DeepMind

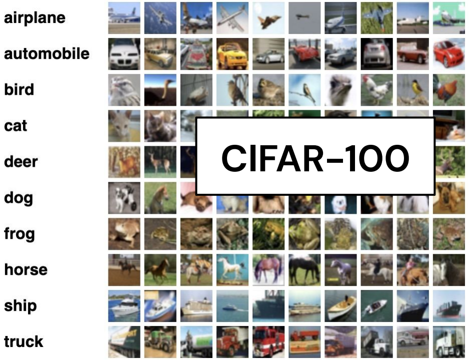
Data Distributional Properties Drive Emergent In-Context Learning in Transformers

Stephanie Chan, Adam Santoro, Andrew Lampinen, Jane Wang, Aaditya Singh, Pierre Richemond, Jay McClelland, Felix Hill

Two distinct modes of learning in neural networks

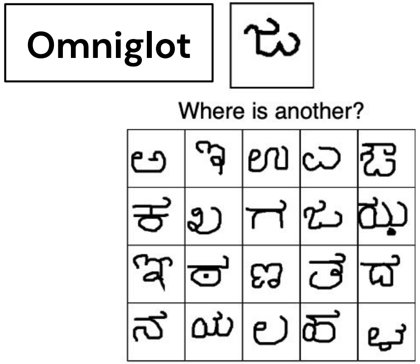
In-weights learning

- gradient-based
- slow: needs many examples
- standard supervised learning

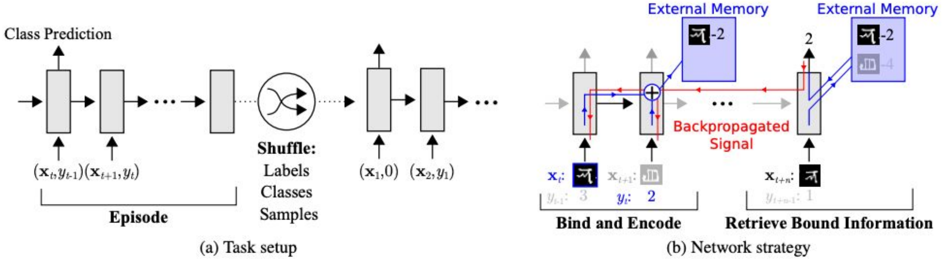


In-context learning

- no gradient updates
- rapid: from a few examples
- few-shot / one-shot learning



Normally, only happens if we **explicitly meta-train** for it:



E.g.: Santoro et al, 2016; Vinyals et al, 2016; Wang et al, 2016

But in-context (few-shot) learning can also *emerge* …when trained on a very different objective!

Language Models are Few-Shot Learners

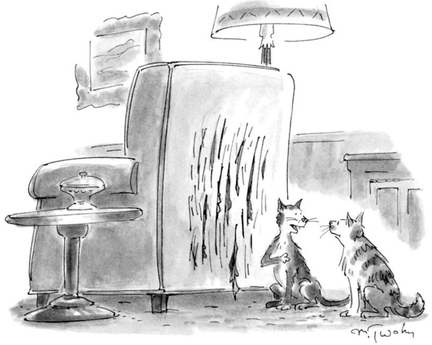
GPT-3

```
sea otter => loutre de mer
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese => ...........................
```

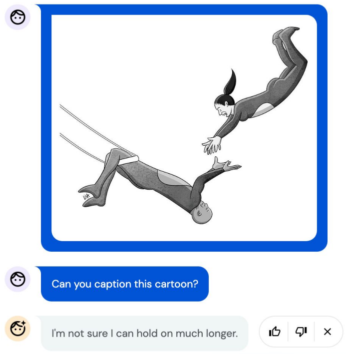
examples

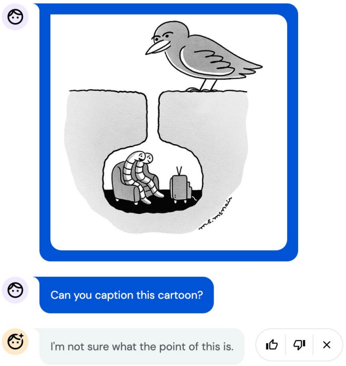
prompt

Flamingo



"I have a couple of other projects I'm excited about."

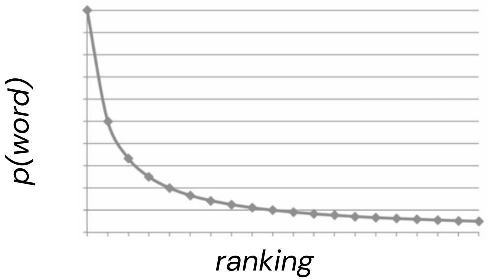


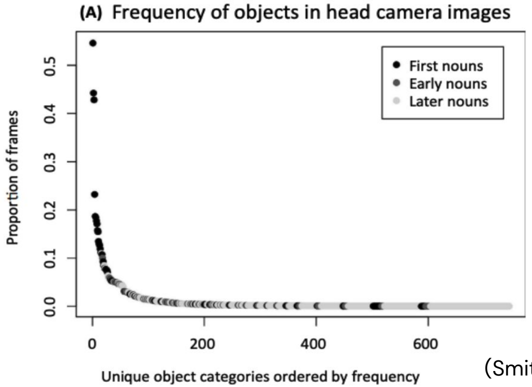


How do large transformer models achieve *emergent* in-context learning?

Hypothesis: Maybe it's because the **distributions of naturalistic data** have special properties

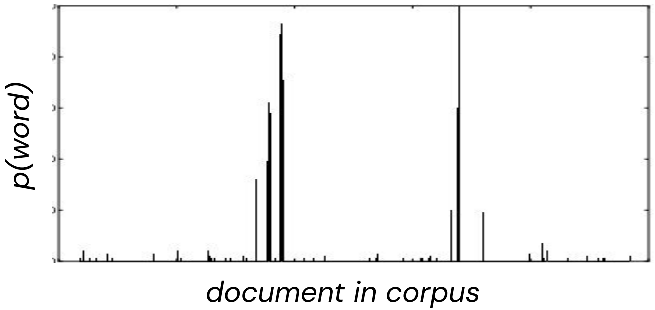
Natural data is long-tailed





(Smith et al, 2018)

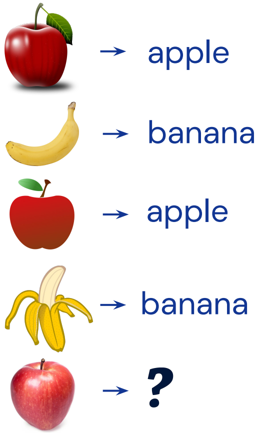
Natural data is bursty



Maybe training on naturalistic data is like an interpolation between supervised and few-shot meta-training...

Standard Supervised

- items recur and are uniform
- label mappings are fixed



Naturalistic data (e.g. language)

- words do recur
- word meanings are somewhat fixed

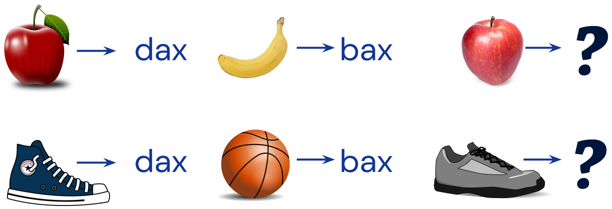
but also:

- rare words do not recur often
- some rare words are bursty
- many-to-many relationships

Few-shot meta-training

explicitly train for few-shot learning

- items differ on every episode
- label mappings are only fixed within episodes



few-shot learning emerges,
even without explicit training

Our Project

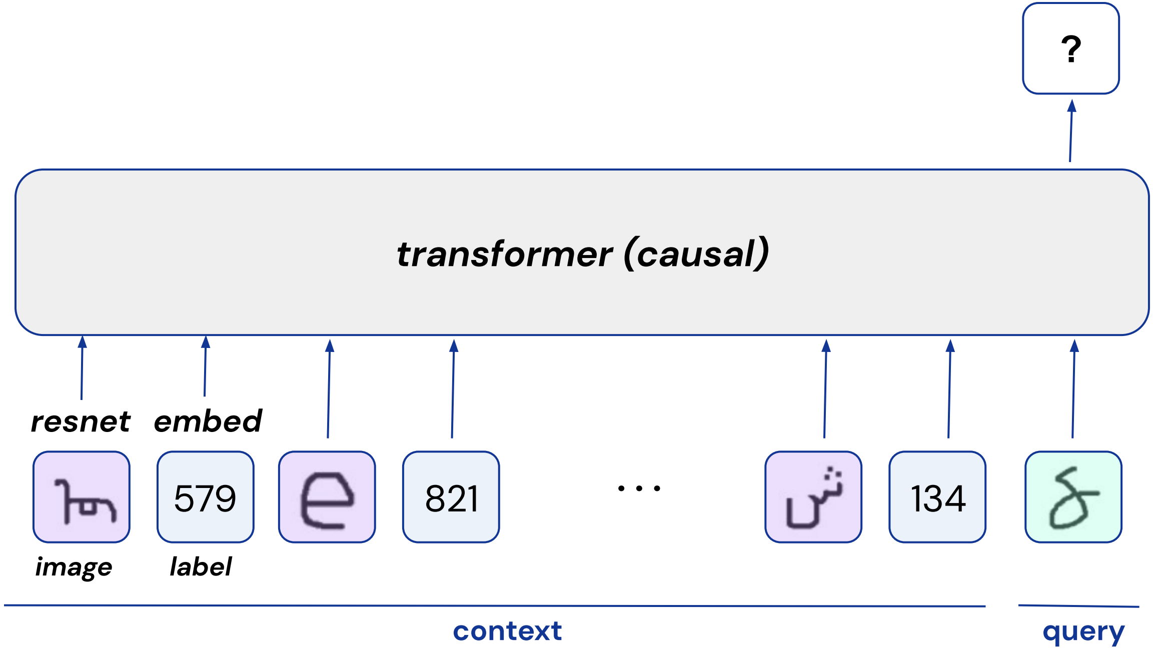
Hypothesis: Certain non-uniformities in data distributions can lead to emergent few-shot (in-context) learning, and this is a general phenomenon.

Experiments: Modify a standard few-shot learning image dataset (Omniglot), to control these distributional properties and measure their effects on few-shot learning.

Implications:

- understanding how we might design or collect datasets to achieve in-context learning in domains outside of language

General structure of the experiments



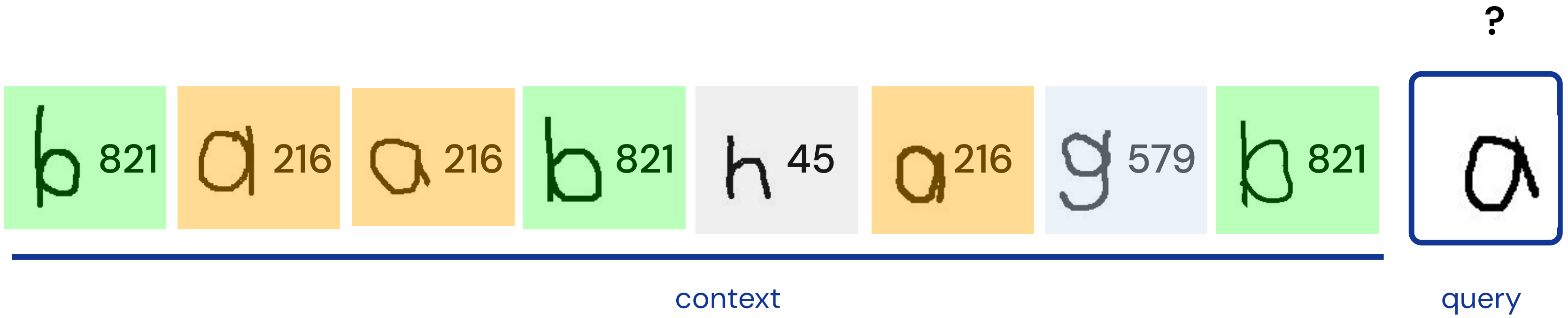
Training data

- labels are fixed across all of training

Two ways to solve:

1. In-weights memorization
2. In-context learning

Example "bursty" sequence

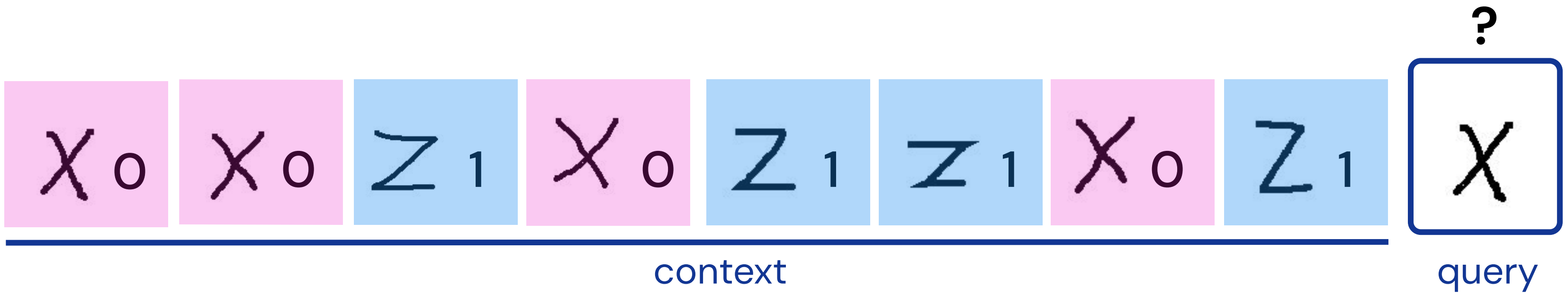


Evaluation data

Example evaluation sequence for <u>in-context learning</u>

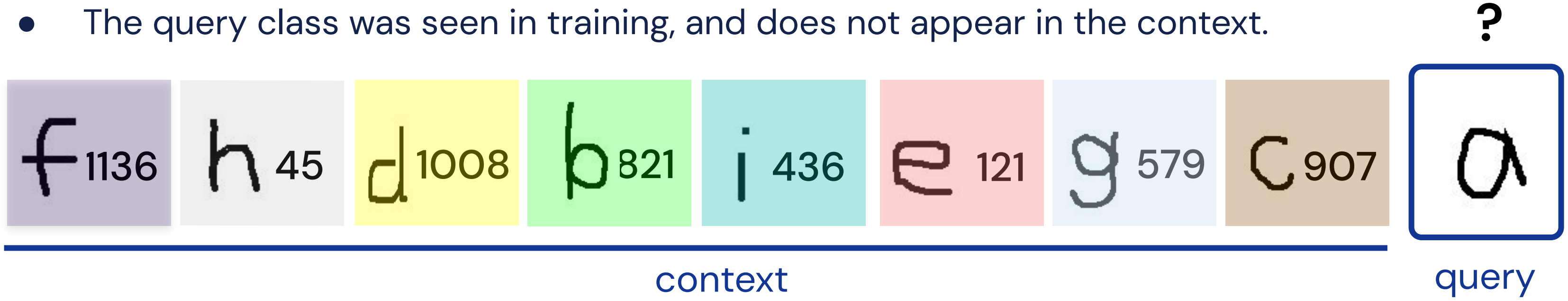
- Two holdout classes, randomly assigned to labels [0, 1]

Compositional binding of images and labels



Example evaluation sequence for <u>in-weights memorization</u>

- The query class was seen in training, and does not appear in the context.



DeepMind

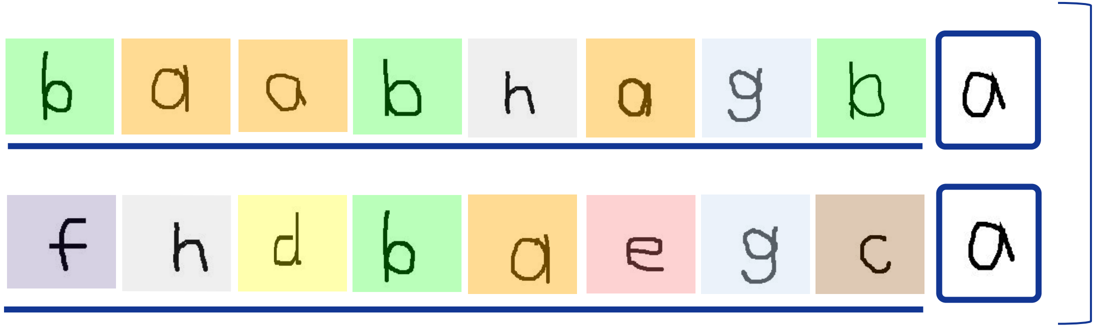
What kinds of training data promote in-context learning?

Importance of burstiness in the data

Train

bursty

non-bursty



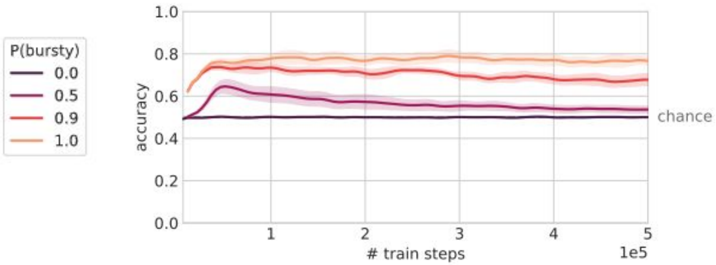
context

query

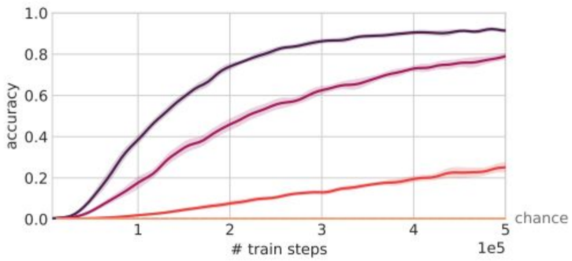
vary the proportions in the training data

Eval

In-context learning on holdout classes.



In-weights learning on trained classes.

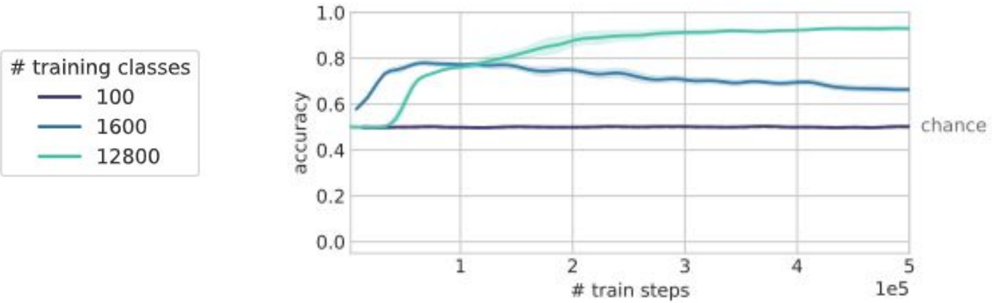


➔ **More burstiness leads to better in-context learning**

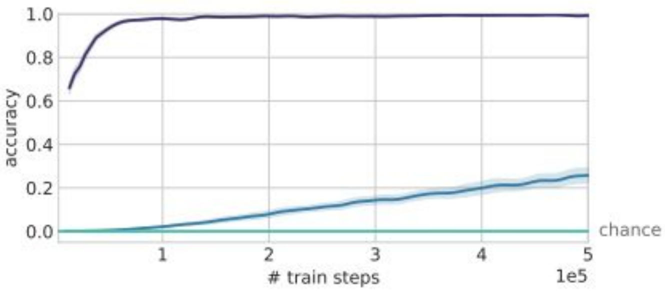
➔ **In-context learning trades off against weights-based learning**

Transformers succeed at the Omniglot challenge: Importance of number of classes in the data

In-context learning on holdout classes.



(b) In-weights learning on trained classes.



➔ More training classes leads to better in-context learning

➔ Again, in-context learning trades off against weights-based learning

Other natural data-inspired distributional properties

Dynamic meaning:

Multiplicity of item-label mappings

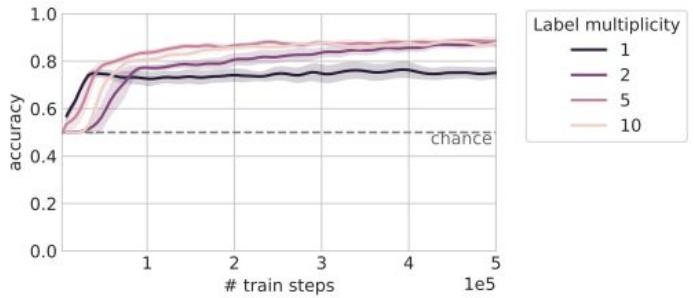
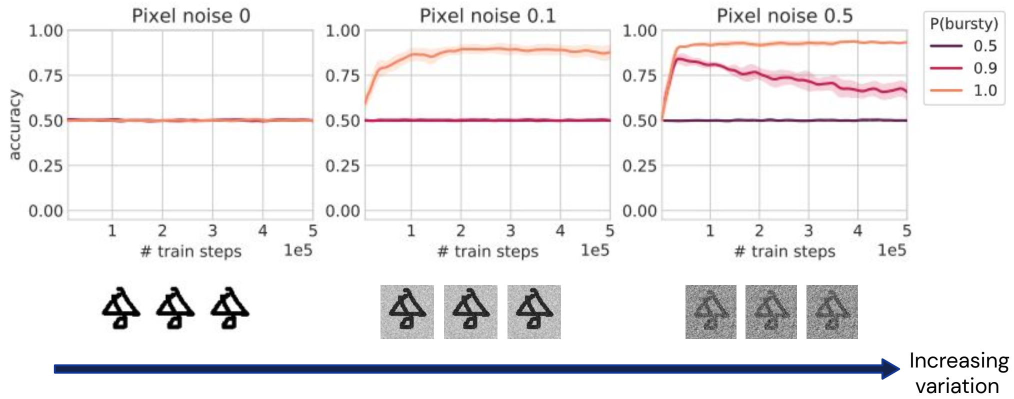


Figure 4: Dynamic meanings improve in-context learning. Increasing the number of labels per class ('label multiplicity') increases in-context learning.

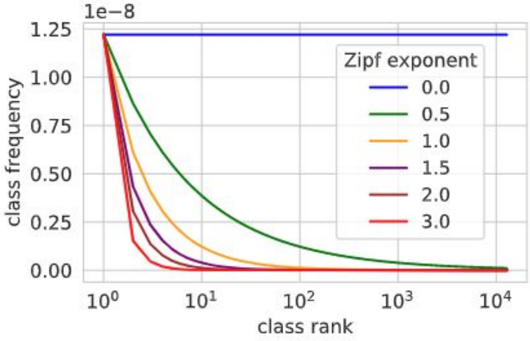
Within-class variation



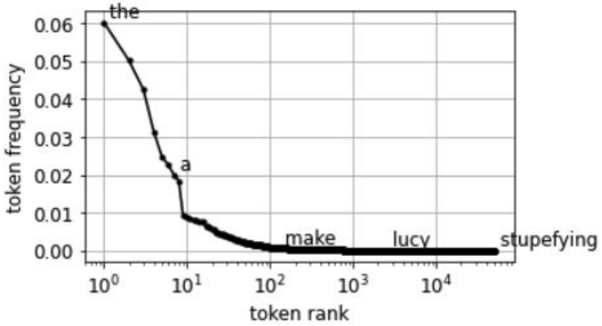
Can in-weights memorization and in-context learning *co-exist* in the same model?

We can achieve both kinds of learning when we train on *skewed* distributions.

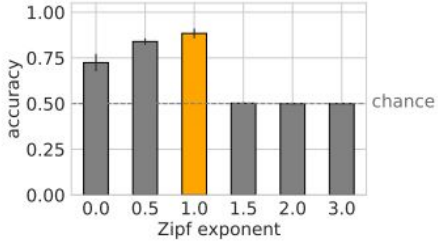
Examples of Zipfian distributions. $p(X = x) \propto \frac{1}{x^{\alpha}}$



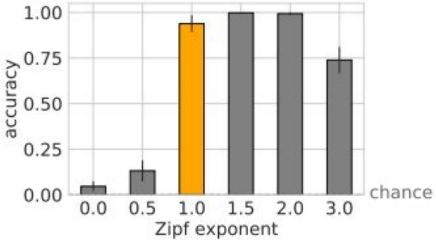
Distribution of tokens in a natural language corpus.



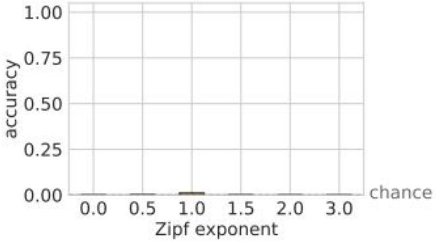
In-context learning on holdout classes.



In-weights learning on common classes.



In-weights learning on rare classes.



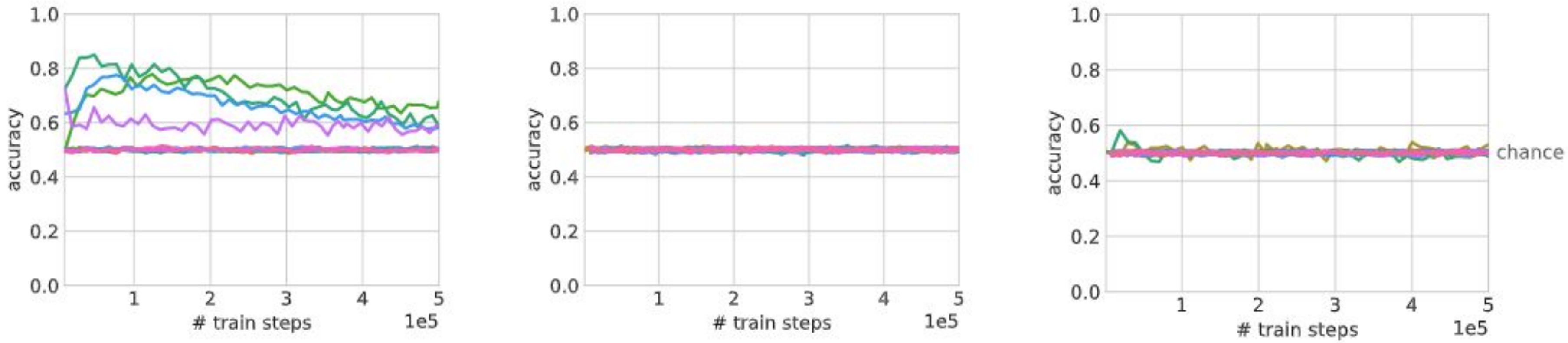
- ➔ There is a sweet spot at Zipf exponent = 1, where we attain both few-shot learning and in-weights memorization
- ➔ Intriguingly, Zipf exponent 1 corresponds approximately to the skew in natural languages

But architecture does matter too…

Transformers vs RNNs

In-context learning on holdout classes.

(a) Transformer. (b) Vanilla RNN. (c) LSTM.



- each line is a hyperparameter setting
- matched on: # params, # layers, hidden size, training data

➔ **Recurrent models never achieve few-shot learning, with the same training data**

➔ **But even though architecture matters, it's not enough – we need the right data, too**

DeepMind

Conclusions

Implications for compositionality

We study in-context few-shot learning, which can be construed as a a narrow instantiation of compositionality.

Our findings on the drivers of this emergent behavior:

- Large-scale data and models are not necessary
- Certain **distributions of training data** promote it
  - these distributional features are present in natural data like language
- **Architecture** matters too
  - Transformers > RNNs

- Andreas 2020
- Akyürek & Andreas, 2022

Implications for compositionality

We study in-context few-shot learning, which can be construed as a a narrow instantiation of compositionality.

Our findings on the drivers of this emergent behavior:

- Large-scale data and models are not necessary
- Certain **distributions of training data** promote it
  - these distributional features are present in natural data like language
- **Architecture** matters too
  - Transformers > RNNs

RNNs with memory augmentation *could* perform well on SCAN (Lake et al 2019)

- memory-augmented NNs help meta-learning because they have both long-term and short-term storage (that is stable + element-wise addressable) (Santoro et al 2013)

Transformers can perform well on certain kinds of composition

- transformers have the desired properties as well
- transformers may perform compositional operations (Elhage et al 2021; Olsson et al 2022)

Implications for compositionality

We study in-context few-shot learning, which can be construed as a a narrow instantiation of compositionality.

Our findings on the drivers of this emergent behavior:

- Large-scale data and models are not necessary
- Certain **distributions of training data** promote it
  - these distributional features are present in natural data like language
- **Architecture** matters too
  - Transformers > RNNs

We need both!

Non-uniformity is important