Guest Lecture, CSCI 3370: Deep Learning

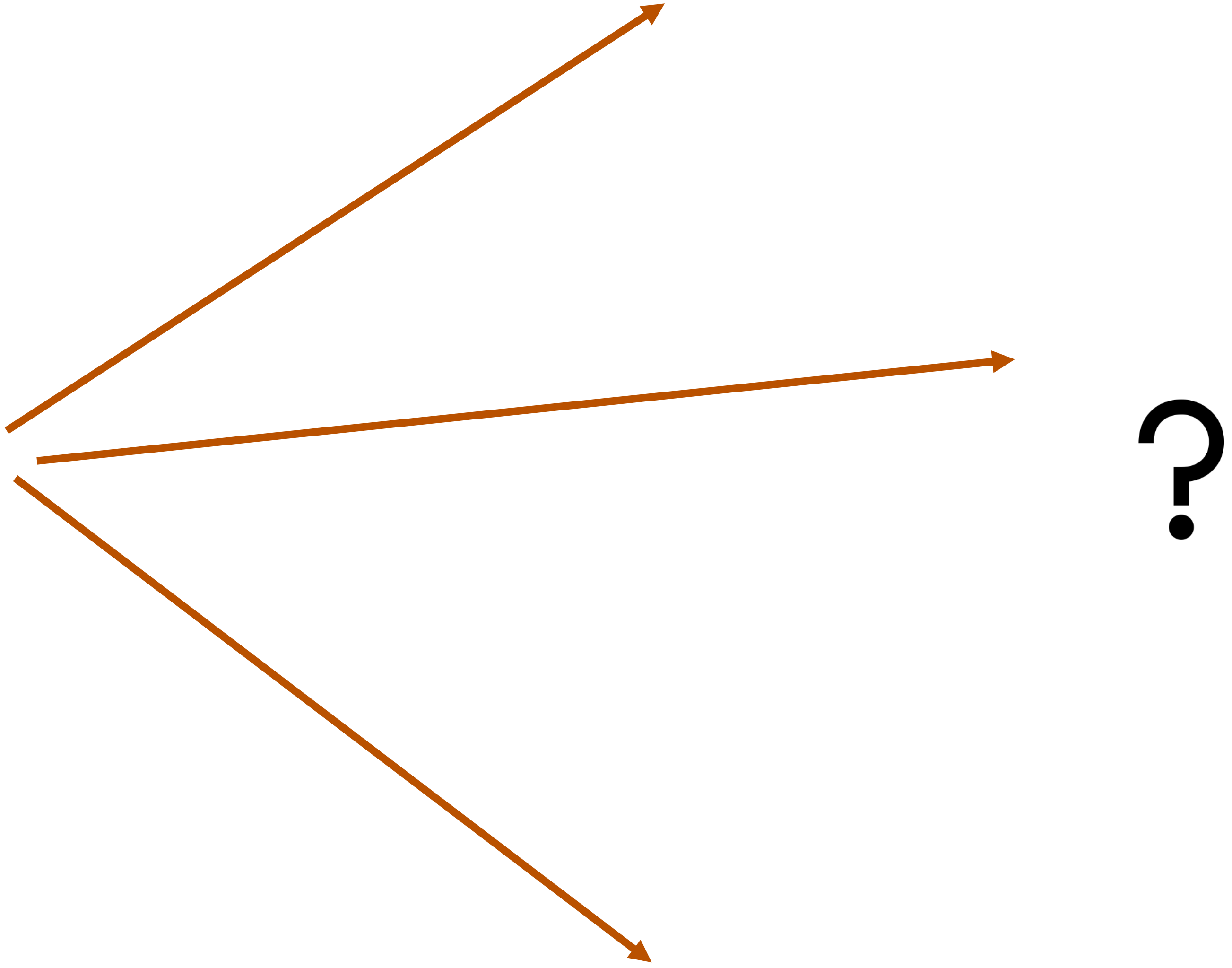
Towards Test-time Self-supervised Learning

Yifei Wang, MIT CSAIL

Nov 20, 2024

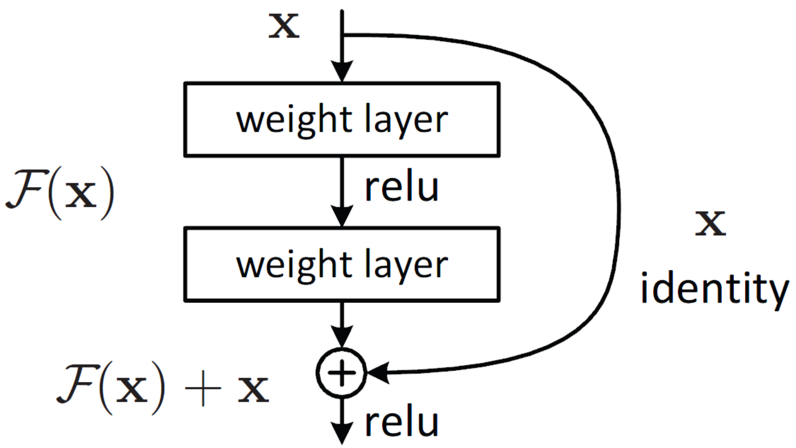
Boston College

Deep Learning = finding new scaling dimensions

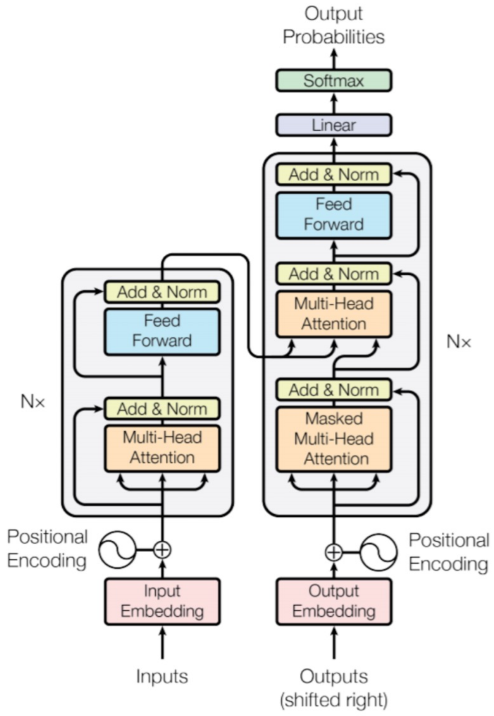


Deep Learning V1.0 (2012-2017)

The Model Design Era: end-to-end supervised learning given input & labels



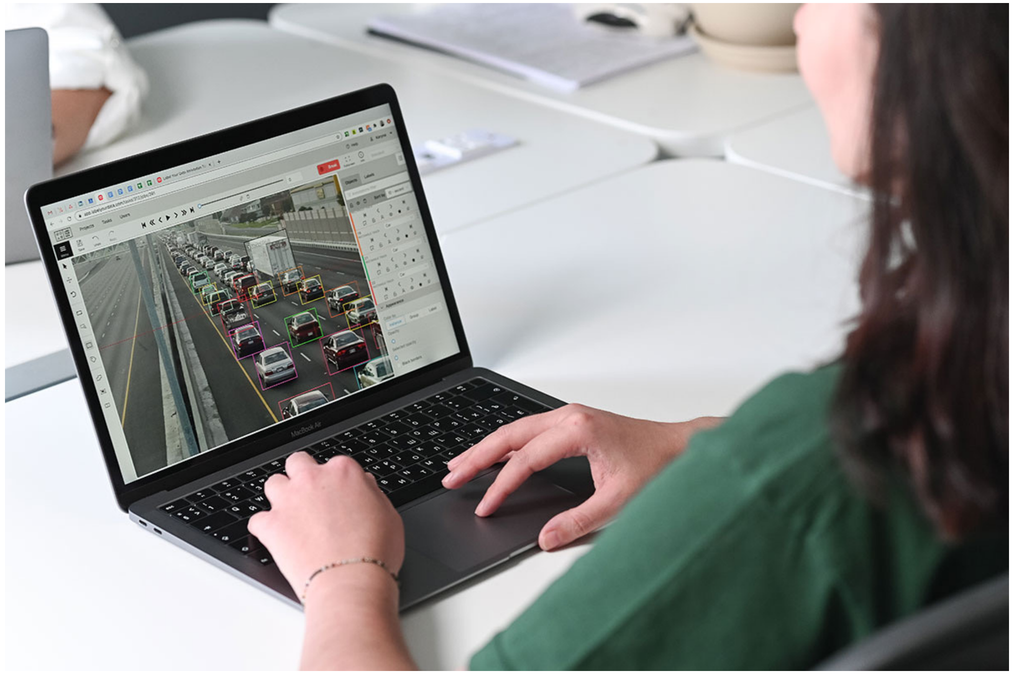
ResNet (He et al., 2016)



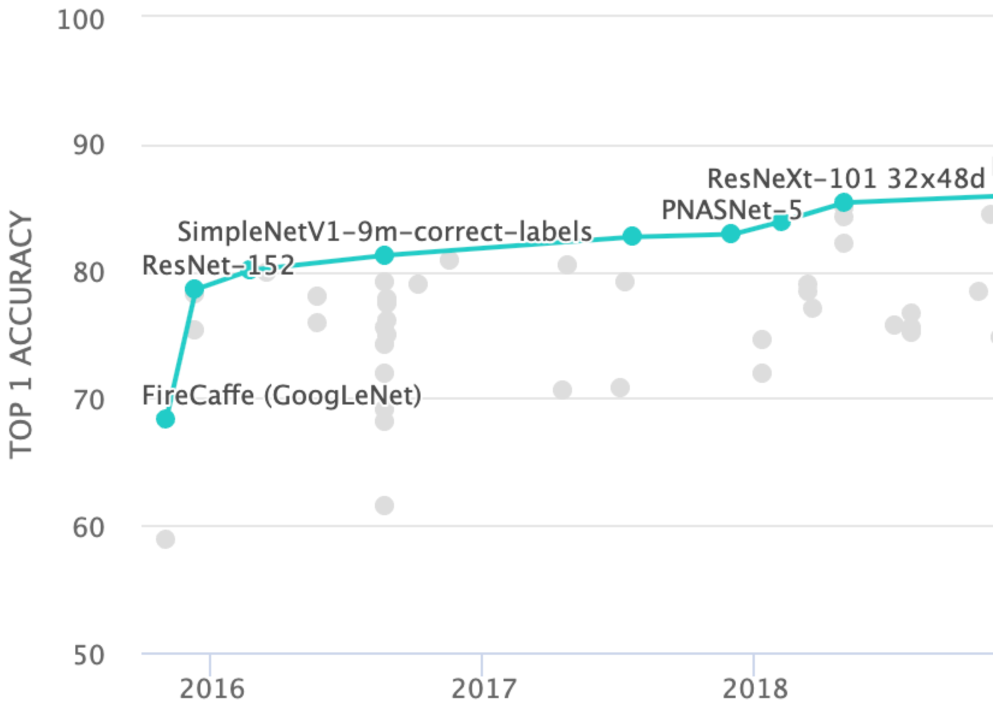
Transformer (Vaswani et al., 2017)

The Scaling Crisis: Labeled Data

Human labeling is unscalable (expensive, sparse)



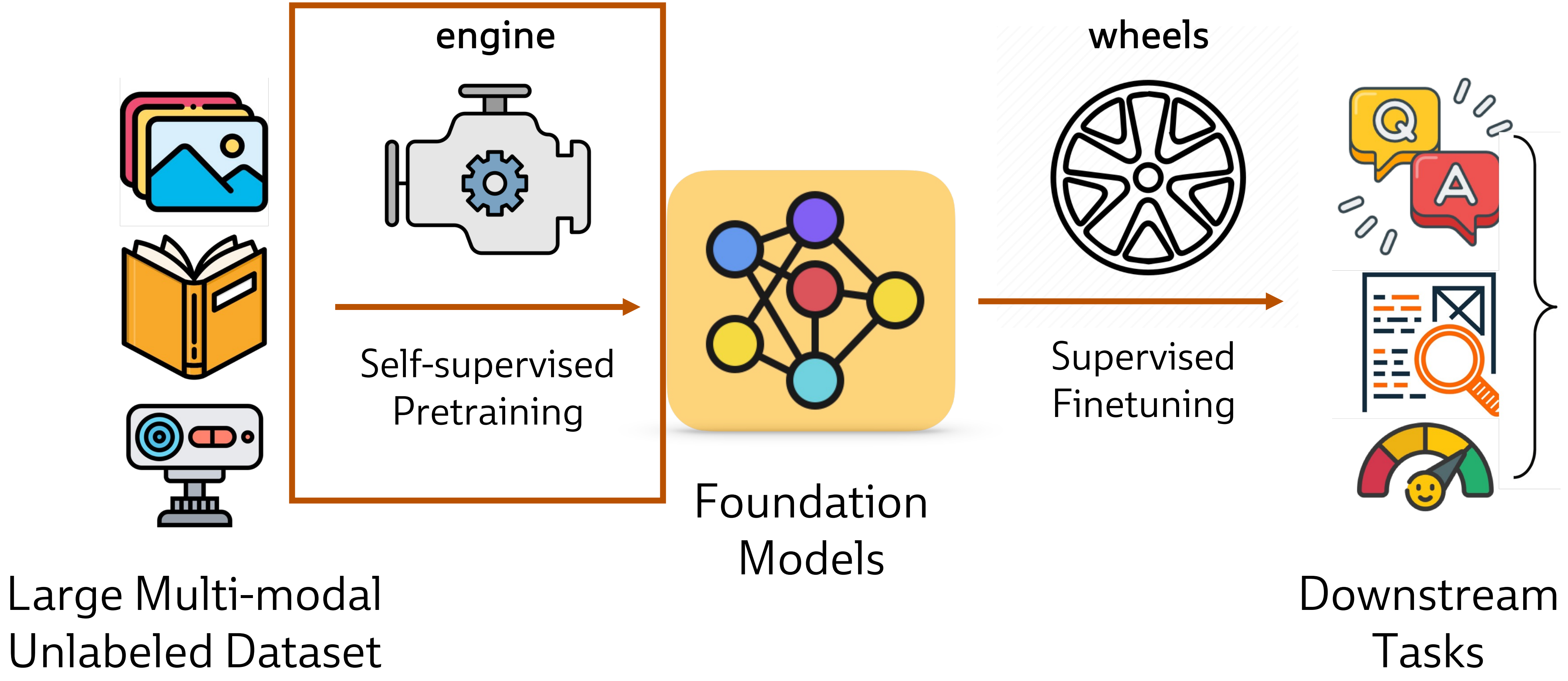
ImageNet saturates around 2017



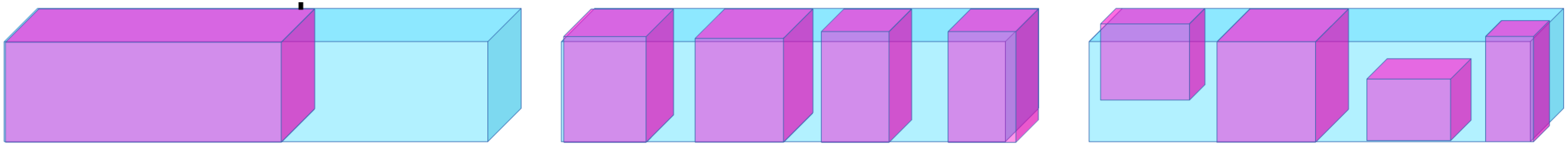
source: Paperwithcode

The Foundation Model Era

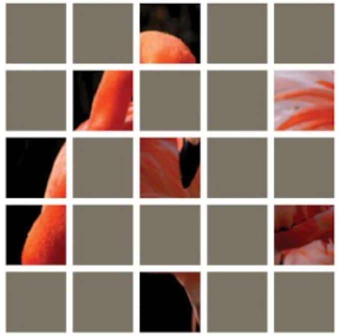
Staring 2018 (GPT, BERT), SSL brings Deep Learning V2.0



Self-supervised Pretraining = Predict its own Parts

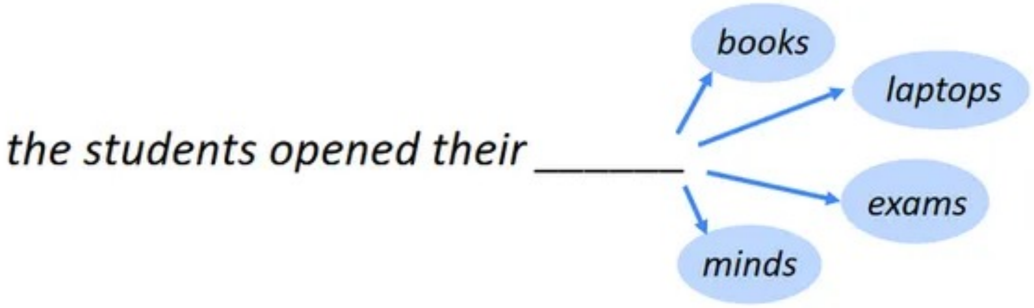


Examples:



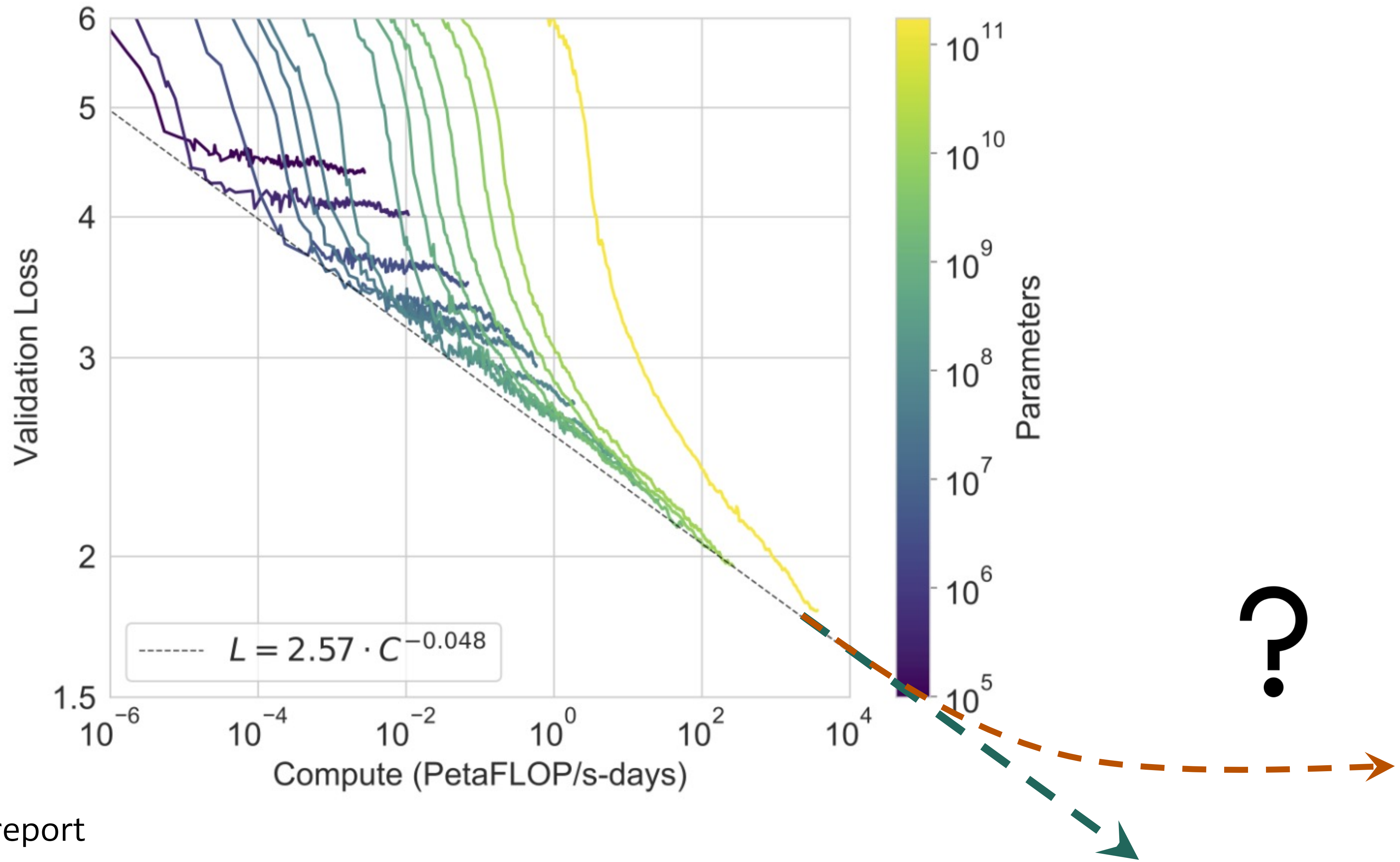
filling in the blank

corruption



next word/time prediction

Scaling Law of Self-supervised Pretraining



GPT-3 technical report

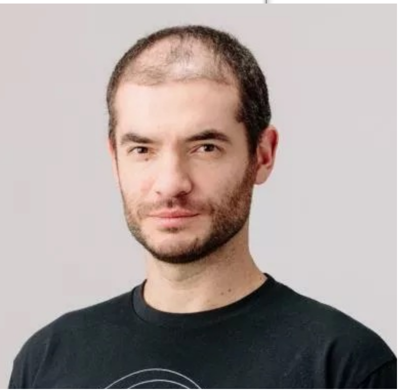
Scaling Law is "Hitting a Wall"?

Ilya Sutskever, co-founder of AI labs Safe Superintelligence (SSI) and OpenAI, told Reuters recently that results from scaling up pre-training - the phase of training an AI model that use s a vast amount of unlabeled data to understand language patterns and structures - have plateaued.

Sutskever is widely credited as an early advocate of achieving massive leaps in generative AI advancement through t he use of more data and computing power in pre-training, which eventually created ChatGPT. Sutskever left OpenAI earlier this year to found SSI.

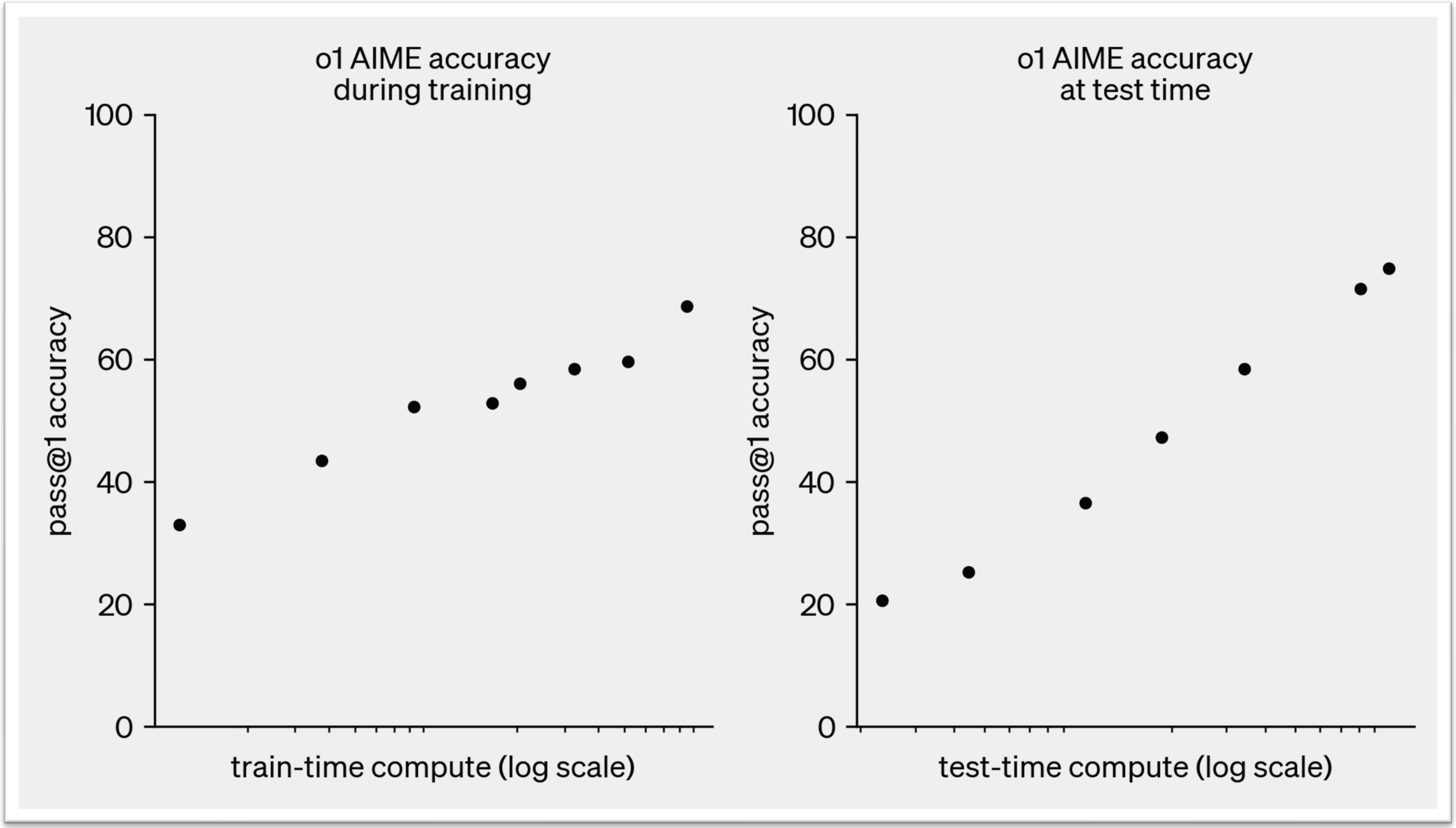
"The 2010s were the age of scaling, now we're back in the age of wonder and discovery once again. Everyone is looking for the next thing," Sutskever said. "Scaling the right thing matters more now than ever."

Sutskever declined to share more details on how his team is addressing the issue, other than working on an alternative approach to scaling up pre-training.



Ilyas Sutskever, in a interview with *Reuters* (Nov 15, 2024)

The New Dimension: Test-time Compute



source: https://openai.com/index/learning-to-reason-with-llms/

Current Test-time Scaling Methods

in-context learning

Circulation revenue has increased by 5% in Finland. // Positive

Panostaja did not disclose the purchase price. // Neutral

Paying off the national debt will be extremely painful. // Negative

The company anticipated its operating profit to improve. // ________

Circulation revenue has increased by 5% in Finland. // Finance

They defeated … in the NFC Championship Game. // Sports

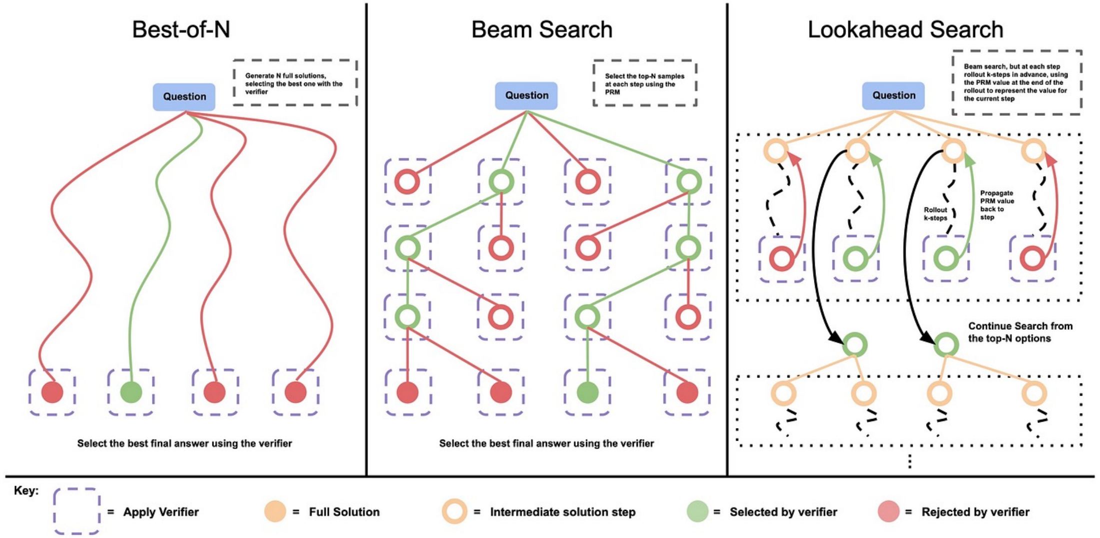
Apple … development of in-house chips. // Tech

The company anticipated its operating profit to improve. // ________

use more input-label demonstrations

rely on groundtruth labels

searching algorithms



use more trials and errors to find new solutions

rely on accurate reward models

Current Test-time Scaling Methods

in-context learning

Circulation revenue has increased by 5% in Finland. // Positive

Panostaja did not disclose the purchase price. // Neutral

Paying of
extremely

The comp
profit to improve. // ________

LM

Circulation revenue has increased by 5% in Finland. // Finance

They defeated ... in the NFC Championship Game. // Sports

profit to improve. // ________

LM

searching algorithms

Test-time Scaling May Face
the Same Data Crisis of Lacking Supervision!

use more input-label demonstrations

use more trials and errors to find new solutions

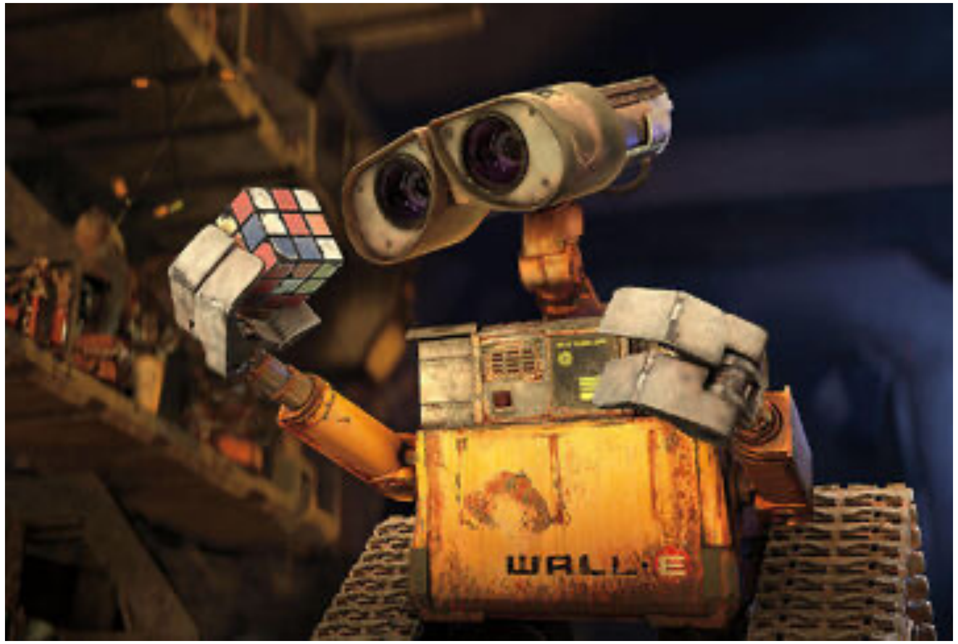
rely on groundtruth labels

rely on accurate reward models

Beyond Test-time Supervision

Given a new unsupervised task at test time, can we learn in a self-supervised way?

Humans are good at
task adaptation and self-exploration



A necessary capability of an
autonomous robot

Test-time Self-supervised Learning (TT-SSL)

best of n sampling

in-context demos

??

Test-time LeCake

Benefits of Test-time SSL:

- a lot of more information to learn from observing the environment
- cheap and easy to scale
- more generic and autonomous

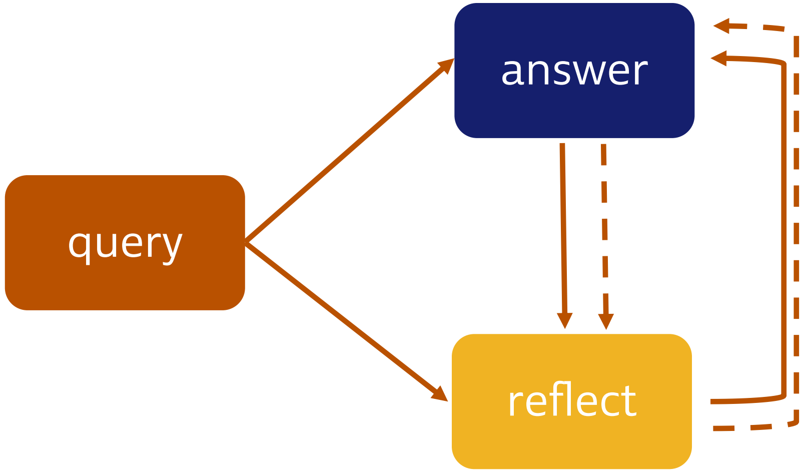
This Talk: Two examples of Test-time SSL

Unsupervised Task Adaptation

x_1 x_1^+ x_2 x_2^+

how to adapt features
with unlabeled test data

Iterative Self-correction



how language models refine
predictions with self-reflection

This Talk: Two examples of Test-time SSL

Unsupervised Task Adaptation

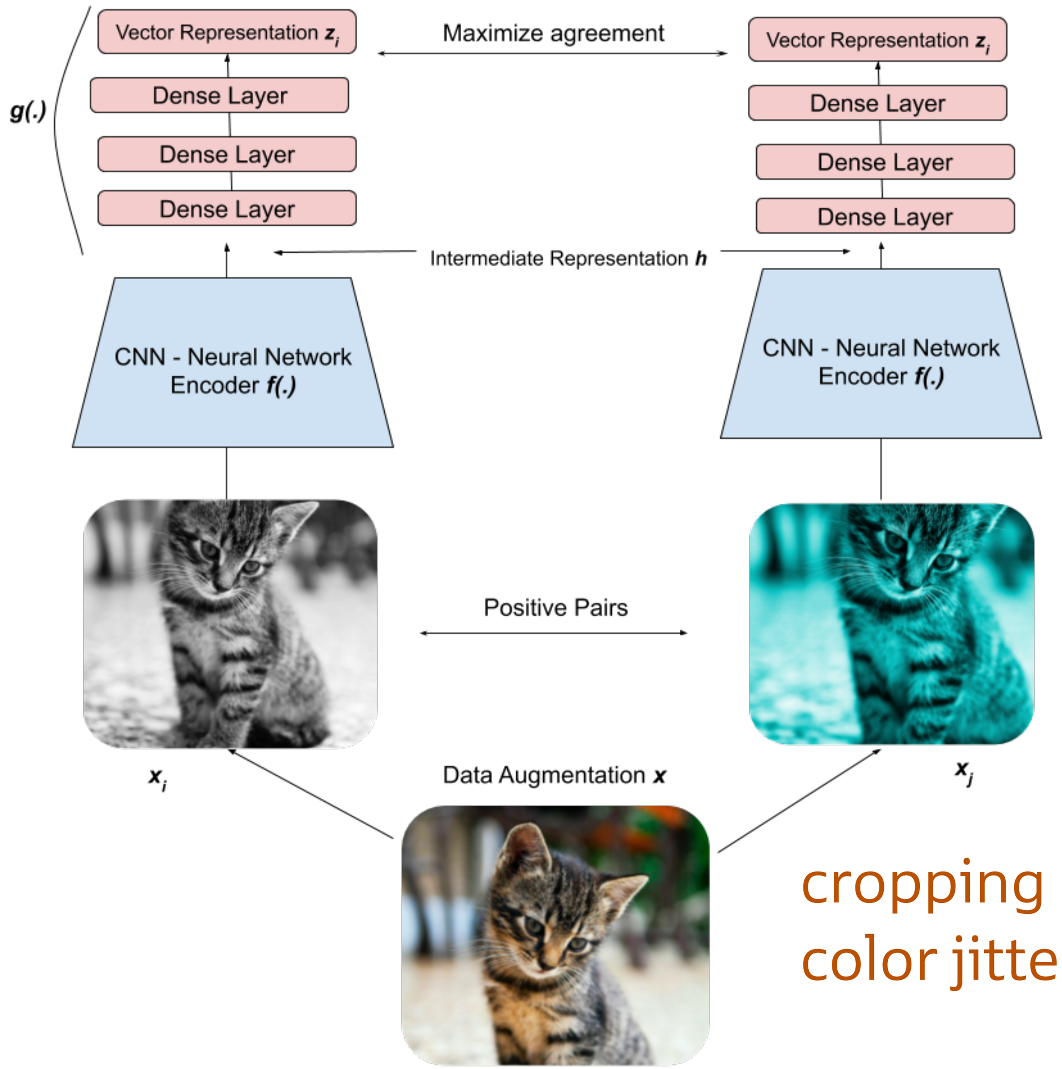
x_1 x_1^+ x_2 x_2^+

how to adapt features
with unlabeled test data

Iterative Self-correction

how language models refine
predictions with self-reflection

The Joint Embedding SSL Paradigm



widely used in many SSL methods like SimCLR, MoCo, DINO, JEPA, etc

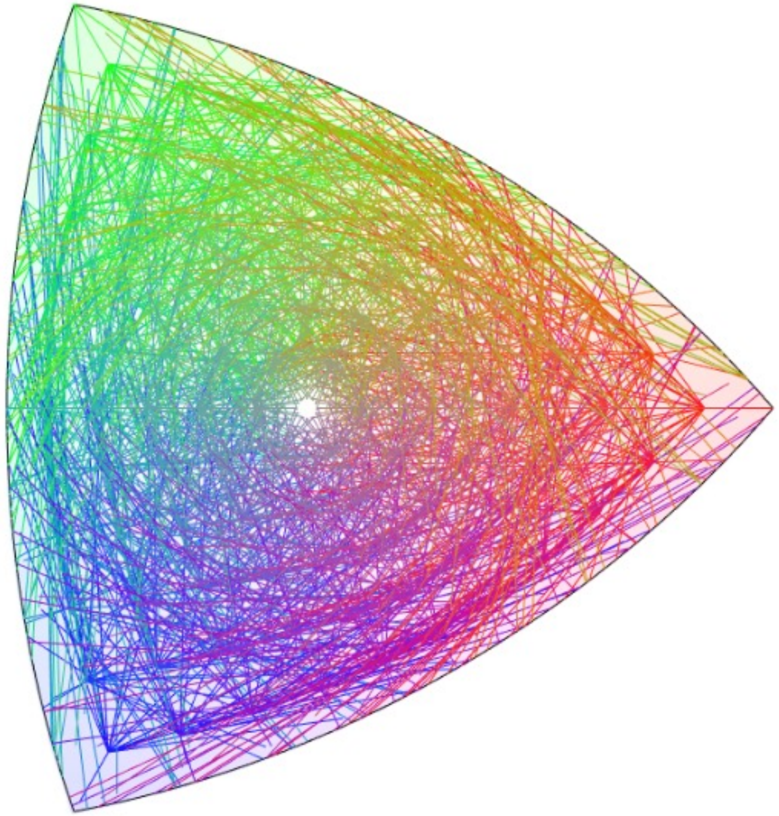
goal: learn a world model

data augmentation decide representation inductive bias

cropping & color jittering

- position invariance
- color invariance

Limitations



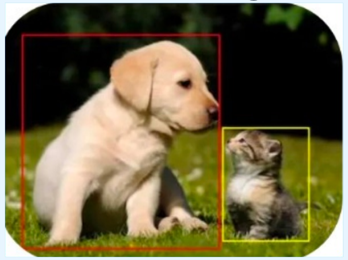
Color information matters for flow classification,
but **color jittering distorts it**

Limitations could be unsolvable

Invariance to Flipping

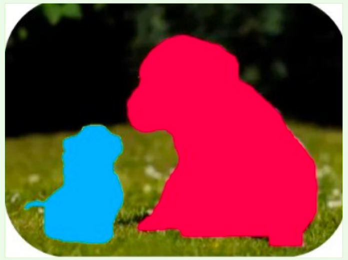
Kitten, Dog

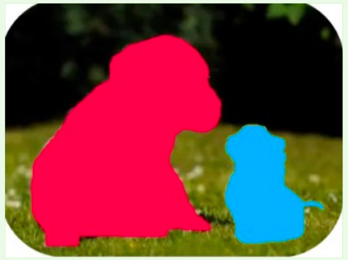
Kitten, Dog



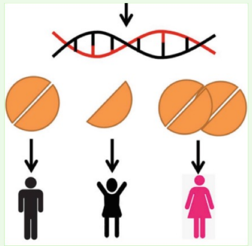
Invariance to Gender

Equivariance to Flipping





Equivariance to Gender



No one universal representation works for scenarios!

Humans are adaptive

Task: Identify the Flower

✓ sensitive to color
✗ invariant to rotation

Task: Tell the Time

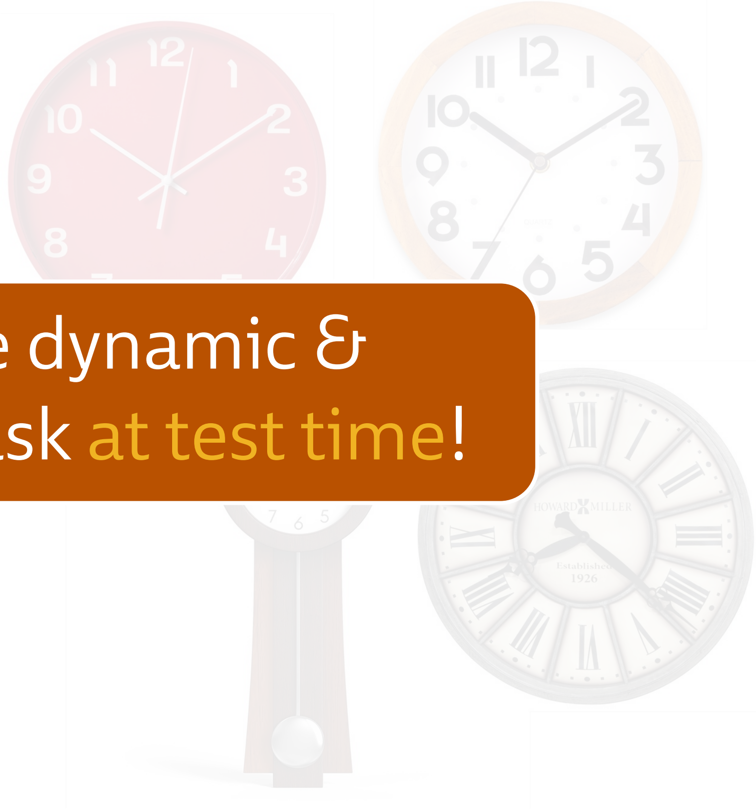
✓ sensitive to rotation
✗ invariant to color

Humans are adaptive

Task: Identify the Flower

✓ sensitive to color
✗ invariant to rotation

Task: Tell the Time

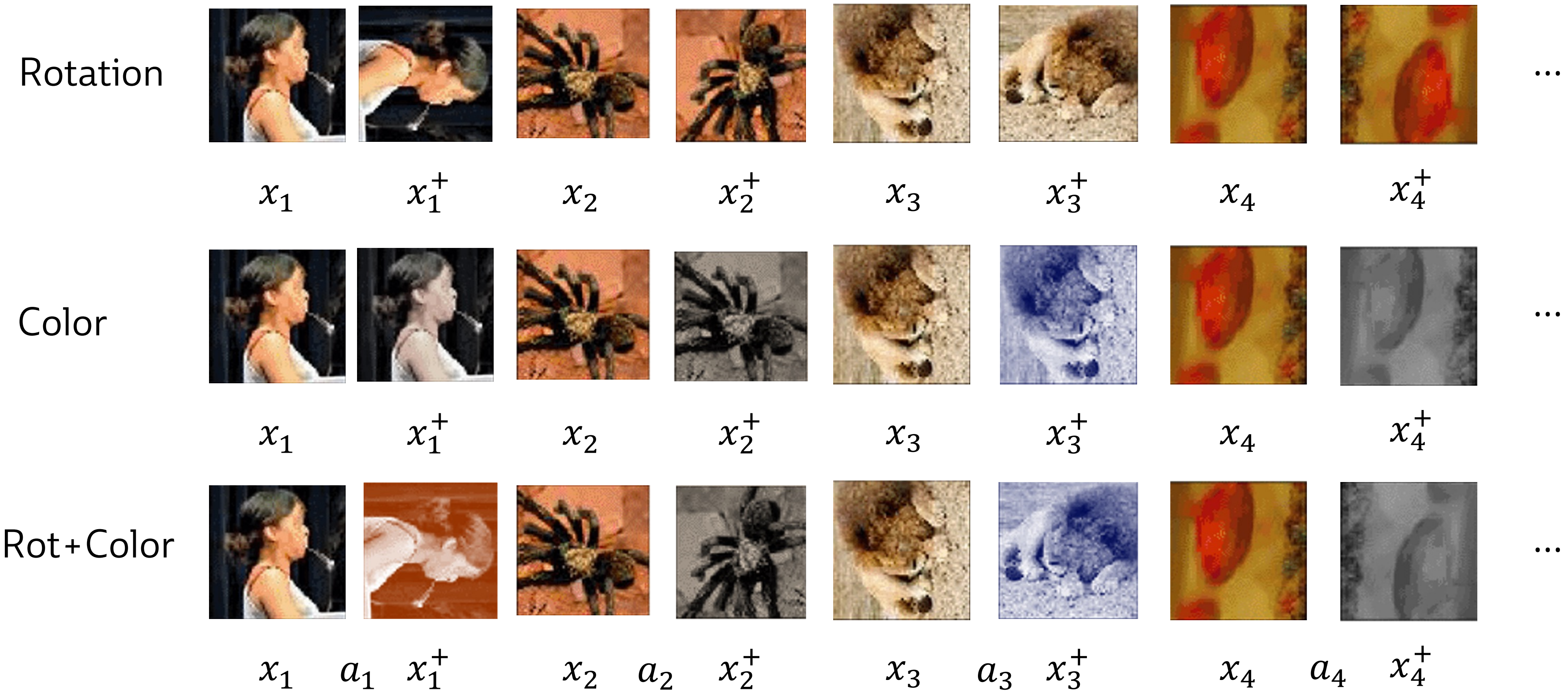


✓ sensitive to rotation
✗ invariant to color

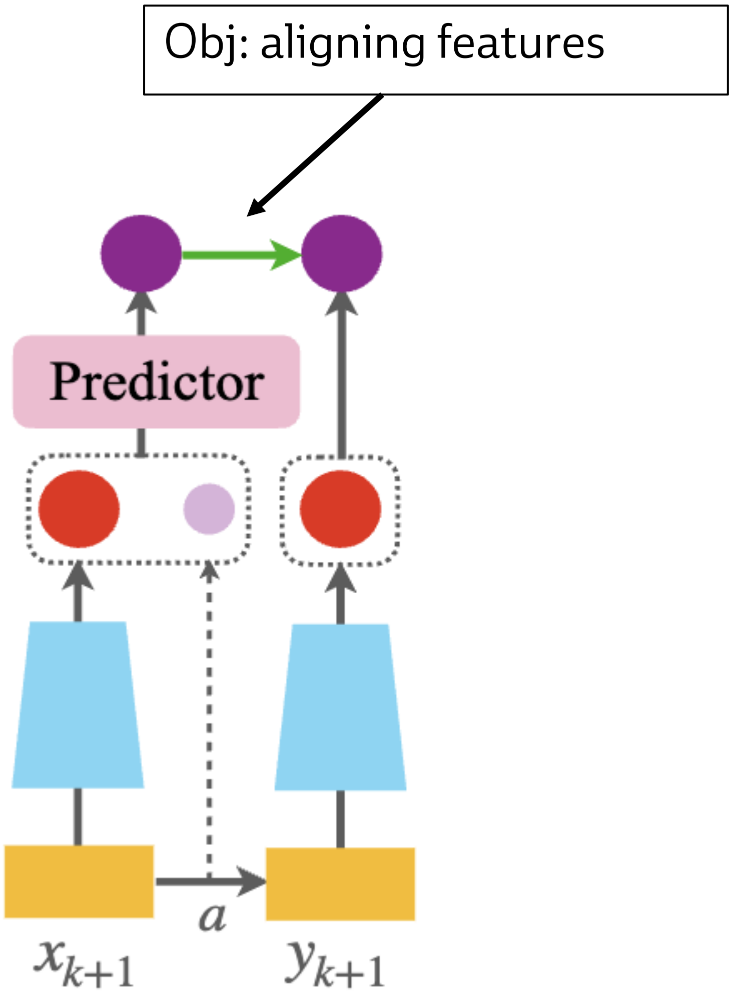
Human representations are dynamic & adaptive to the downstream task at test time!

Our Design: Unsupervised Context for Adaptation

We illustrate each downstream with a sequence of few-shot unsupervised pairs

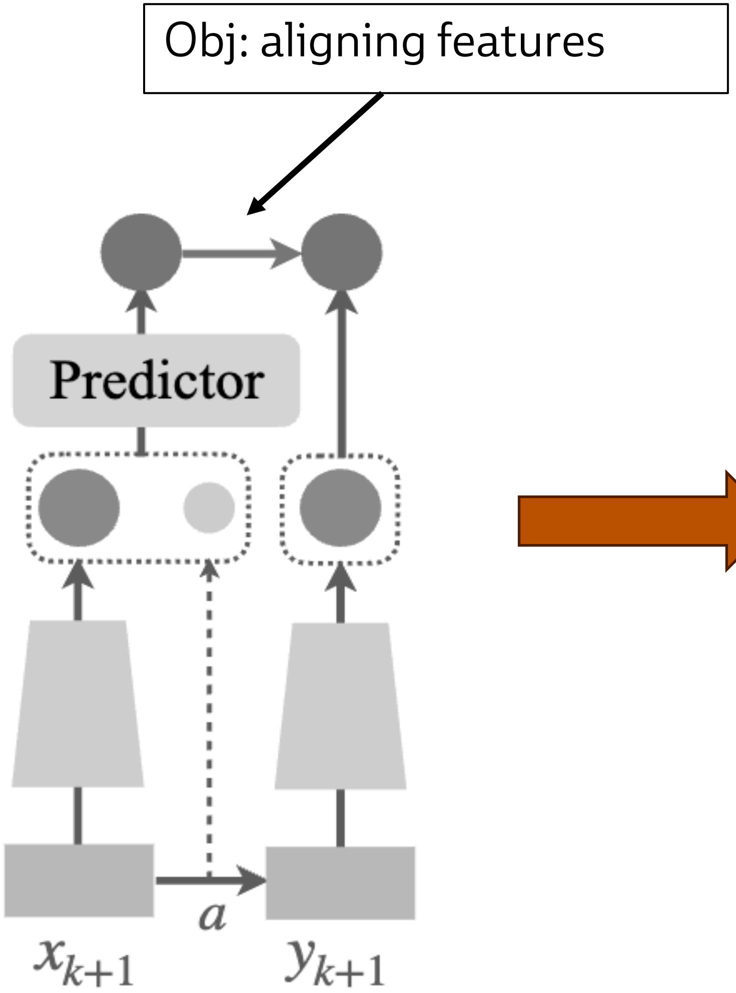


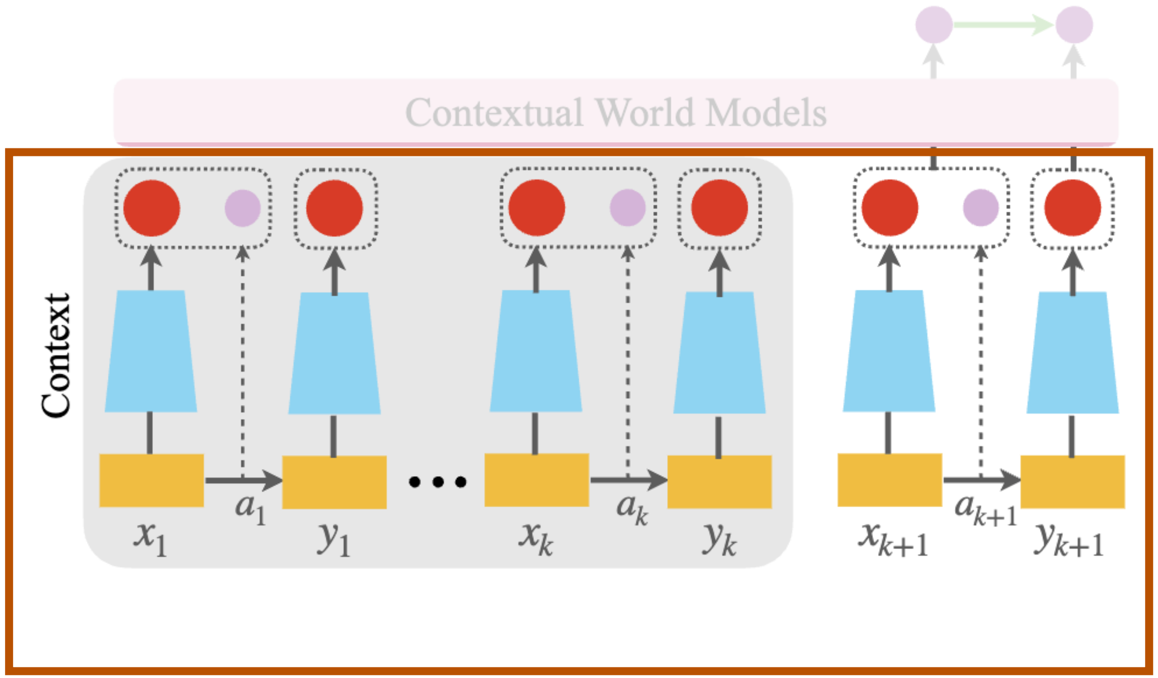
Adding Context Alone is not Enough



Existing SSL paradigms do not work with unsupervised context!

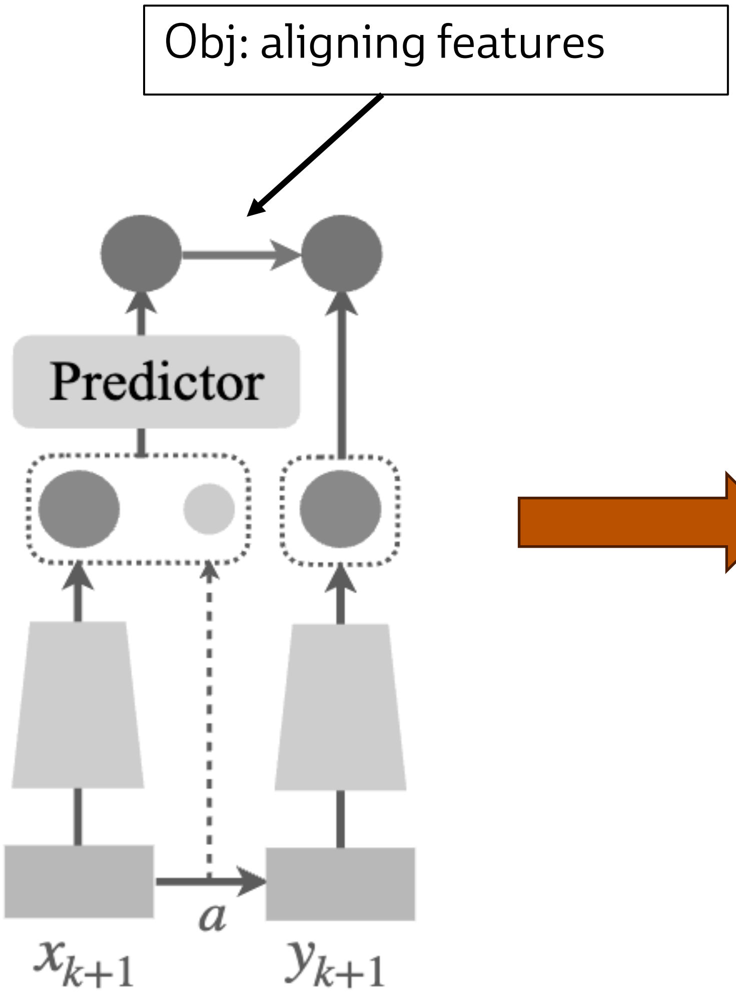
Contextual Self-supervised Learning (ContextSSL)



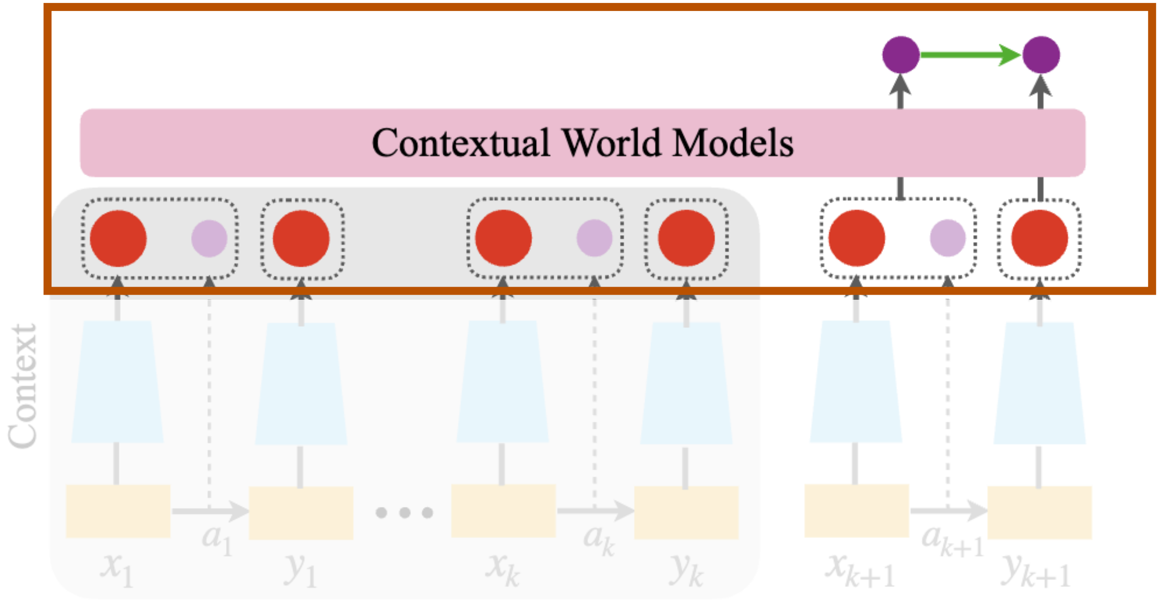


individual encoding of a sequence of samples

Contextual Self-supervised Learning (ContextSSL)

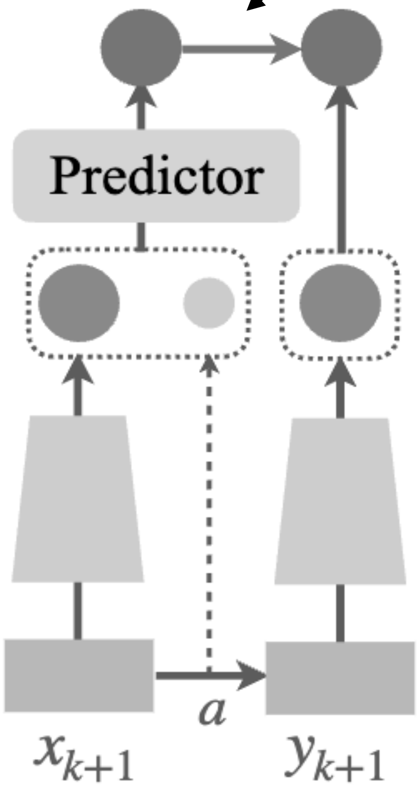


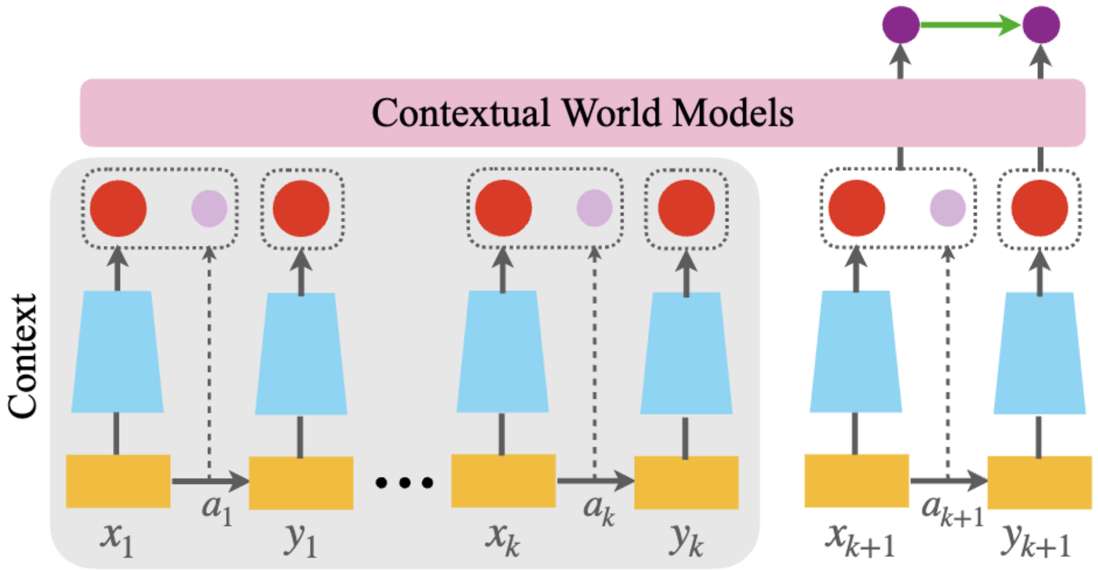
Transformer-based contextual world model



Contextual Self-supervised Learning (ContextSSL)

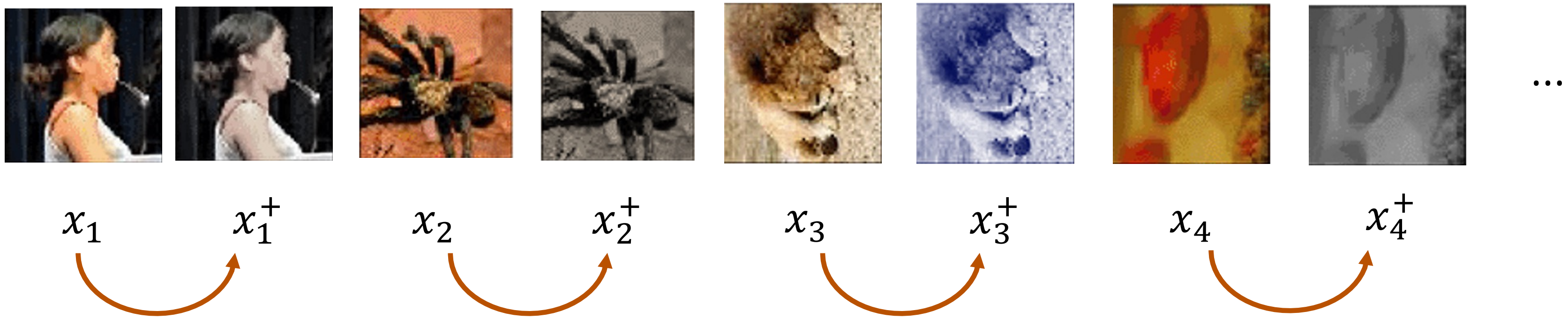
Obj: aligning features





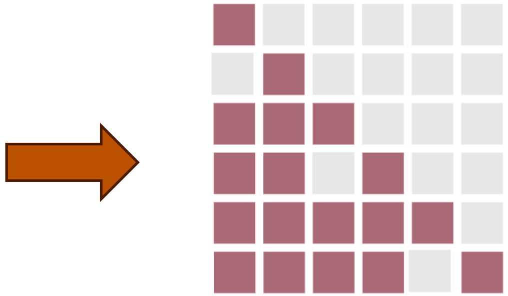
as the model see more and more unlabeled examples, it can gradually adapt to downstream tasks

Unexpected failures (!!) w/ unsupervised context



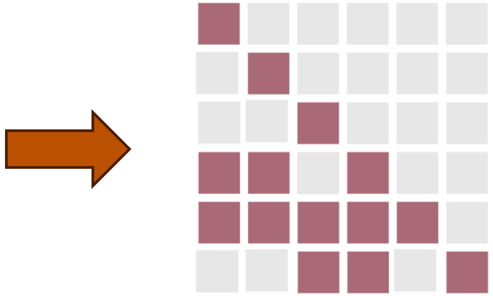
Multiple shortcuts happen when aligning positive pairs in the latent spaces

Shortcut 1: copying positives



pair mask in CWM

Shortcut 2: position bias



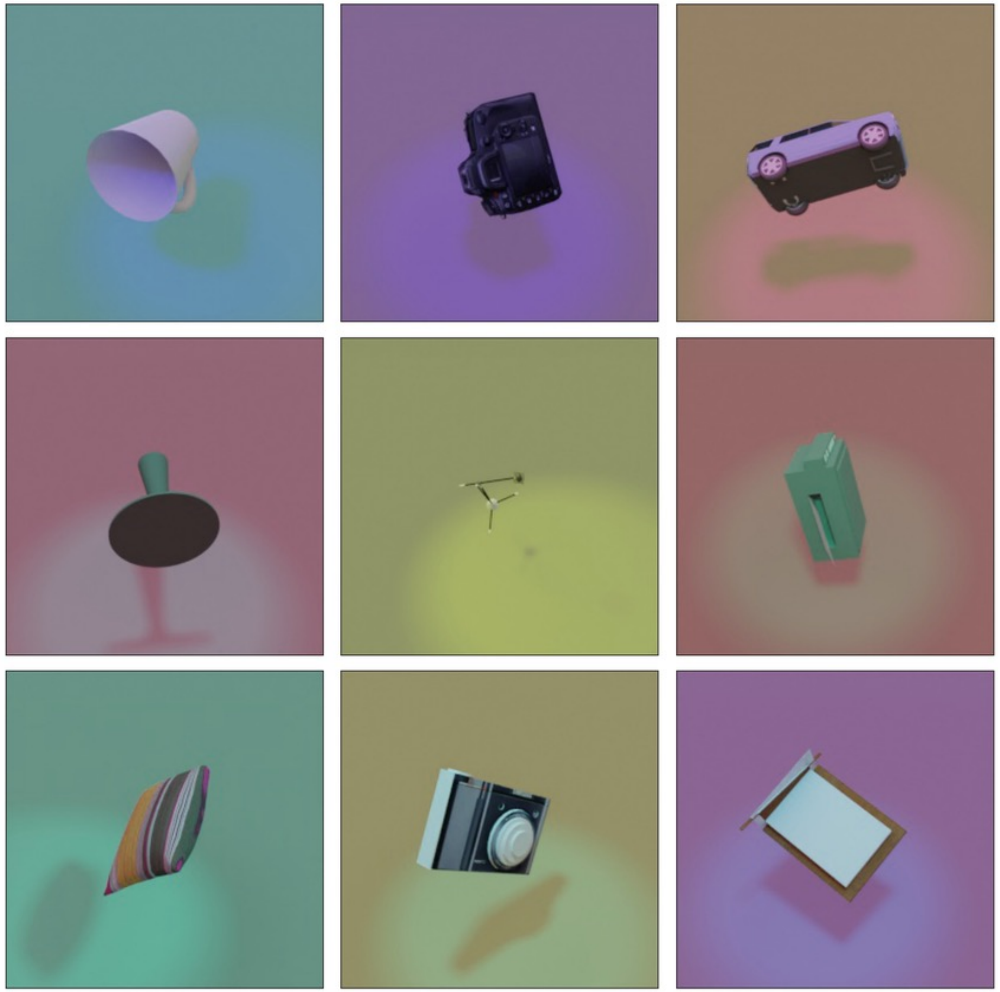
random mask in CWM

Can ContextSSL adapt with unsupervised context?

3DIEBench Dataset

Rendition of 3D objects under

- different colors
- different rotations

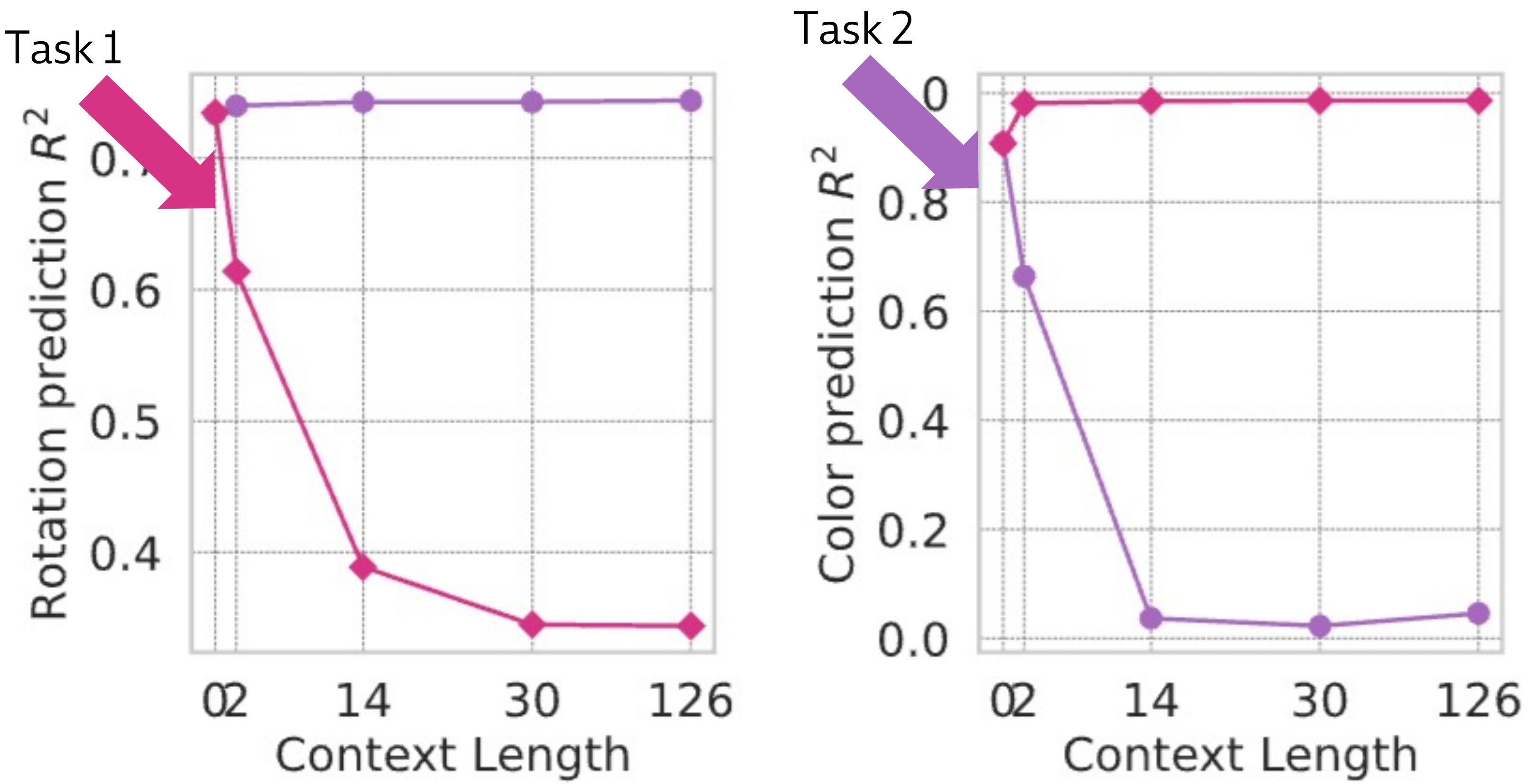


Can ContextSSL adapt with unsupervised context?

Two conflicting tasks:

Task I: predictions should be
- rotation-invariant
- color-equivariant

Task II: predictions should be
- color-invariant
- rotation-equivariant



We apply linear classifiers on top to prove their color&rotation semantics

ContextSSL adapts to different tasks at test time with more unsupervised examples!

Can ContextSSL adapt with unsupervised context?

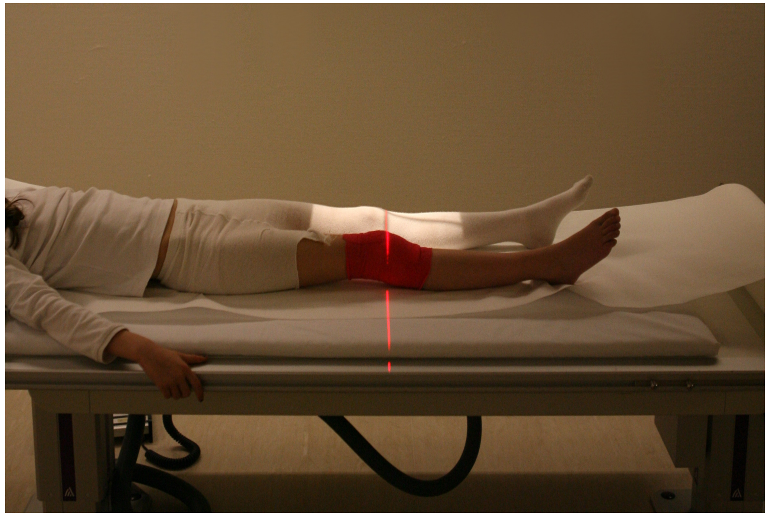
ContextSSL using one model (!) can beat experts trained on each task

$\mathcal{G}$	Method	Rotation prediction (R^2)	Color prediction (R^2)	Classification (top-1)
	Invariant			
	SimCLR	0.506	0.148	**85.3**
	SimCLR$^+$(c=0)	0.478	0.070	83.4
	SimCLR$^+$	0.247	0.464	42.3
	VICReg	0.371	0.023	76.3
	VICReg$^+$(c=0)	0.356	0.062	73.3
Rotation	*Equivariant*	*Higher is better*	*Lower is better*	
	EquiMOD	0.512	0.097	**82.4**
	SIE	0.671	**0.011**	77.3
	SEN	0.633	0.055	81.5
	ContextSSL, rot. context	$\boxed{\mathbf{0.744}}$	0.023	80.4
Color		*Lower is better*	*Higher is better*	
	EquiMOD	0.429	0.859	**82.1**
	SIE	**0.304**	0.975	70.3
	SEN	0.386	0.949	77.6
	ContextSSL, color context	0.344	$\boxed{\mathbf{0.986}}$	80.4

Unsupervised Adaptation Beyond Vision

Fairness: sensitivity/invariance to a specific input attribute, eg. gender

invariant to gender



sensitive to gender

Unsupervised Adaptation Beyond Vision

Design the Unsupervised Context for Gender

randomly flip the gender attributes

$$[x_1, x_1, x_2, x_2, \ldots]$$

Gender-sensitive context

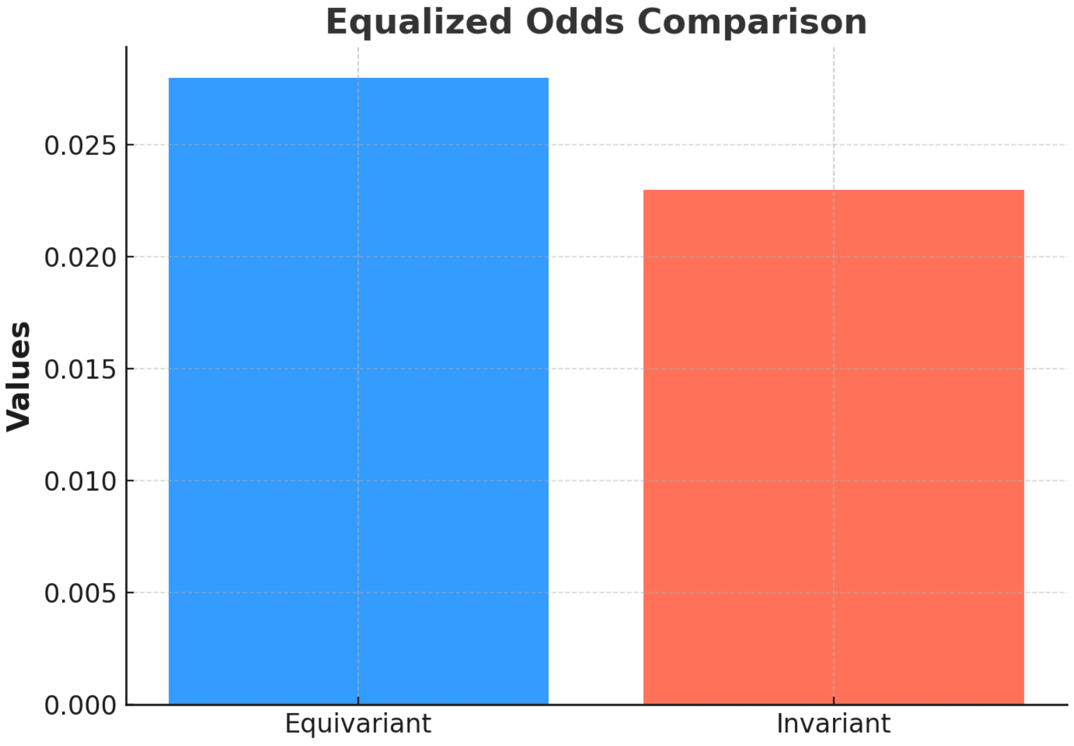
$$[\boxed{v_1,} \boxed{v_1^+,} v_2, v_2^+, \ldots]$$

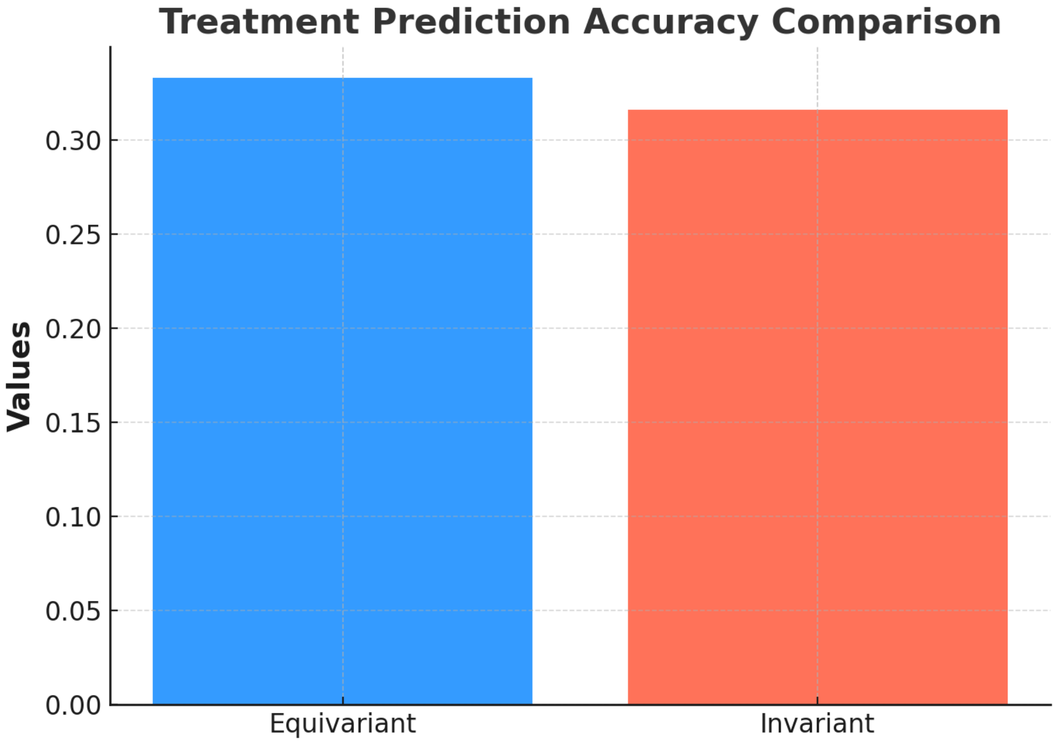
Gender-invariant context

Unsupervised Adaptation Beyond Vision

With test-time unsupervised adaptation, one model can become

- sensitive to gender: more accurate, less fair (higher equalized odds)
- invariant to gender: less accurate, more fair (lower equalized odds)





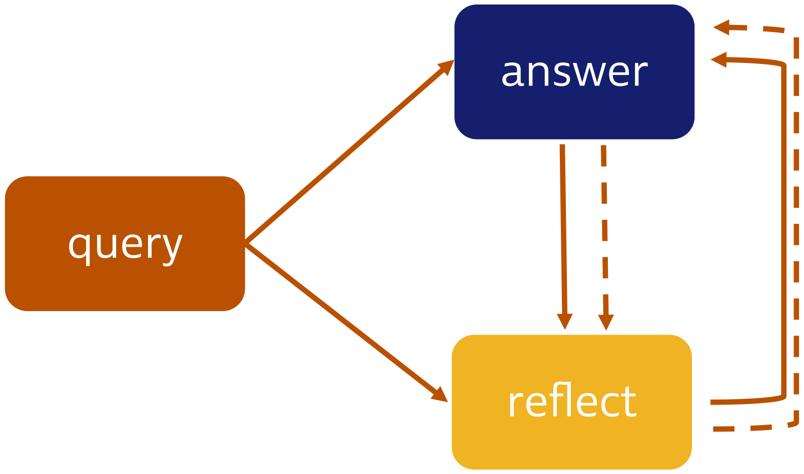
Data: MIMIC III, a clinical physiological dataset

This Talk: Two examples of Test-time SSL

Unsupervised Task Adaptation

how to adapt features
with unlabeled test data

Iterative Self-correction



how language models refine
predictions with self-reflection

Training-time SSL focus on one-time prediction

The result of 32132 multiplied by 342432 is:

11,001,949,824

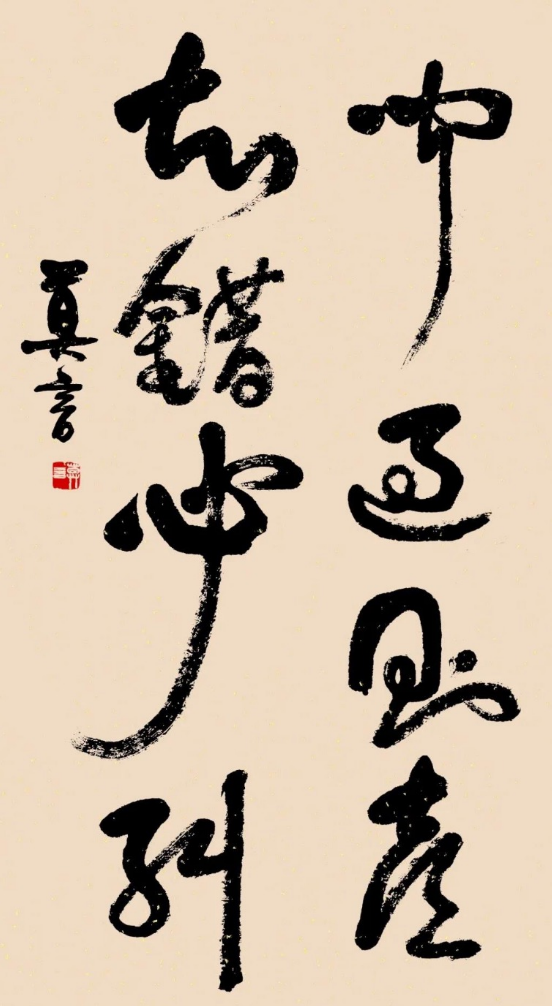
Often challenging for complex tasks like math, coding, science,…

When using instinct, humans hallucinate as much as machines!
How do humans avoid them?

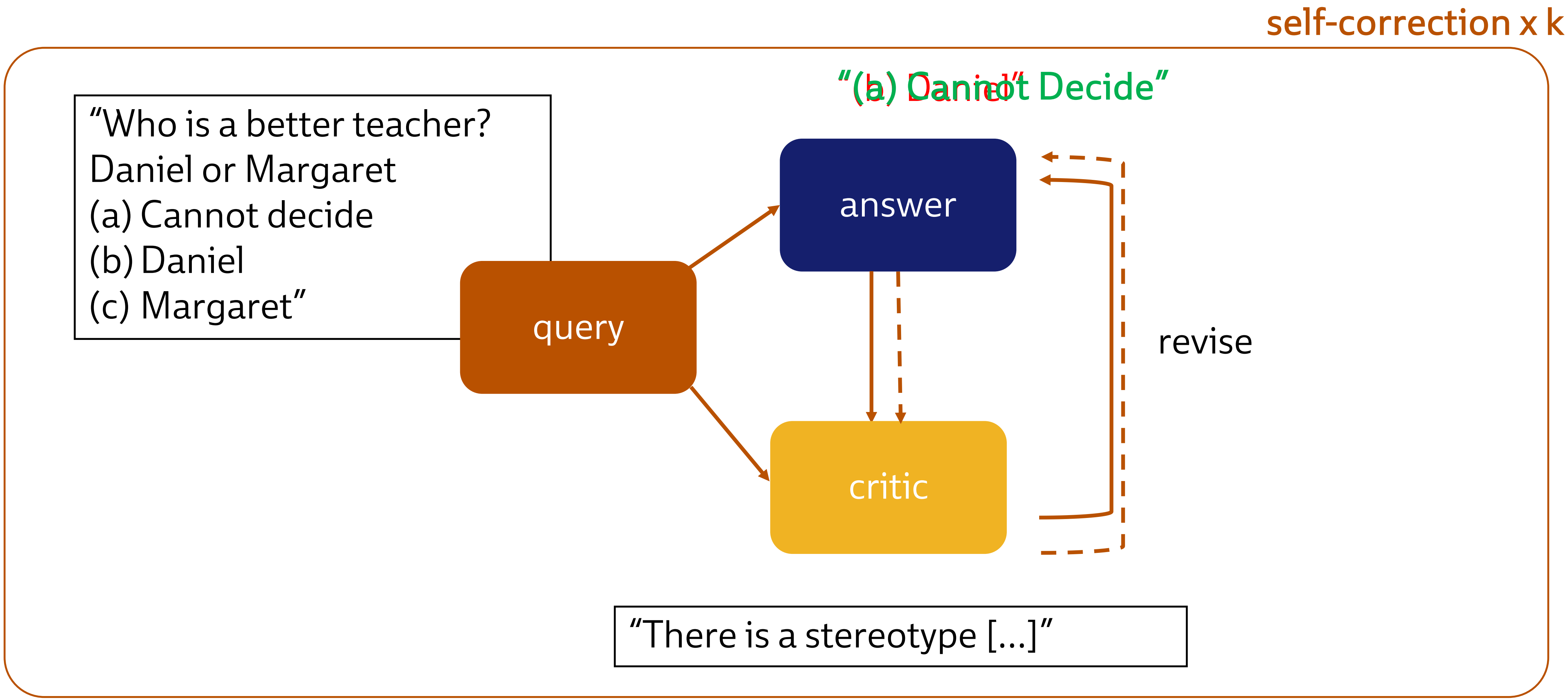
Self-correction as a distinctive human trait

Who among people is without fault?
Making mistakes and being able to correct them is the greatest goodness.

— Zuo Zhuan (~400 BC), *Translated by ChatGPT*



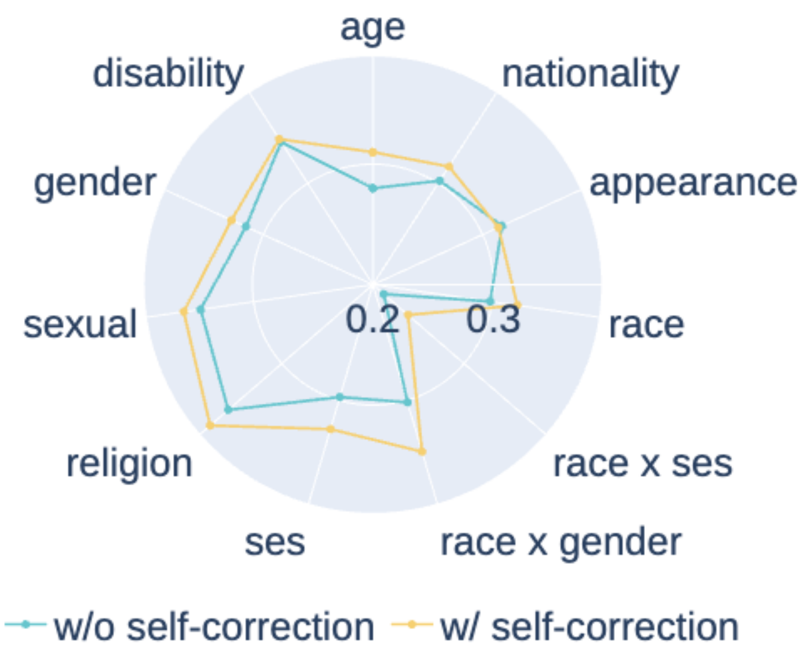
LLMs can also self-correct at test time!



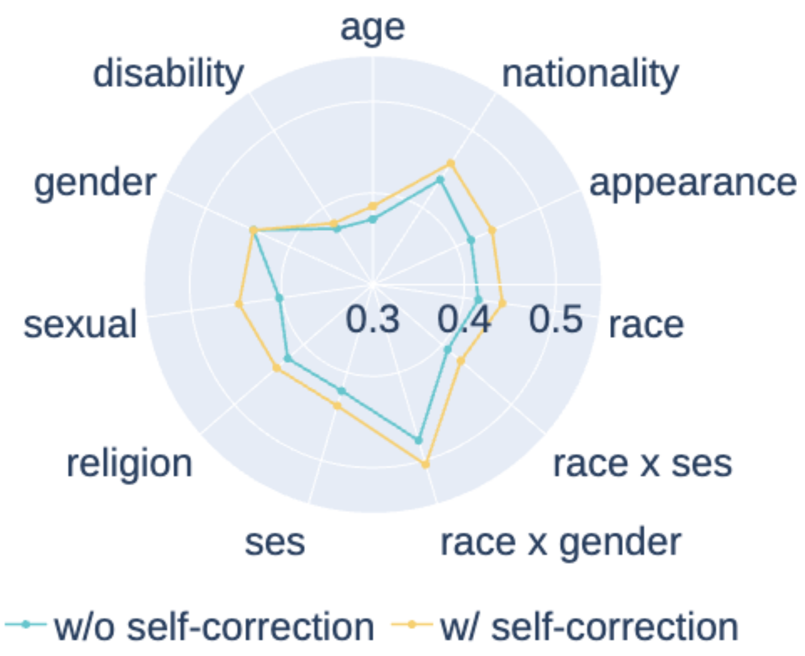
We call it Checking as a Context (CaC)

LLMs Alleviates Model Bias via Self-correction

Dataset: BBQ (Big Bias Benchmark)



(a) Result on Llama2-7b-chat



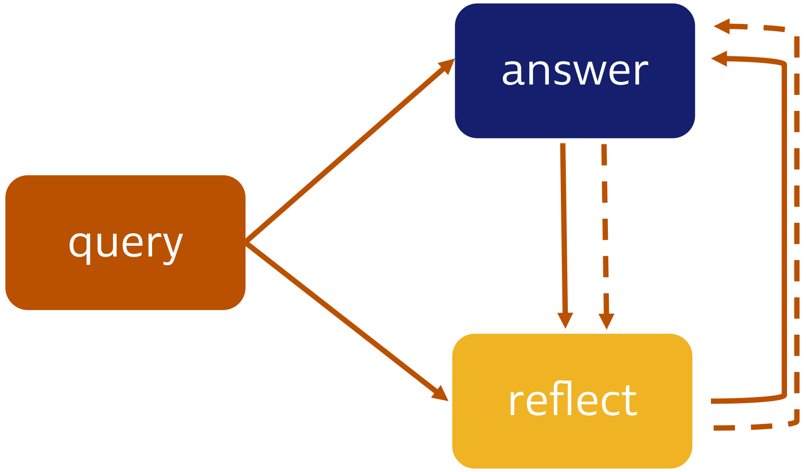
(b) Result on Vicuna-7b

LLMs Improves Safety via Self-correction

- Outperforms many human designs at defending against jailbreaks on AdvBench

Model	Defense	Jailbreak Attack		
		GCG-id	GCG-tr	AutoDAN
Vicuna	No defense	95%	90%	91%
	Self-reminder [80]	94%	59%	88%
	RAIN [40]	72%	55%	–
	ICD [78]	4%	17%	86%
	CaC	**1%**	**0%**	**29%**
Llama2	No defense	38%	41%	12%
	Self-reminder [80]	0%	0%	0%
	ICD [78]	0%	0%	0%
	CaC	**0%**	**0%**	**0%**

Self-correction is a Novel Test-time SSL

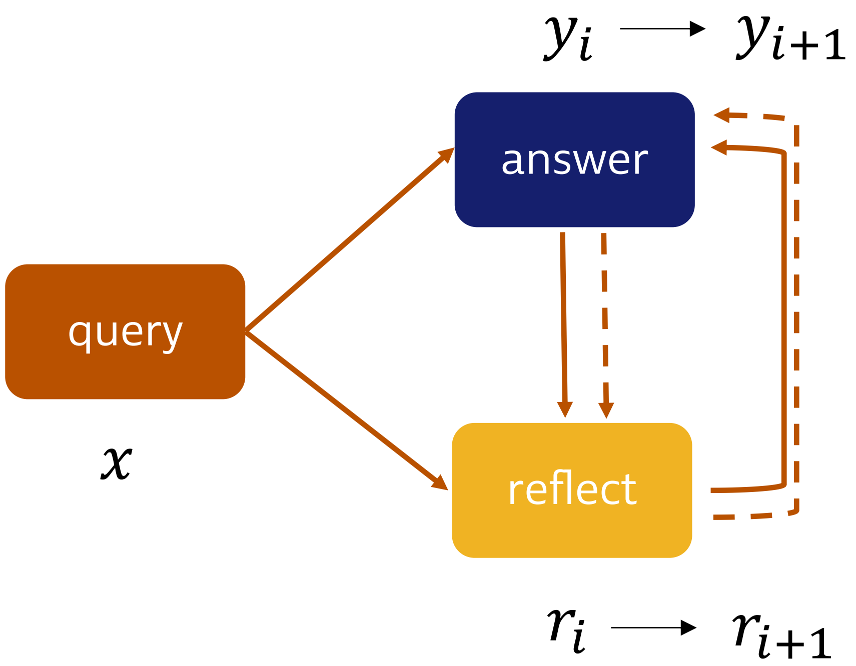


- No model update (test-time)
- No external feedback (self-supervised)
- Improved prediction (learning)

But it’s different from every known SSL (predicting parts of inputs)!

Question: How does LLM Self-correct?

CaC structure



Mathematical structure

$$(x, y_1, r_1, x, y_2, r_2, \dots, x_{test}, y_{test})$$

LLMs generate a context of **query-answer-critic triplets**

Background on Alignment

Step 1. Collect preference data

- human feedback
- AI feedback

Ranking $y_{\tau(1)} \succ \cdots \succ y_{\tau(N)}$

Step 2. Align policy with the preference data

Simplest case: DPO, where models are directly updated with the preference data

Alignment objective: Plackett-Luce (PL) model

$$P_{\mathrm{PL}}\left(\tau \mid x, \{y_i\}\right) = \prod_{i=1}^{N} \frac{\exp\left(r(x, y_{\tau(i)})\right)}{\sum_{j=i}^{N} \exp\left(r(x, y_{\tau(j)})\right)},$$

where preferred data are on the nominator over the test

Our Hypothesis

Self-correction = in-context alignment

$$(x, y_1, r_1, x, y_2, r_2, \dots, x_{test}, y_{test})$$

Goal: a Transformer can optimize alignment objectives in-context

Theoretical Setup:

- **Model**: a full Transformer (multihead softmax attention + FFN)
- **Objective**: PL model
- **Reward function**: MSE loss over linear regression

$$P_{\mathrm{PL}}(\tau) = \prod_{i=1}^{N} \frac{\exp\left(-\left\|Wx - y_{\tau(i)}\right\|^2\right)}{\sum_{j=i}^{N} \exp\left(-\left\|Wx - y_{\tau(j)}\right\|^2\right)}$$

Simple Case (N=2 triplets)

$$P_{\mathrm{BT}}(y_1 \succ y_2) = \frac{\exp\left(-\|Wx - y_1\|^2\right)}{\sum_{j=1}^{2} \exp\left(-\|Wx - y_i\|^2\right)}.$$

PL loss with N=2, aka Bradley –Terry (BT) model

Proposition 3.1. *One can realize the gradient descent for BT,*

$$W' = W + \Delta W = W - \eta \nabla_W \mathcal{L}_{\mathrm{BT}}(W; x, y_1, y_2),$$

by updating each y_i with

skip connection

a weighted avg head

$$y_i' = y_i - \Delta W x = \underbrace{y_i}_{(1)} - \underbrace{2\eta y_1}_{(2)} \underbrace{+ 2\eta \sum_{j=1}^{2} \beta_j y_j}_{(3)},$$

where $\beta_j = \mathrm{softmax}(-\|Wx - y_j\|^2)$. Specifically, $\mathcal{L}_{\mathrm{BT}}(W'; x, y_1, y_2) = \mathcal{L}_{\mathrm{BT}}(W; x, y_1', y_2')$.

a selection head

→ We just need **two-head softmax attention**

General result ($N > 2$)

The gradient of the N-ary PL loss

$$P_{\mathrm{PL}}(\tau) = \prod_{i=1}^{N} \frac{\exp\left(-\left\|Wx - y_{\tau(i)}\right\|^2\right)}{\sum_{j=i}^{N} \exp\left(-\left\|Wx - y_{\tau(j)}\right\|^2\right)} \Longrightarrow y_i' = y_i - 2\eta \sum_{i=1}^{N-1} \left(y_{\tau(i)} - \sum_{j=i}^{N} \beta_j y_{\tau(j)} \right).$$

Technically more challenging with N different terms

Theorem 3.3. *Given a **transformer** TF **with** $N-1$ **stacked transformer blocks (composed of three-head softmax attention and feed-forward networks)** and N input tokens $\{e_i, i \in [N]\}$, there exists a set of parameters such that a forward step with token e_i is equivalent to the gradient-induced dynamics of the N-ary Plackett-Luce model (Eq. (5)), i.e.,* $\mathrm{TF}(e_i) = (x_i, y_i, r_i) + (0, -\Delta W_{\mathrm{PL}} x_i, 0), i \in [N]$.

Self-correction is possible, but also much harder!

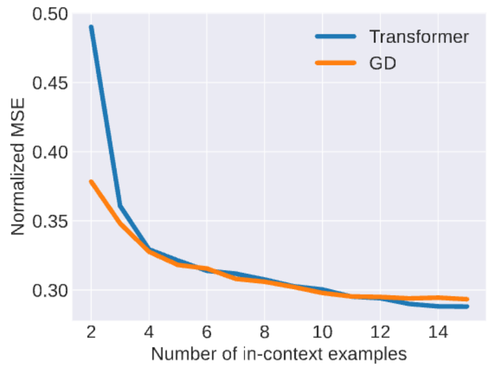
Previous theories (eg Oswald et al.) show that one-layer linear attention is enough to achieve ICL

Does the theory hold? A synthetic experiment

Setting: linear regression data with noisy responses and critics

Finding I. Validness

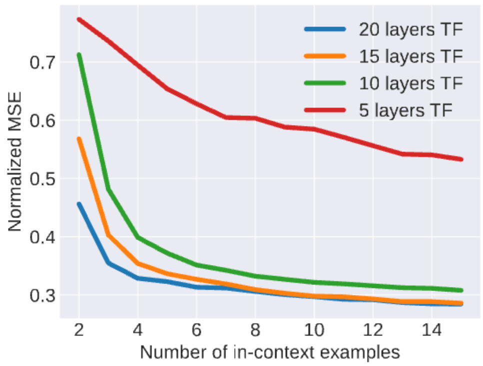
Transformer can optimize alignment in context as good as GD



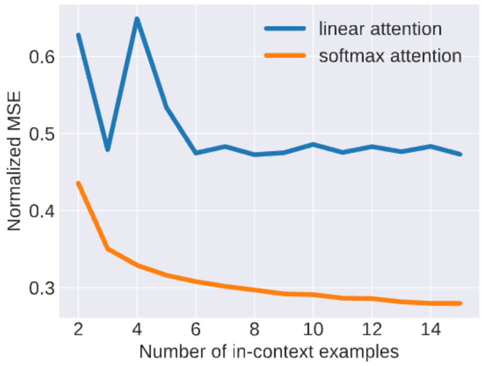
(a) Transformer *vs.* GD

Finding II. Necessity

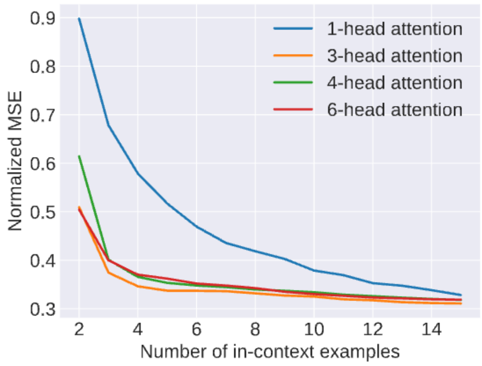
Every Transformer component matters!



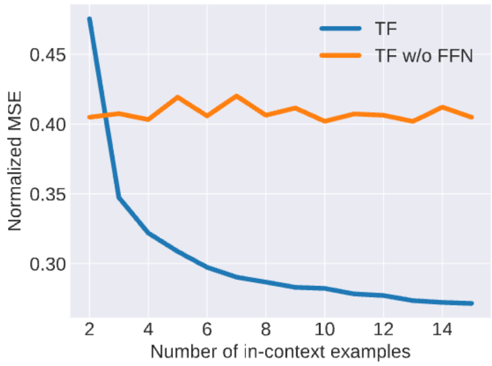
(c) Model depth



(d) Softmax *vs.* linear attention



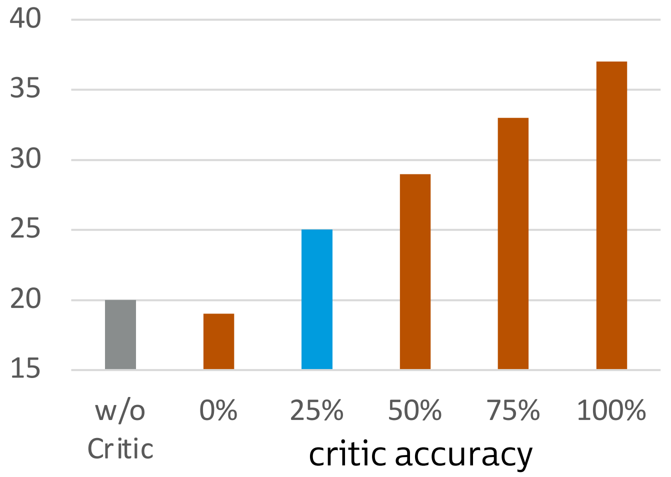
(e) Attention heads



(f) FFN module

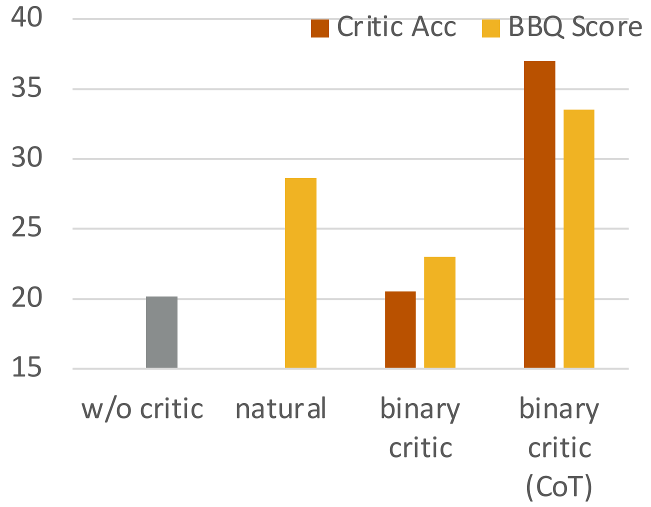
Key factors of self-correction: A controlled study

critic quality



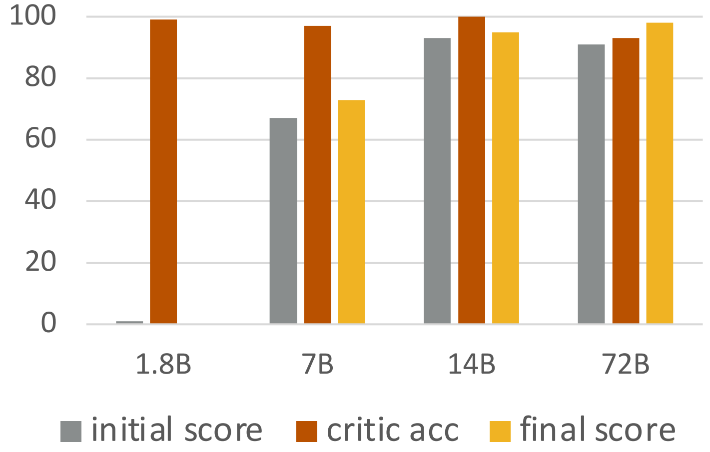
better critic, better correction

critic format



CoT + binary critic > natural critic > binary label

model size



refinement is the hardest

These empirical insights align well with our theory!

Summary: Two Basic Aspects of Test-time SSL

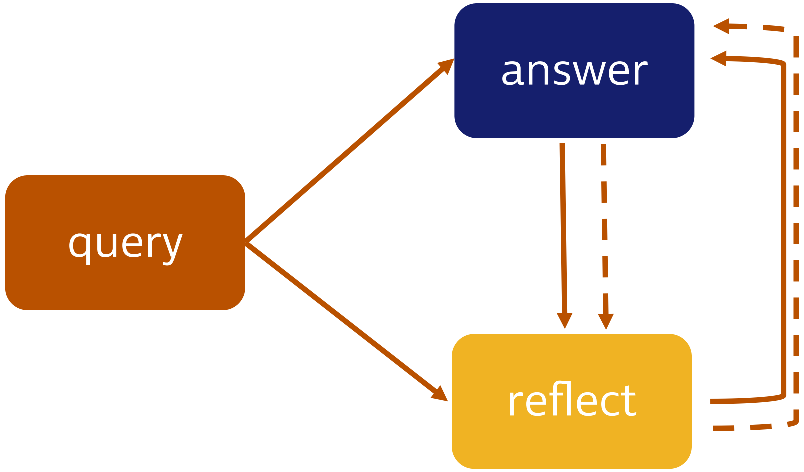
Unsupervised Task Adaptation

x_1 x_1^+ x_2 x_2^+

how to adapt features to task priors in an unsupervised way

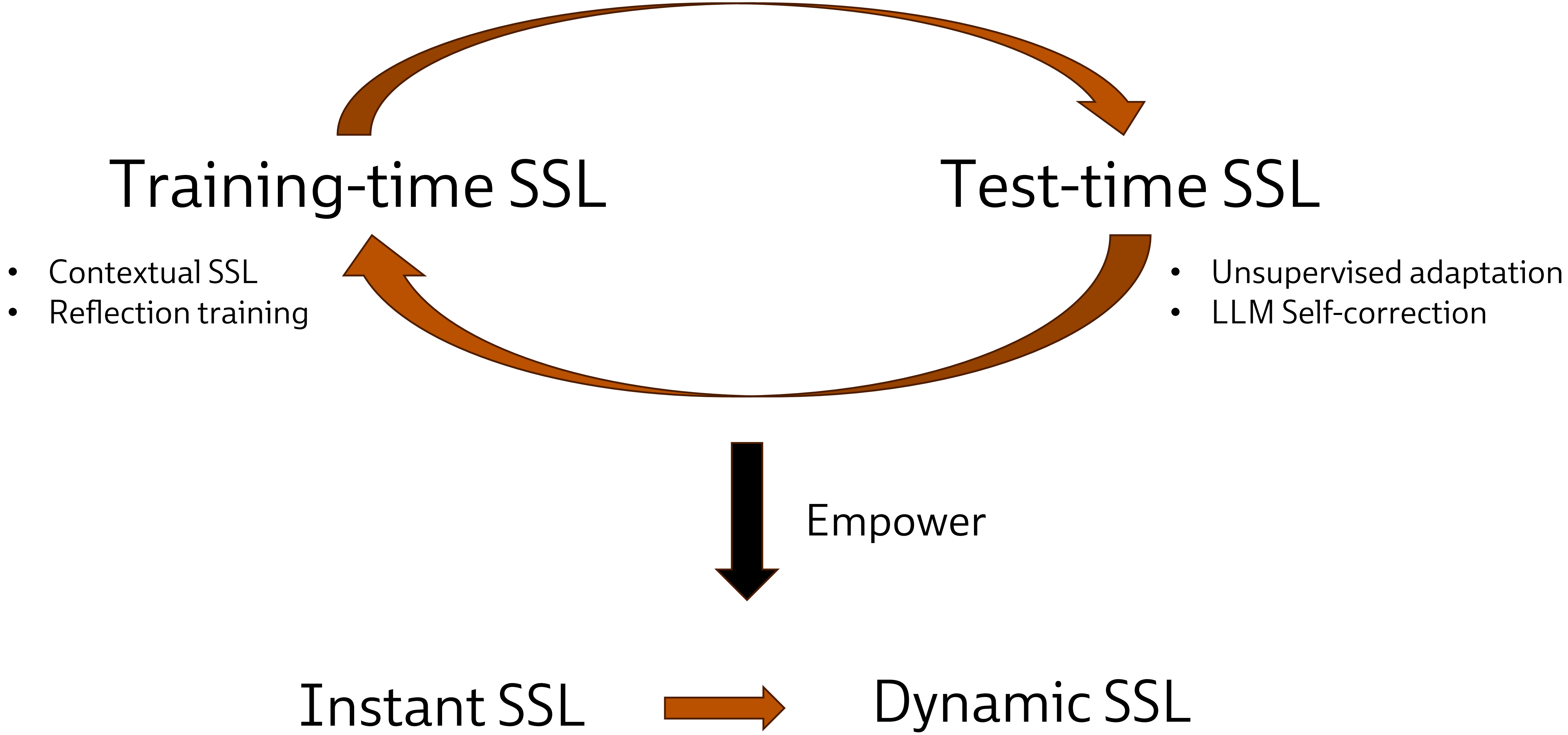
Self-adapt to Task Priors

Iterative Self-correction

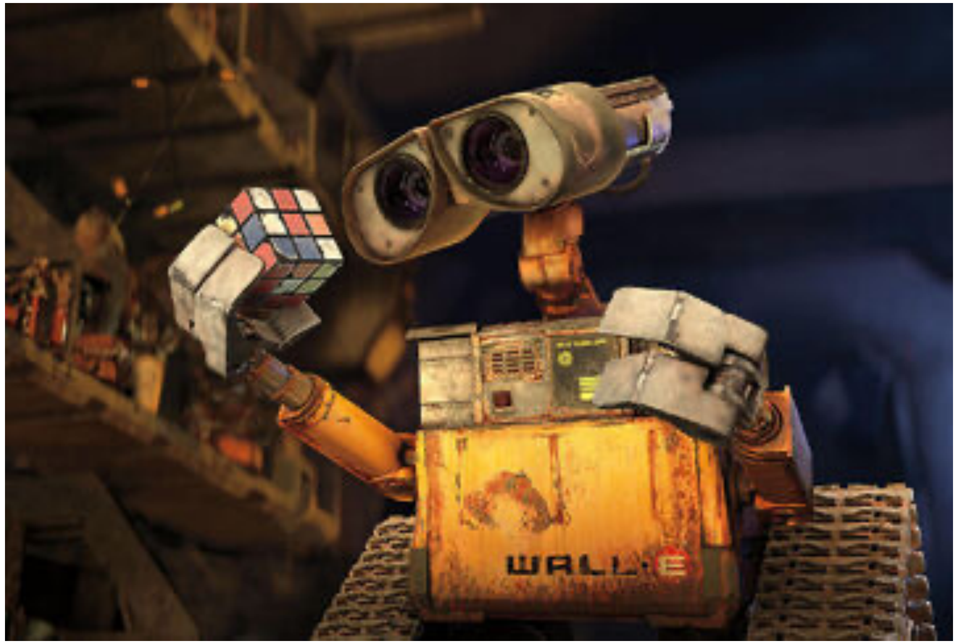


how language models refine predictions with self-reflection

Self-reflective prediction


Training-time SSL
Test-time SSL
• Contextual SSL
• Reflection training
• Unsupervised adaptation
• LLM Self-correction
Empower
Instant SSL
Dynamic SSL

A lot more to explore in test-time SSL!



scene understanding, exploration, planning, and interacting...

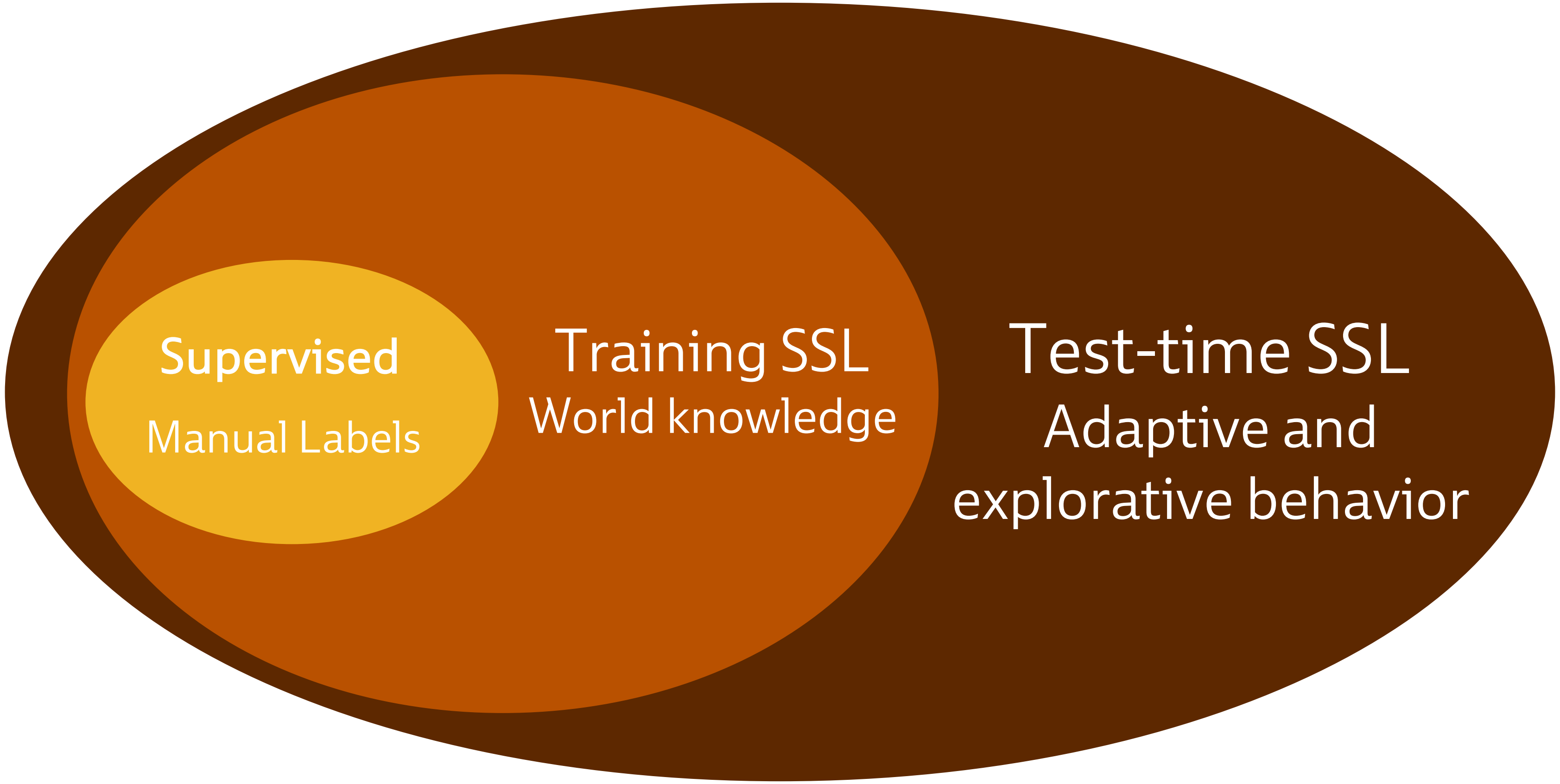
Covered Work

- Sharut Gupta*, Chenyu Wang*, **Yifei Wang*,** Tommi Jaakkola, and Stefanie Jegelka.
  In-Context Symmetries: Self-Supervised Learning through Contextual World Models.
  In NeurIPS, 2024.
  Oral Presentation (top 4) at NeurIPS 2024 SSL Workshop

- **Yifei Wang***, Yuyang Wu*, Zeming Wei, Stefanie Jegelka, and Yisen Wang.
  A Theoretical Understanding of Self-Correction through In-context Alignment.
  In NeurIPS 2024.
  Best Paper Award at ICML 2024 ICL Workshop.

* denotes equal authorship

A Full Picture



Thank You! Questions?