Skill-it! A data-driven skills framework for understanding and training language models

Fred Sala
University of Wisconsin-Madison

Joint work with Mayee F. Chen, Nicholas Roberts, Kush Bhatia, Jue Wang, Ce Zhang, Christopher Ré

NeurIPS 2023 Spotlight
Arxiv: https://arxiv.org/abs/2307.14430
Github: https://github.com/HazyResearch/skill-it

Bonus: making large pretrained models more robust
Zero-Shot Robustification of Zero-Shot Models

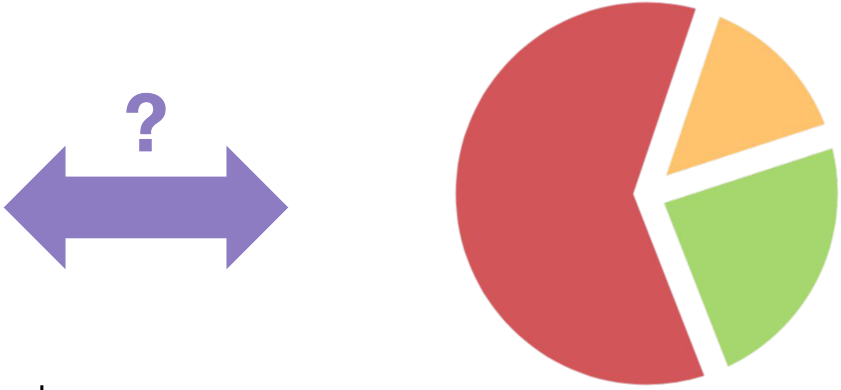
ICLR 2024
Arxiv: https://arxiv.org/pdf/2309.04344
Github: https://github.com/SprocketLab/roboshot

together.ai

Motivation

Large language models (LLMs) can do many things:

- Write code
- Chat with users
- Generate creative content

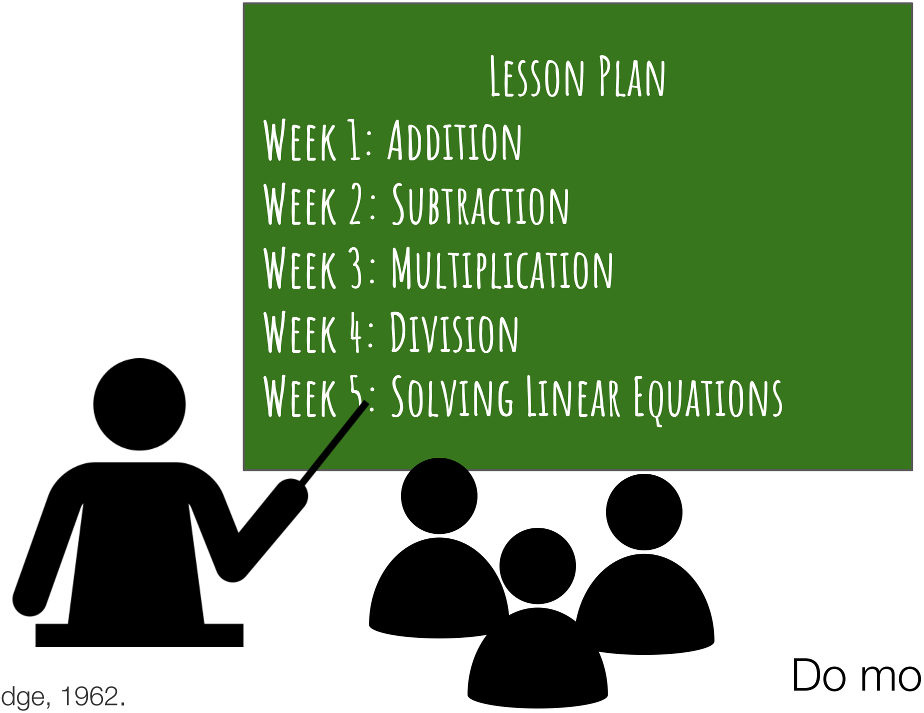


Training data mixture

How do we select data for them? What is the relationship between data and these capabilities?

Motivation

How do humans learn from data? Learn skills in a certain *order.*[1]



Lesson Plan

Week 1: Addition
Week 2: Subtraction
Week 3: Multiplication
Week 4: Division
Week 5: Solving Linear Equations

Do models also learn like this?

---

[1] Gagne. The acquisition of knowledge, 1962.

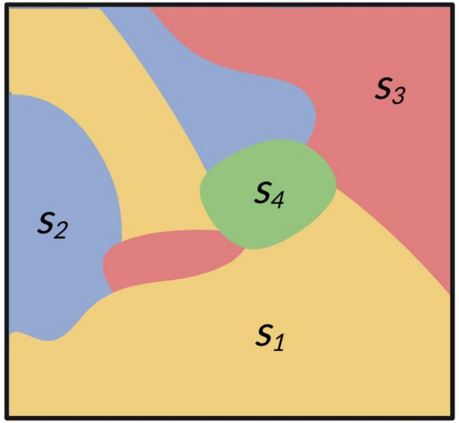
Our findings

1) There exist sets of *skills* associated with data that the LLM learns most efficiently in some particular order.
2) We can learn this order and exploit it to better select training data.

Definition: what is a skill?

Training data can be partitioned into subsets associated with ***skills***.

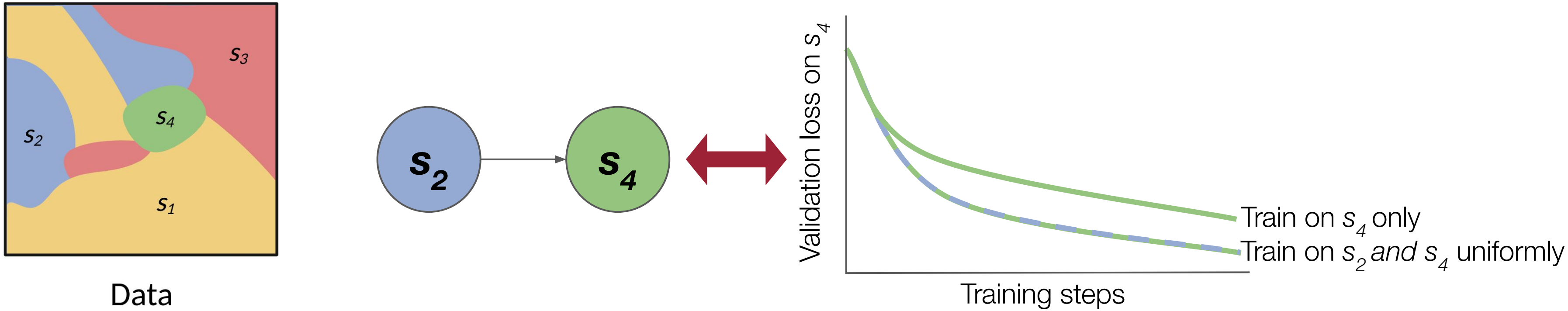
Examples: tasks, data sources, task categories.



Data

How do we define a meaningful order over skills?

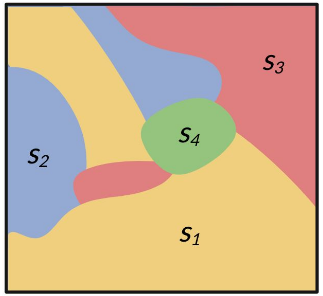
Given a set of skills S, we have a ***skills graph*** G encoding how skills are learned.



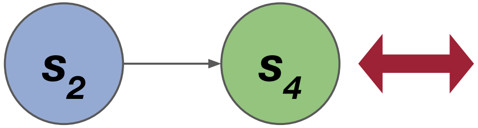
How do we define a meaningful order over skills?

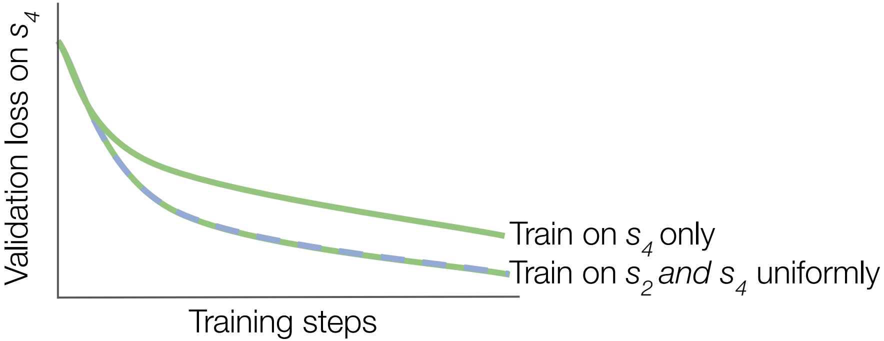
Given a set of skills S, we have a ***skills graph*** G encoding how skills are learned.

Ordered skills set: S whose skills graph is neither complete nor empty



Data



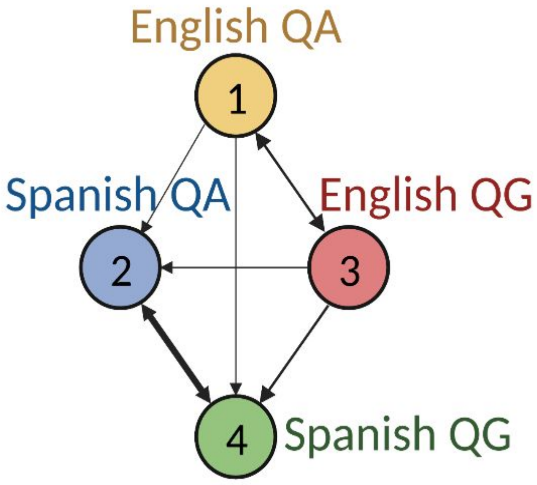


Ordered skill set example

Four skills: Spanish/English question answering and question generation from Natural Instructions[1]

Ordered skill set example

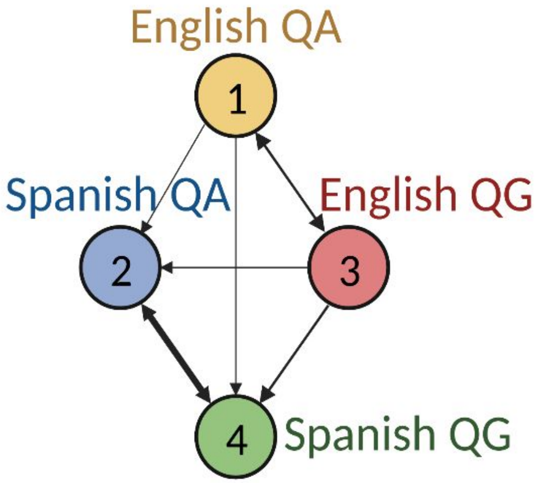
Four skills: Spanish/English question answering and question generation from Natural Instructions[1]



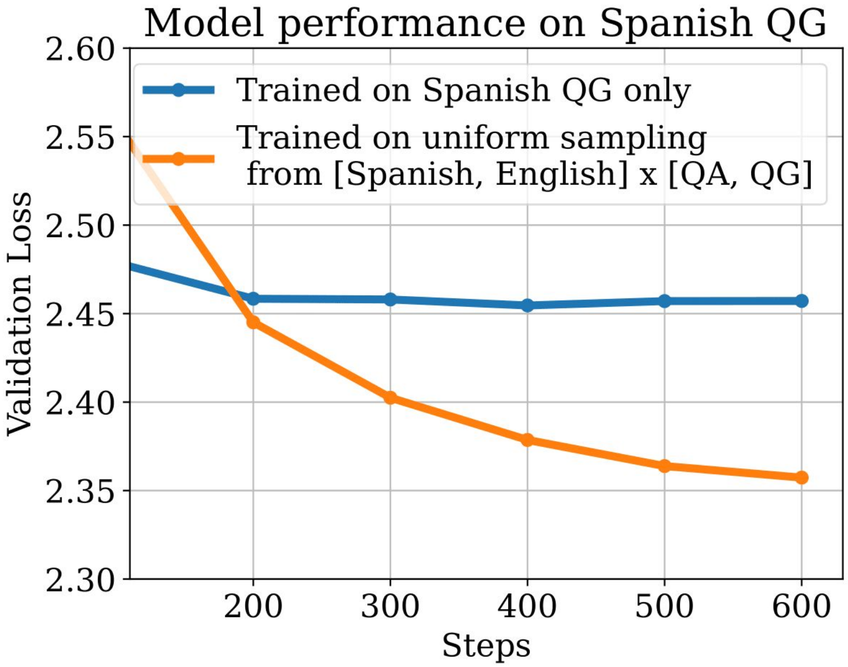
Ordered skill set

Ordered skill set example

Four skills: Spanish/English question answering and question generation from Natural Instructions[1]

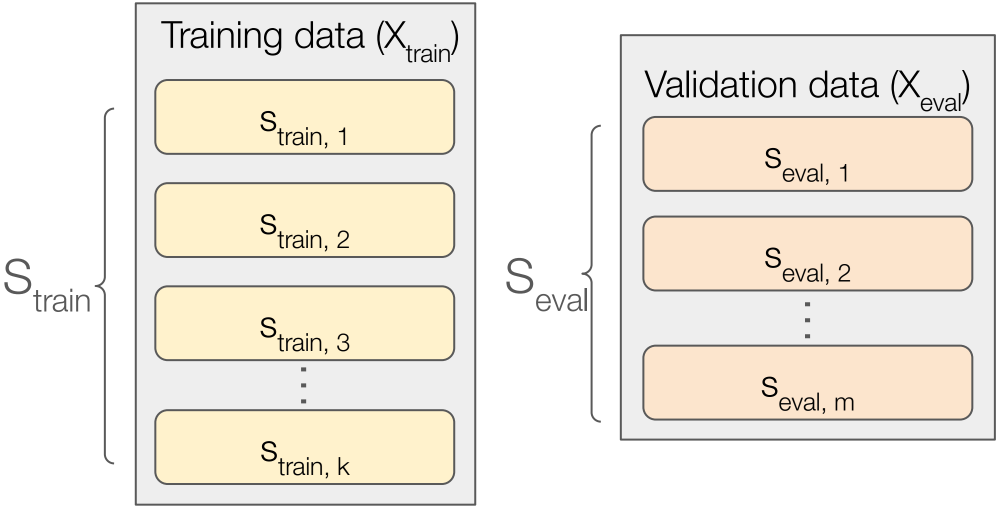


Ordered skill set



Skill-It data selection algorithm

Problem setup



Training data (X_{train})

S_{train}

$s_{train,1}$

$s_{train,2}$

$s_{train,3}$

⋮

$s_{train,k}$

Validation data (X_{eval})

S_{eval}

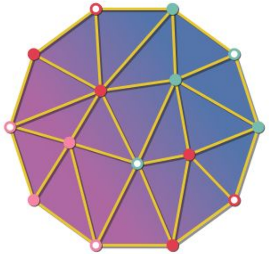
$s_{eval,1}$

$s_{eval,2}$

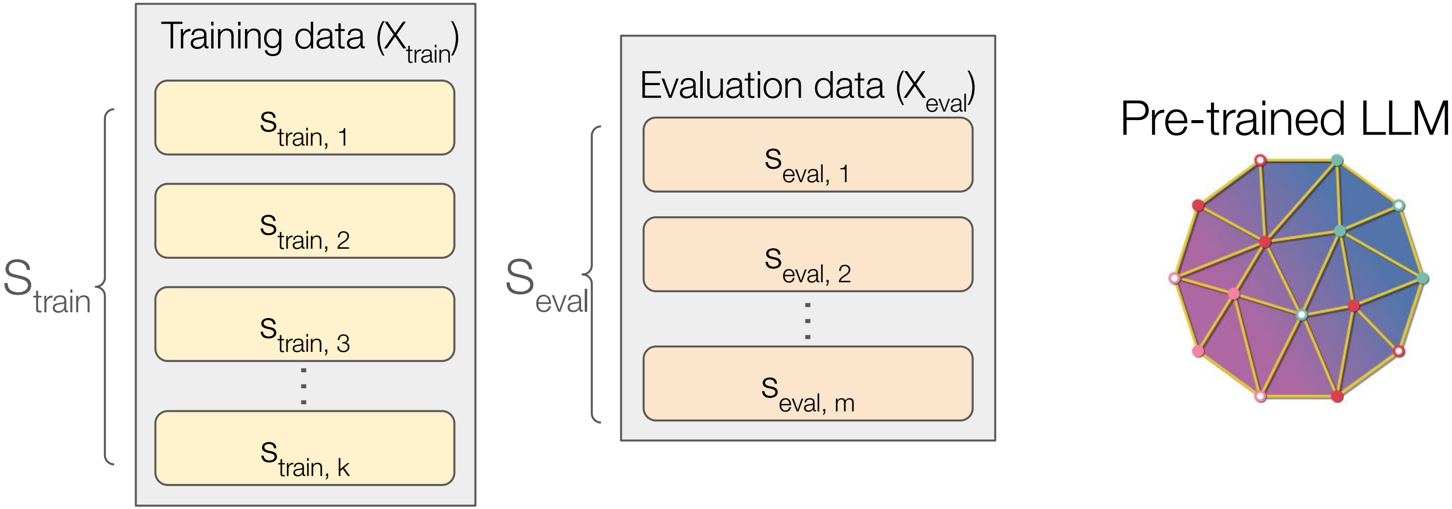
⋮

$s_{eval,m}$

Pre-trained LLM f



Problem setup



S_{train}

Training data (X_{train})

$s_{train, 1}$

$s_{train, 2}$

$s_{train, 3}$

⋮

$s_{train, k}$

S_{eval}

Evaluation data (X_{eval})

$s_{eval, 1}$

$s_{eval, 2}$

⋮

$s_{eval, m}$

Pre-trained LLM f

Goal: how to order and select n samples from X_{train} for f to perform well on X_{eval}

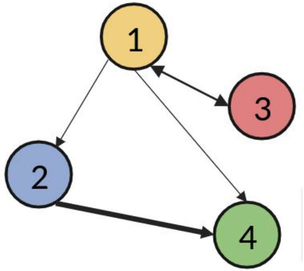
Skill-It algorithm

Online optimization problem: split training into *T* rounds, adjust mixture of skills at each round

Skill-It algorithm

Online optimization problem: split training into *T* rounds, adjust mixture of skills at each round

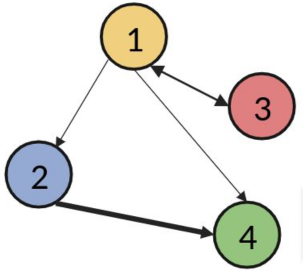
1. Learn skills graph



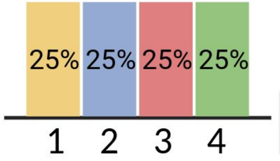
Skill-It algorithm

Online optimization problem: split training into *T* rounds, adjust mixture of skills at each round

1. Learn skills graph



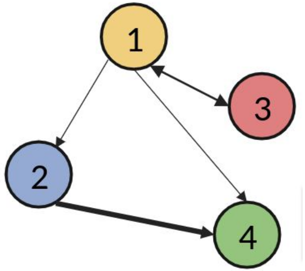
2. Initialize training data mixture p_1



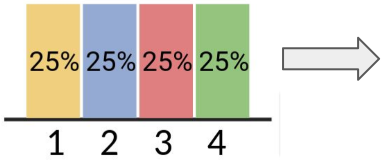
Skill-It algorithm

Online optimization problem: split training into T rounds, adjust mixture of skills at each round

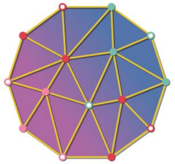
1. Learn skills graph



2. Initialize training data mixture p_1

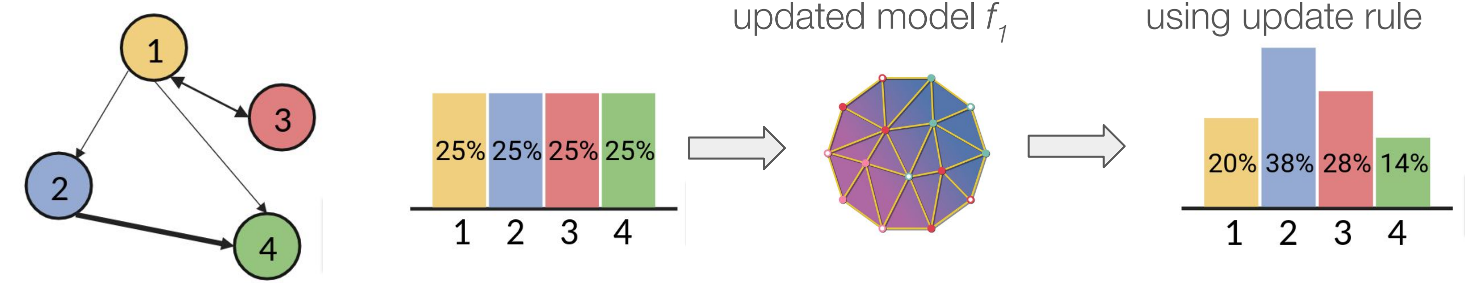


3. Train model for one round with p_1 and get updated model f_1



Skill-It algorithm

Online optimization problem: split training into T rounds, adjust mixture of skills at each round



1. Learn skills graph

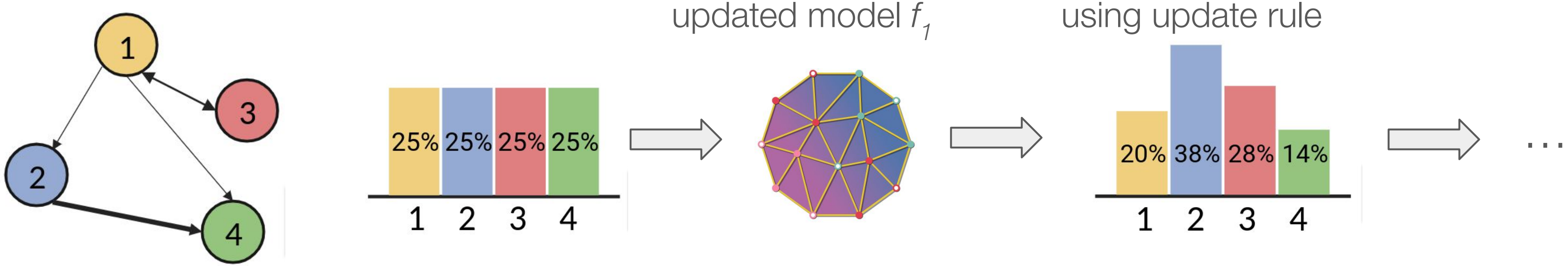
2. Initialize training data mixture p_1

3. Train model for one round with p_1 and get updated model f_1

4. Construct next round's data mixture p_2 using update rule

Skill-It algorithm

Online optimization problem: split training into T rounds, adjust mixture of skills at each round

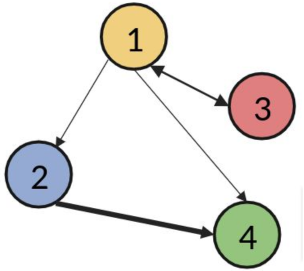


1. Learn skills graph
2. Initialize training data mixture p_1
3. Train model for one round with p_1 and get updated model f_1
4. Construct next round's data mixture p_2 using update rule
5. Repeat for T total rounds

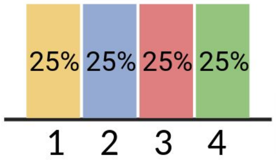
Skill-It algorithm

Online optimization problem: split training into *T* rounds, adjust mixture of skills at each round

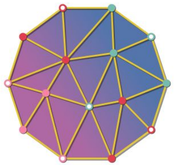
1. Learn skills graph



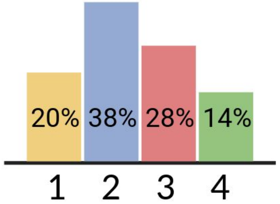
2. Initialize training data mixture p_1



3. Train model for one round with p_1 and get updated model f_1



4. Construct next round's data mixture p_2 using update rule



5. Repeat for *T* total rounds

…

Update rule:

$$p_{t+1}^{i} = p_t^{i} \exp\left(\eta \sum_{j=1}^{m} A_{ij} L_{\text{eval},j}(f_t)\right)$$

Skill-It algorithm

Online optimization problem: split training into *T* rounds, adjust mixture of skills at each round

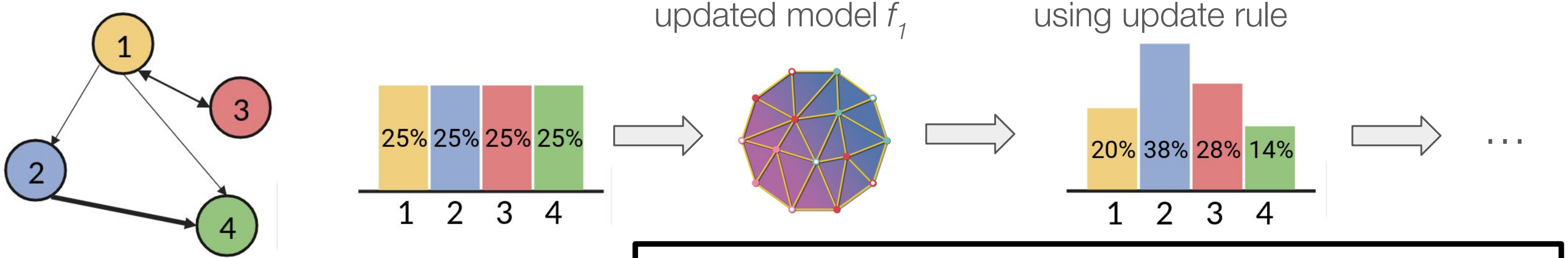
1. Learn skills graph

2. Initialize training data mixture p_1

3. Train model for one round with p_1 and get updated model f_1

4. Construct next round's data mixture p_2 using update rule

5. Repeat for *T* total rounds



Update rule:

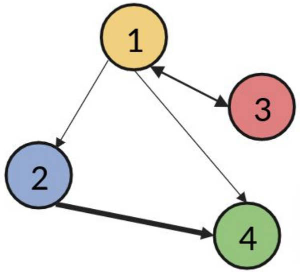
$$p^i_{t+1} = p^i_t \exp\left(\eta \sum_{j=1}^{m} \boxed{A_{ij}} L_{\text{eval},j}(f_t)\right)$$

Weight on $s_{train,\, i}$ depends on what $s_{eval,\, j}$ it influences in skills graph

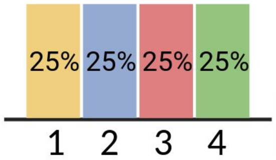
Skill-It algorithm

Online optimization problem: split training into T rounds, adjust mixture of skills at each round

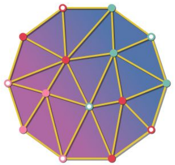
1. Learn skills graph



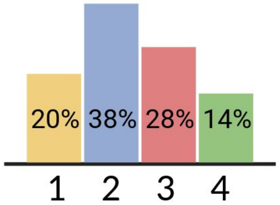
2. Initialize training data mixture p_1



3. Train model for one round with p_1 and get updated model f_1



4. Construct next round's data mixture p_2 using update rule



5. Repeat for T total rounds

…

Update rule:

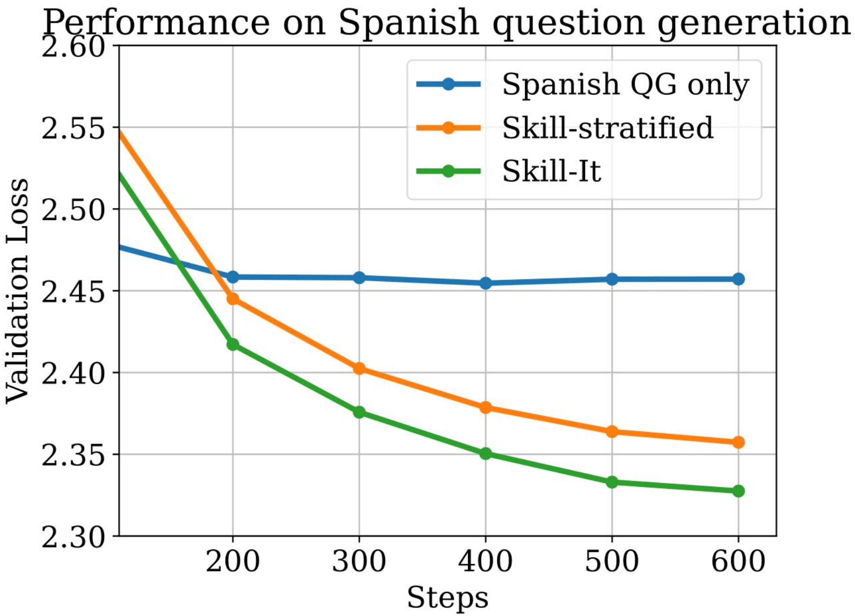
$$p_{t+1}^i = p_t^i \exp\left(\eta \sum_{j=1}^m \boxed{A_{ij}} \boxed{L_{\text{eval},j}(f_t)}\right)$$

Weight on $s_{train, i}$ depends on what $s_{eval, j}$ it influences in skills graph

Weight is proportional to how well model is learning each $s_{eval, j}$

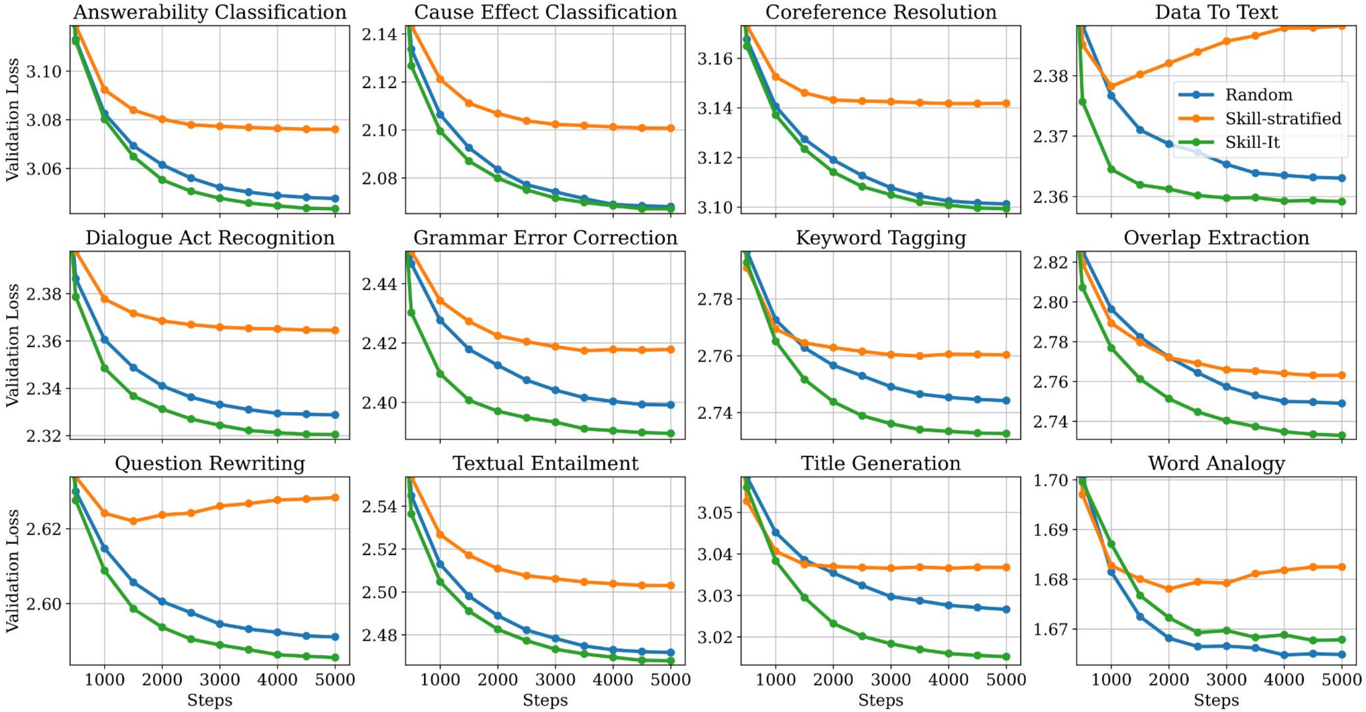
Results

Natural Instructions fine-tuning results



Natural instructions out-of-domain results

Skill-It outperforms baselines on 11/12 task categories



Skill-It!

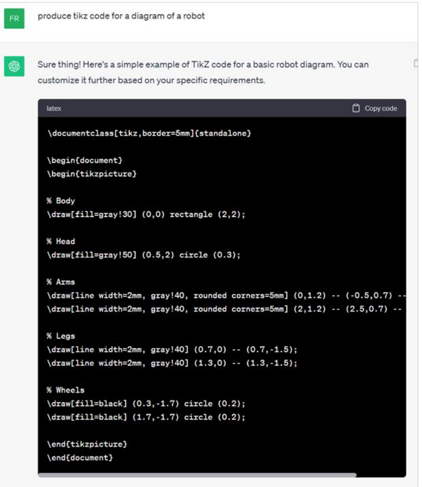
See paper for:

- How to efficiently learn a skills graph (on smaller models)
- Insights into unsupervised skills discovery
- Additional experimental results (more synthetics, real datasets, and a 3B parameter model trained on RedPajama)

Arxiv: https://arxiv.org/abs/2307.14430

Zero-Shot Robustification of Zero-Shot Models

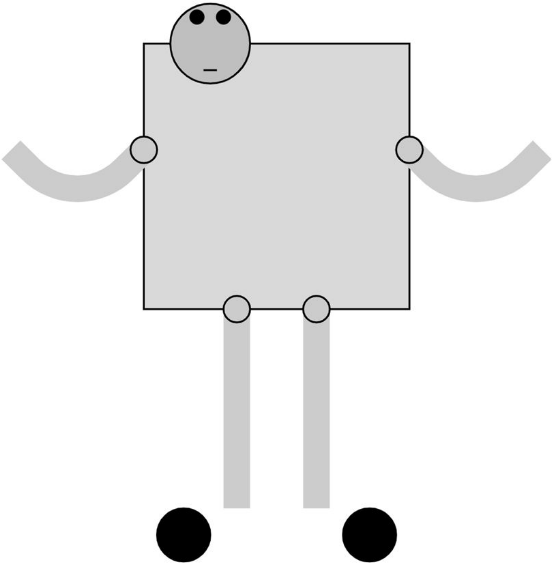
Zero-shot models are a powerful new paradigm,



produce tikz code for a diagram of a robot

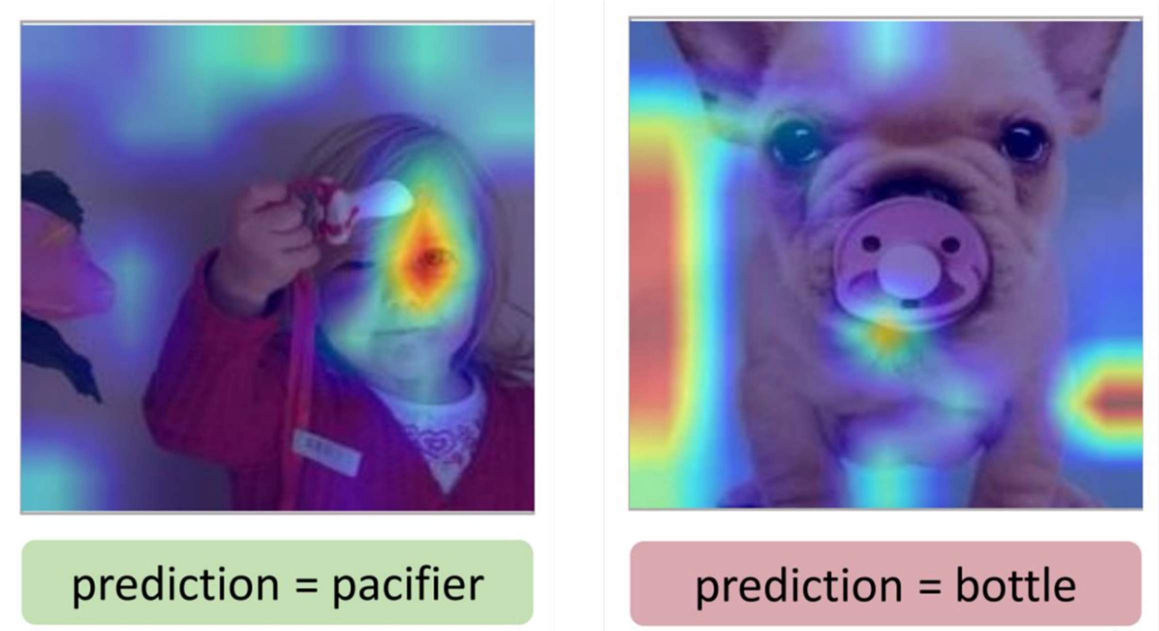
Sure thing! Here's a simple example of TikZ code for a basic robot diagram. You can customize it further based on your specific requirements.

```latex
\documentclass[tikz,border=5mm]{standalone}

\begin{document}
\begin{tikzpicture}

% Body
\draw[fill=gray!30] (0,0) rectangle (2,2);

% Head
\draw[fill=gray!50] (0.5,2) circle (0.3);

% Arms
\draw[line width=2mm, gray!40, rounded corners=5mm] (0,1.2) -- (-0.5,0.7) --
\draw[line width=2mm, gray!40, rounded corners=5mm] (2,1.2) -- (2.5,0.7) --

% Legs
\draw[line width=2mm, gray!40] (0.7,0) -- (0.7,-1.5);
\draw[line width=2mm, gray!40] (1.3,0) -- (1.3,-1.5);

% Wheels
\draw[fill=black] (0.3,-1.7) circle (0.2);
\draw[fill=black] (1.7,-1.7) circle (0.2);

\end{tikzpicture}
\end{document}
```



But Their Performance Isn’t Always Great

Pretraining data leads to all sorts of potential quirks,

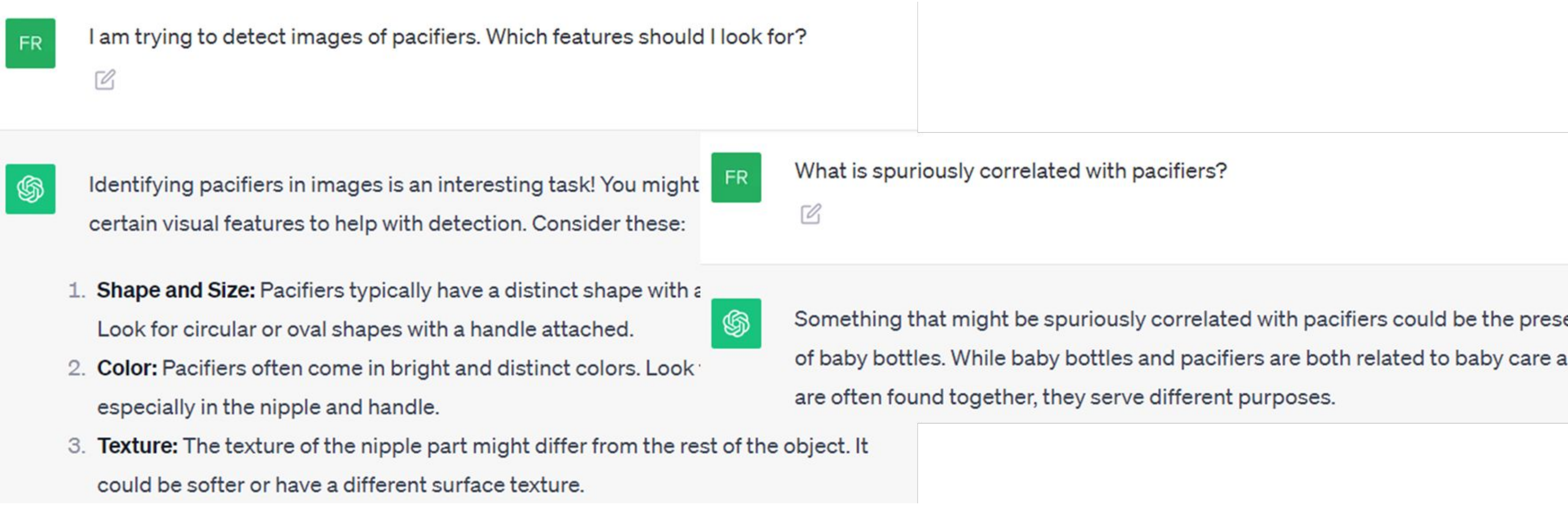


Yang et ‘23

How Do We Fix These?

One idea: these models might know useful concepts

- But don’t use these during prediction



FR I am trying to detect images of pacifiers. Which features should I look for?

Identifying pacifiers in images is an interesting task! You might certain visual features to help with detection. Consider these:

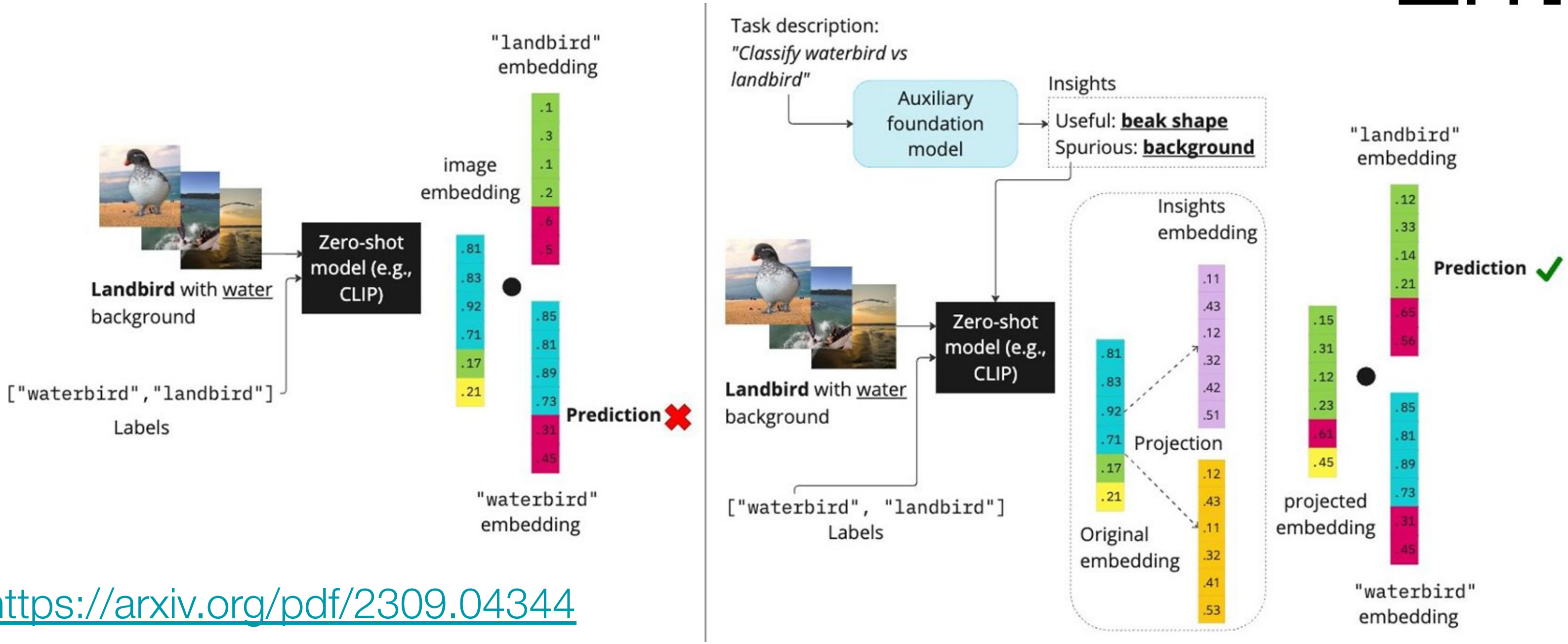
1. **Shape and Size:** Pacifiers typically have a distinct shape with Look for circular or oval shapes with a handle attached.
2. **Color:** Pacifiers often come in bright and distinct colors. Look especially in the nipple and handle.
3. **Texture:** The texture of the nipple part might differ from the rest of the object. It could be softer or have a different surface texture.

FR What is spuriously correlated with pacifiers?

Something that might be spuriously correlated with pacifiers could be the prese of baby bottles. While baby bottles and pacifiers are both related to baby care a are often found together, they serve different purposes.

How Do We Fix These?

Our work: two-stage process: obtain insights, modify representations



Arxiv: https://arxiv.org/pdf/2309.04344

Thank You!

fredsala@cs.wisc.edu