts typing words in the text input box, the typed content will be rendered immediately as replacement of the corresponding blank.

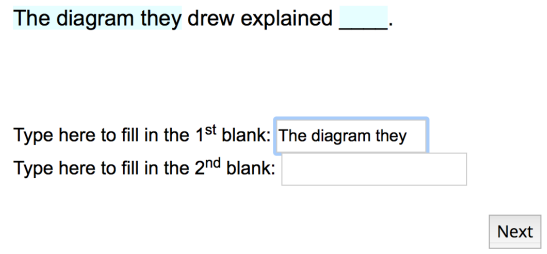


Figure 6: Screenshot of an experimental trial of the fragment completion task for human participants.

B Human Evaluation of Completion Quality

We collected human rating of completion quality for Evaluation I. Figure 7 shows the screenshot of one trial in the completion quality judgment task for human raters.

C Model Implementation

C.1 Model Size

Table 1 lists the number of parameters in each model. Notice that the count of total parameters for GibbsComplete includes the autoregressive language modeling motif (124M) and the masked language modeling motif (109M), both of which has 12 layers.

	# of layers	# of parameters
GibbsComplete	12	233M
InfillT5	12	223M
InfillBART	24	406M
ILM	12	124M

Table 1: Model size comparison.

C.2 Sampling Infilling Pairs for Model Training

To train InfillT5, InfillBART, and ILM directly on the text infilling task, we employ a data generation process to randomly sample training data and prepare validation data. Given a sentence, we randomly replace spans of the sentence with `[BLANK]` symbol and append the original contents of the blanks in order separated by `[FILLER]` symbol. The total number of spans is uniformly sampled between 2 and 9. For a sentence, the maximal number of spans to be sampled is capped at the sentence length if the total number of words in the sentence is less than 9. This process gives us a fragmentary context C and one of its plausible completion B. The paired C and B sampled from a corpus are used to provide task supervision signal.

For example, given the original sentence in the corpus: "The camera had been operating on its backup electrical system since last summer, however, when electrical problems in its main system caused it to shut down for a while.", a training instance after randomly cropping the sentence may consist of C as "The camera had been `[BLANK]` backup electrical `[BLANK]` electrical problems in its main system caused it to shut `[BLANK]` while." and B as "operating on its `[FILLER]` system since last summer, however, when `[FILLER]` down for a `[FILLER]`", where the cropped spans of words to be filled in are simply concatenated with `[FILLER]` marking the end of each span.

Notice that pretrained T5 uses a numbered list of tokens (e.g. `<extra_id_0>`, `<extra_id_1>`, etc) as delimiters of the spans to be filled into the blanks. Hence we follow the same convention when generating infilling pair data for fine-tuning pretrained T5 or training InfillT5 from scratch.

D Test Items in Evaluation I

Here we list the 30 items of the form "α ____ β" used in Evaluation I:

(1) He is one of _____ jazz on a violin.
(2) It is unclear _____ will have on the elections.
(3) The prices - even the special offers - _____ thought reasonable.
(4) Giving up the violin _____ had predicted would have a promising career on the concert stage.
(5) A difference of opinion arose between _____ the vote is handled.
(6) He was quiet, polite, _____ he met.
(7) The local authority _____ applied refused to provide the necessary grant.
(8) They argued _____ proposed implies a competitive outcome.
(9) Vineyards were found scattered throughout the region _____ visited grew any grapes.
(10) She knew the rents _____ find any good, affordable space in town.
(11) Due to the fact that building codes _____ what they are.

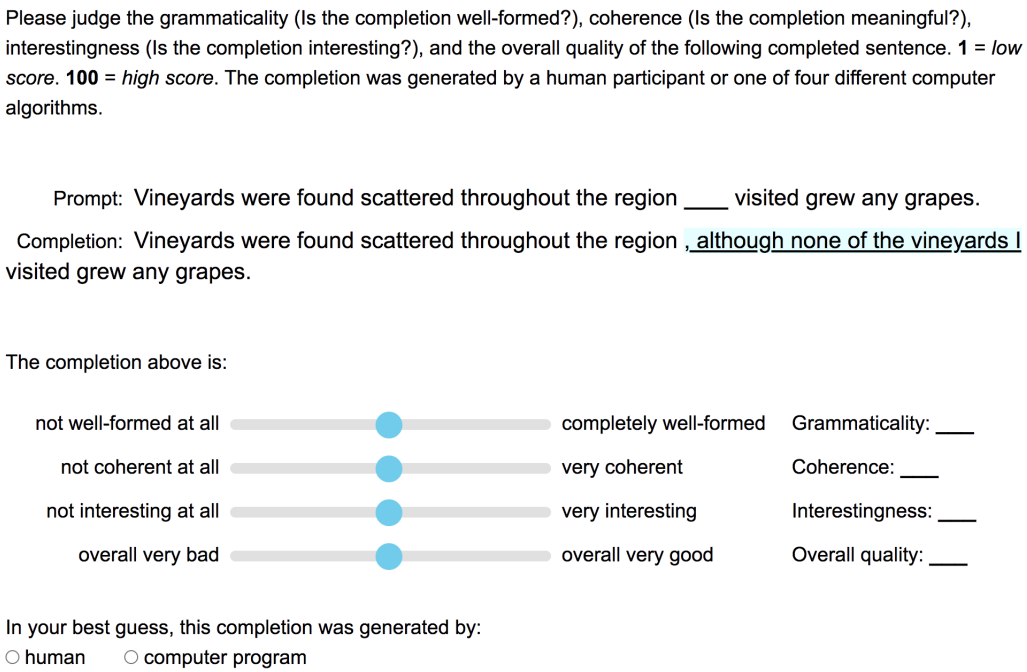


Figure 7: Screenshot of a trial of the completion quality evaluation task for human raters.

(12) The fact that _____ but no less important in the long run.
(13) It is _____ ever existed outside the imaginations of historians.
(14) He _____ any of the Beatles songs.
(15) Even _____ say the findings should be taken with caution.
(16) The manager put _____ table.
(17) Each of the children invited to _____ marked with a red, white and blue ribbon.
(18) A sports club located _____ to keep up with the rent.
(19) A man believed involved _____ yesterday.
(20) The famous writer _____ a publisher to write two more books.
(21) If these services are _____ revenues to make them possible.
(22) It does indeed _____ through no fault of their own shoulder the cost.
(23) Although they emphasizes that the opportunities _____ them worth considering.
(24) This building has been consistently _____ and businesses for ten years.
(25) The team admitted it _____ a tough year.
(26) The performance marked _____ most promising young conductor.
(27) Previous experience _____ the best equipment possible.
(28) There appears to be enough _____ the interest deficiency for eight more years.
(29) They do not _____ not open to debate.
(30) The government _____ not addressing the poverty.

E Syntactic Reasoning Tests

Here we briefly introduce the 26 sets of tests, using one stimulus from each test to illustrate the syntactic constraints imposed by the fragments. Tests (1–5) target SUBJECT–PREDICATE AGREEMENT in English. These tests target the number mismatch of a noun phrase and a predicate, with the noun phrase embedded in various local syntactic structures, including (1) a bare noun phrase, (2) a noun phrase with a relative clause, (3) a noun phrase in complement clause, (4) a noun phrase in a coordinate structure, and (5) a noun phrase with a partially complete prepositional-phrase modifier.

(1) The reporter _____ are _____ .
(2) The paintings that the artist gave to _____ is _____ .
(3) _____ insist that the jury _____ have _____ .
(4) The bike and _____ has _____ .
(5) The museums in _____ is _____ .

Tests (6–15) target a variety of CLAUSAL STRUCTURE. Tests (6–11) require that the completion place the first verb inside an embedded clause that ends where the second verb's VP begins. Tests (12–13) feature a sentence-final verbal phrase with a missing obligatory object, requiring a completion in which that object has been extracted. Tests (14–15) target resultative structure.

(6) _____ published won _____ .
(7) _____ introduced into the classroom has _____ .
(8) _____ visited during the trip to the beautiful valley grew _____ .
(9) _____ owned the trademark began _____ .
(10) _____ brought out the plan for protecting the habitat of the endangered birds argued _____ .
(11) Giving up the violin _____ predicted would _____ .
(12) _____ relied on.

(13) _____ the founder of the research institute has never dreamed of.
(14) _____ growth steady.
(15) _____ decision of selling the old car unwise.

Tests (16–22) target COORDINATION of different phrasal categories, including (16) coordinated clauses, (17) coordinated VPs, (18) coordinated NPs in an embedded clause, (19) coordinated VPs in an embedded clause, and (22) coordinated subjects in a complement clause. Tests (20) and (21) feature collocations "either ... or ..." and "neither ... nor ... ". The intuition behind (20–21) is that the "either" or "neither", if used as conjunction words, requires an "or" or "nor" to follow respectively.

(16) _____ problem but no one took _____ .
(17) _____ standards and improved the _____ .
(18) _____ the symbols and the patterns carved on the surface of the clay bowl stood for.
(19) _____ studied the formation of galaxies and explored the mystery of the universe was _____ .
(20) _____ either book a ticket from the official website of _____ .
(21) _____ neither give up too quickly simply because of the failures encountered at the initial _____ .
(22) _____ what the senator and those who supported _____ steps moving forward.

Tests (23–26) target FILLER-GAP DEPENDENCIES. The basic idea is that the wh-word requires a gap to appear at an appropriate position.

(23) _____ how the _____ has ever thought about.
(24) _____ what the _____ the sculpture.
(25) _____ who _____ bake different kinds of bread.
(26) _____ why _____ benefited from.

F Comparing Automated Tree-Search Evaluation with Human Judgment

	Precision	Recall	Accuracy
Human	0.938	0.894	0.865
GibbsComplete	0.974	0.841	0.846
InfillT5	0.978	0.889	0.875
InfillBART	0.966	0.934	0.913
ILM	0.928	0.842	0.837

Table 2: Performance of tree search pattern-based evaluation by model types.

To show the effectiveness of Tregex-based automatic evaluation with hand-designed tree search pattern, we conduct a small-scale comparison of human evaluation of the targeted structure in a completion to an item of the syntactic reasoning test. We select a subset of completions generated by humans and models. For each model along with humans, we randomly choose two items out of each test and randomly select two completions to each of the item. There are 520 completions in total for us to annotate human judgment.

For each of the 520 completions, we evaluate whether it resolves the challenge associated with the particular test and annotate human judgment as binary outcome, with 1 indicating success and 0 indicating failure. During the annotation process, the human annotator was blind to the judgment of Tregex-based evaluation as well as which model the completion came from. Completions were also shuffled before the annotation. Table 2 shows that the tree search pattern-based evaluation generally aligns with human judgment with high accuracy for all models. Table 3 shows that tree search pattern-based automated evaluation result aligns with that of human judgments for most tests.

G Additional Results

G.1 Evaluation I

Table 4, 5, and 6 list one sampled completion for four selected items from the 30 stimuli used in Evaluation I. Completions in Table 4 are generated by pretrained/fine-tuned models. Completions in Table 5 are generated by models trained from scratch on NYT. Completionns in Table 6 are generated by models trained from scratch on BLLIP.

G.2 Evaluation II

Figure 8 shows model performance on each syntactic reasoning test. Figure 9 and Figure 10 show accuracies on each syntactic reasoning test and the aggregated performance respectively, based on results where the outputs of the directly-specialized models are composed with the reranking process of GibbsComplete with the same reranker.

G.3 Evaluation III

Figure 11 shows the comparison among GibbsComplete and other models with similar reranking process applied. The qualitative pattern is similar to those reported in Section 3.3. Figure 12, 13, and 14 plot the relative frequency of specific LCA category in human-written completions against that of model-generated completions, which are estimated from pretrained/fine-tuned models, models trained from scratch on NYT, and models trained from scratch on BLLIP respectively.

Test Index	Test Name	Precision	Recall	Accuracy
(1)	Number Agreement	0.875	0.875	0.800
(2)	Number Agreement (Long Subject)	0.938	0.882	0.850
(3)	Number Agreement (Embedded Clause)	1.000	0.900	0.900
(4)	Number Agreement (Coordination)	1.000	0.727	0.850
(5)	Number Agreement (with PP)	0.938	1.000	0.950
(6)	Clausal Structure	1.000	0.882	0.900
(7)	Clausal Structure (PP Adjunct)	0.950	1.000	0.950
(8)	Clausal Structure (Long Adjunct)	0.938	0.833	0.800
(9)	Clausal Structure (Complement)	1.000	0.944	0.950
(10)	Clausal Structure (Long Complement)	1.000	0.944	0.950
(11)	Gerund	0.846	0.733	0.700
(12)	Phrasal Verb	0.938	0.789	0.750
(13)	Phrasal Verb (with NP)	1.000	0.875	0.900
(14)	Resultative	1.000	0.938	0.950
(15)	Resultative (Long NP)	1.000	0.867	0.900
(16)	S Coordiation	1.000	0.789	0.800
(17)	VP Coordination	1.000	1.000	1.000
(18)	Embedded NP Coordination	1.000	0.500	0.700
(19)	Embedded VP Coordination	0.947	1.000	0.950
(20)	Coordination (either)	1.000	0.833	0.850
(21)	Coordination (neither)	0.727	0.667	0.650
(22)	Coordination in wh-clause	1.000	0.842	0.850
(23)	Filler-Gap (Adjunct)	0.941	0.889	0.850
(24)	Filler-Gap (Object)	1.000	1.000	1.000
(25)	Filler-Gap (Subject)	0.947	0.947	0.900
(26)	Filler-Gap (Phrasal Verb)	0.882	1.000	0.900

Table 3: Performance of tree search pattern-based evaluation by individual test.

	He is one of _____ jazz on a violin.
GibbsComplete	He is one of the few jazz musicians to have played jazz on a violin.
InfillT5	He is one of the first to play jazz on a violin.
InfillBART	He is one of the few people in the country with a master of jazz on a violin.
ILM	He is one of only three players to have taught jazz on a violin.
	Vineyards were found scattered throughout the region _____ visited grew any grapes.
GibbsComplete	Vineyards were found scattered throughout the region, but none of the vineyards she had visited grew any grapes.
InfillT5	Vineyards were found scattered throughout the region but no one ever visited grew any grapes.
InfillBART	Vineyards were found scattered throughout the region, but none of the farms I visited grew any grapes.
ILM	Vineyards were found scattered throughout the region, and once they had visited grew any grapes.
	The famous writer _____ a publisher to write two more books.
GibbsComplete	The famous writer, however, moved on and finally found a publisher to write two more books.
InfillT5	The famous writer, he said he is considering a bid from a publisher to write two more books.
InfillBART	The famous writer Ivanov has just been hired by a publisher to write two more books.
ILM	The famous writer Samuel Beckett became a publisher to write two more books.
	Giving up the violin _____ had predicted would have a promising career on the concert stage.
GibbsComplete	Giving up the violin, she asked her brother, Francis, who she had predicted would have a promising career on the concert stage.
InfillT5	Giving up the violin to pursue a more eminent musical career, he made Mr. Orbach a friend and friend who he had predicted would have a promising career on the concert stage.
InfillBART	Giving up the violin, she gave up her training to concentrate on her son's music studies, which she had predicted would have a promising career on the concert stage.
ILM	Giving up the violin he had predicted would have a promising career on the concert stage.

Table 4: Completions generated by pretrained/fine-tuned models for selected stimuli in Evaluation I.

	He is one of _____ jazz on a violin.
GibbsComplete	He is one of some of the most popular musicians in the history of jazz on a violin.
InfillT5	He is one of the best and brightest young singers, though the music is a bit of a sardonic evocation of jazz on a violin.
InfillBART	He is one of several young singers nominated for the prize, whether with a soloist or with the jazz on a violin.
ILM	He is one of four music students competing with jazz on a violin.
	Vineyards were found scattered throughout the region _____ visited grew any grapes.
GibbsComplete	Vineyards were found scattered throughout the region by people who pointed out that no one they visited grew any grapes.
InfillT5	Vineyards were found scattered throughout the region, and none of the pigs he visited grew any grapes.
InfillBART	Vineyards were found scattered throughout the region, and no one in the group of counties we visited grew any grapes.
ILM	Vineyards were found scattered throughout the region and not in as far north as the New Hampshire border, but the discovery of more vines than anyone else hasd visited grew any grapes.
	The famous writer _____ a publisher to write two more books.
GibbsComplete	The famous writer then told us that it would be a mistake for a publisher to write two more books.
InfillT5	The famous writer in the novel–"Fountains and Mysteries"–really was looking for a publisher to write two more books.
InfillBART	The famous writer who helped write "Municipal Bondage," she also said she had a intention of waiting for a publisher to write two more books.
ILM	The famous writer for The Times has a publisher to write two more books.
	Giving up the violin _____ had predicted would have a promising career on the concert stage.
GibbsComplete	Giving up the violin was a good example of what some critics had predicted would have a promising career on the concert stage.
InfillT5	Giving up the violin in his youth, he wore a long jacket and tie and was a fixture of a young man that he had predicted would have a promising career on the concert stage.
InfillBART	Giving up the violin part of the album, Mr. Carter, 61, earned a reputation as a musician who he had predicted would have a promising career on the concert stage.
ILM	Giving up the violin is the only kind of performance Mr. Levine had predicted would have a promising career on the concert stage.

Table 5: Completions generated by models trained from scratch on NYT for selected stimuli in Evaluation I.

	He is one of _____ jazz on a violin.
GibbsComplete	He is one of the few who think the big players are playing jazz on a violin.
InfillT5	He is one of the leading characters of jazz on a violin.
InfillBART	He is one of the few players to play jazz on a violin.
ILM	He is one of the most popular jazz on a violin.
	Vineyards were found scattered throughout the region _____ visited grew any grapes.
GibbsComplete	Vineyards were found scattered throughout the region and almost a third of the homes he had visited grew any grapes.
InfillT5	Vineyards were found scattered throughout the region, but the food I visited grew any grapes.
InfillBART	Vineyards were found scattered throughout the region, but neither those they visited grew any grapes.
ILM	Vineyards were found scattered throughout the region as soon as the trees visited grew any grapes.
	The famous writer _____ a publisher to write two more books.
GibbsComplete	The famous writer lives in Beverly Hills and is trying to persuade a publisher to write two more books.
InfillT5	The famous writer wants a publisher to write two more books.
InfillBART	The famous writer is a woman who has to get a book to get a publisher to write two more books.
ILM	The famous writer, a former writer for the New York Times, recently asked a publisher to write two more books.
	Giving up the violin _____ had predicted would have a promising career on the concert stage.
GibbsComplete	Giving up the violin was part of a major art collection that I had predicted would have a promising career on the concert stage.
InfillT5	Giving up the violin, which was not at all a good ad, was the key factor behind a successful series on MGM, which Mr. Ross had predicted would have a promising career on the concert stage.
InfillBART	Giving up the violin to the piano for a complete play was tough, since he was the conductor, a man you had predicted would have a promising career on the concert stage.
ILM	Giving up the violin, he was a bit uncomfortable and had predicted would have a promising career on the concert stage.

Table 6: Completions generated by models trained from scratch on BLLIP for selected stimuli in Evaluation I.

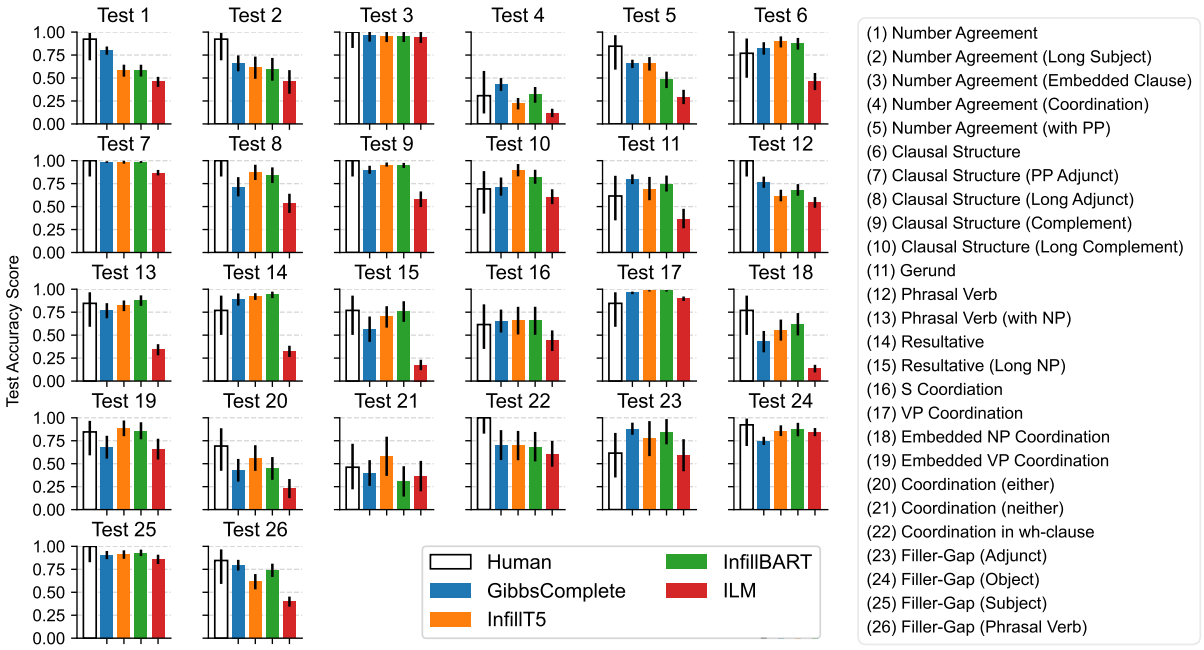


Figure 8: Model performances by each syntactic reasoning test, corresponding to example stimuli in Appendix E. Error bars represent 95% confidence intervals. The results are based on pretrained model or models fine-tuned on pretrained representation.

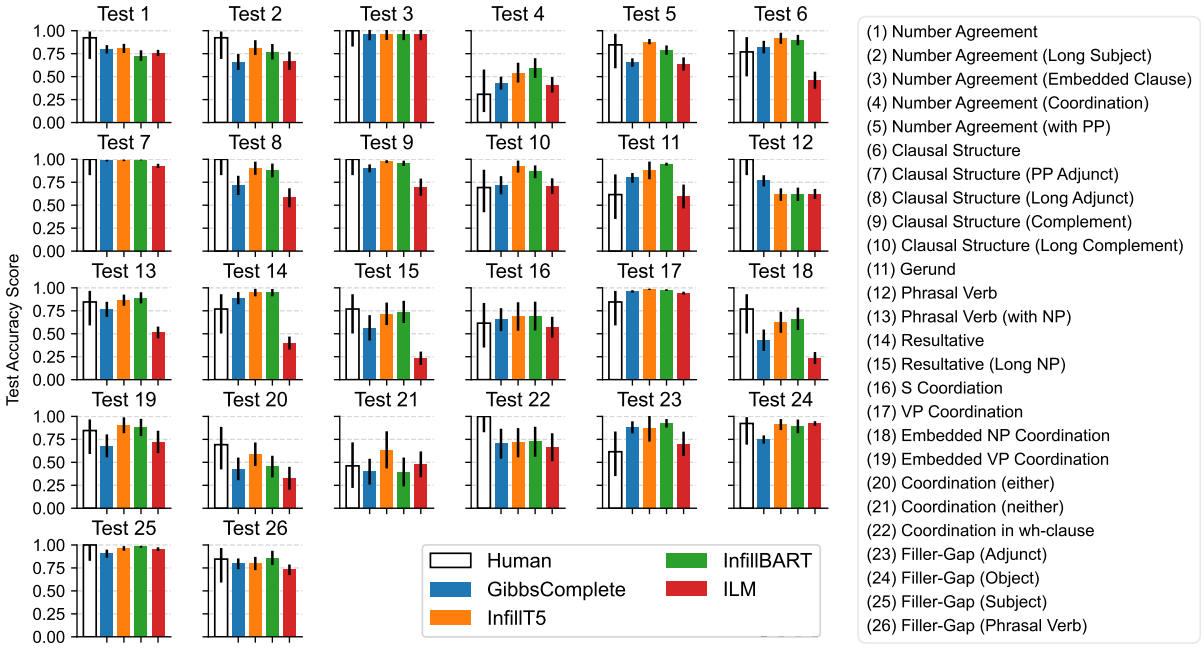


Figure 9: Model performances by each syntactic reasoning test, corresponding to example stimuli in Appendix E. Error bars represent 95% confidence intervals. The results are based on pretrained model or models fine-tuned on pretrained representation. Reranking is applied to InfillT5, InfillBART, and ILM.

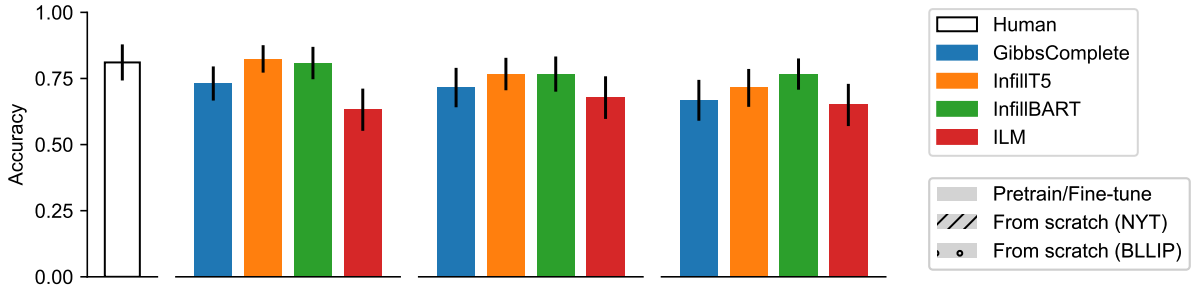


Figure 10: Aggregated performance on syntactic reasoning tests. Error bars represent asymptotic 95% confidence intervals. Reranking is applied to InfillT5, InfillBART, and ILM.

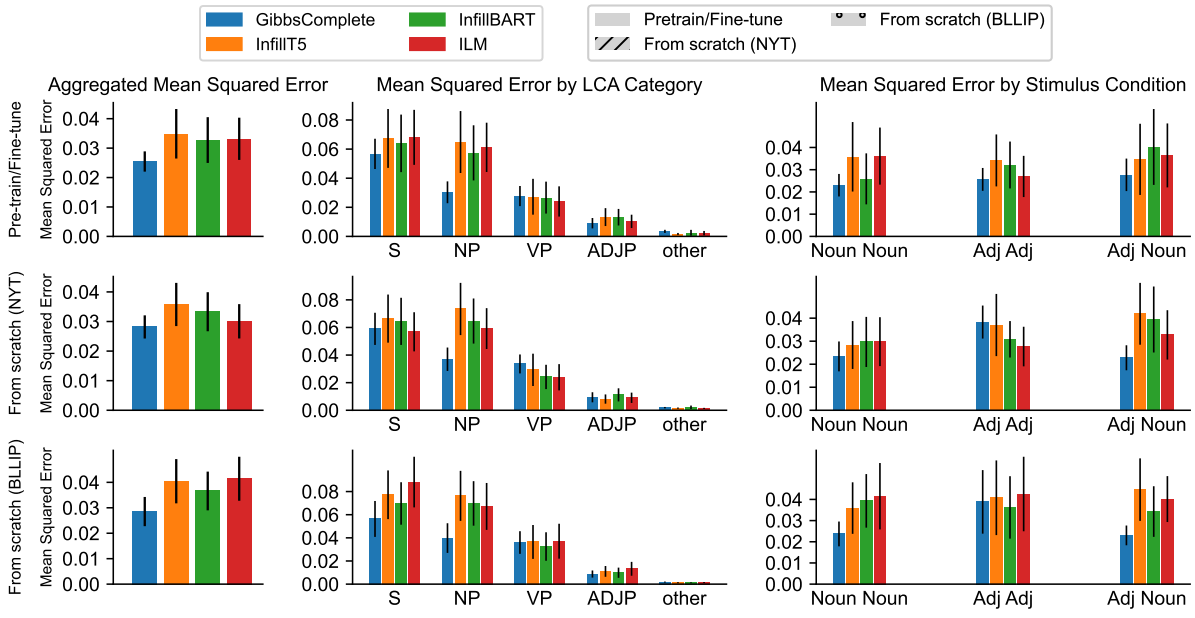


Figure 11: Comparing structural statistics in model's completions with that of the human-written completions. Error bars represent asymptotic 95% confidence intervals. Reranking is applied to InfillT5, InfillBART, and ILM.

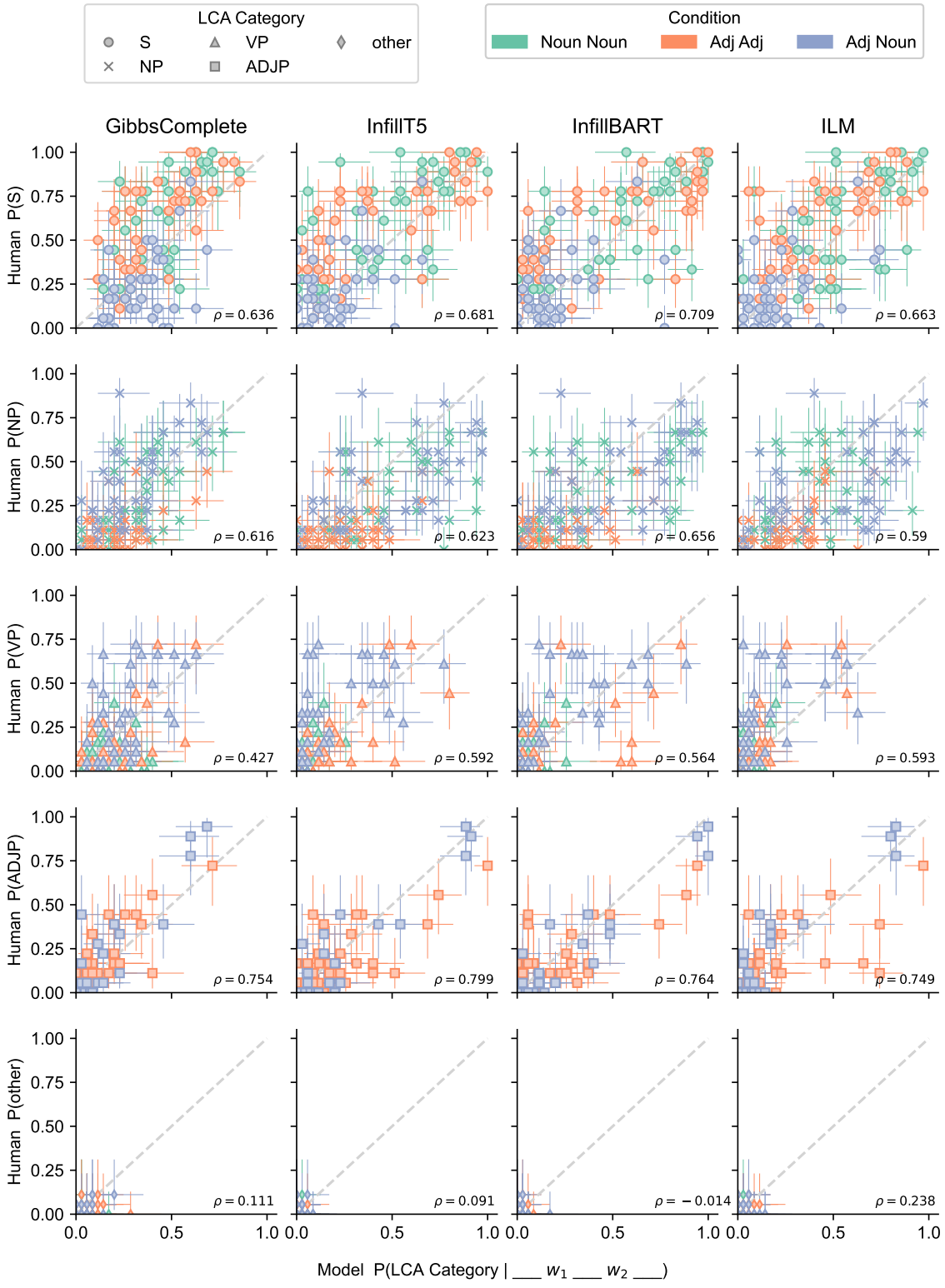


Figure 12: Scatter plots of structural statistics in human-written completions and that of model-generated completions. Error bars represent 95% confidence intervals. Spearman's ρ between the frequency of specific LCA category estimated from model and human completions across all the items is annotated in each subplot. Models plotted here are pretrained/fine-tuned, where GibbsComplete is composed of pretrained models without additional parameter tuning while InfillT5, InfillBART, and ILM are fine-tuned on a subset of New York Times corpus.

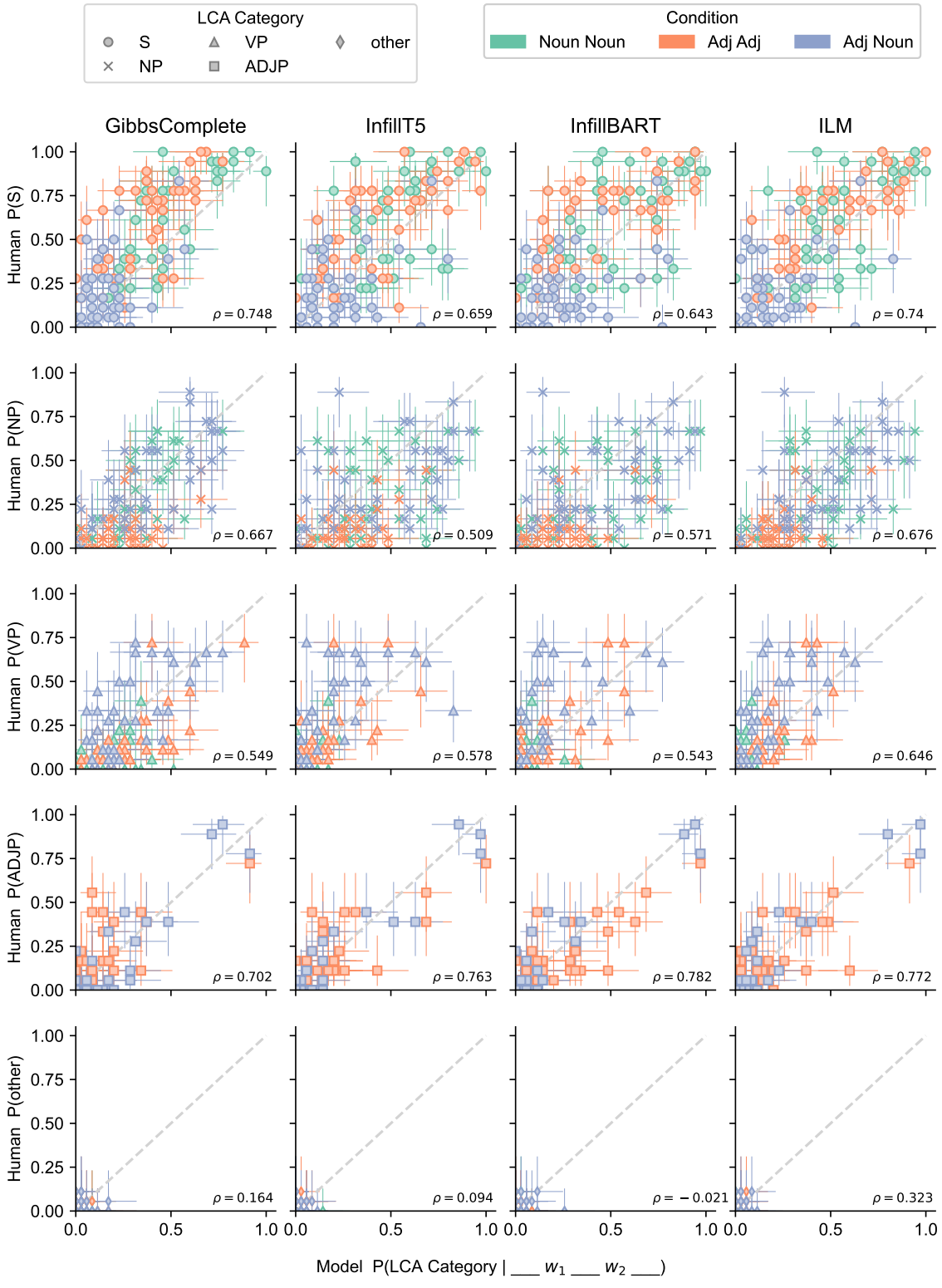


Figure 13: Scatter plots of structural statistics in human-written completions and that of model-generated completions. Error bars represent 95% confidence intervals. Spearman's ρ between the frequency of specific LCA category estimated from model and human completions across all the items is annotated in each subplot. Models plotted here are trained from scratch on NYT.

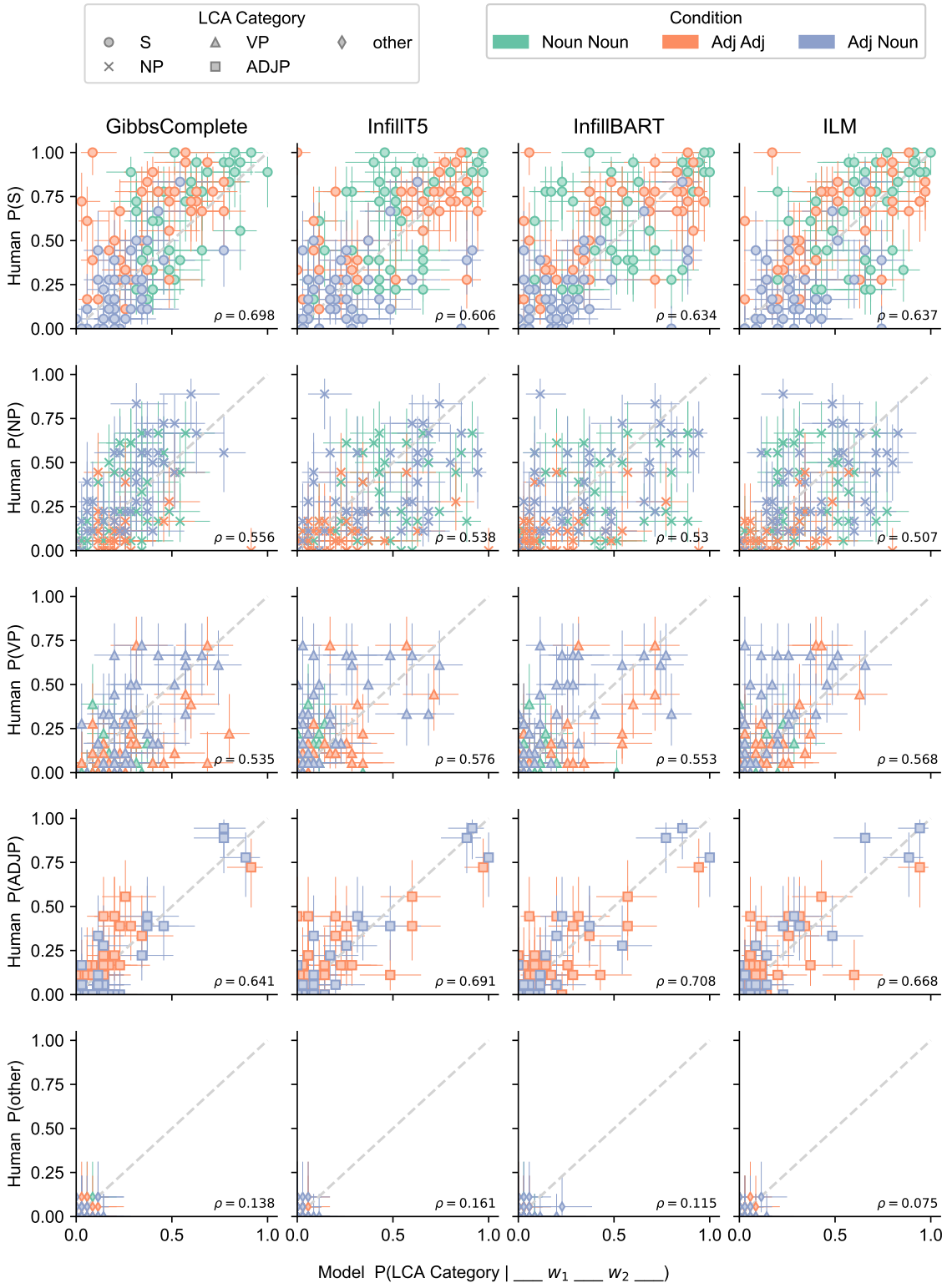


Figure 14: Scatter plots of structural statistics in human-written completions and that of model-generated completions. Error bars represent 95% confidence intervals. Spearman's ρ between the frequency of specific LCA category estimated from model and human completions across all the items is annotated in each subplot. Models plotted here are trained from scratch on `BLLIP`.