Transformers generalize differently from information stored in context vs weights

Stephanie C.Y. Chan* DeepMind
Ishita Dasgupta* DeepMind
Junkyung Kim DeepMind
Dharshan Kumaran DeepMind
Andrew K. Lampinen DeepMind
Felix Hill DeepMind

Abstract

Transformer models can use two fundamentally different kinds of information: information stored in weights during training, and information provided "in-context" at inference time. In this work, we show that transformers exhibit different inductive biases in how they represent and generalize from the information in these two sources. In particular, we characterize whether they generalize via parsimonious rules (*rule-based generalization*) or via direct comparison with observed examples (*exemplar-based generalization*). This is of important practical consequence, as it informs whether to encode information in weights or in context, depending on how we want models to use that information. In transformers trained on controlled stimuli, we find that generalization from weights is more rule-based whereas generalization from context is largely exemplar-based. In contrast, we find that in transformers pre-trained on natural language, in-context learning is significantly rule-based, with larger models showing more rule-basedness. We hypothesise that rule-based generalization from in-context information might be an emergent consequence of large-scale training on language, which has sparse rule-like structure. Using controlled stimuli, we verify that transformers pretrained on data containing sparse rule-like structure exhibit more rule-based generalization.

1 Introduction

Transformer-based architectures have an impressive ability to use both information stored in weights during training ("in-weights learning"), and information stored only in the inputs provided at inference time (without any gradient updates to the weights of the model; "in-context learning") [Chan et al., 2022]. In-context learning on pretrained models enables learning efficiently from a few examples ("few-shot learning") [Brown et al., 2020], or even efficiently compressing a large dataset ("prompt tuning") [Li and Liang, 2021, Lester et al., 2021, Sun et al., 2022]. Given the evident current and future potential for this new learning paradigm, it is important and useful to understand its inductive biases, especially how it differs from in-weights learning.

One way to understand inductive bias is by examining how models *generalize* to held-out data. In this work, we adapt the experimental paradigm in Dasgupta et al. [2022] that pose a classification task that distinguishes between two previously defined kinds of generalization behaviors (see 1). A "rule-based" decision is made on the basis of minimal features that support the category boundary [Ashby and Townsend, 1986], while an "exemplar-based" decision generalizes on the basis of similarity to examples from training data [Shepard and Chang, 1963], invoking many or all features available.

*Equal contributions

Preprint. Under review.

This distinction is particularly interesting when comparing in-weights vs in-context learning. Exemplar-based generalization (that uses all available features) is useful in a low-data regime where there is not enough information to form an abstract sparse rule [Feldman, 2020]. On the other hand, sparser rule-based generalization may help avoid sensitivity to spurious correlation when training with large, noisy, naturalistic datasets (that are commonly used to train in-weights learning).

We find that transformers exhibit a striking difference in their generalization from in-context vs in-weights information. Transformers display a strong inductive bias towards exemplar-based generalization from in-context information. In contrast, transformers display a strong inductive bias towards rule-based generalization from in-weights information.

However, when we pose a similar task to large transformer models pretrained on language, they exhibit stronger rule-based generalization from in-context information. One interpretation of these results is that the distribution of natural language is more compatible with rule-based generalization from context (rule-based generalization is in fact optimal in compositional domains like langauge [Arjovsky et al., 2019]), and such patterns might present strong enough learning pressure to overcome – and even reverse – transformers' inherent bias towards exemplar-based generalization from context.

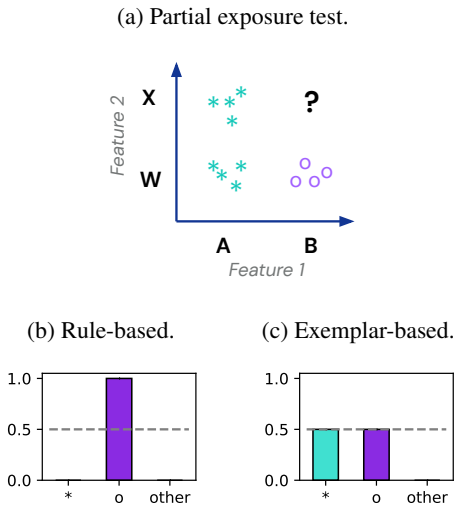


Figure 1: Partial exposure test for differentiating rule-based vs exemplar-based generalization. Stimuli have two features. The model sees three combinations (`AX`, `AW`, and `BW`) in training or in context (depending on experiment), and is evaluated on a held-out (test) combination `BX`. **(b)** A rule-based model uses a parsimonious decision boundary that explains the data (here, based only on Feature 1), classifying the test as `o`. **(c)** An exemplar-based model computes the similarity between test and training examples using all features. Since `BX` is equally similar to `AX` and `BW`, it is equally likely to classify it as `*` or `o`.

2 Experimental Design

We adapted the "partial exposure" paradigm from Dasgupta et al. [2022] where each stimulus has two features; only one of the features predicts the label. We evaluate how the model generalizes to a held-out combination (using sparse rules or similarity to exemplars), see Fig 1.

First, we explored generalization in transformers trained on controlled synthetic data, where we can examine generalization from both in-weights and in-context information and directly compare them. Second, we repeat this experiment on pretrained language models and characterize their in-context generalization. Finally, we compare the patterns observed and investigate factors that explain the differences observed.

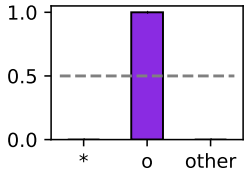
3 Results

3.1 Trained-from-scratch transformers

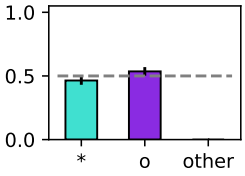
For the trained-from-scratch transformers, we passed sequences of stimulus-label pairs as inputs to the transformer model [Vaswani et al., 2017]. The sequences consisted of two parts: a context (24 tokens; i.e. 12 stimulus-label pairs) and a query (stimulus). The model was trained to minimize a softmax cross-entropy loss on the prediction for the final (query) stimulus. Each stimulus consists of two subvectors concatenated together into a single token (Fig 4c) – these subvectors comprise the two features of the partial exposure paradigm. See Appendix A for further details.

Generalization from in-weights information is rule-based. To investigate generalization from in-weights information, we trained the model on partial exposure data, and evaluated on the held-out combination. During training, the label for each stimulus class was fixed, so that the stimulus-label mappings were stored in weights. The context tokens are uninformative for the query. After training,

(a) From in-weights.



(b) From in-context; few-shot training.



(c) From in-context; rule-based training.

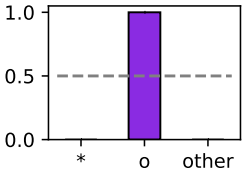


Figure 2: Generalization patterns of transformer models trained on synthetic data: frequency of various model outputs when presented with the held-out stimulus of the partial exposure paradigm (Fig 1). **(a)** Generalization from weights is completely rule-based. **(b)** In contrast, generalization from context is exemplar-based. **(c)** The exemplar-based bias in in-context learning can be overcome by pretraining the model on sequences that explicitly require rule-based generalization.

we measured the model's biases by evaluating on the held-out class combination. See Appendix Fig 4d for details. When trained and evaluated in this way, transformers displayed fully rule-based generalization (Fig 2a), i.e. based on a sparse rule that only took the first feature into account.

Generalization from in-context information is exemplar-based. To investigate generalization from in-context information, we first pretrained the model for few-shot in-context learning (see A.2 for more details), i.e. to refer to information provided in-context when making a query prediction. Importantly, pretraining for few-shot learning imparts no bias towards either rule-based or exemplar-based generalization, because either is an equally valid strategy for solving the few-shot problems. Then we evaluated the model on partial exposure sequences. See Appendix Fig 4b for example sequences used for training and evaluation. When trained and evaluated in this way, transformers displayed totally exemplar-based generalization, in striking contrast to the rule-based generalization that was observed from weights. That is, when queried on the held-out combination `BX`, the models were equally likely to output the labels associated with either `AX` and `BW` (Fig 2b), indicating that they are comparing the query directly (and along all features) to examples that were shown in context.

3.2 Pretrained language models

Do these patterns hold when evaluating on pretrained language models, arguably the most-used transformer-based model at the moment? We used a large (70B parameters) pre-trained language model (LM) trained for autoregressive language prediction on large-scale web data [Hoffmann et al., 2022]. To investigate in-context generalization in this model, we use the same "partial exposure" experimental paradigm, but instead using shape and color words as the two features comprising the stimuli, and nonsense words for the labels (see Fig 5 for an example). We take the completion generated by the LM as the predicted query label.

With the synthetic data, we could ensure no a priori bias towards any feature; we have no such guarantee here. We account for any possible bias toward generalizing along a specific feature with a control condition from [Dasgupta et al., 2022] where neither feature is more predictive in the training data, so the model must use pure exemplar-based generalization. Performance in this condition is now used as the baseline for pure exemplar-based generalization, and deviation from this baseline is the measure for rule-based generalization. See Appendix B for further details. We don't investigate in-weights generalization in these models, since that would require manipulation and control of the training data and large-scale re-training.

In language models, generalization from in-context information is partially rule-based. First, in the control condition where the model is forced to use exemplar-based generalization, we find that the language model prefers generalizing along the color dimension rather than the shape dimension (Fig 3a). Since bias towards shape gives better classification performance on naturalistic visual stimuli [Landau et al., 1988, Geirhos et al., 2018], it is interesting that we see the opposite bias in a system trained on naturalistic text data; future work should look into possible explanations (e.g. the "pragmatic" nature of language; Degen et al. [2019]). The LM predominantly produces one of the two labels provided in context, but does sometimes produce an unseen word ('other' in Fig 3). The LM

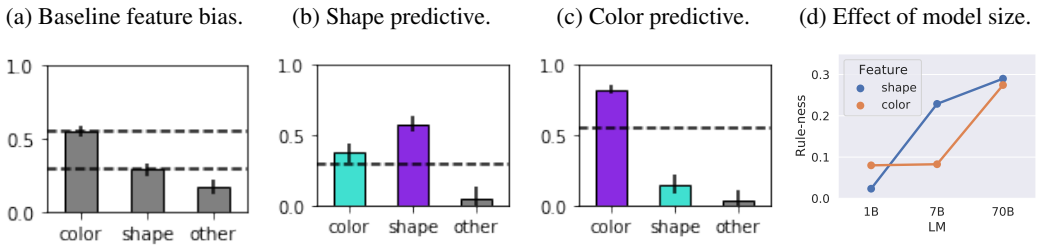


Figure 3: Generalization from in-context information in a pretrained LM. We classify LM responses by whether it gives the label consistent with generalizing along color, shape, or neither. **(a)** Measuring feature-level bias with the Control condition; the model prefers to generalize along color. We use these results as baselines for the partial exposure conditions. **(b)** When a sparse rule-based decision boundary supports shape as predictive, the model classifies along shape more often than in the baseline control (dotted line). **(b)** Similarly when color is predictive, the model classifies along color more often than in the baseline control (dotted line). **(d)** Smaller LMs are less rule-based.

produces an unseen word more frequently when the query is also unseen in-context (not reported), suggesting a mutual exclusivity bias [Gandhi and Lake, 2020], also worth future investigation.

To measure the degree of rule-based generalization, we compared each partial exposure result to the respective control: for instance, we compared the probability of generalizing along color in the color predictive partial exposure evaluation condition (Fig 3c) vs in the control condition. If a model uses pure exemplar-based generalization, there will be no difference in the classification pattern between the partial exposure and the control conditions. Alternatively, rule-based generalization predicts an increased sensitivity to the predictive feature dimension (either shape or color) compared to control. Here, we found evidence of rule-based generalization for both color and shape. This is still in contrast with the purely exemplar-based in-context generalization we saw in our synthetic experiment.

Smaller models are less rule-based. We investigate this further by evaluating language models of different sizes. We measure 'rule-ness' as how much more likely the model is the generalize along the predictive feature (as supported by a sparse rule) in the partial exposure condition, compared to the corresponding (model-specific) control condition. This corresponds to the difference between the purple bar and dotted control in Figs 2 and 3. We find that smaller models (1B and 7B parameters) are steadily less rule-based, with the 1B model effectively performing exact exemplar-based generalization – similar to those trained from scratch (Fig 3d, further details in Appendix 6).

3.3 In-context generalization can be made more rule-based with pre-training.

We did not observe the same effect of scale in transformers that were trained from scratch on synthetic data – increasing number of layers, number of attention heads, and number of classes did not lead to more rule-basedness. This may be because we were not able to achieve the necessary scale with those experiments or due to synergistic effects between scale and the type of training data.

To evaluate the role of training data, we evaluated in-context generalization on a transformer trained on synthetic stimuli where the query explicitly required rule-based classification (more details in Appendix A.4). With this training data, the transformer learns rule-based generalization to held-out sequences 2c. Thus, while transformers exhibit inherent bias towards exemplar-based generalization from context, when trained on data that encourages rule-based generalization from context, they can learn to do so. This supports the interpretation that pretrained language model show rule-based generalization because natural language contains implicitly rule-based data, but crucially, this structure only seems to be picked up and used by larger models.

4 Conclusions

We found distinct patterns of generalization when transformers generalize from information stored in weights vs in context. When trained on synthetic data, generalization from in-weights information is completely rule-based, whereas generalization from in-context information is almost entirely exemplar-based. However, a pre-trained language model is surprisingly rule-based when generalizing

from in-context information, and this is increasingly true for larger models. We find that it is indeed possible to induce rule-based generalization from in-context information by pretraining a transformer on an explicitly rule-based classification problem. Together, these findings support the possibility that natural language data (perhaps because of its combinatorial nature) provides a strong learning pressure towards rule-like generalization, which works in concert with model scale.

References

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

F. Gregory Ashby and James T. Townsend. Varieties of perceptual independence. *Psychological Review*, 93:154–179, 1986. ISSN 1939-1471. doi: 10.1037/0033-295X.93.2.154. Place: US Publisher: American Psychological Association.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. *arXiv:2005.14165 [cs]*, July 2020. URL `http://arxiv.org/abs/2005.14165`. arXiv: 2005.14165.

Stephanie C. Y. Chan, Adam Santoro, Andrew K. Lampinen, Jane X. Wang, Aaditya Singh, Pierre H. Richemond, Jay McClelland, and Felix Hill. Data Distributional Properties Drive Emergent In-Context Learning in Transformers, May 2022. URL `http://arxiv.org/abs/2205.05055`. arXiv:2205.05055 [cs].

Ishita Dasgupta, Erin Grant, and Tom Griffiths. Distinguishing rule and exemplar-based generalization in learning systems. In *Proceedings of the 39th International Conference on Machine Learning*, pages 4816–4830. PMLR, June 2022. URL `https://proceedings.mlr.press/v162/dasgupta22b.html`. ISSN: 2640-3498.

Judith Degen, Robert D. Hawkins, Caroline Graf, Elisa Kreiss, and Noah D. Goodman. When redundancy is useful: A Bayesian approach to 'overinformative' referring expressions, December 2019. URL `http://arxiv.org/abs/1903.08237`. arXiv:1903.08237 [cs].

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 954–959, 2020.

Kanishk Gandhi and Brenden M Lake. Mutual exclusivity as a challenge for deep neural networks. *Advances in Neural Information Processing Systems*, 33:14182–14192, 2020.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and L. Sifre. Training compute-optimal large language models. *ArXiv*, abs/2203.15556, 2022.

Barbara Landau, Linda B Smith, and Susan S Jones. The importance of shape in early lexical learning. *Cognitive development*, 3(3):299–321, 1988.

Brian Lester, Rami Al-Rfou, and Noah Constant. The Power of Scale for Parameter-Efficient Prompt Tuning, September 2021. URL `http://arxiv.org/abs/2104.08691`. arXiv:2104.08691 [cs].

Xiang Lisa Li and Percy Liang. Prefix-Tuning: Optimizing Continuous Prompts for Generation. *arXiv:2101.00190 [cs]*, January 2021. URL `http://arxiv.org/abs/2101.00190`. arXiv: 2101.00190.

Roger N. Shepard and Jih-Jie Chang. Stimulus generalization in the learning of classifications. *Journal of Experimental Psychology*, 65:94–102, 1963. ISSN 0022-1015. doi: 10.1037/h0043732. Place: US Publisher: American Psychological Association.

Tianxiang Sun, Yunfan Shao, Hong Qian, Xuanjing Huang, and Xipeng Qiu. Black-Box Tuning for Language-Model-as-a-Service. In *Proceedings of the 39th International Conference on Machine Learning*, pages 20841–20855. PMLR, June 2022. URL `https://proceedings.mlr.press/v162/sun22e.html`. ISSN: 2640-3498.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. *arXiv:1706.03762 [cs]*, page 11, 2017.

A Experiment details: Trained-from-scratch transformers

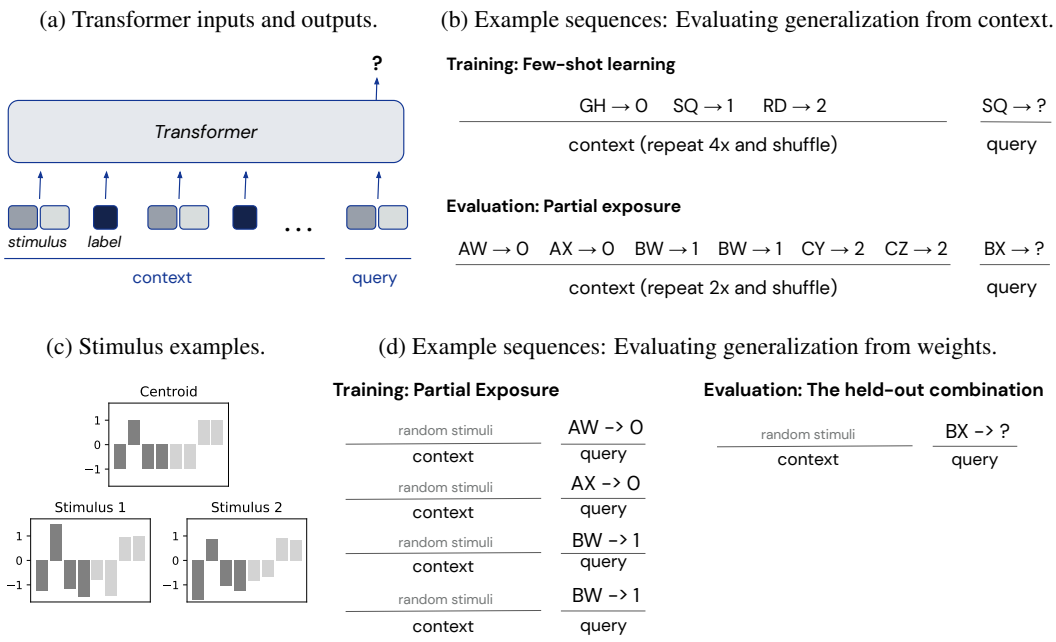


Figure 4: **(a)** Sequences of alternating stimuli and labels are passed to a transformer. Each sequence consists of a "context" (12 stimulus-label pairs) and a "query" stimulus. The model is trained to minimize the loss on the query prediction. **(b)** To evaluate generalization from context, the model is first pretrained to perform in-context learning by training on few-shot sequences; stimulus classes and labels are randomly chosen for every sequence, so that the model must perform few-shot learning from context. Inductive biases are evaluated on "partial exposure" sequences, where one combination is held out for evaluation ("BX"). Consistent selection of the label associated with "B" indicates a rule-based bias, while equal selection of the labels associated with "A" and "B" indicates an exemplar-based bias (since "BX" is equally similar to "AX" and "BW"). **(c)** Each stimulus consists of two subvectors concatenated together into a single token. Each subvector belongs a particular class, and each class is characterized by a different centroid. The subvectors are sampled from a multivariate normal centered on that centroid. The subvectors have length 32, but here we only show 4 values. **(d)** To evaluate generalization from weights, the model is instead trained directly on partial exposure data, and inductive biases are evaluated on the held-out combination. (The context consisted of random samplings of the stimulus classes, and were irrelevant to the query prediction.)

A.1 Subvector stimuli

Each subvector belongs a particular "subvector class", and each subvector class is characterized by a different centroid. The subvectors are sampled from a multivariate normal centered on that centroid. A "stimulus class" is the concatenation of two subvector classes, and a stimulus is a sampling from a

stimulus class. Example stimuli are shown in Fig 4. This design is a 64-dimensional generalization of the 2-D classification example from Dasgupta et al. [2022], and ensures that there is no a priori bias towards one feature or another.

Number of features per stimulus: 2 | Feature length: 32 | Number of classes per feature: 10 | Number of values per class: 100 | Covariance scaling on the per-class normal distributions: 0.1

A.2 Pretraining for few-shot learning

To pretrain the model for in-context learning, we pretrained the model on few-shot (4-shot 3-way) sequences, i.e. sequences for which the context consisted of 3 different stimulus classes each repeated 4 times, and where the query class was one of those 3 classes. The classes and labels were randomly assigned for each sequence.

A.3 Evaluating inductive biases

Models were trained and evaluated as shown in Figs 4b and 4d.

An additional label ("2", in the example in Fig 4b) and corresponding examples were included in the partial exposure sequences, to ensure that if the model is equally likely to select the labels associated with "A" and "B", it is because of those examples' similarity to the query stimulus, rather than because the model is selecting labels at chance. This was similarly done for training on partial exposure data in weights (not shown in Fig 4d).

Note also that the `BW` stimulus was shown twice as often as other stimuli in the partial exposure data, as was done in Dasgupta et al. [2022], in order to ensure that there is no bias induced by having one label more frequent than another. We also ran experiments where `BW` was not repeated, and the pattern of results was the same.

A.4 Pretraining for rule-based generalization

To pretrain the model for rule-based generalization, we used the partial exposure sequences shown in 4b, but for training *as well as* for evaluation. In training, the label for the query `BX` was always the same as the label associated with `BW` in context (note that the stimuli and labels are randomly assigned, so that `BX` could be manifested as any of the stimulus classes, and the query label could be any of [0, 1, 2].)

A.5 Architecture, training, and evaluation details

Num layers: 12 | Embedding size: 64 | Optimizer: Adam | Batch size: 32 | Learning rate schedule: Linear warmup and square root decay, described as `min(3e-4 / 4000 * global_step, power(4000, 0.5) * 3e-4 * power(global_step, -0.5))`

For each experiment, we ran 16 TPUv3 cores and 4 v100 GPU cores for 200,000 training steps.

Models are evaluated on evaluation data throughout training, and bar plots in Fig 2 show evaluation outputs averaged over the last half of training (100k-200k steps). Error bars indicate 1.96 standard errors across 10 training runs (there is zero variance for the results in Figs 2a and 2c).

B Experiment details: Large language model experiments

See Fig 5 for example evaluation sequences.

In order to account for potential low-level word-order effects, we evaluated on four different formats for the stimuli ('red circle', 'a circle that is red', 'an object that is circular and red', 'an object that is red and circular'). We found no significant differences in qualitative patterns across the different word orderings, so we report the average across all four formats in our results.

We also include more detailed information about the experiments using different model sizes in Fig 6. These show the raw performances (including model-specific controls) on all model sizes and feature sets.

Evaluation: Partial exposure (color predictive)

a red triangle is hib, a blue square is fep, a red square is hib	a blue triangle is ?
context (repeat 6x and shuffle)	**query**

Evaluation: Partial exposure (shape predictive)

a red triangle is hib, a blue square is fep, a red square is fep	a blue triangle is ?
context (repeat 6x and shuffle)	**query**

Evaluation: Control

a red triangle is hib, a blue square is fep	a blue triangle is ?
context (repeat 6x and shuffle)	**query**

Figure 5: To evaluate generalization from context in a pretrained language model, the model is evaluated on partial exposure sequences where the features are instead text features (shape and color words). The control condition allows us to evaluate the model's baseline bias towards shape or color.

(a) Control condition. (b) Shape predictive. (c) Color predictive.

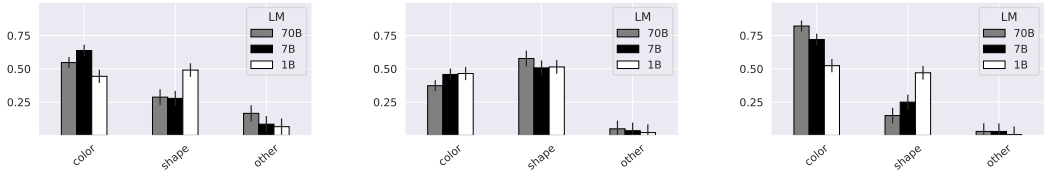


Figure 6: Results on the pretrained language models for different sizes.