META-LEARNING GENERAL-PURPOSE LEARNING ALGORITHMS WITH TRANSFORMERS

Presenter: Mert Kilickaya, **Paper**: Link, ICLR'23 Submission
Louis Kirsch et al. (slide set)

RQ: What phase transitions do black-box models undergo while achieving meta-learning?

Phases: Memorization to Learning to Meta Learning

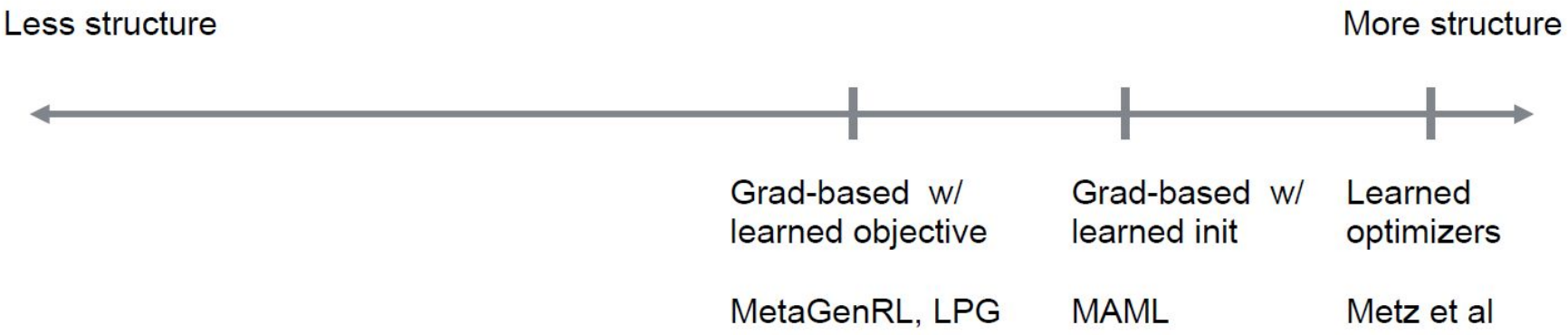
Black-box models: RNN, LSTM, Transformers

How did meta learning generalize so far?

With algorithmic / architectural structure at meta-test time!

How did meta learning generalize so far?

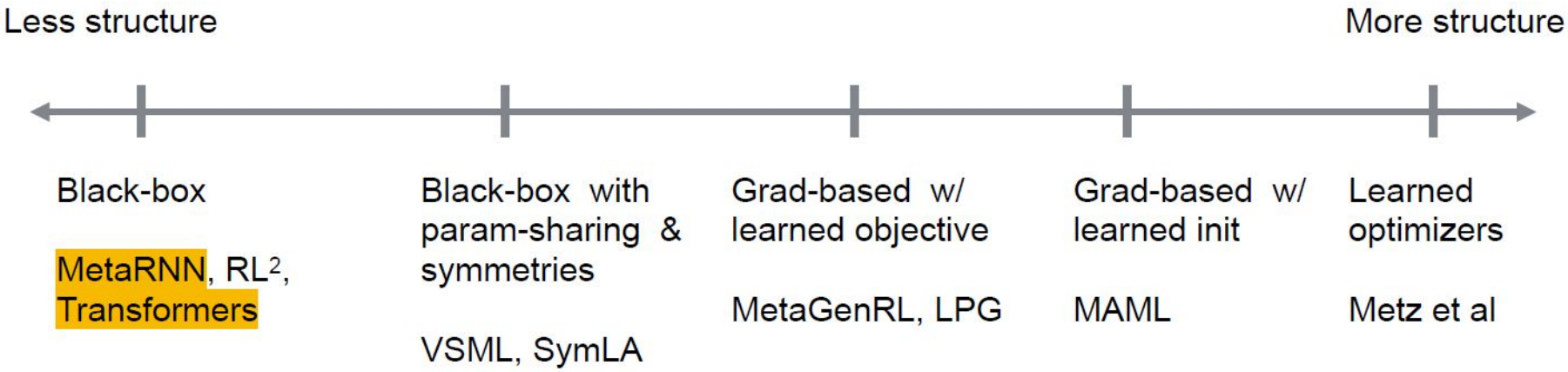
With algorithmic / architectural structure at meta-test time!



Google Research

Can we do this data-driven instead?

Using the data distribution to enable generalization!



What is an In-Context Learning Algorithm?

In supervised learning $\left(\{x_i, y_i\}_{i=1}^{N_D}, x' \right) \mapsto y'$

Learning = Improving predictions y' with larger $D = \{x_i, y_i\}_{i=1}^{N_D}$

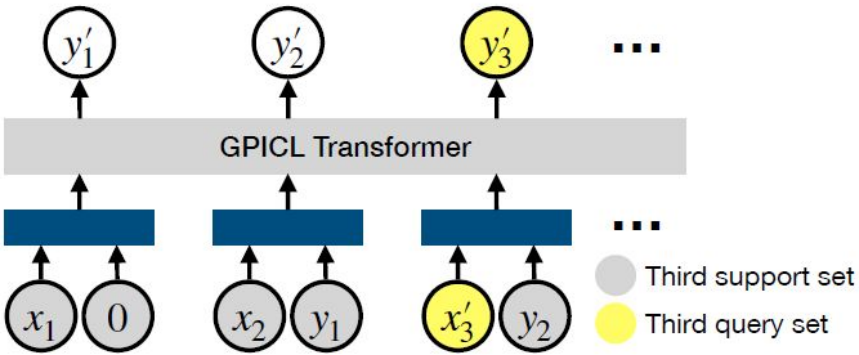
With black-box models such as LSTMs or **Transformers**

What is an In-Context Learning Algorithm?

In supervised learning $\left(\{x_i, y_i\}_{i=1}^{N_D}, x' \right) \mapsto y'$

Learning = Improving predictions y' with larger $D = \{x_i, y_i\}_{i=1}^{N_D}$

With black-box models such as LSTMs or **Transformers**

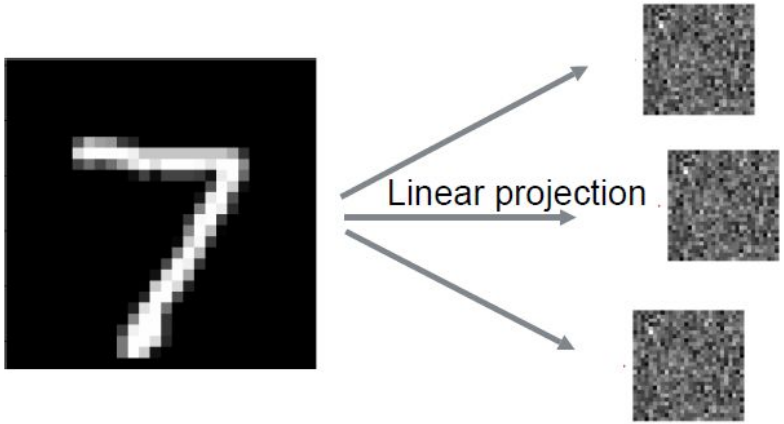


Generating Tasks for Learning-To-Learn

MNIST dataset

$$\bar{D} = \{\bar{x}_i, \bar{y}_i\}$$

Generating Tasks for Learning-To-Learn



MNIST dataset

$$\bar{D} = \{\bar{x}_i, \bar{y}_i\}$$

Create n tasks

$$D = \{A\bar{x}_i, \rho(\bar{y}_i)\} \quad A_{ij} \sim \mathcal{N}\left(0, \frac{1}{N_x}\right)$$

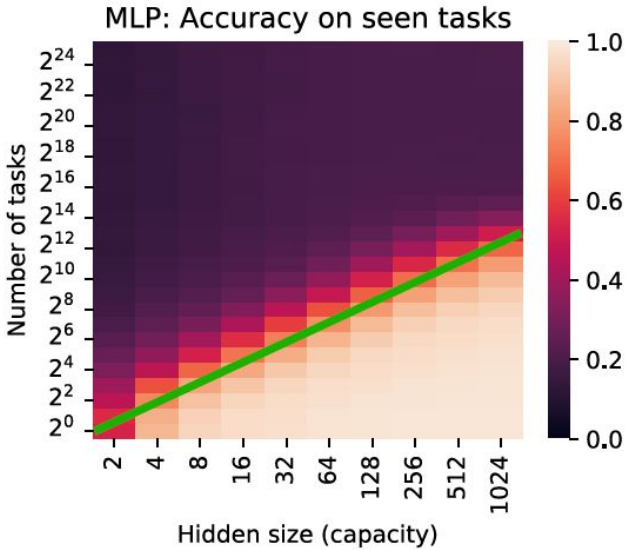
Linear projection

Label Permutation

Label $\mapsto$ one-hot index

Large Sequence Models and Data

MLP: $x' \mapsto y'$

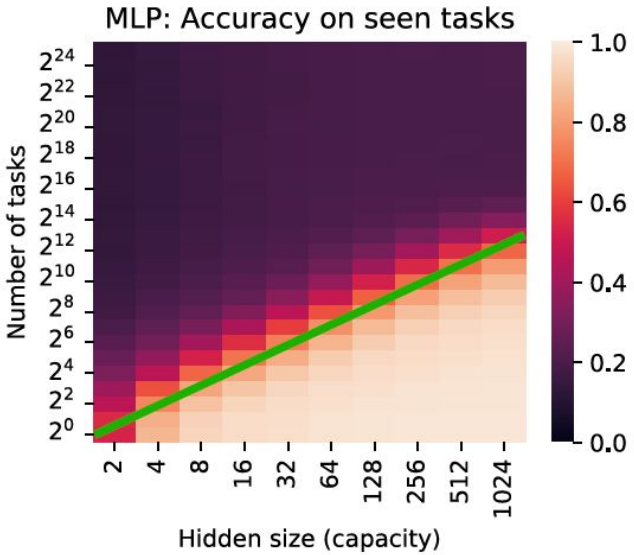


Large Sequence Models and Data

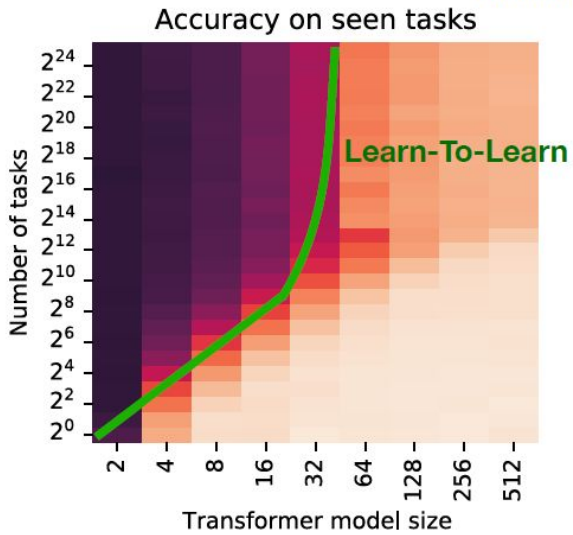
MLP: $x' \mapsto y'$

Each element in the sequence is from the same task (projection)

Transformer: $\left(\{x_i, y_i\}_{i=1}^{N_D}, x'\right) \mapsto y'$



Transformer



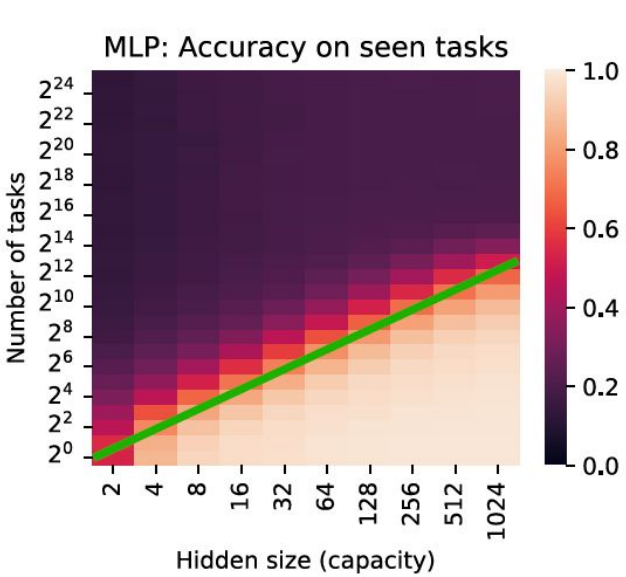
At a certain model size and number of tasks, the Transformer generalizes to a seemingly unbounded number of tasks.

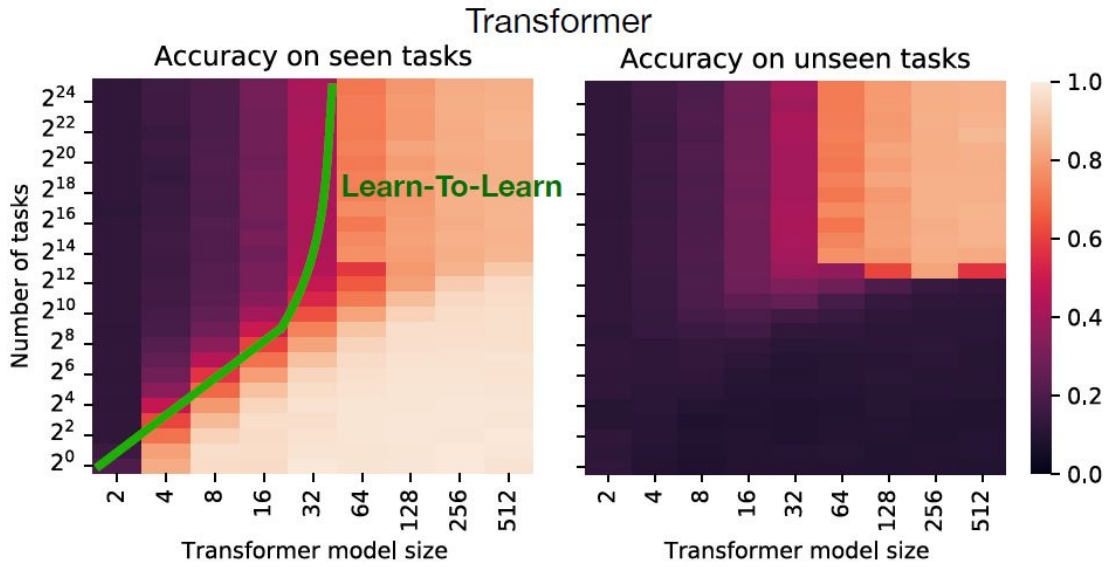
Large Sequence Models and Data

MLP: $x' \mapsto y'$

Each element in the sequence is from the same task (projection)

Transformer: $\left(\{x_i, y_i\}_{i=1}^{N_D}, x'\right) \mapsto y'$





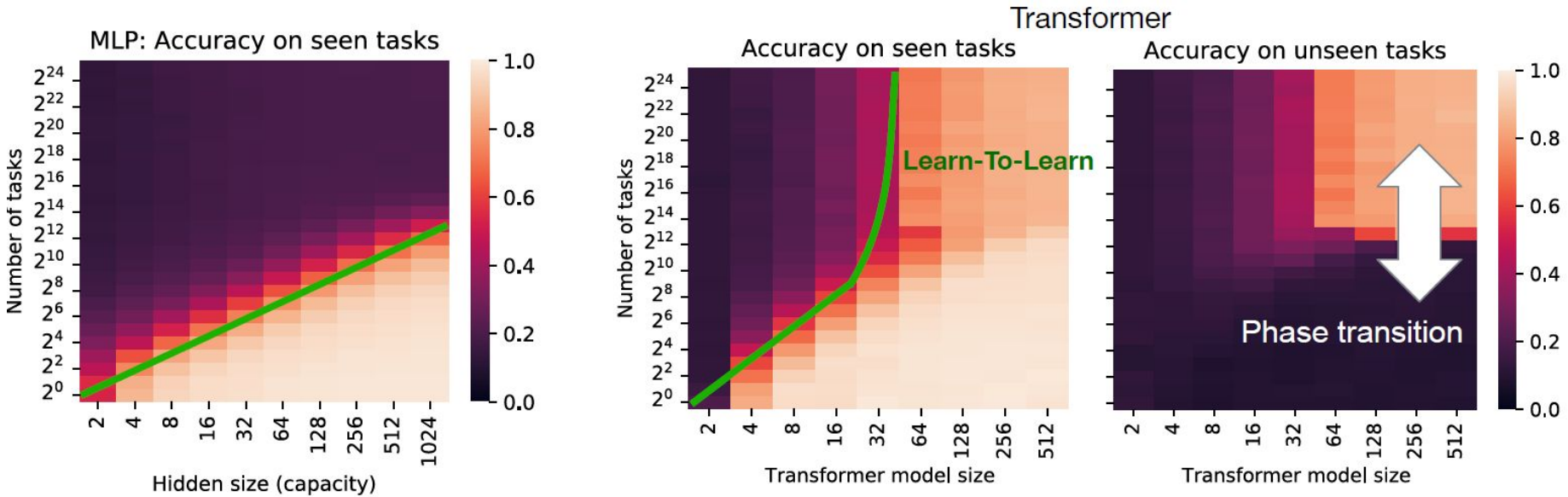
At a certain model size and number of tasks, the Transformer generalizes to a seemingly unbounded number of tasks.

Large Sequence Models and Data

MLP: $x' \mapsto y'$

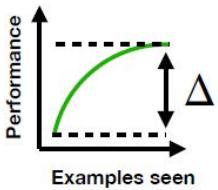
Each element in the sequence is from the same task (projection)

Transformer: $\left(\{x_i, y_i\}_{i=1}^{N_D}, x'\right) \mapsto y'$



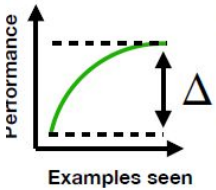
At a certain model size and number of tasks, the Transformer generalizes to a seemingly unbounded number of tasks.

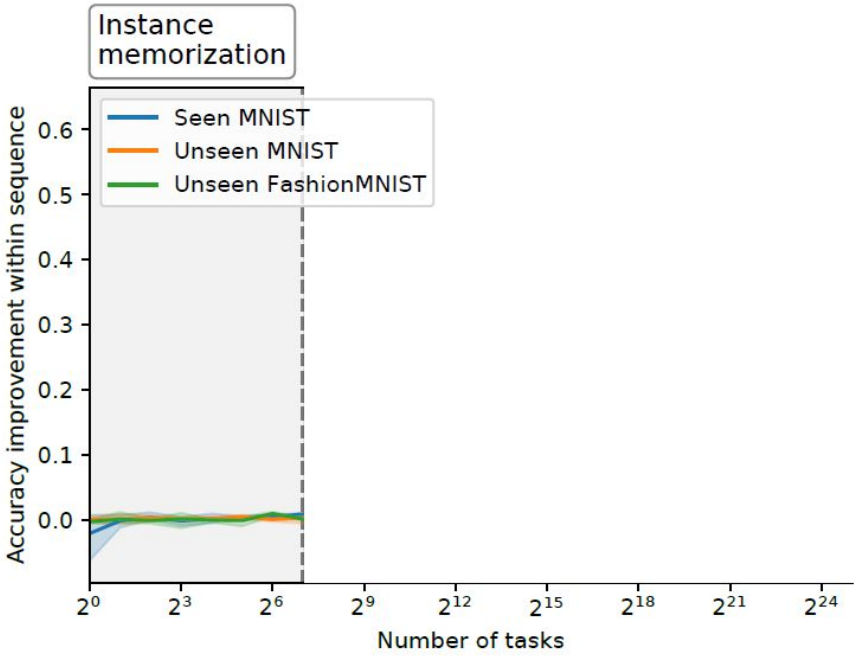
Transitioning from Memorization to Learning



Learning	Generalization	Algorithm Description

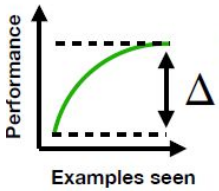
Transitioning from Memorization to Learning

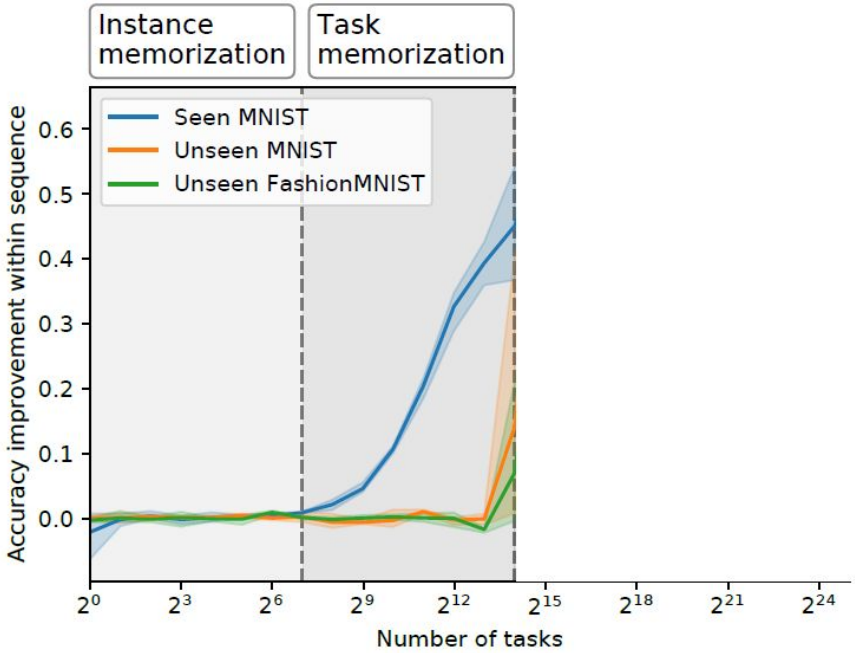




Learning	Generalization	Algorithm Description
× No	× No	Instance memorization

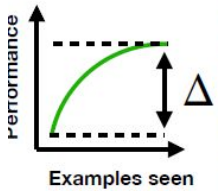
Transitioning from Memorization to Learning

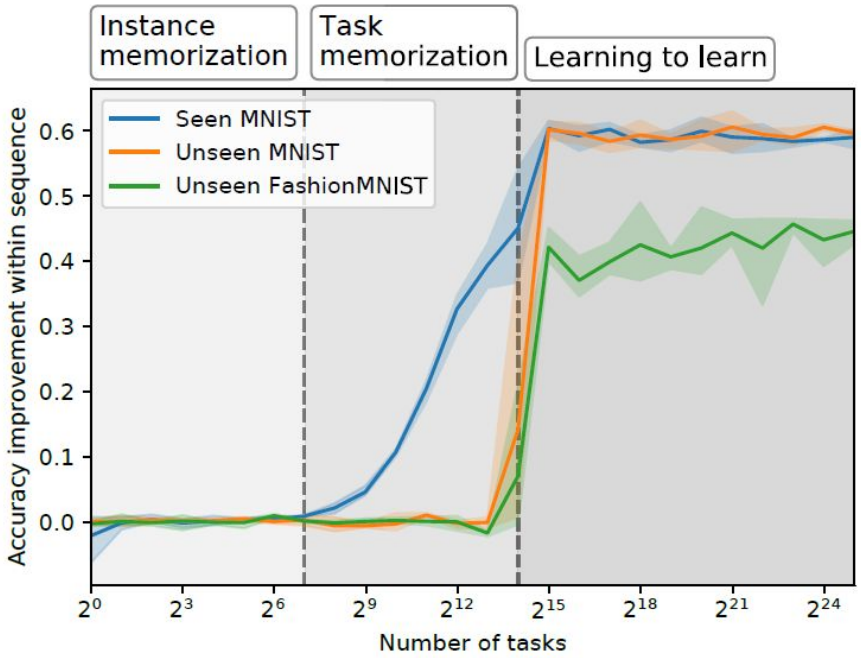




Learning	Generalization	Algorithm Description
× No	× No	Instance memorization
✓ Yes	× No	System identification / Task memorization

Transitioning from Memorization to Learning

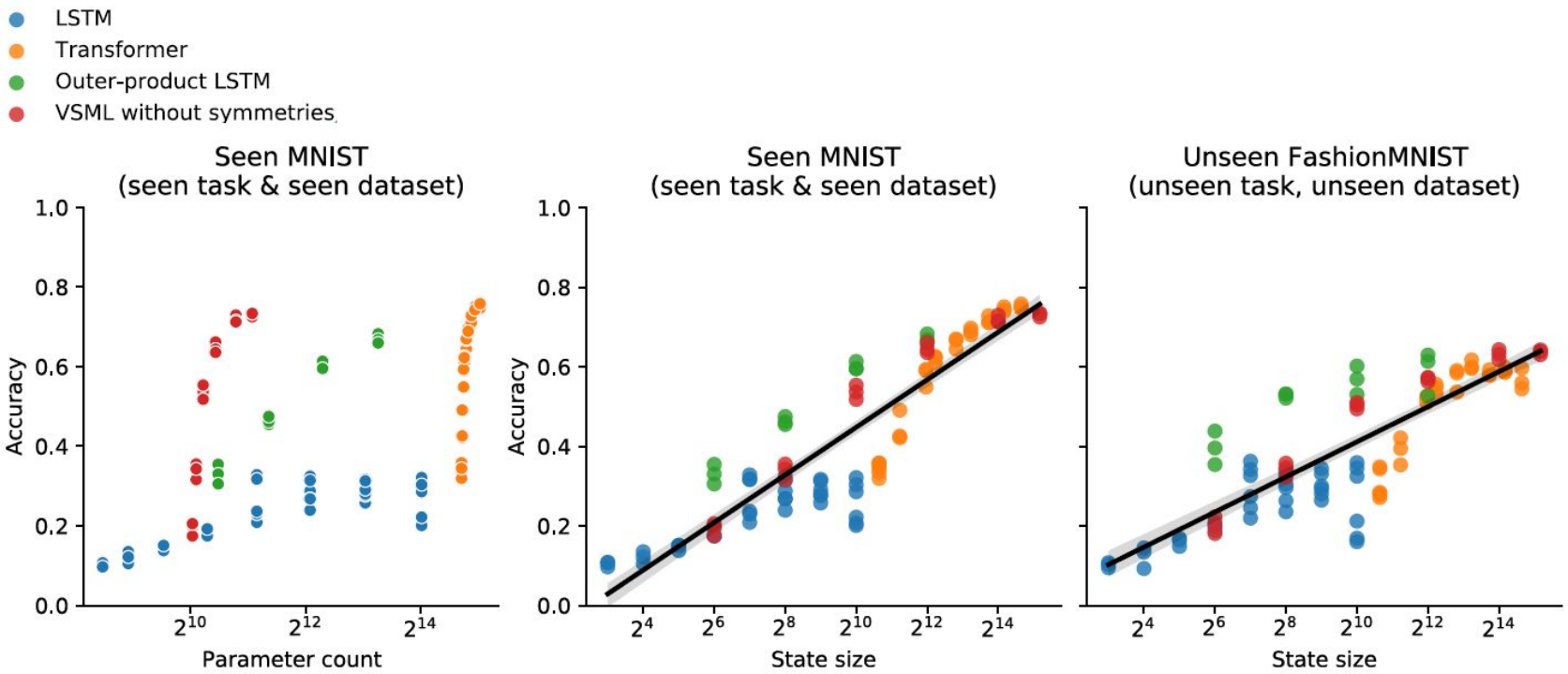




Learning	Generalization	Algorithm Description
× No	× No	Instance memorization
✓ Yes	× No	System identification / Task memorization
✓ Yes	✓ Yes	General-purpose learning algorithm

Transformers exhibit three different phases in terms of meta-learned behavior.

Architecture: A Large State is Crucial for Learning



The state size (accessible memory) of an architecture most strongly predicts its performance as a general-purpose learning algorithm.

Google Research

Task generation beyond random projections?

- Generating tasks from scratch
- Other ways of augmenting existing data
- Create huge datasets of naturally existing tasks
  - We need in the order of $2^{13} = 8192$ tasks

Summary

Transformers can discover general-purpose learning algorithms from the data.

Generalization emerges in three steps: Memorize instance, Memorize task, Generalize.

Parameters of Generalization: Number of tasks, Size of batch, Dimensionality of embedding.

Reviews: 8-6-5

(+) *Very interesting, purely empirical study of learning dynamics of black-box meta-learning.*

(+) *Findings: i) Existence of phase shifts in learning, ii) Learning is controlled by embedding size (rather than model size)*

(-) *Findings may be specific to the paper's setup, and not generic.*

(-) *Why not use a large-scale dataset like ImageNet with abundant tasks (i.e. 15k)?*