Online Representation Learning on the Open Web

Ellis Brown
Advisor: Deepak Pathak
Computer Science Department, School of Computer Science
Carnegie Mellon University

Carnegie Mellon University
Computer Science Department

Committee
Deepak Pathak
Deva Ramanan
Alexei A. Efros

Consider this scenario:

Consider this scenario:

Task: classify bird species

Consider this scenario:

Task: classify bird species

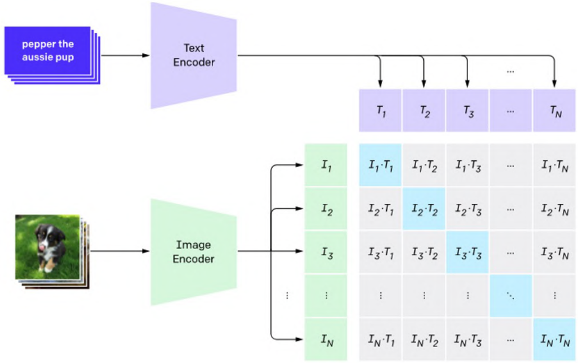
Question: what do you do to get max performance?

Current Paradigm: Transfer Learning

Current Paradigm: Transfer Learning

1. Some large dataset

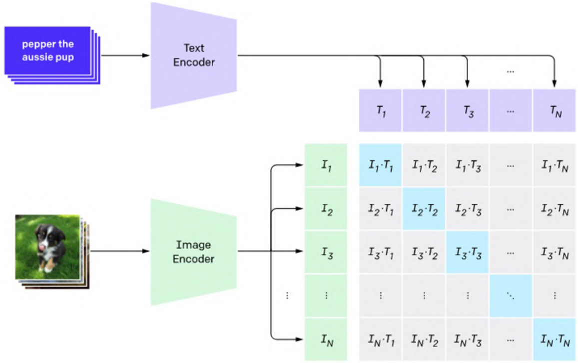
Current Paradigm: Transfer Learning



1. Some large dataset

2. Pretrained Model
(AlexNet, ResNet, CLIP)

Current Paradigm: Transfer Learning



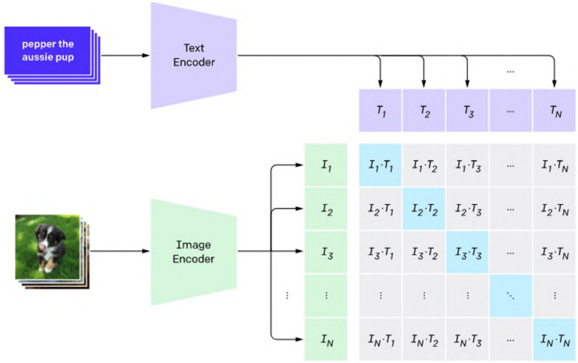
1. Some large dataset

2. Pretrained Model (AlexNet, ResNet, CLIP)

3. Fine-tune on target

Current Paradigm: Transfer Learning

1. Some large dataset



2. Pretrained Model (AlexNet, ResNet, CLIP)

3. Fine-tune on target

Let's talk about this

Scale is getting bigger and bigger…

Scale is getting bigger and bigger…

1.2M

Scale is getting bigger and bigger…

OpenAI
CLIP

1.2M **400M**

Scale is getting bigger and bigger…

OpenAI
CLIP

LAION-5B
Large-scale Artificial Intelligence Open Network

1.2M	400M	5,000M

OpenAI

CLIP

LAION-5B

Large-scale Artificial Intelligence Open Network

IMAGENET

OpenAI
CLIP

LAION-5B
Large-scale Artificial Intelligence Open Network

Static Datasets

IMAGENET

OpenAI

CLIP

LAION-5B

Large-scale Artificial Intelligence Open Network

Static Datasets

- Snapshot of the internet

IMAGENET

OpenAI

CLIP

LAION-5B

Large-scale Artificial Intelligence Open Network

Static Datasets

- Snapshot of the internet
- Instantly stale

IMAGENET

OpenAI

CLIP

LAION-5B

Large-scale Artificial Intelligence Open Network

Static Datasets

- Snapshot of the internet
- Instantly stale
- Curator's bias

IMAGENET

OpenAI

CLIP

LAION-5B

Large-scale Artificial Intelligence Open Network

Static Datasets

- Snapshot of the internet
- Instantly stale
- Curator's bias
- Worse for long-tail tasks

IMAGENET

OpenAI

CLIP

LAION-5B

Large-scale Artificial Intelligence Open Network

Static Datasets

- Snapshot of the internet
- Instantly stale
- Curator’s bias
- Worse for long-tail tasks
- …

Static Datasets

Internet: Billions of images uploaded **each day**

IMAGENET

OpenAI

CLIP

LAION-5B

Large-scale Artificial Intelligence Open Network

Static Datasets

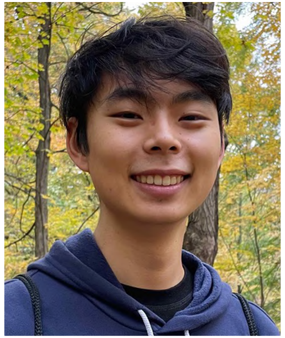
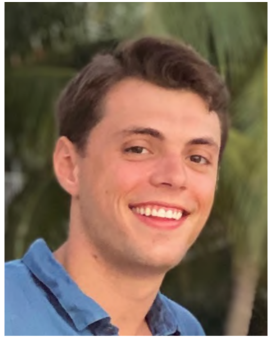
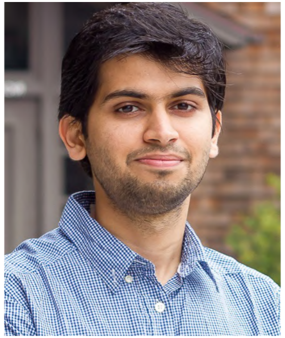
Internet: Billions of images uploaded **each day**

Static datasets are <u>miniscule and out-of-date</u> in comparison to the Internet!

Internet Explorer

Targeted Representation Learning on the Open Web

Alexander C. Li*, Ellis Brown*, Alexei A. Efros, Deepak Pathak

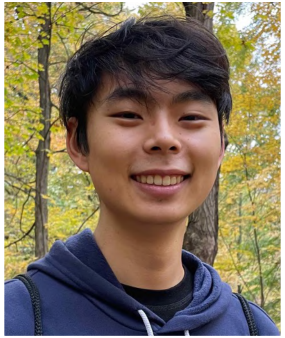
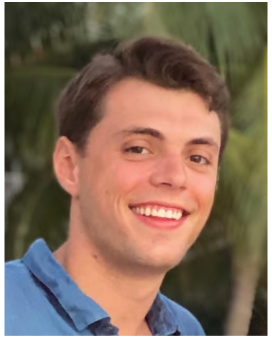
Accepted at **ICML 2023**

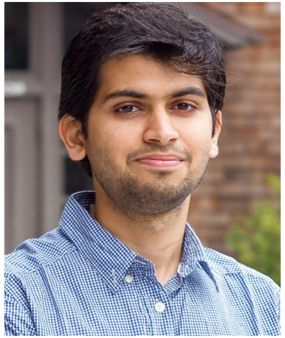
ICML
International Conference
On Machine Learning

Internet Explorer

Targeted Representation Learning on the Open Web

Alexander C. Li*, Ellis Brown*, Alexei A. Efros, Deepak Pathak



Our proposal

Our proposal

Treat *Internet* itself as a dataset

Our proposal

Treat *Internet* itself as a dataset

open-ended

Our proposal

Treat *Internet* itself as a dataset

open-ended

constantly growing

Our proposal

Treat *Internet* itself as a dataset

open-ended

constantly growing

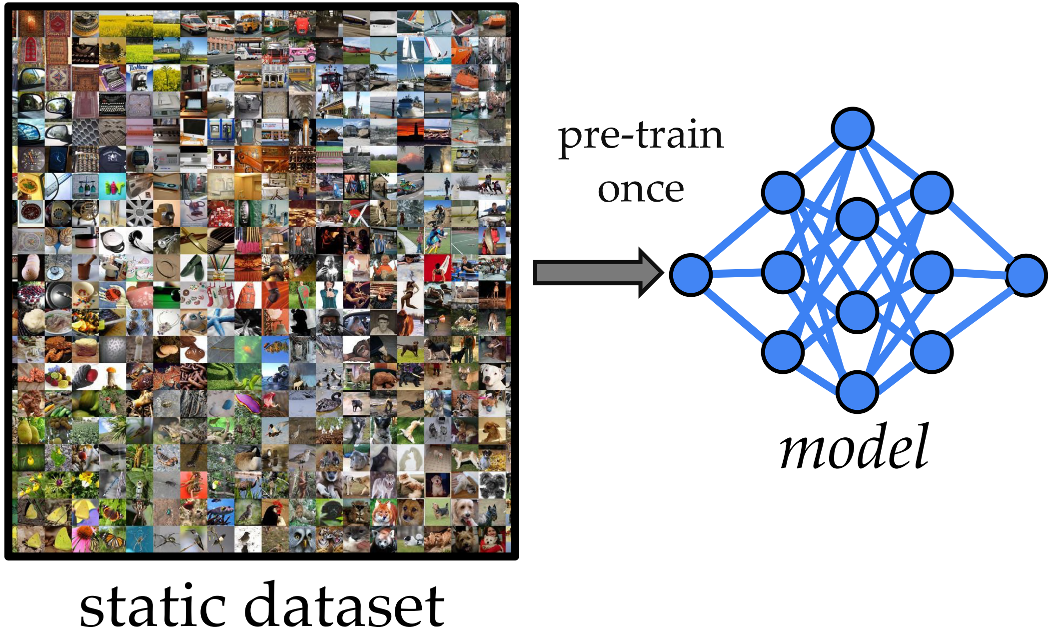
always up-to-date

Current paradigm

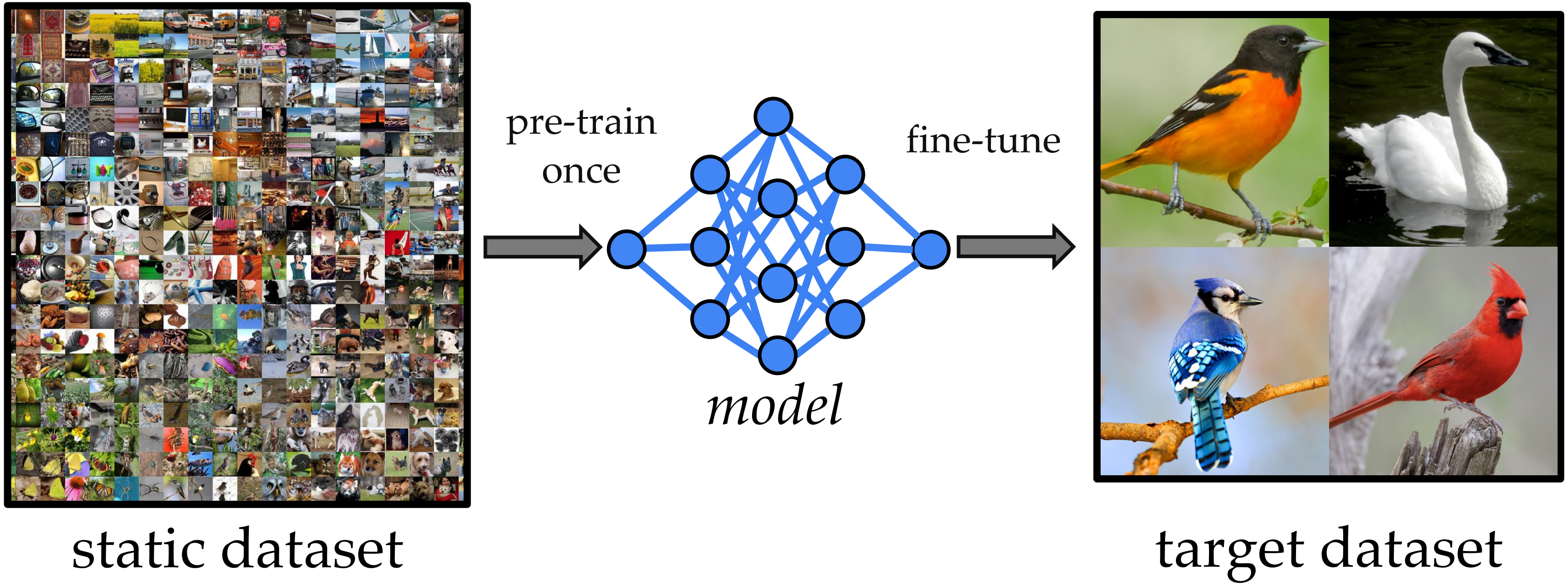
Current paradigm

static dataset

Current paradigm



Current paradigm

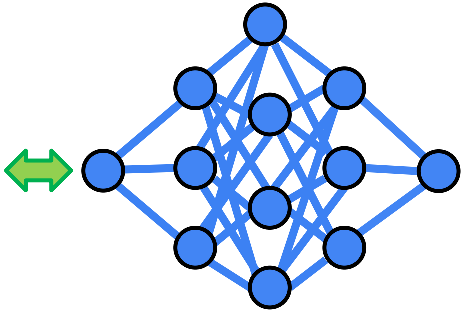


Our setting

target dataset

Our setting

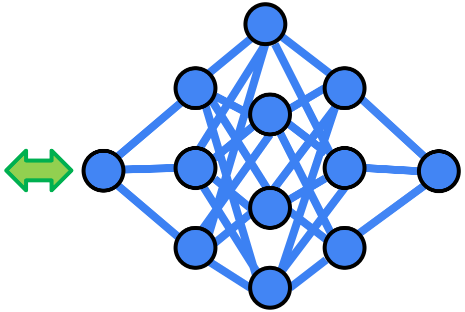
target dataset



model

Our setting

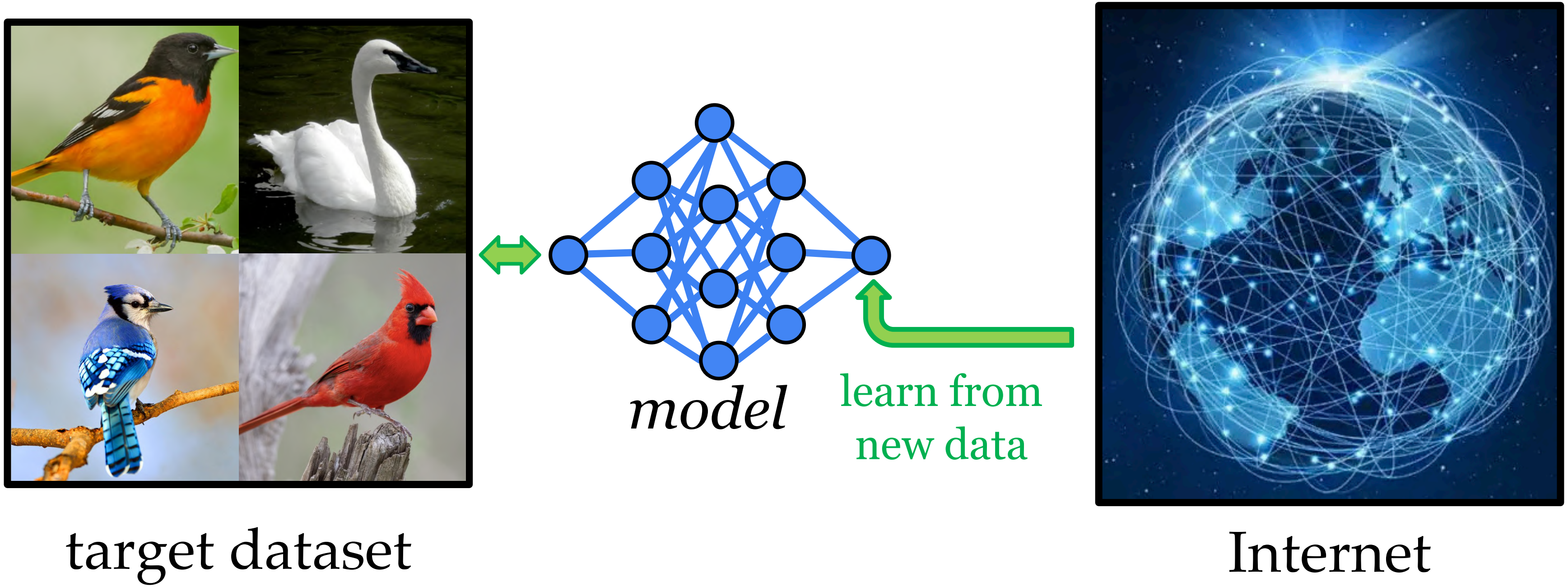
target dataset



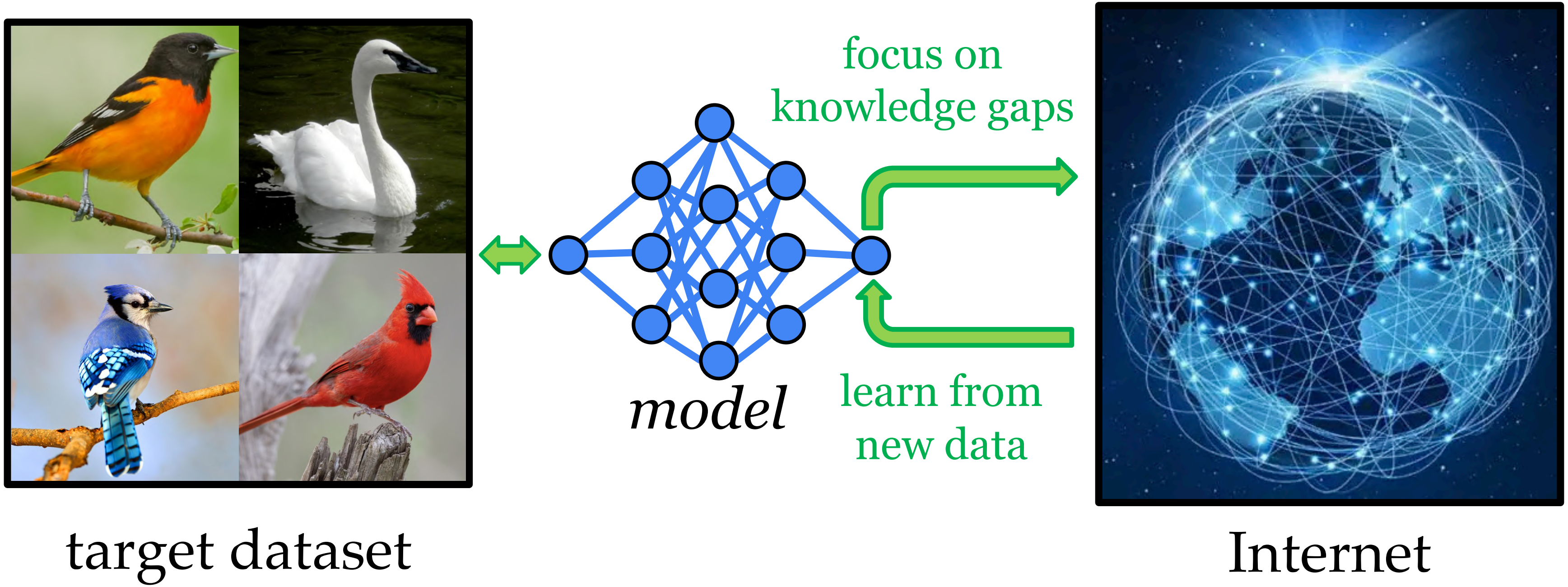
model

Internet

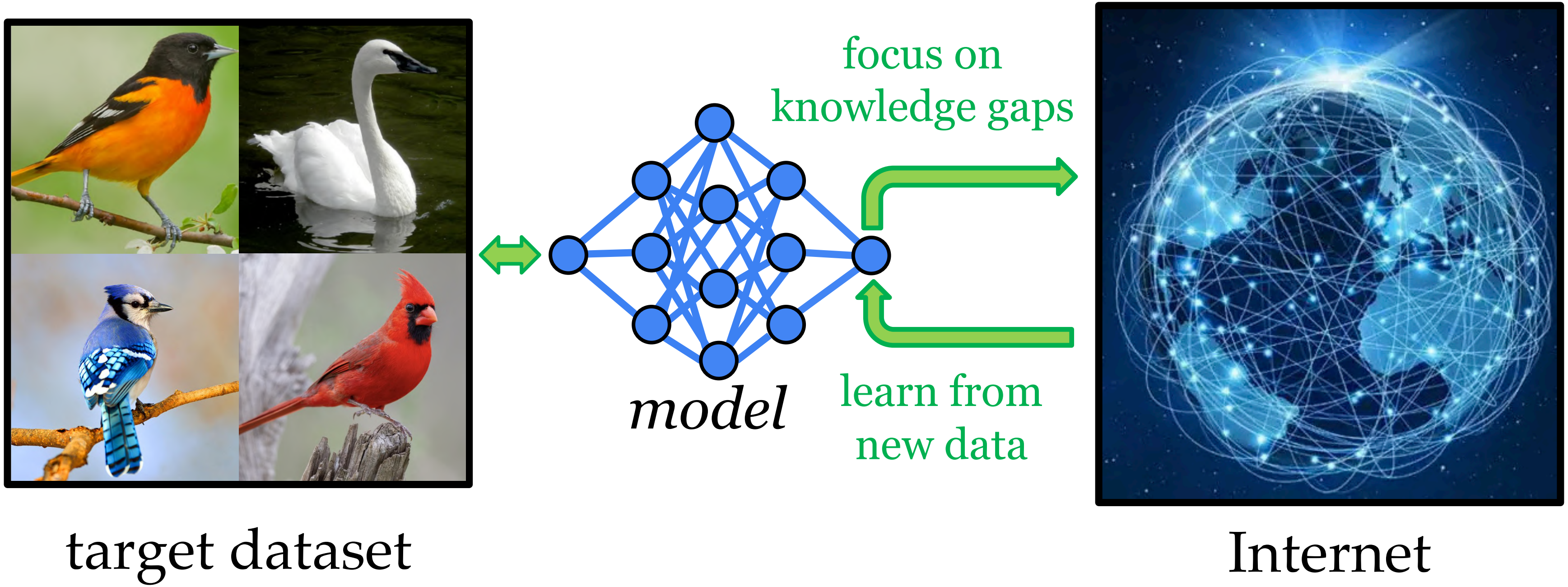
Our setting



Our setting


focus on
knowledge gaps
model
learn from
new data
target dataset
Internet

Our setting



“Internet Explorer”

What can we do with the full breadth of the Internet?

What can we do with the full breadth of the Internet?

Learn features for any task

What can we do with the full breadth of the Internet?

Learn features for any task

Cover long-tail corner cases

What can we do with the full breadth of the Internet?

Learn features for any task

Cover long-tail corner cases

Find up-to-date data

Challenges

Challenges

- What to search for?

Challenges

- What to search for?
- How to search for it?

Challenges

- What to search for?
- How to search for it?
- What data is good?

Challenges

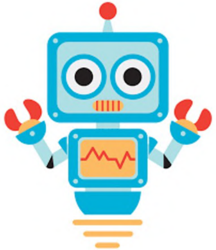
- What to search for?
- How to search for it?
- What data is good?
- How to integrate the data into our model?

Analogy to Reinforcement Learning

Analogy to Reinforcement Learning

Robot Explorer

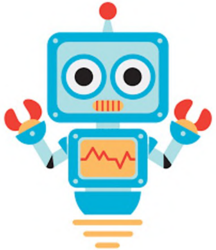
Analogy to Reinforcement Learning



Agent

Robot Explorer

Analogy to Reinforcement Learning

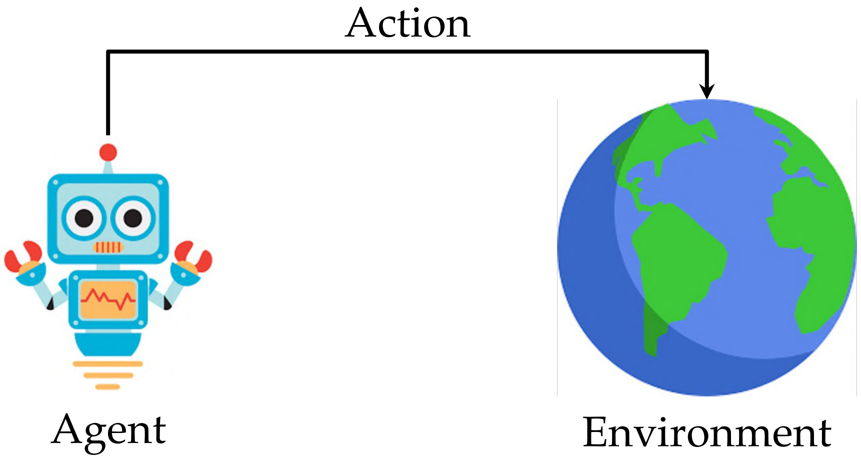


Agent

Environment

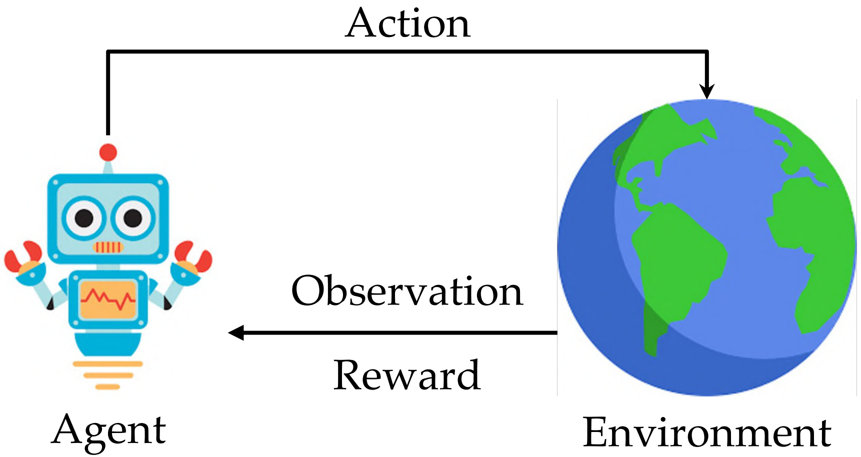
Robot Explorer

Analogy to Reinforcement Learning



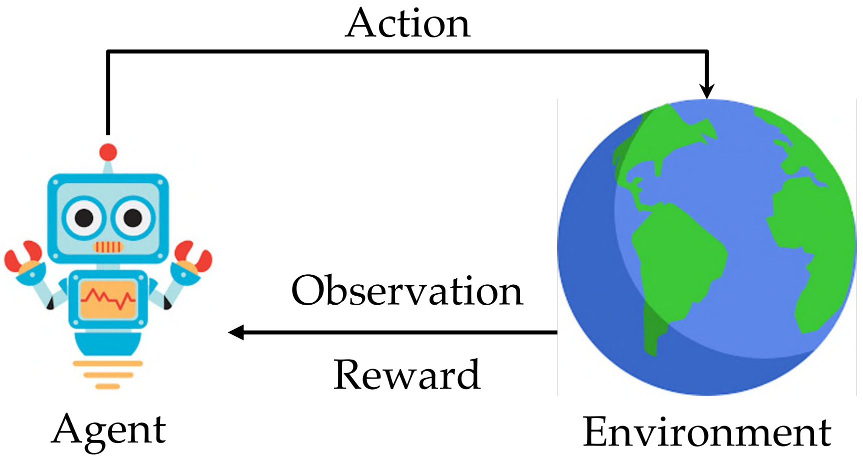
Robot Explorer

Analogy to Reinforcement Learning



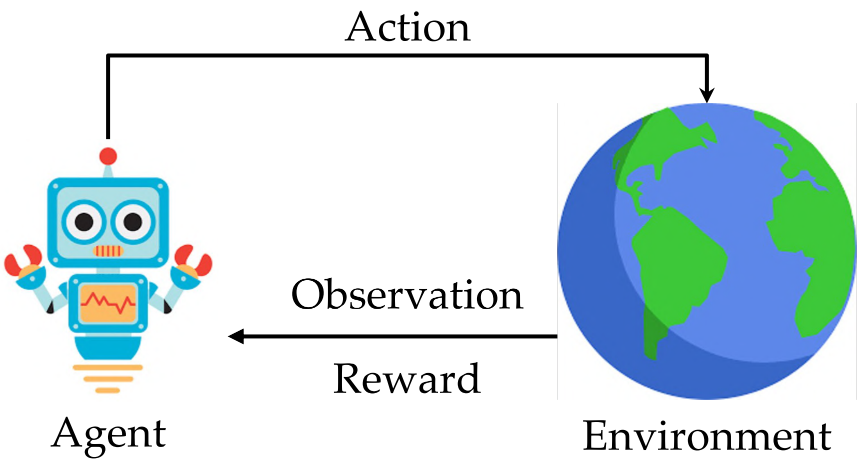
Robot Explorer

Analogy to Reinforcement Learning



Robot Explorer

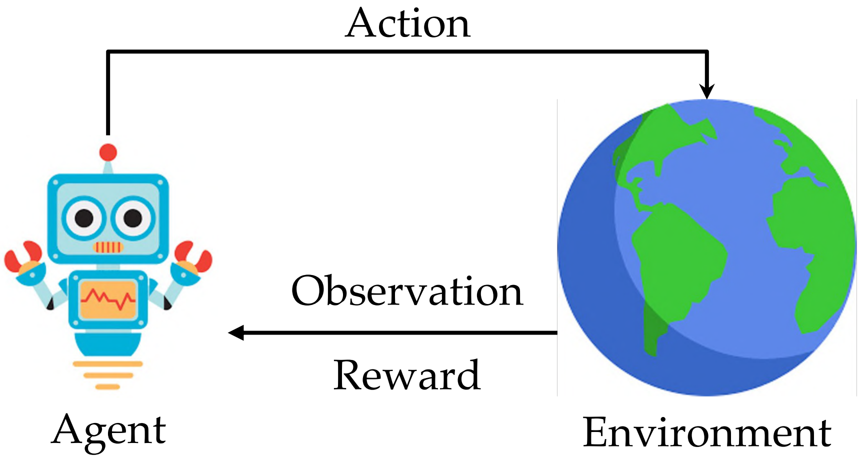
Analogy to Reinforcement Learning



Robot Explorer

Internet Explorer

Analogy to Reinforcement Learning

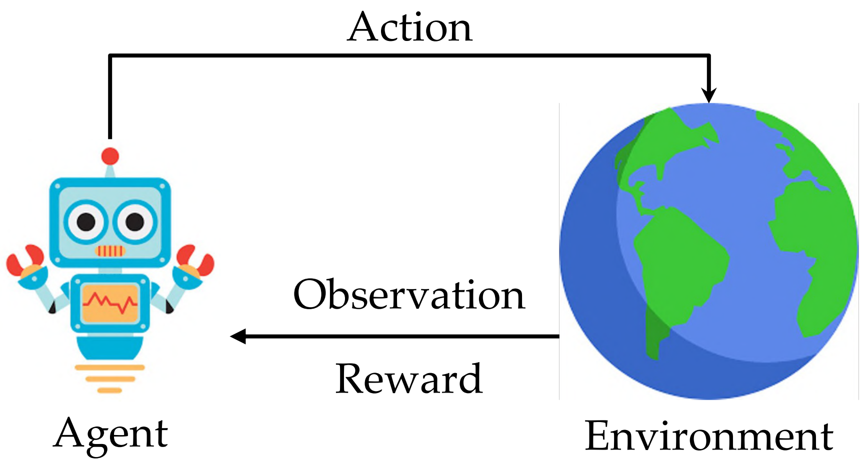


Environment → Internet

Robot Explorer

Internet Explorer

Analogy to Reinforcement Learning



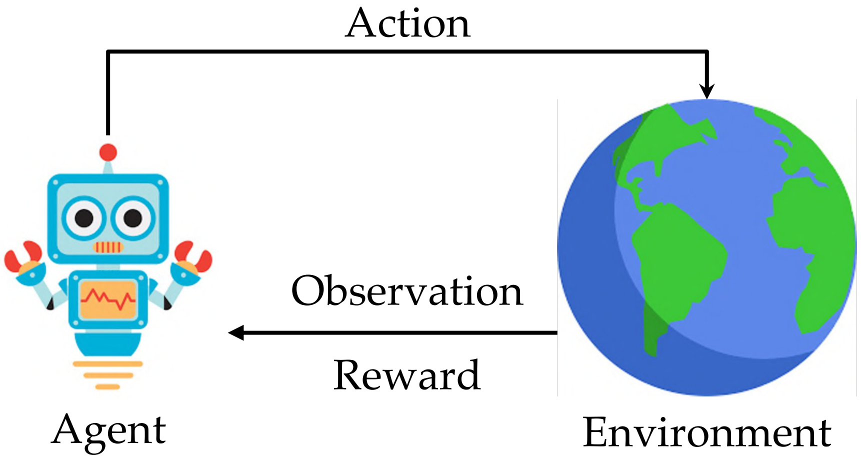
Robot Explorer

Environment → Internet

Action → search engine queries

Internet Explorer

Analogy to Reinforcement Learning



Robot Explorer

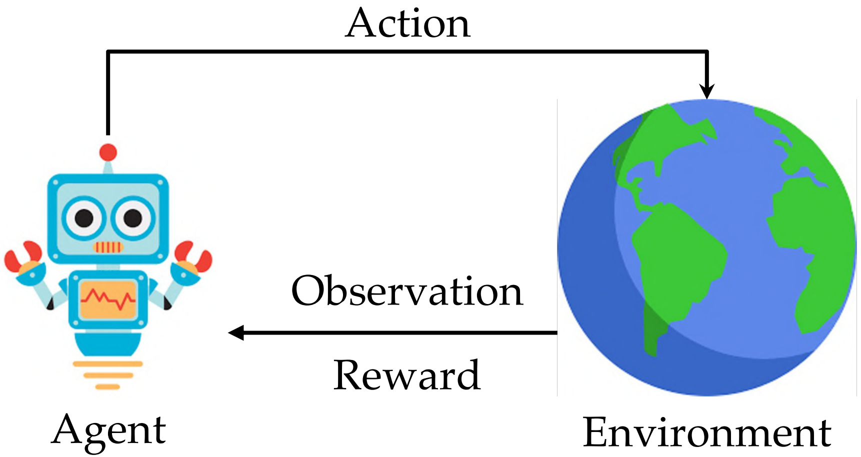
Environment → Internet

Action → search engine queries

Observation → Internet results

Internet Explorer

Analogy to Reinforcement Learning



Robot Explorer

Environment → Internet

Action → search engine queries

Observation → Internet results

Reward → relevant training data

Internet Explorer

Internet Explorer Method

Internet Explorer Method

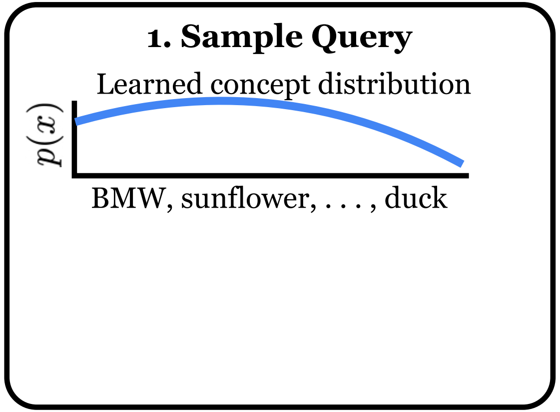
1. Sample Query

Internet Explorer Method

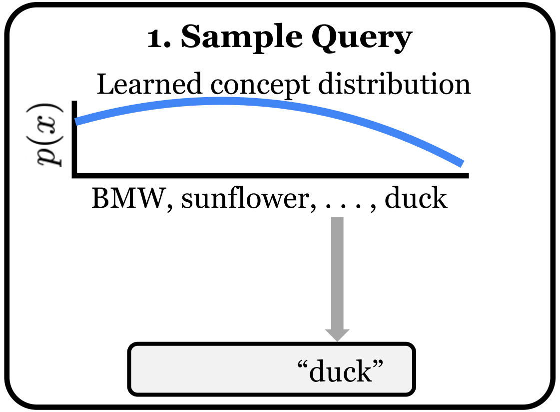
1. Sample Query

BMW, sunflower, . . . , duck

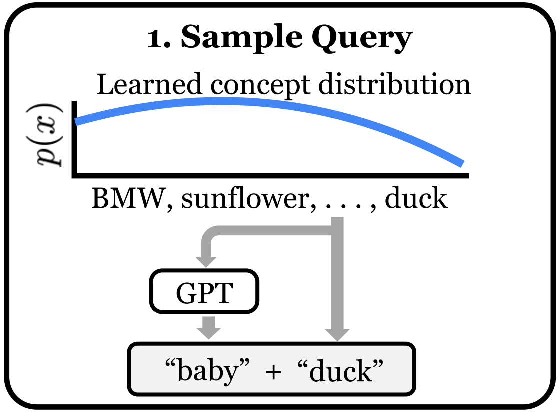
Internet Explorer Method



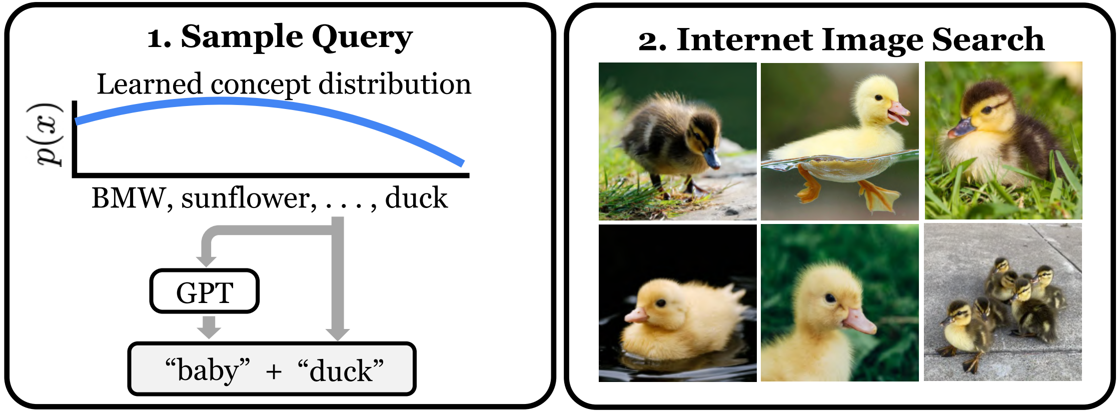
Internet Explorer Method



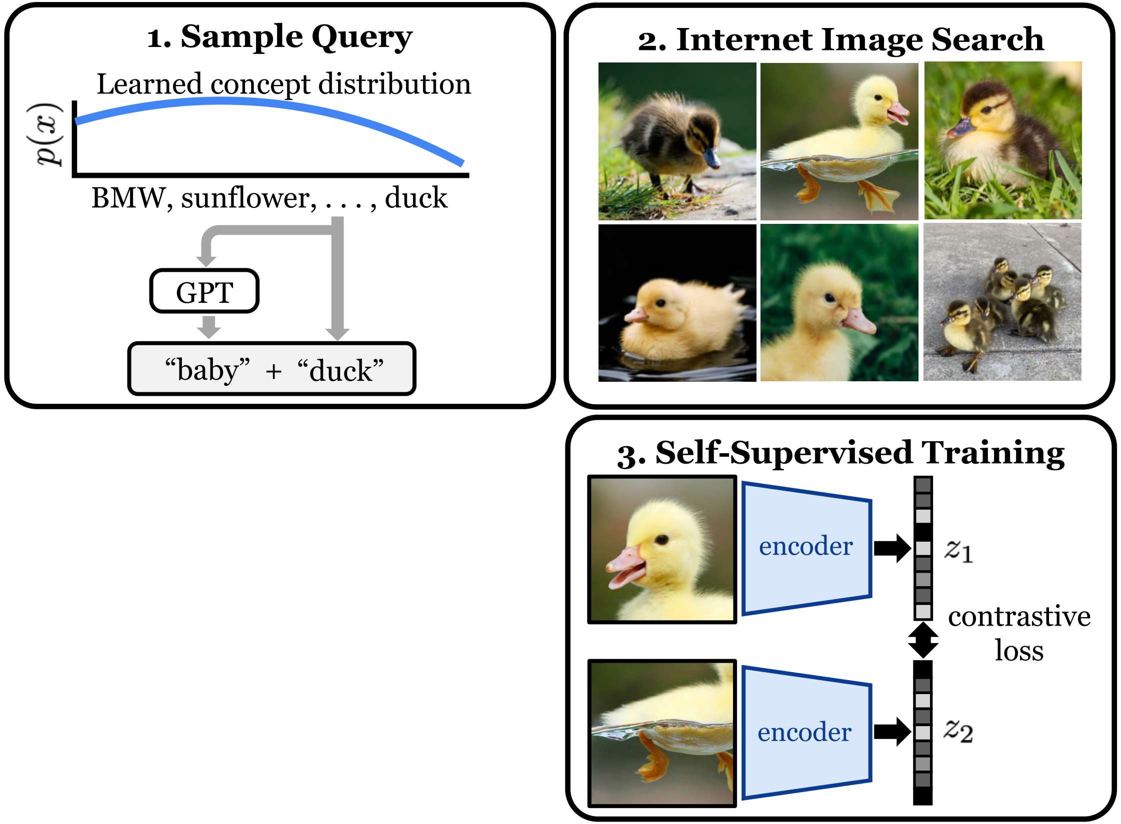
Internet Explorer Method



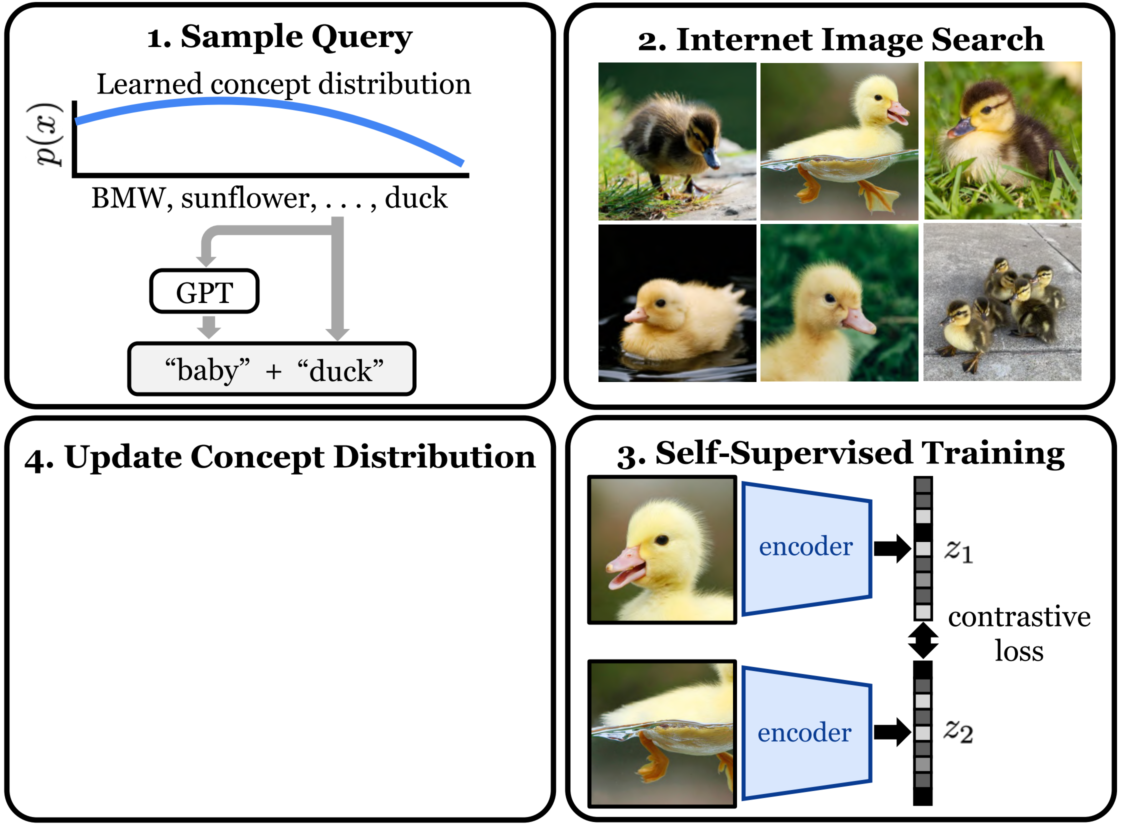
Internet Explorer Method



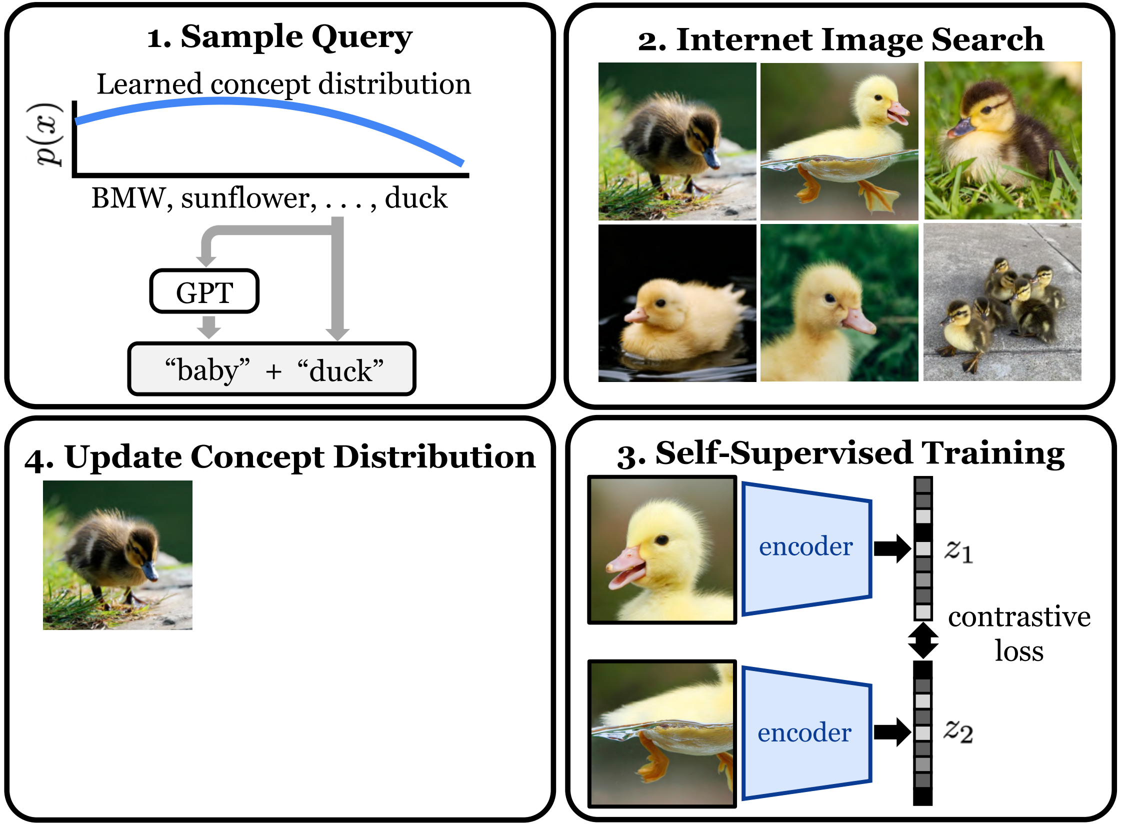
Internet Explorer Method



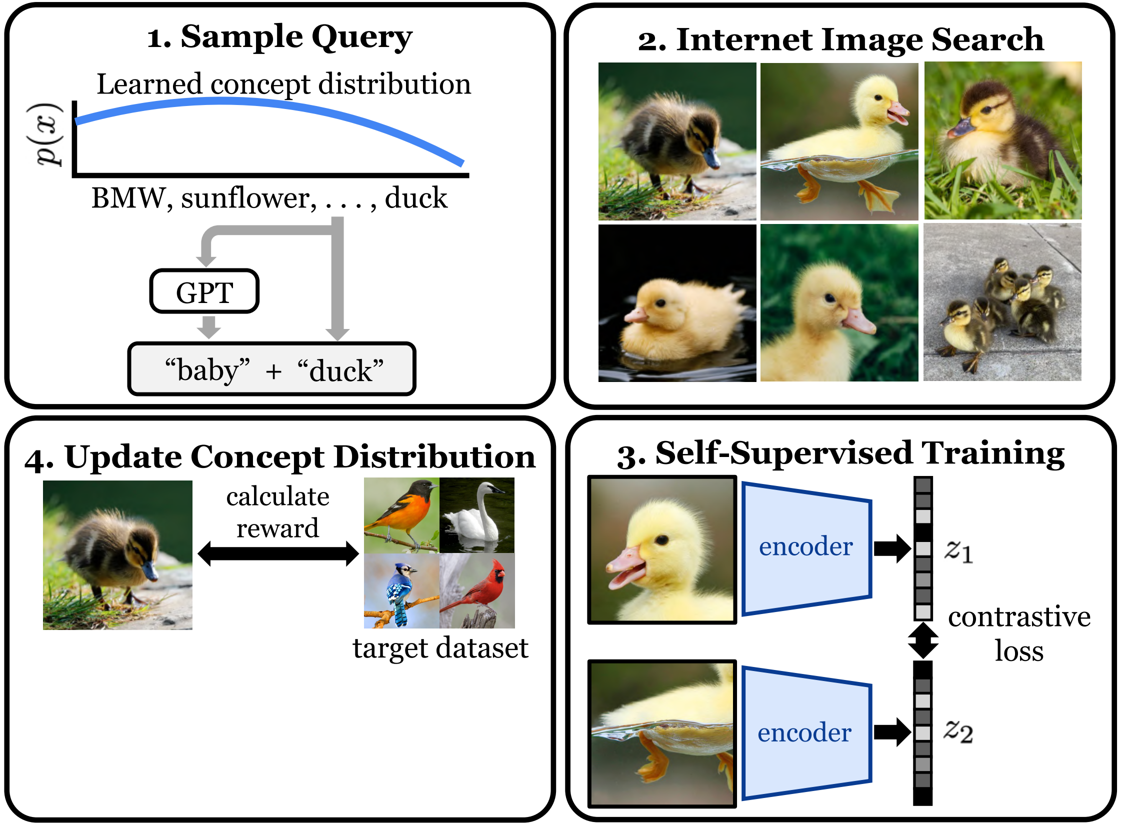
Internet Explorer Method



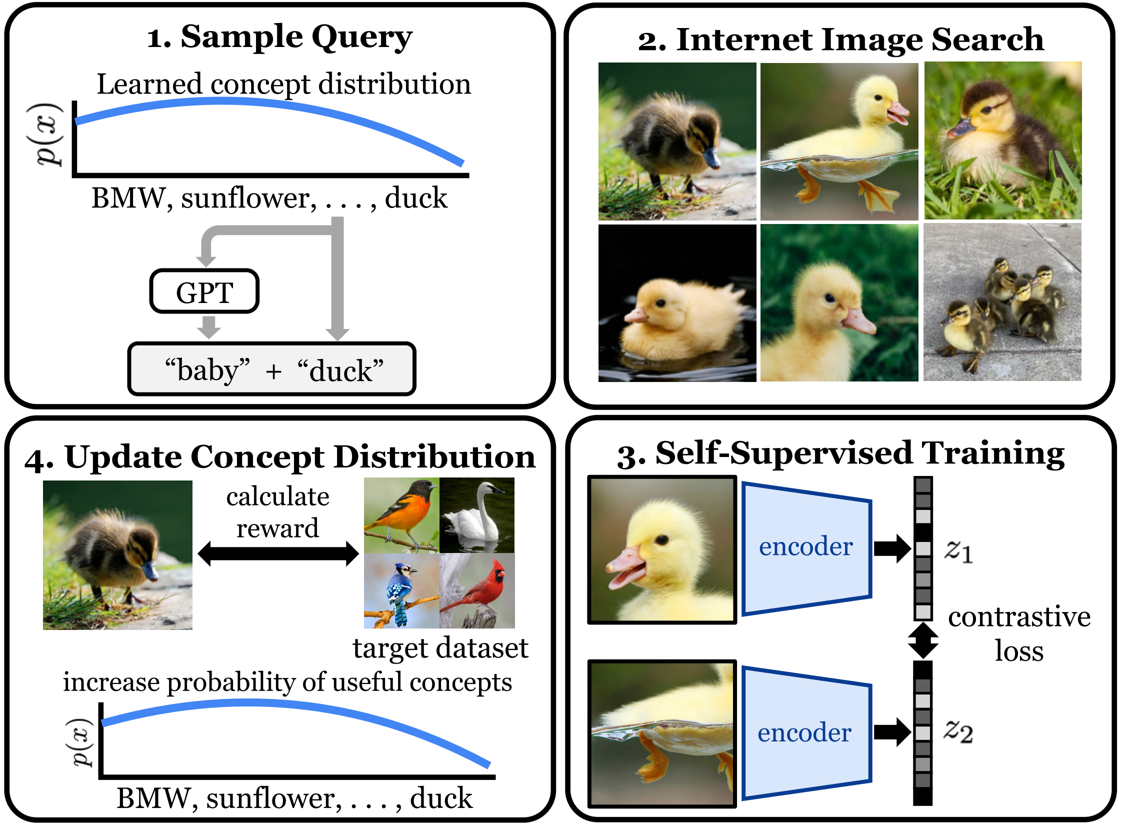
Internet Explorer Method



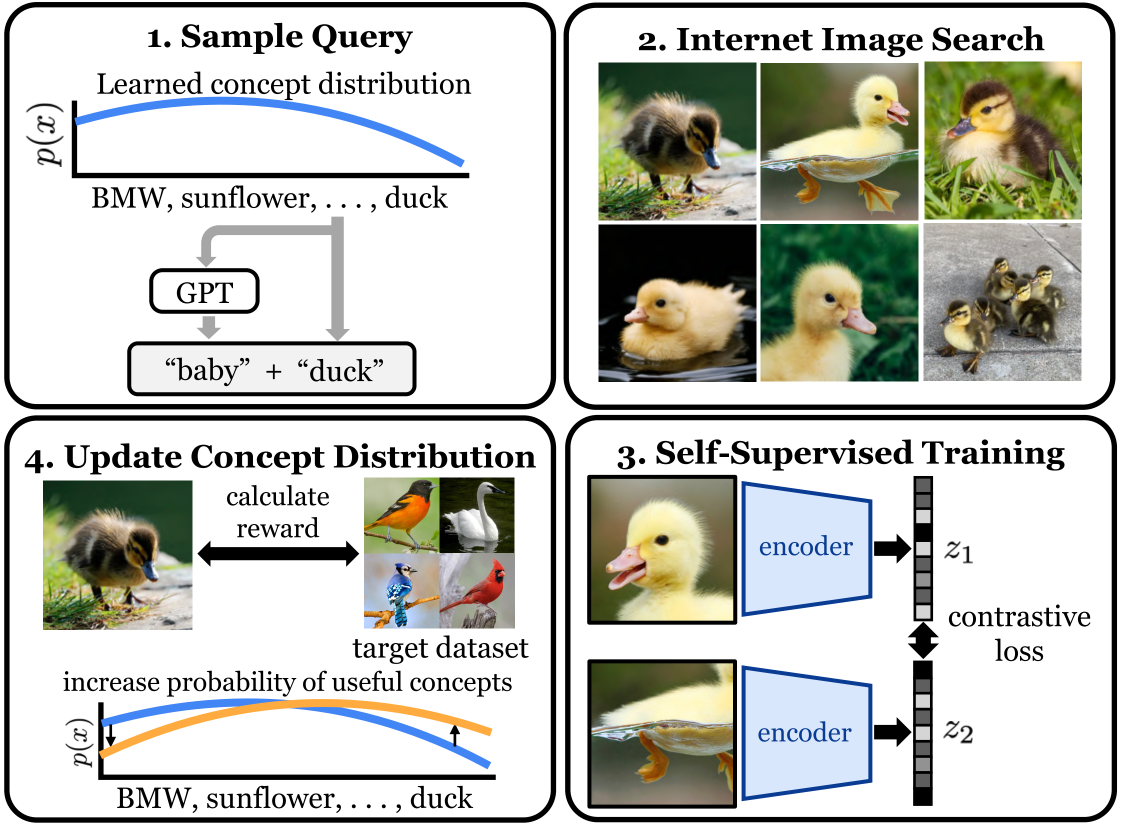
Internet Explorer Method



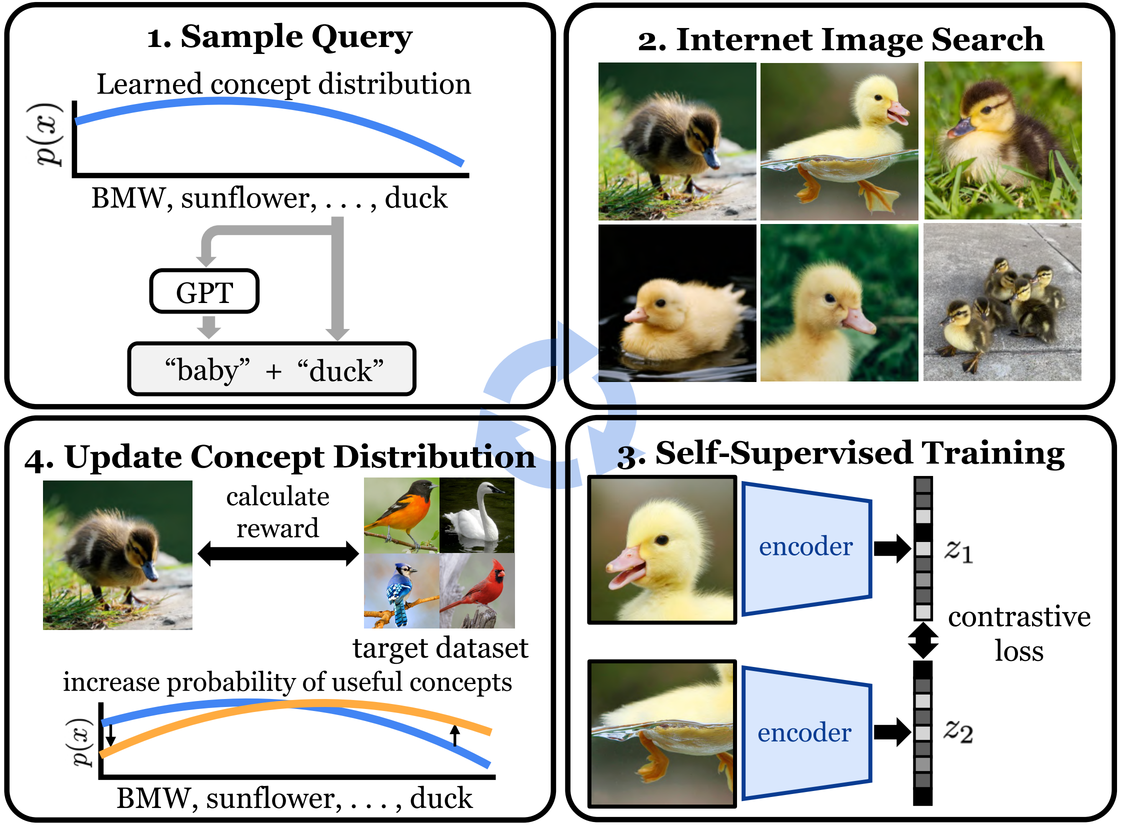
Internet Explorer Method



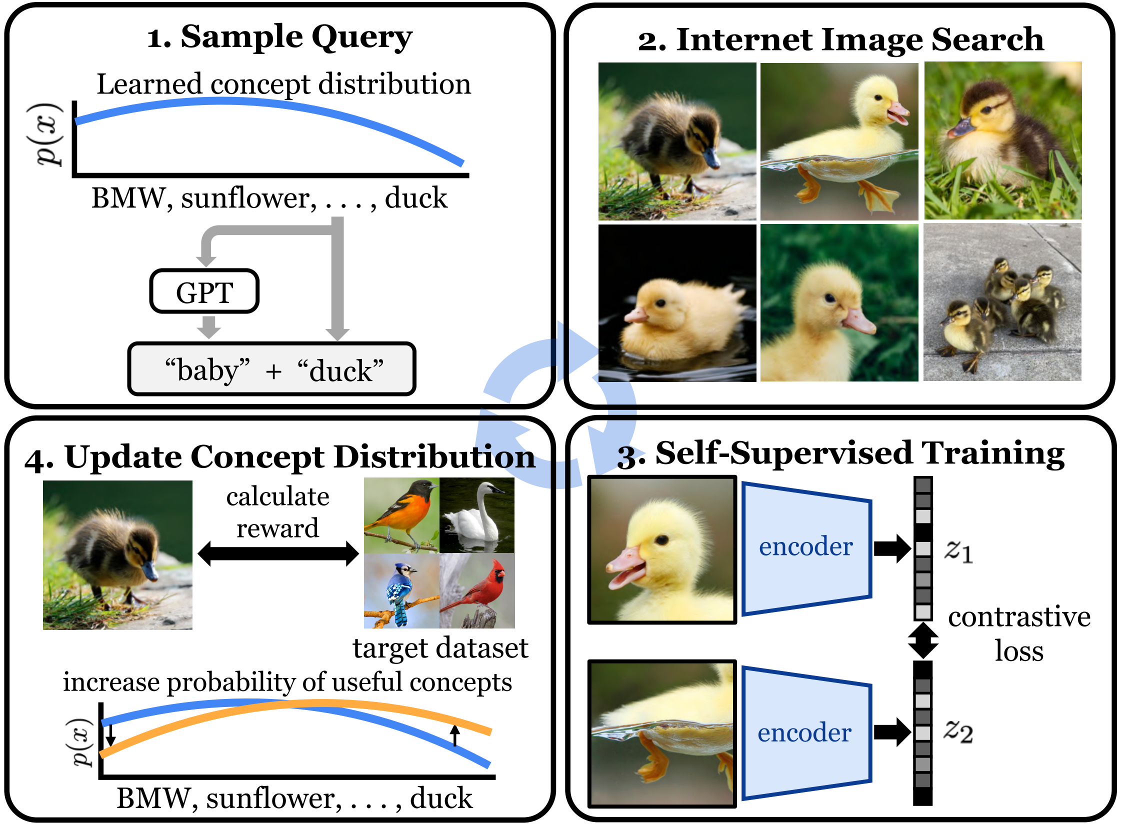
Internet Explorer Method



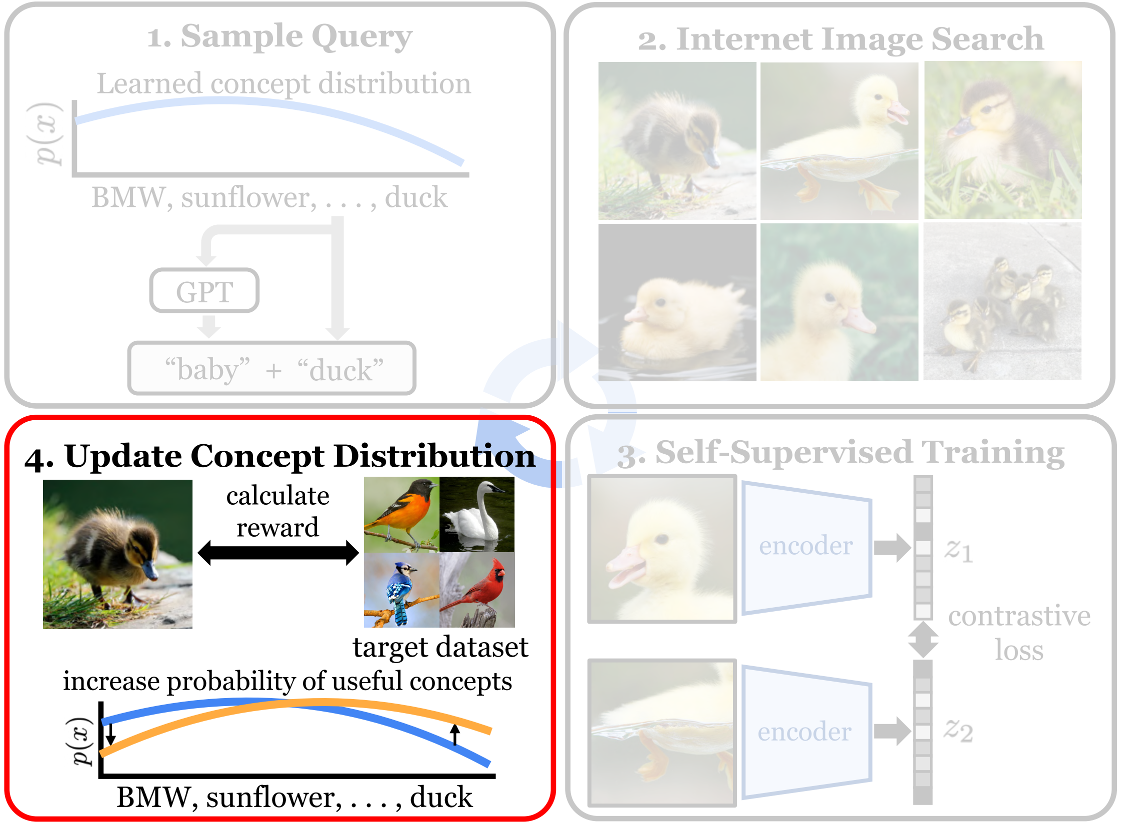
Internet Explorer Method



Internet Explorer Method



Internet Explorer Method



Internet Explorer Method

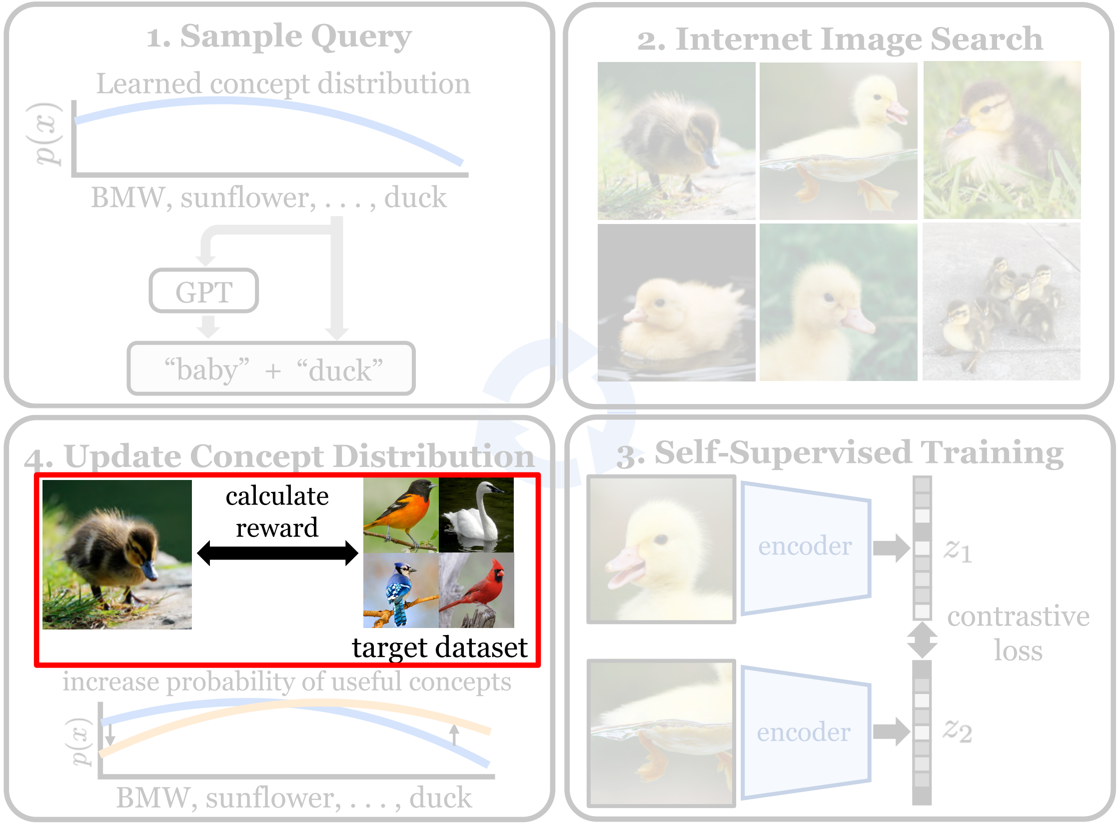


Image Reward (prioritize relevant *images*)

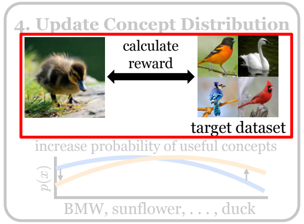
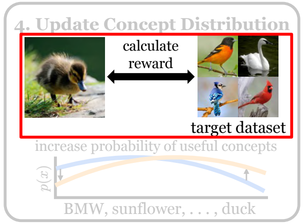
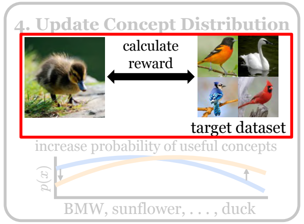


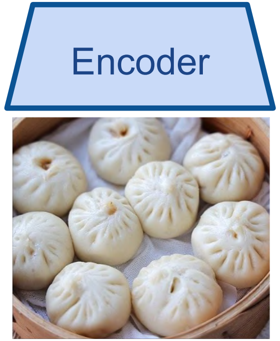
Image Reward (prioritize relevant *images*)



“steam buns”
downloaded
image #1

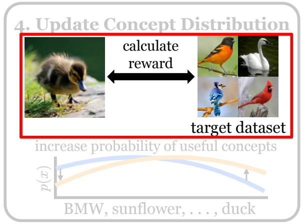
Image Reward (prioritize relevant *images*)

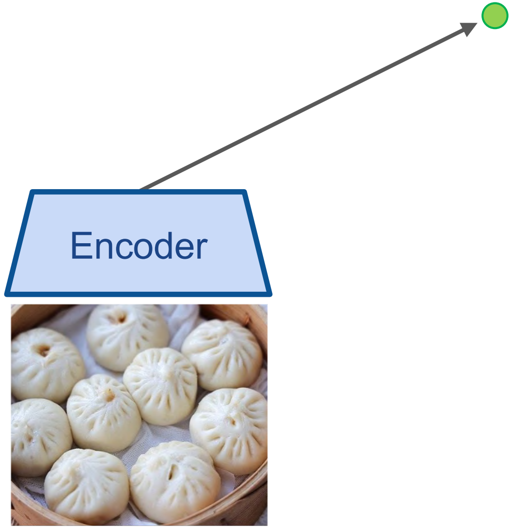




“steam buns” downloaded image #1

Image Reward (prioritize relevant *images*)





“steam buns”
downloaded
image #1

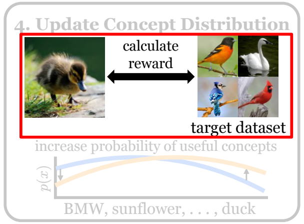
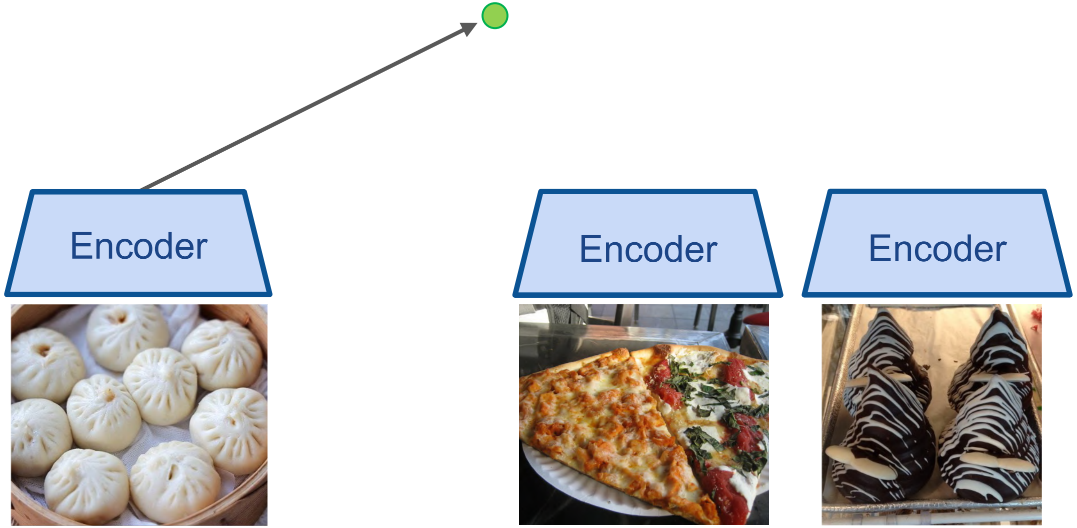


Image Reward (prioritize relevant *images*)



"steam buns"
downloaded
image #1

Food101
Target dataset images

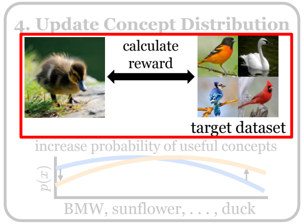
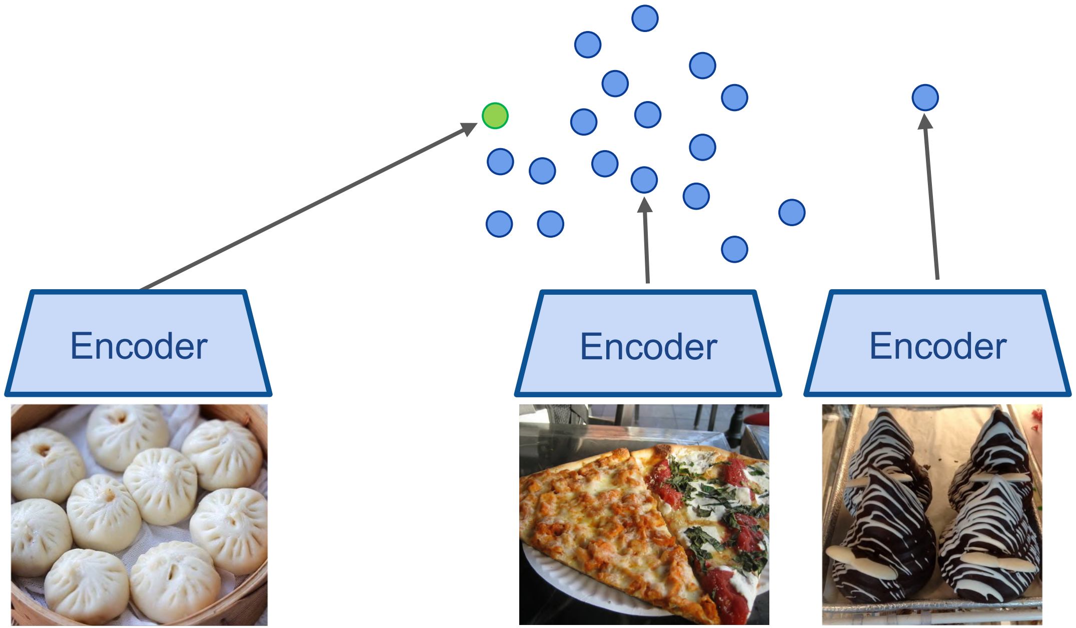


Image Reward (prioritize relevant *images*)



"steam buns" downloaded image #1

Food101
Target dataset images

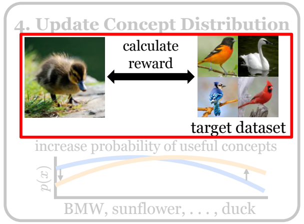
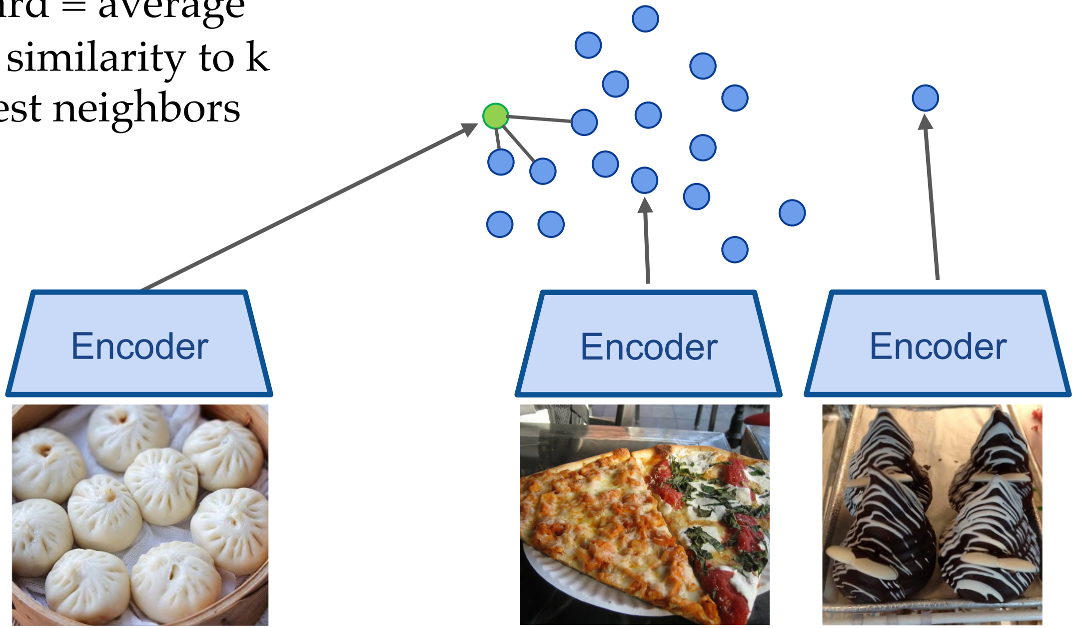


Image Reward (prioritize relevant *images*)

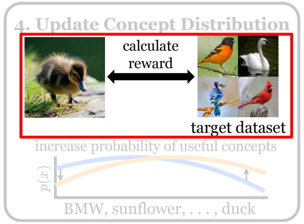
Reward = average cosine similarity to k nearest neighbors



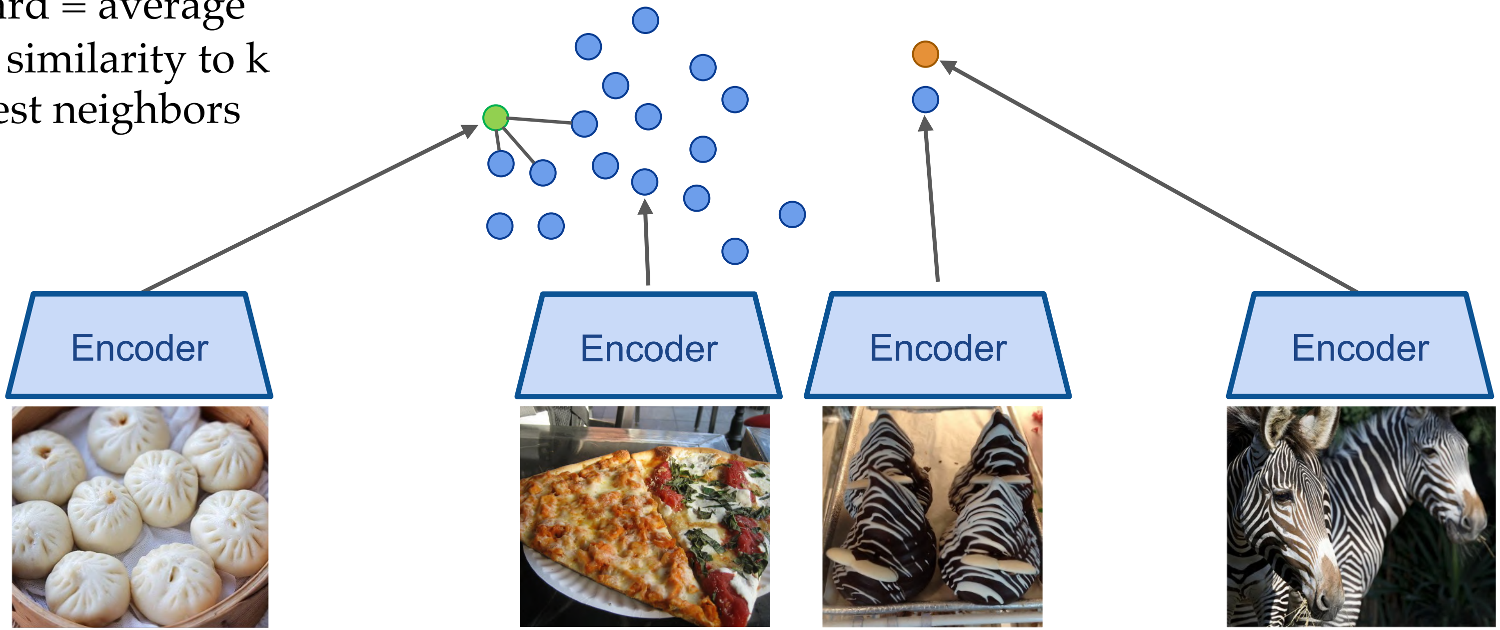
"steam buns" downloaded image #1

Food101
Target dataset images

Image Reward (prioritize relevant *images*)



Reward = average cosine similarity to k nearest neighbors

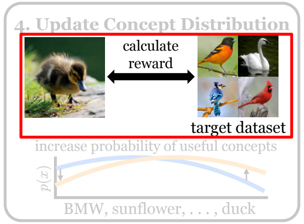


"steam buns" downloaded image #1

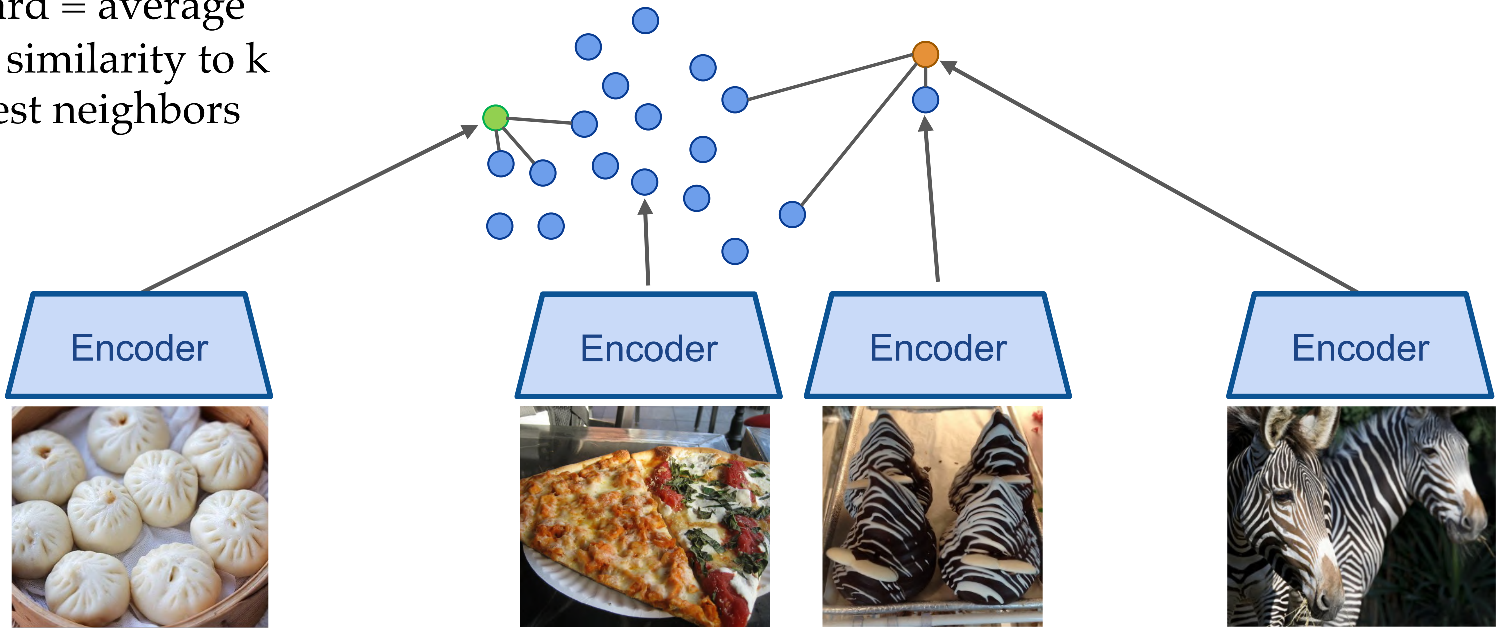
Food101
Target dataset images

"zebra" downloaded image #2

Image Reward (prioritize relevant *images*)



Reward = average cosine similarity to k nearest neighbors

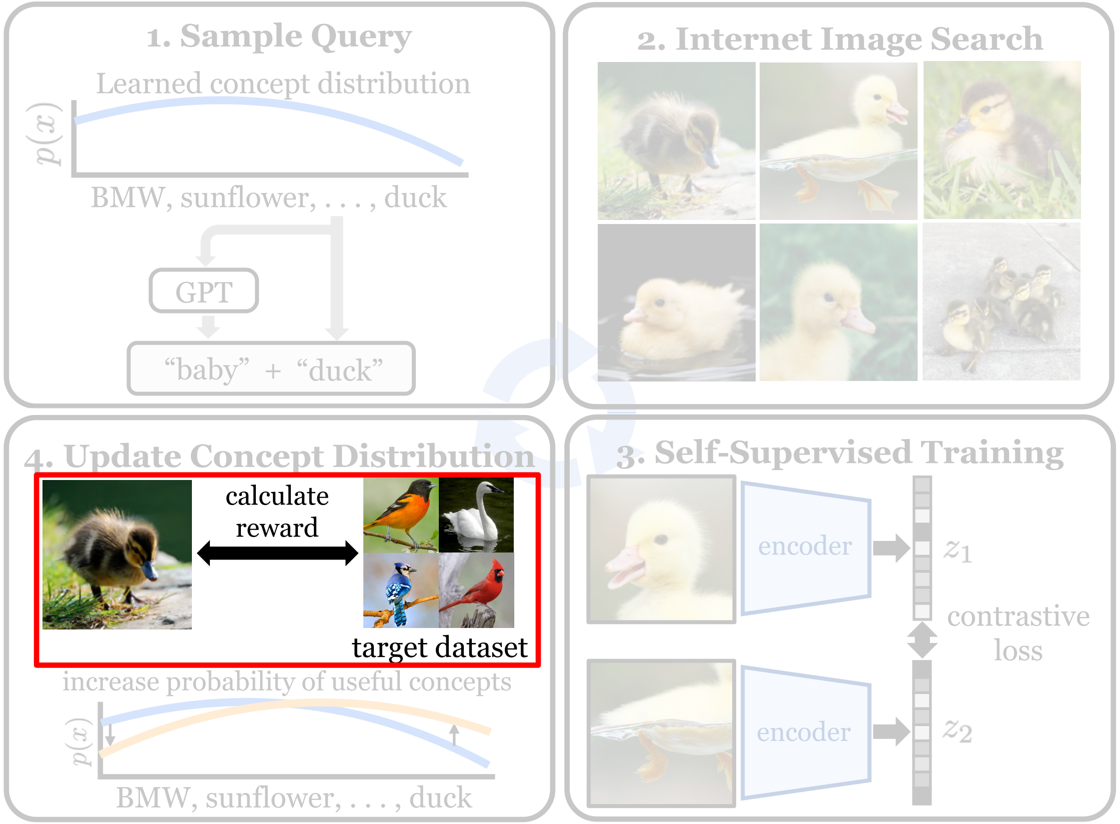


"steam buns" downloaded image #1

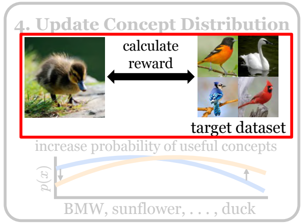
Food101
Target dataset images

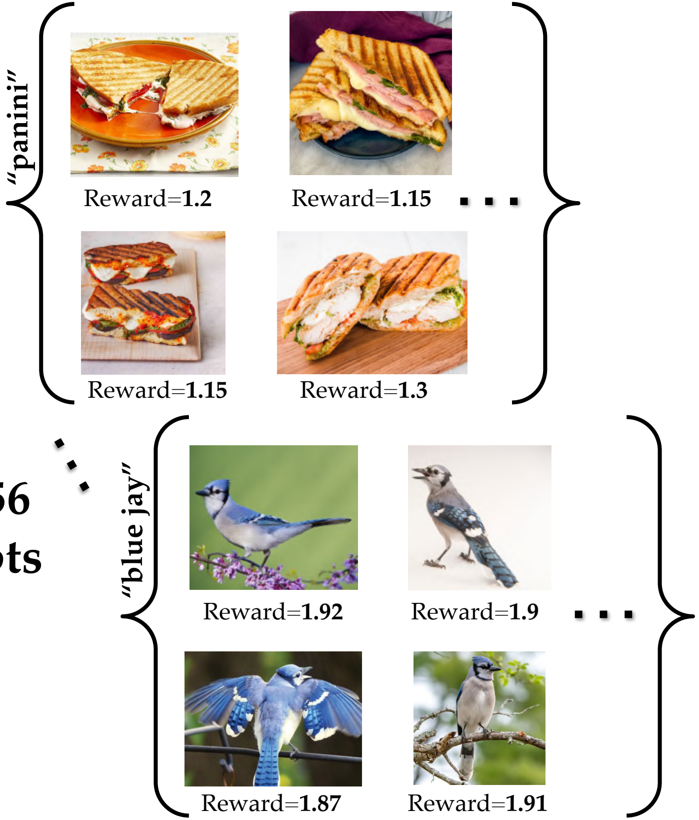
"zebra" downloaded image #2

Internet Explorer Method



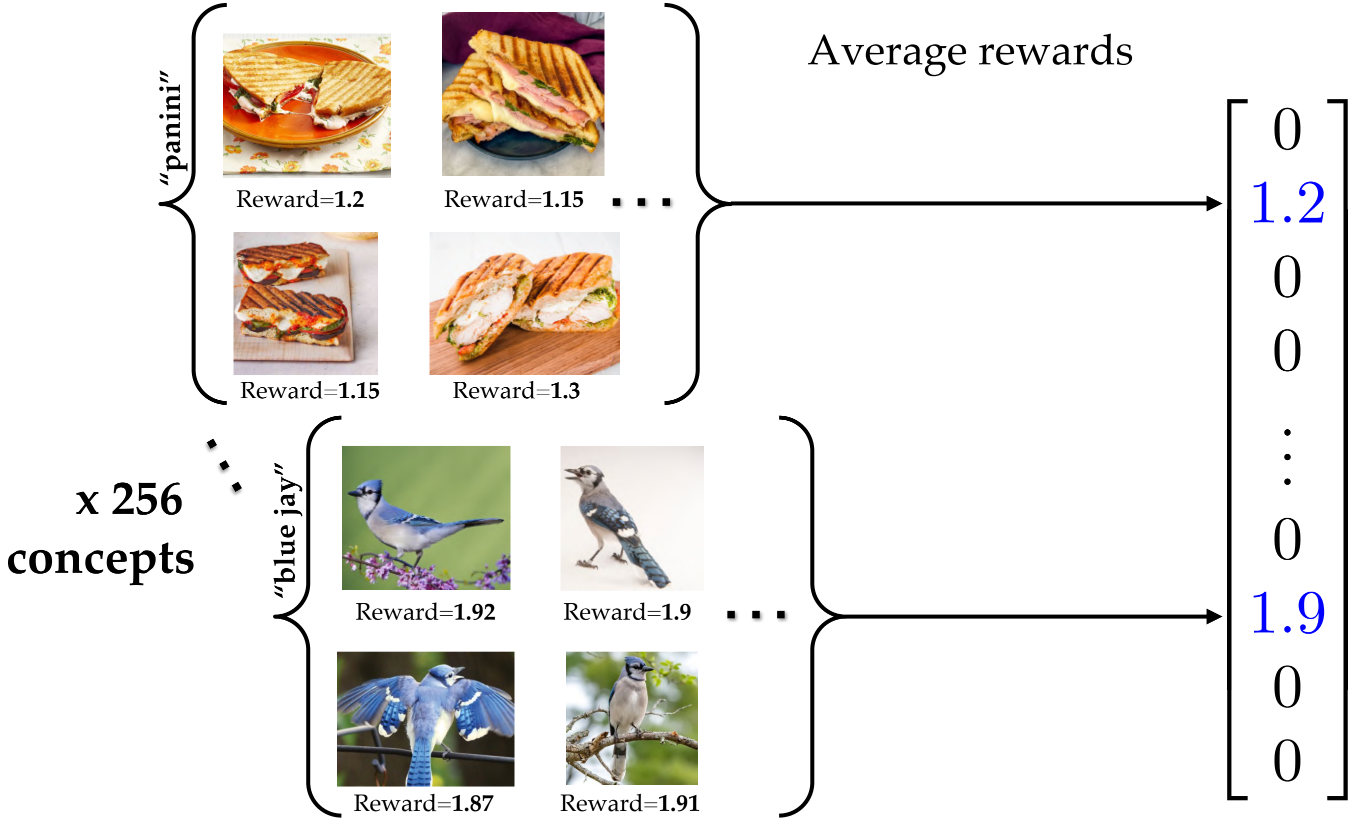
Concept Reward (prioritize relevant *concepts*)

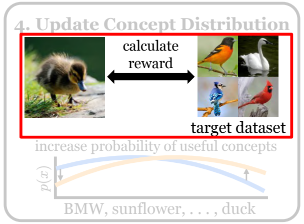




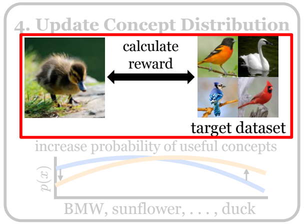
x 256 concepts

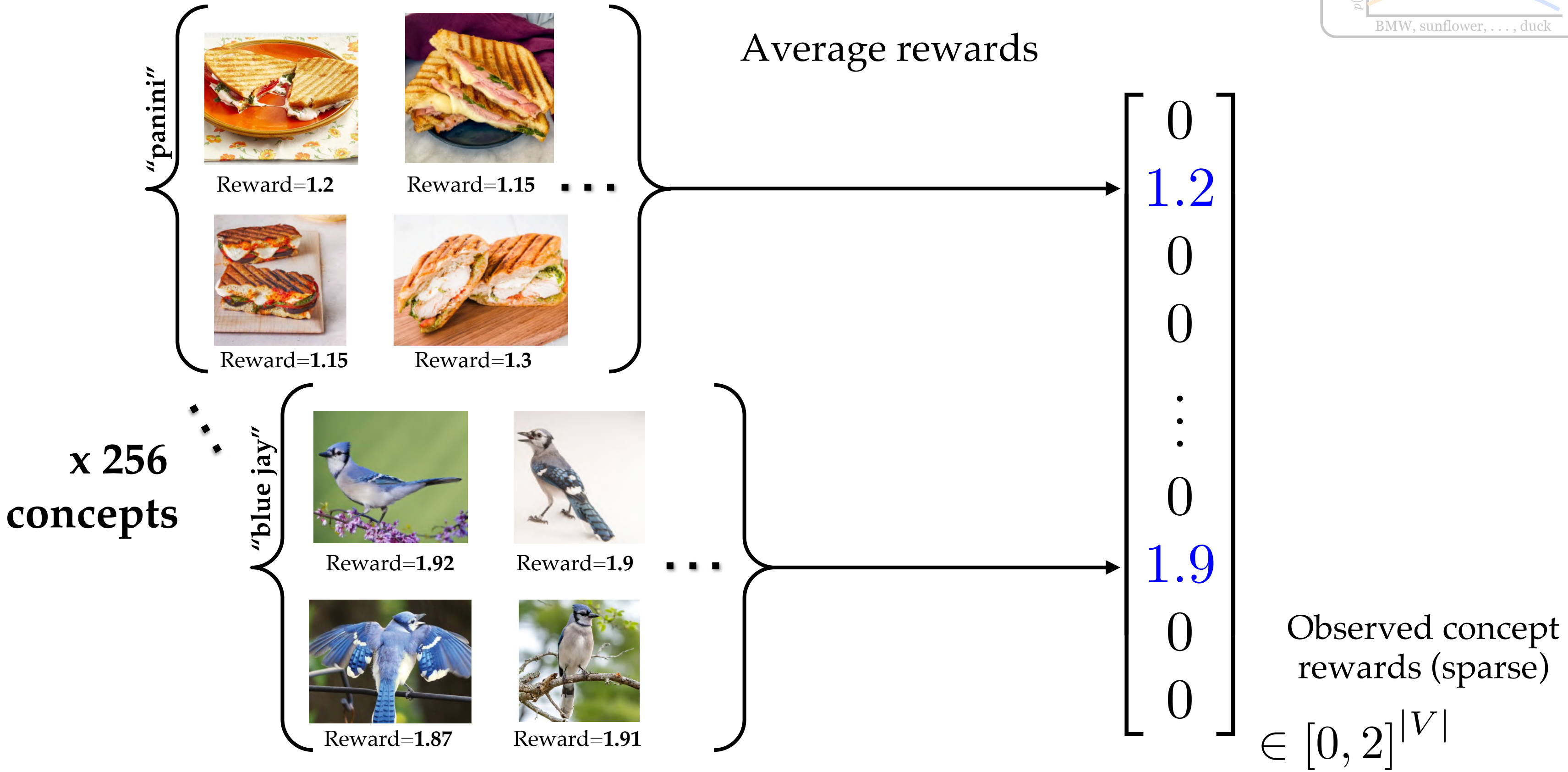
Concept Reward (prioritize relevant *concepts*)



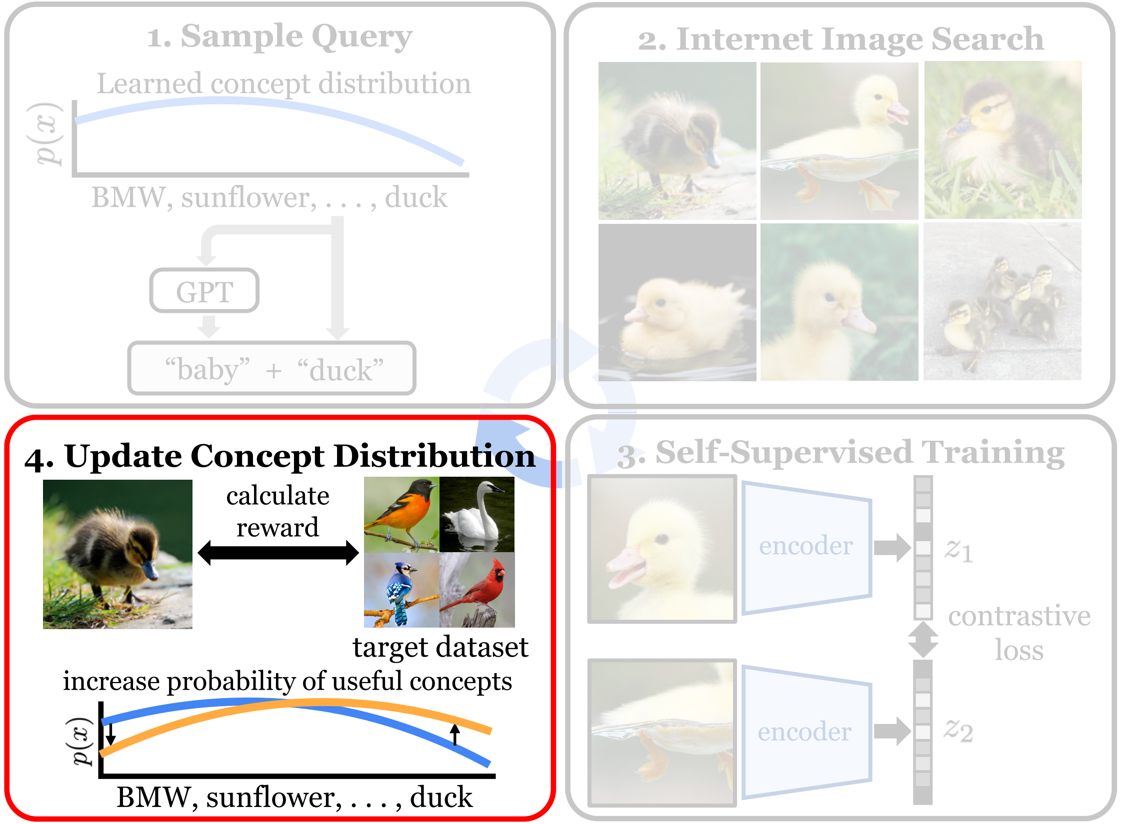


Concept Reward (prioritize relevant *concepts*)

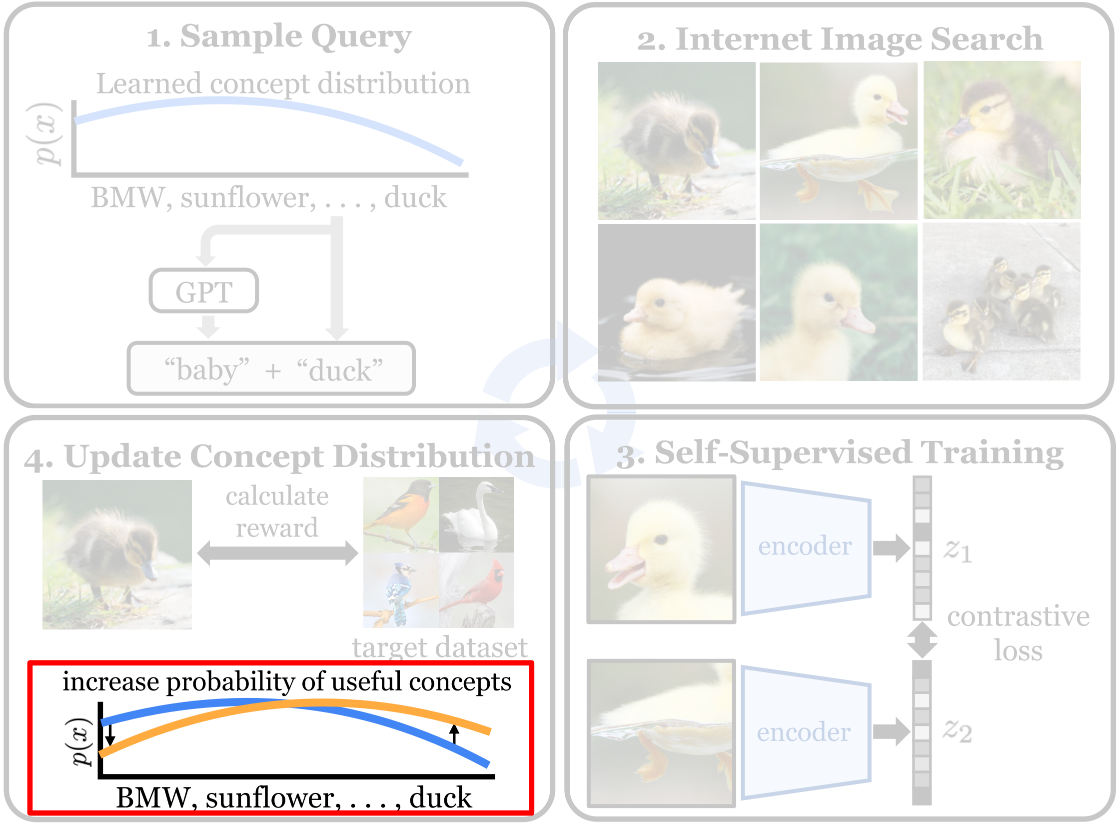




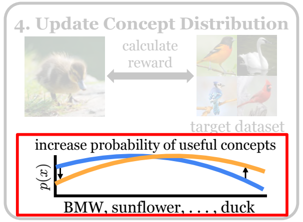
Internet Explorer Method

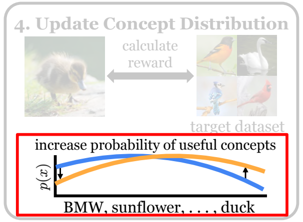


Internet Explorer Method



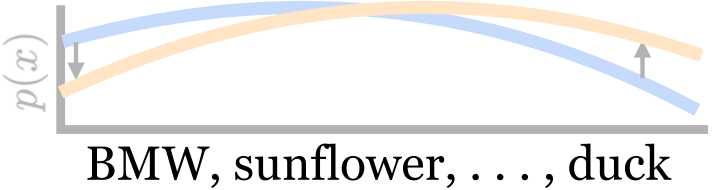
Concept Distribution



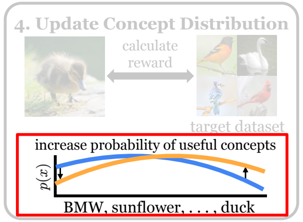


Concept Distribution

- Vocabulary size: $|V| \approx 150$k concepts

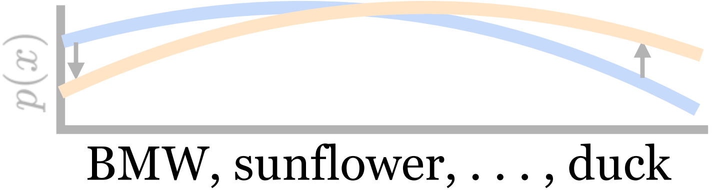


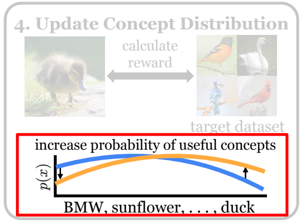
WORDNET



Concept Distribution

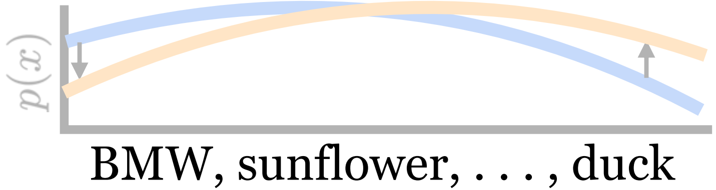
- Vocabulary size: $|V| \approx 150$k concepts
- Want to estimate value of unseen concepts from just a few thousand results

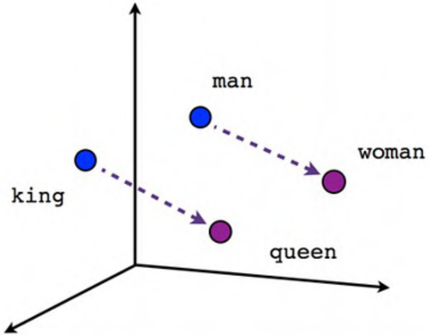




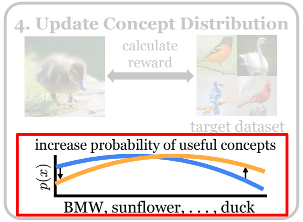
Concept Distribution

- Vocabulary size: $|V| \approx$ 150k concepts
- Want to estimate value of unseen concepts from just a few thousand results



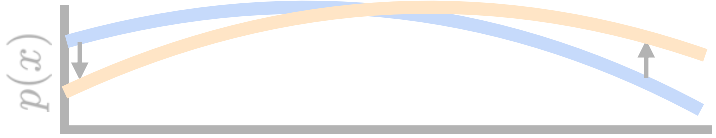


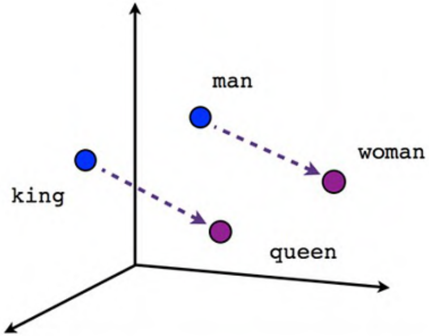
WORDNET

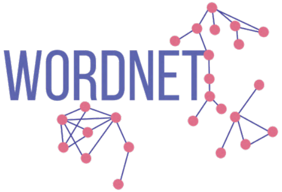


Concept Distribution

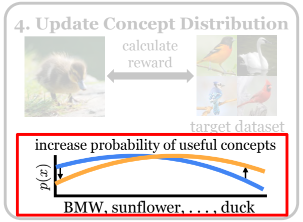
- Vocabulary size: |V|≈ 150k concepts
- Want to estimate value of unseen concepts from just a few thousand results





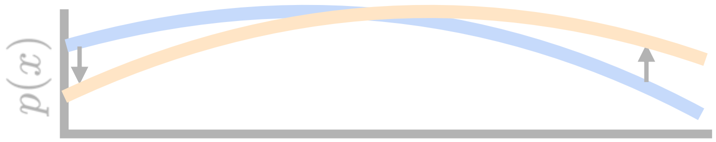


name + description

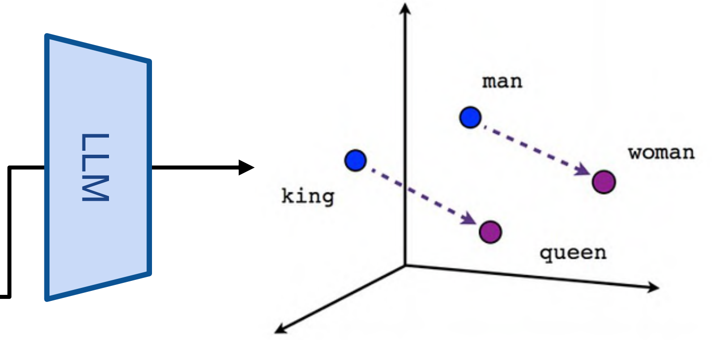


Concept Distribution

- Vocabulary size: |V| ≈ 150k concepts
- Want to estimate value of unseen concepts from just a few thousand results



BMW, sunflower, . . . , duck

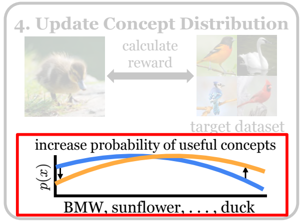


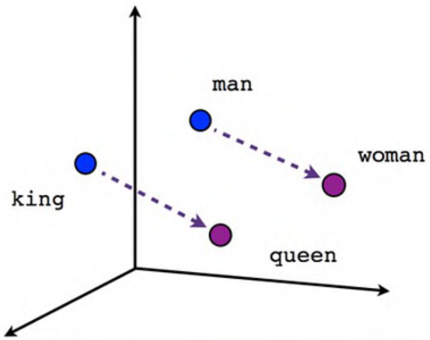
WORDNET

name + description

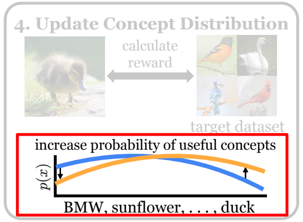
Sentence Transformer
Embedding Space

“Prospecting” in concept-embedding space





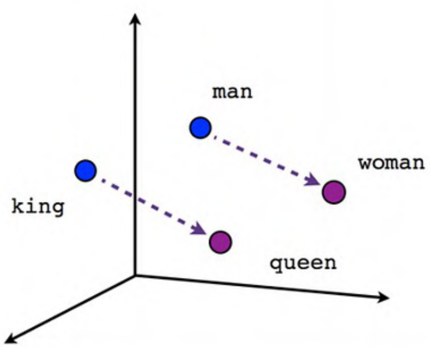
Sentence Transformer
Embedding Space



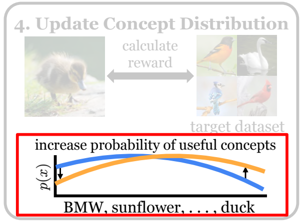
"Prospecting" in concept-embedding space

$$\begin{bmatrix} 0 \\ \color{blue}{1.2} \\ 0 \\ 0 \\ \vdots \\ 0 \\ \color{blue}{1.9} \\ 0 \\ 0 \end{bmatrix}$$

Observed concept rewards (sparse)

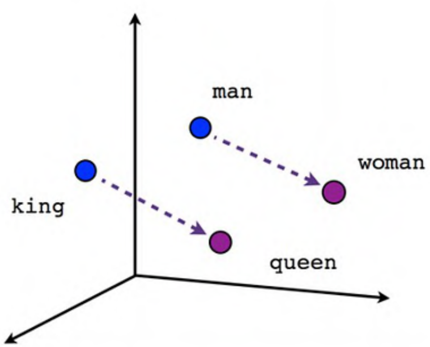


Sentence Transformer Embedding Space



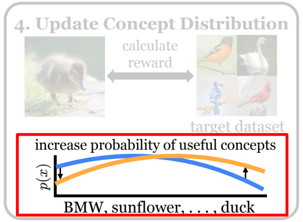
“Prospecting” in concept-embedding space

$$\left\{ \begin{bmatrix} 0 \\ 1.2 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1.9 \\ 0 \\ 0 \end{bmatrix} , \right\} \Rightarrow$$



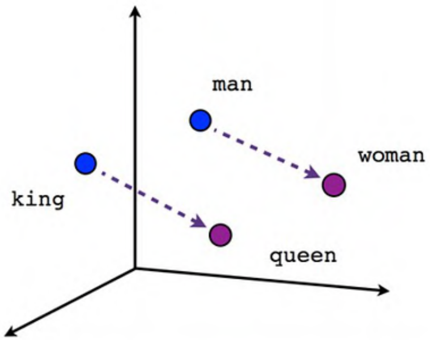
Observed concept rewards (sparse) , Sentence Transformer Embedding Space

“Prospecting” in concept-embedding space

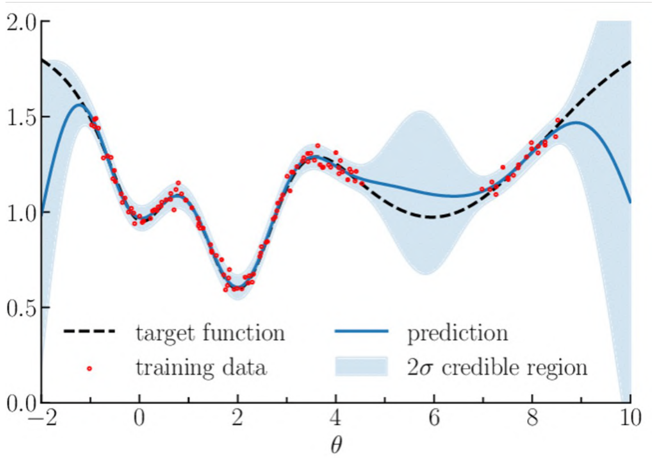


$$\begin{bmatrix}0\\1.2\\0\\0\\\vdots\\0\\1.9\\0\\0\end{bmatrix}$$

Observed concept rewards (sparse) ,

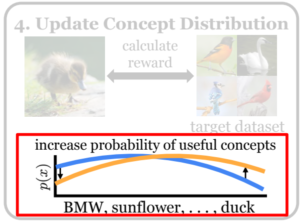


Sentence Transformer Embedding Space



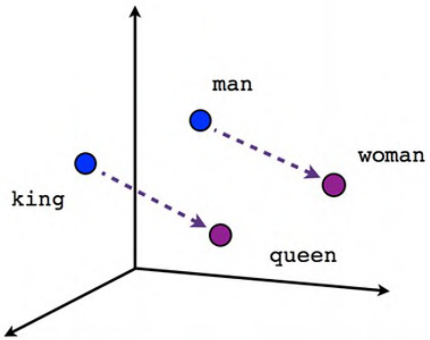
Gaussian Process Regression

“Prospecting” in concept-embedding space



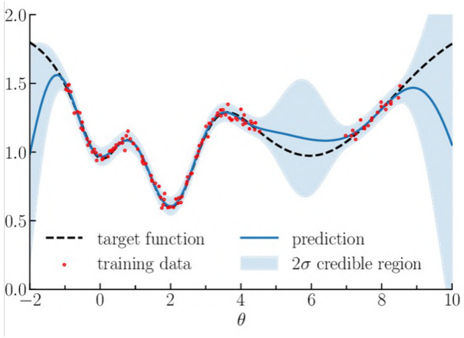
$$\begin{bmatrix} 0 \\ \color{blue}{1.2} \\ 0 \\ 0 \\ \vdots \\ 0 \\ \color{blue}{1.9} \\ 0 \\ 0 \end{bmatrix}$$

Observed concept rewards (sparse) ,

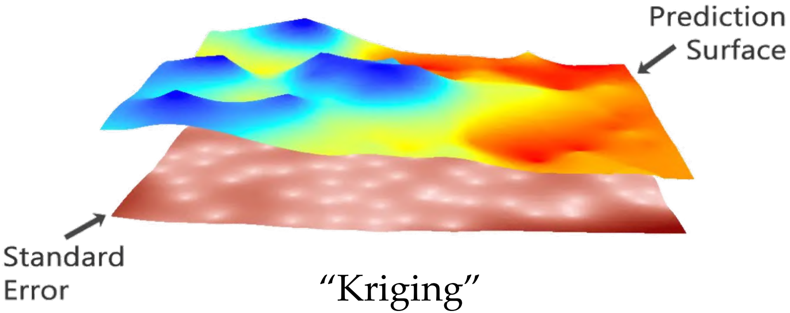


Sentence Transformer Embedding Space

➡

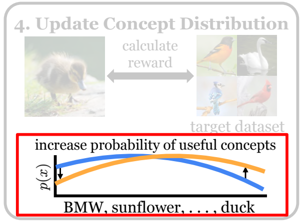


Gaussian Process Regression

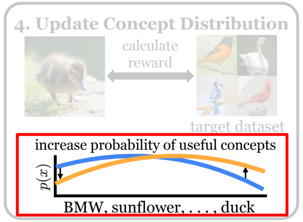


“Kriging”

Predicting Rewards / Forming Distribution



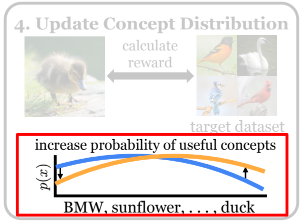
Predicting Rewards / Forming Distribution



$$\mu(\mathbf{e}) + \sigma(\mathbf{e})$$

Predicted concept reward means & stds. from GPR

Predicting Rewards / Forming Distribution

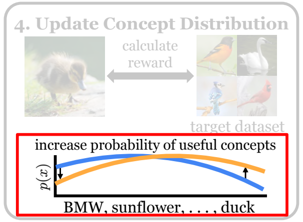


$$\mu(\mathbf{e}) + \sigma(\mathbf{e}) \longrightarrow \begin{bmatrix} 1.4 \\ 1.2 \\ 1.3 \\ 1.5 \\ \vdots \\ 1.5 \\ 1.9 \\ 1.5 \\ 1.7 \end{bmatrix}$$

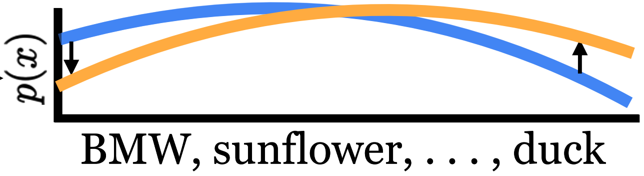
Predicted concept reward means & stds. from GPR

Predicted concept rewards (***dense***)

Predicting Rewards / Forming Distribution



$$\mu(\mathbf{e}) + \sigma(\mathbf{e}) \longrightarrow \begin{bmatrix} {\color{red}1.4} \\ {\color{blue}1.2} \\ {\color{red}1.3} \\ {\color{red}1.5} \\ \vdots \\ {\color{red}1.5} \\ {\color{blue}1.9} \\ {\color{red}1.5} \\ {\color{red}1.7} \end{bmatrix} \xrightarrow{\text{Softmax}}$$

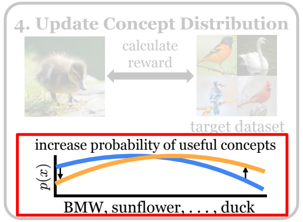


Predicted concept reward means & stds. from GPR

Predicted concept rewards (***dense***)

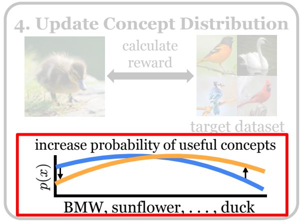
Next iteration's concept distribution

Tiering

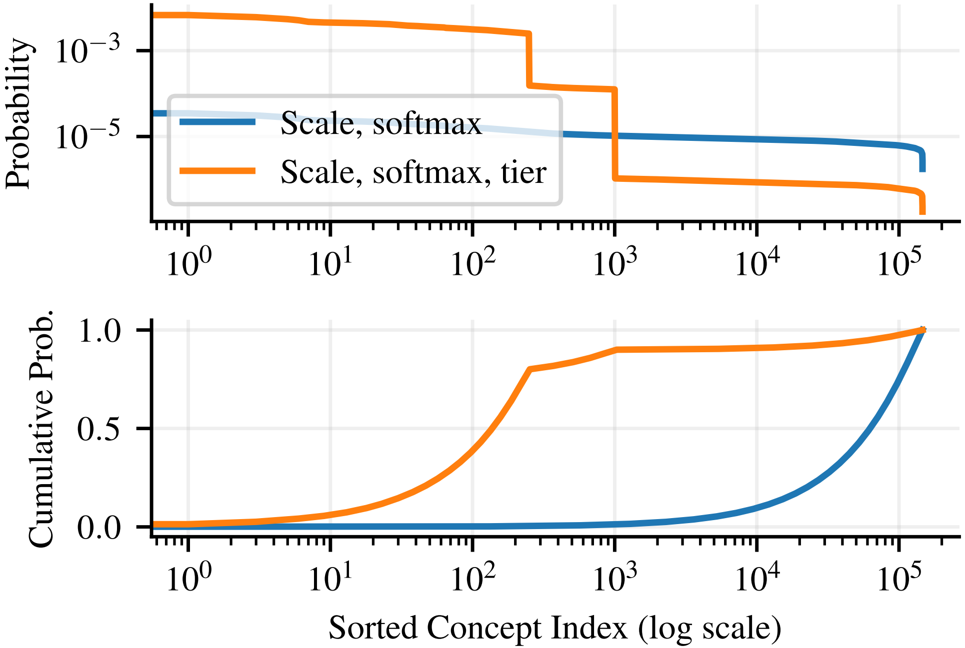


150k concepts! Most relevant are *still* rarely sampled…

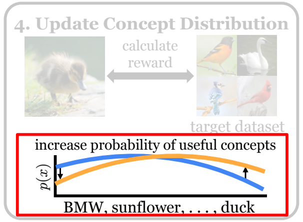
Tiering



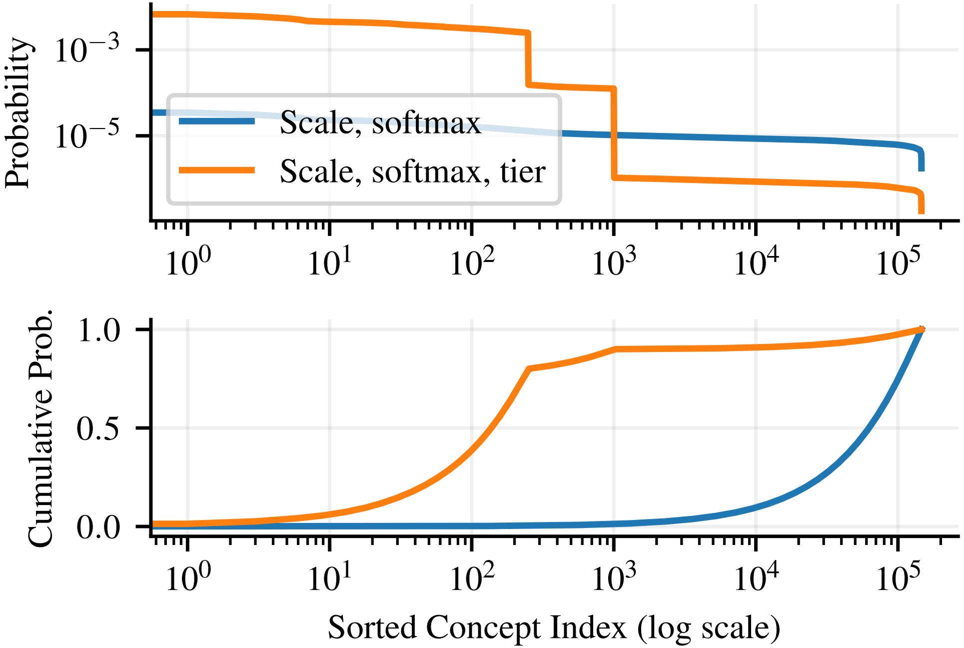
150k concepts! Most relevant are *still* rarely sampled…



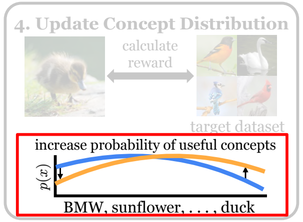
Tiering



150k concepts! Most relevant are *still* rarely sampled…

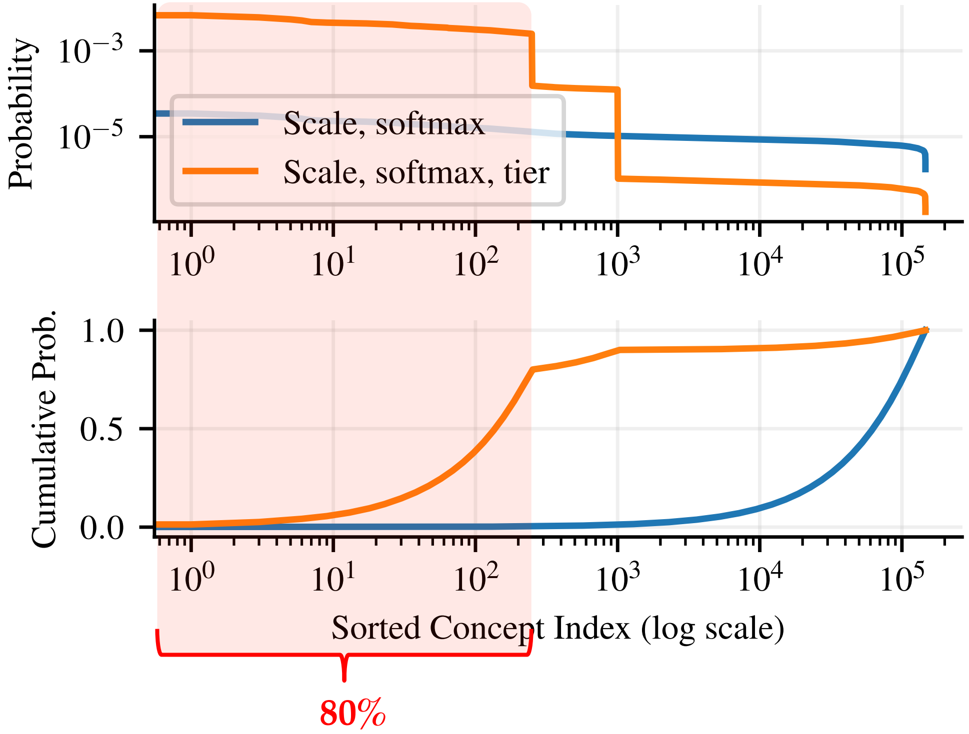


Tiering

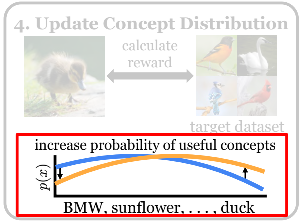


150k concepts! Most relevant are *still* rarely sampled…

- Top 250 concepts sampled 80% of the time

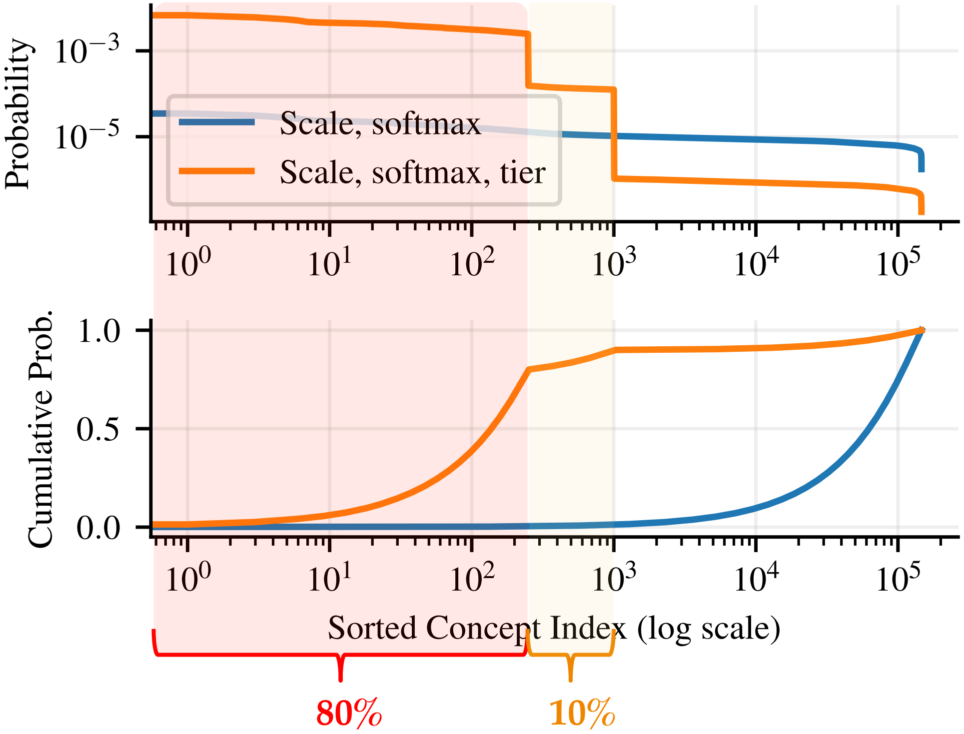


Tiering

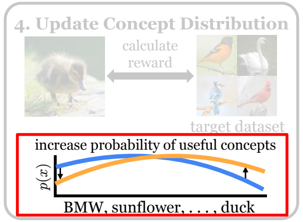


150k concepts! Most relevant are *still* rarely sampled…

- Top 250 concepts sampled 80% of the time
- 251–1000 ranked concepts sampled 10% of the time

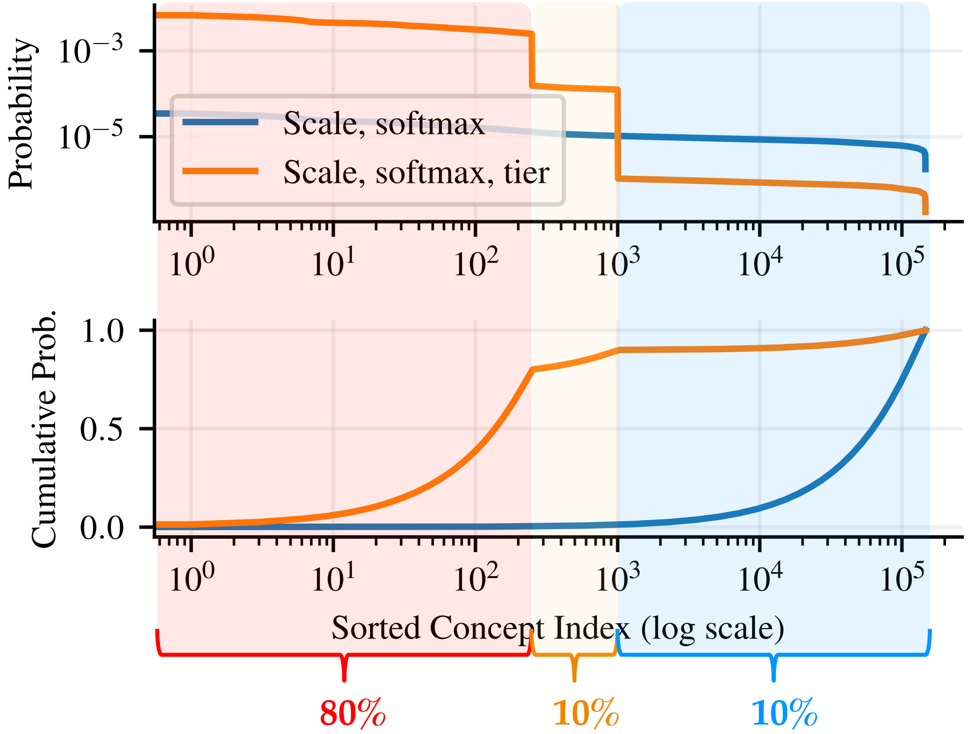


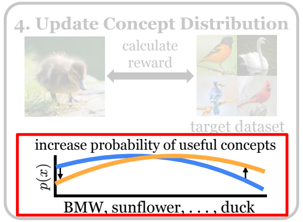
Tiering



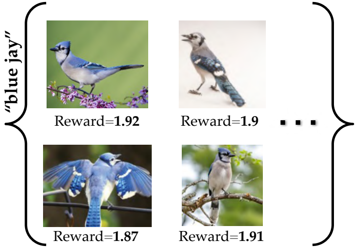
150k concepts! Most relevant are *still* rarely sampled…

- Top 250 concepts sampled 80% of the time
- 251–1000 ranked concepts sampled 10% of the time
- Remaining concepts sampled 10% of the time

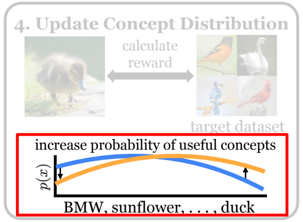


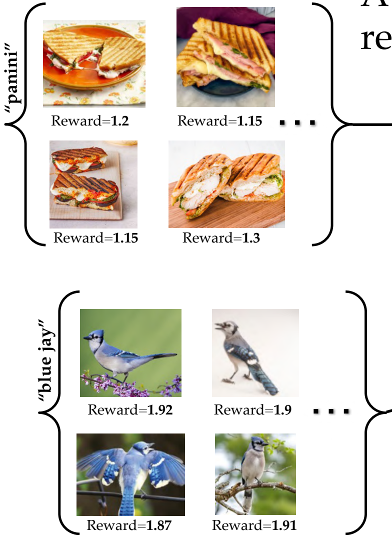


Recap: update dist. by predicting rewards



Recap: update dist. by predicting rewards



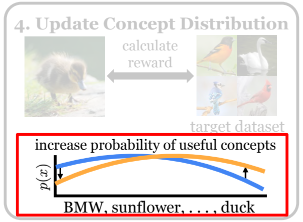


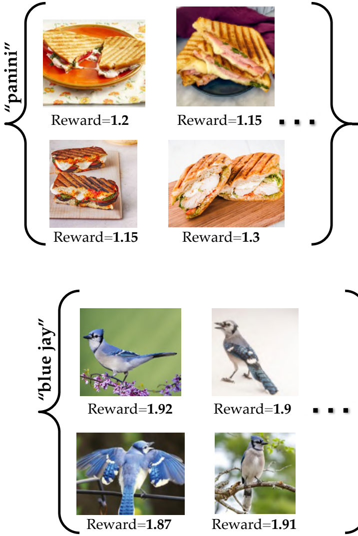
Average rewards

$$\begin{bmatrix} 0 \\ \color{blue}{1.2} \\ 0 \\ 0 \\ \vdots \\ 0 \\ \color{blue}{1.9} \\ 0 \\ 0 \end{bmatrix} \in [0, 2]^{|V|}$$

Observed concept rewards (*sparse*)

Recap: update dist. by predicting rewards

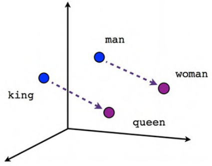




Average rewards

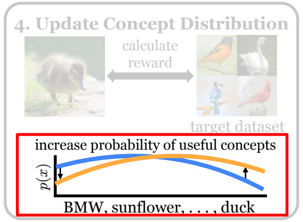
$$\begin{bmatrix} 0 \\ \color{blue}{1.2} \\ 0 \\ 0 \\ \vdots \\ 0 \\ \color{blue}{1.9} \\ 0 \\ 0 \end{bmatrix} \in [0,2]^{|V|}$$

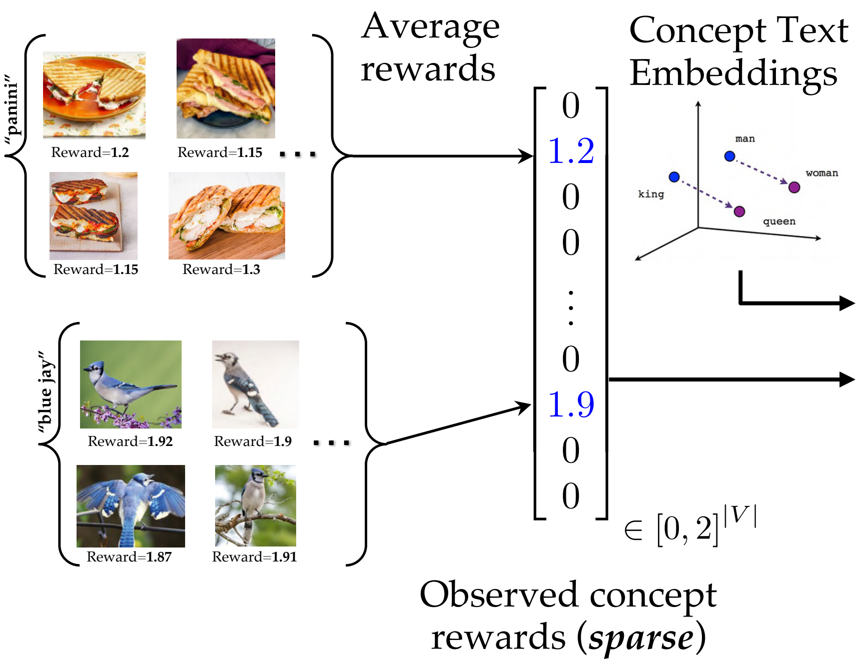
Concept Text Embeddings

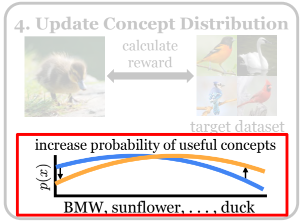


Observed concept rewards (*sparse*)

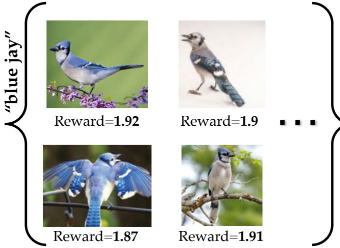
Recap: update dist. by predicting rewards







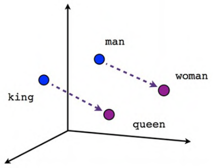
Recap: update dist. by predicting rewards

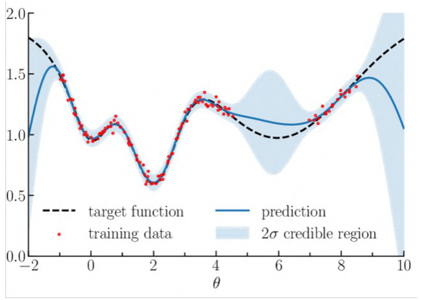


Average rewards

$$\begin{bmatrix} 0 \\ \color{blue}{1.2} \\ 0 \\ 0 \\ \vdots \\ 0 \\ \color{blue}{1.9} \\ 0 \\ 0 \end{bmatrix} \in [0,2]^{|V|}$$

Concept Text Embeddings

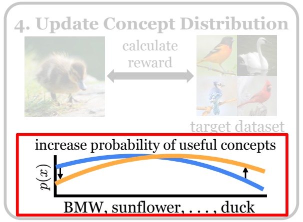


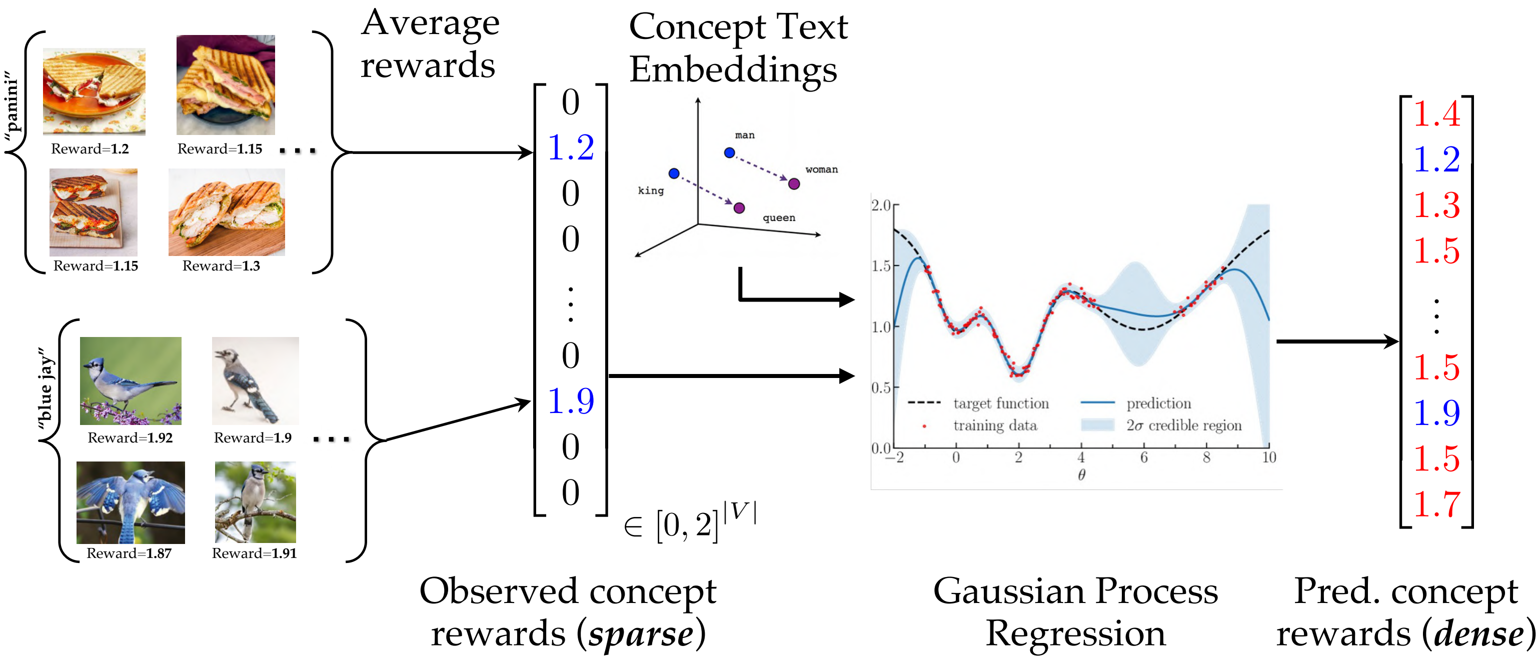


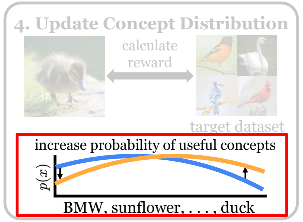
Observed concept rewards (*sparse*)

Gaussian Process Regression

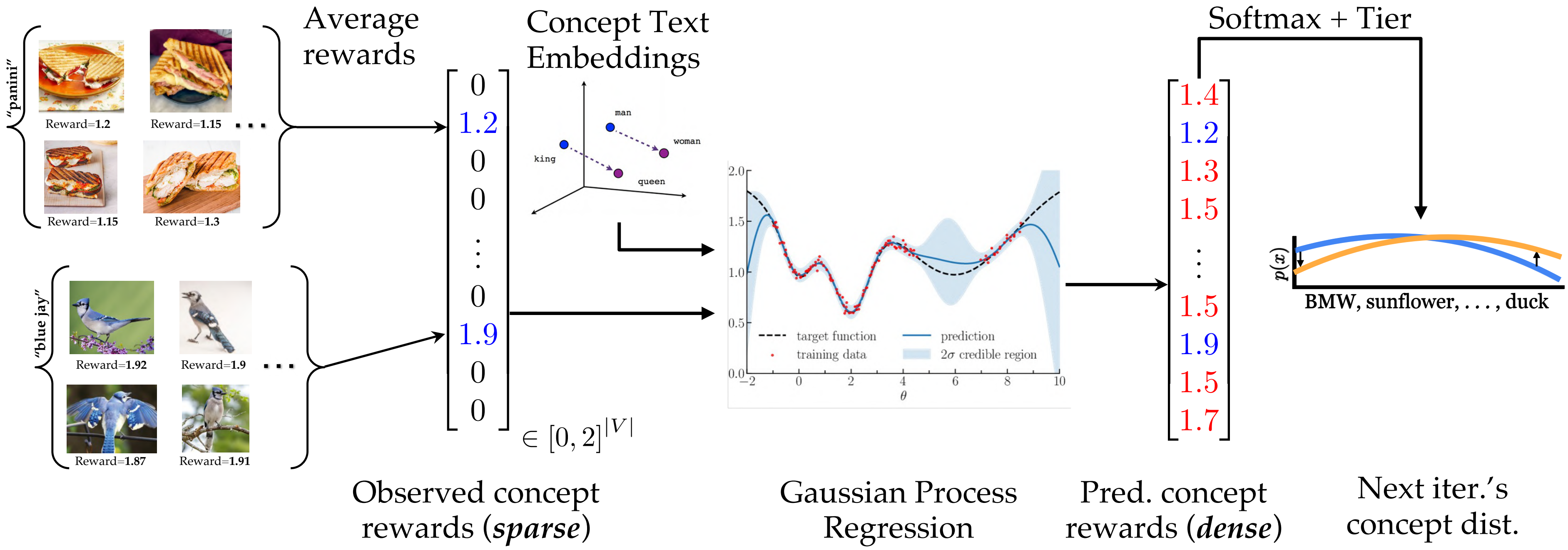
Recap: update dist. by predicting rewards



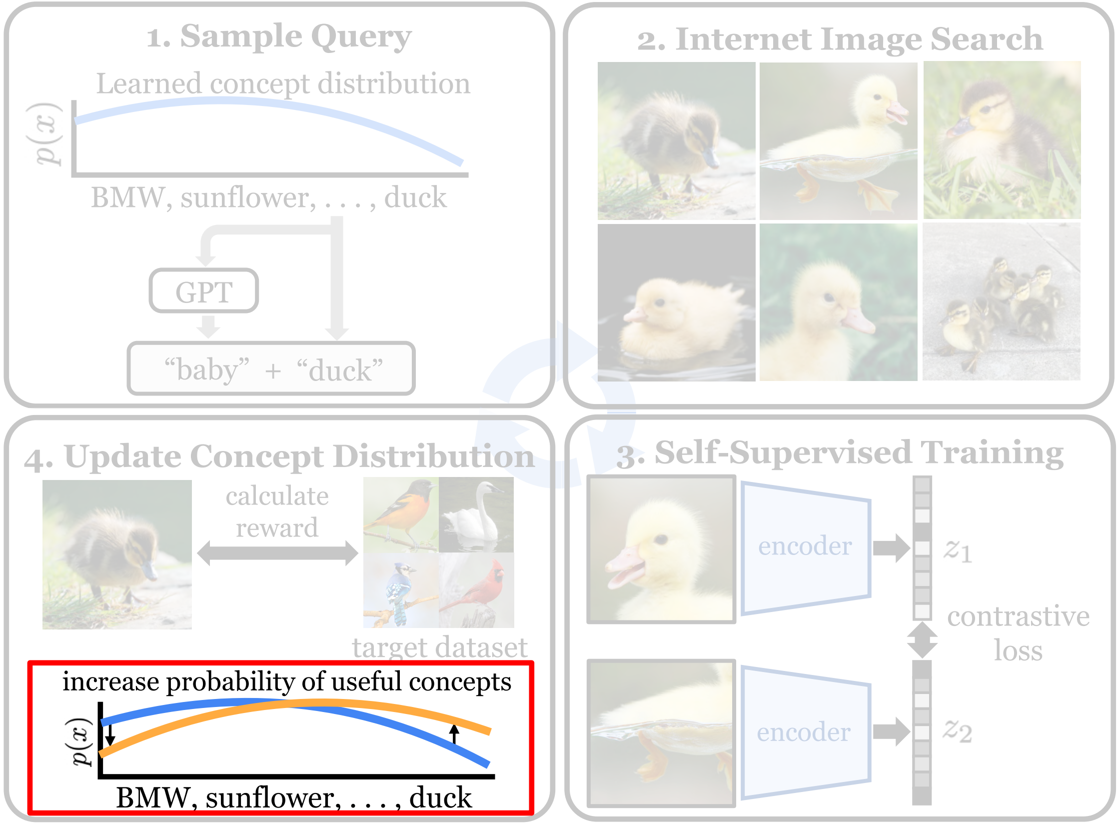




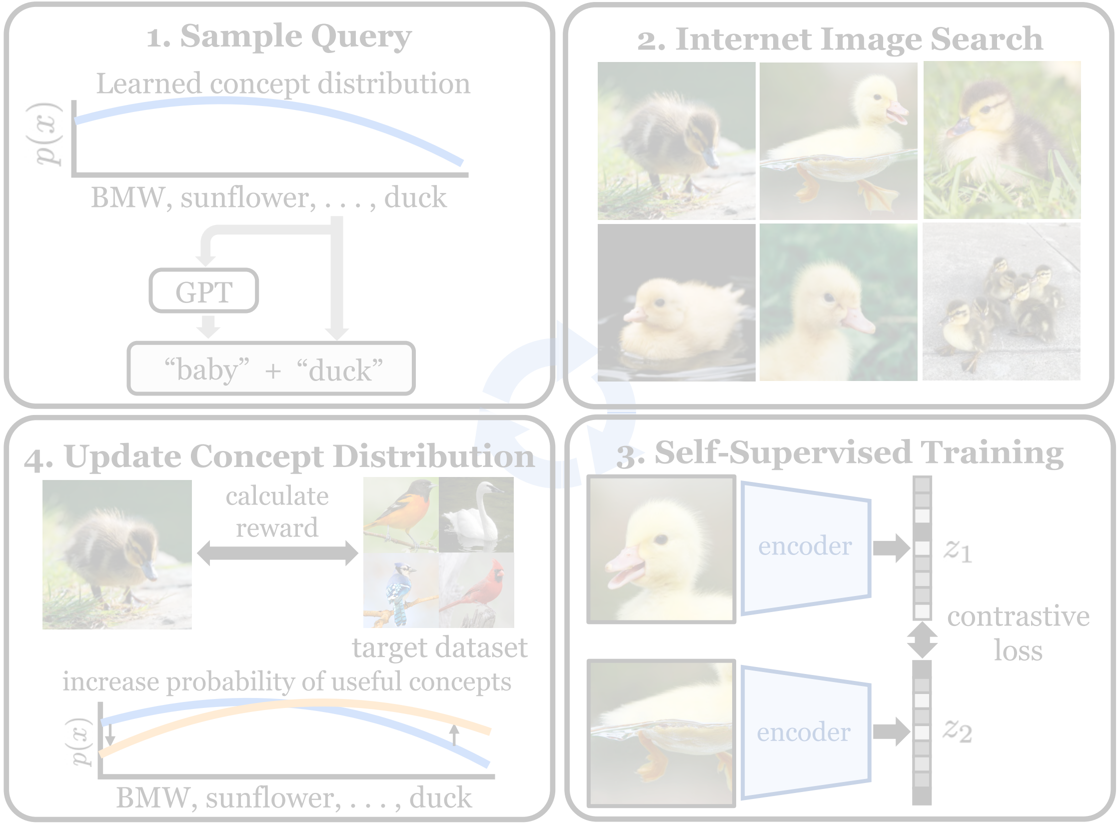
Recap: update dist. by predicting rewards



Internet Explorer Method



Internet Explorer Method



What’s changing over time?

What’s changing over time?

What’s changing over time?

Images that we search for and download

What's changing over time?

Images that we search for and download

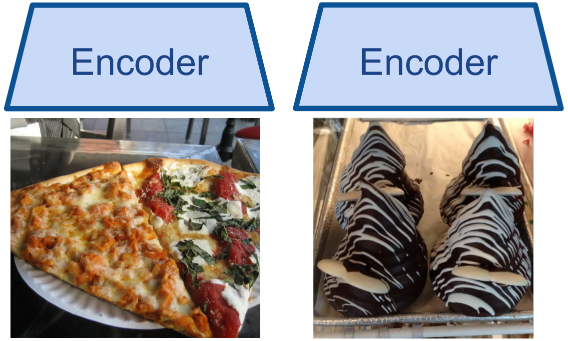
Representation space in which we compare images

Embedding space (and image reward) improves over time

Iteration 0:

Embedding space (and image reward) improves over time

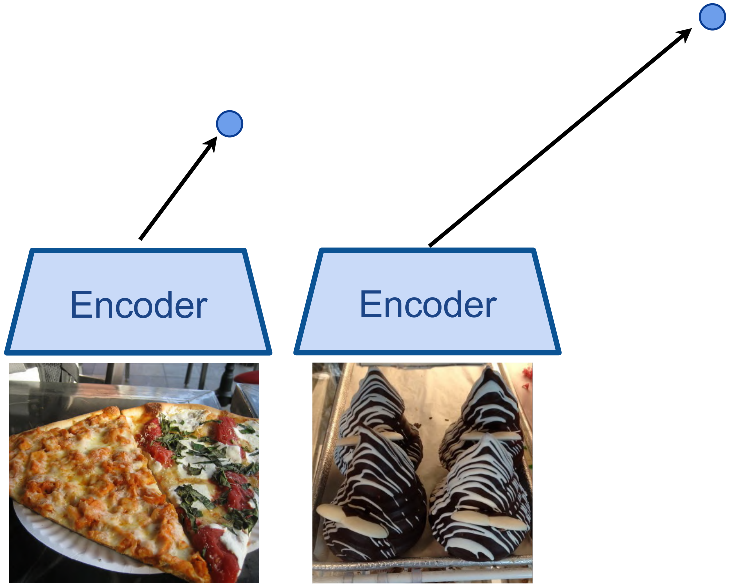
Iteration 0:



Target dataset images

Embedding space (and image reward) improves over time

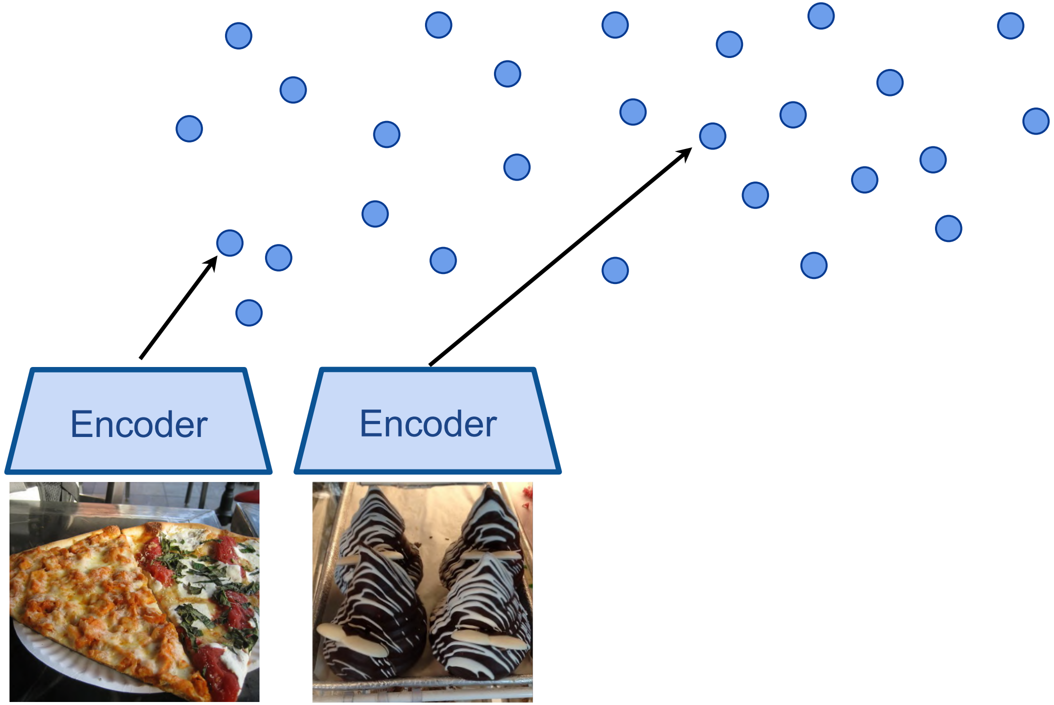
Iteration 0:



Target dataset images

Embedding space (and image reward) improves over time

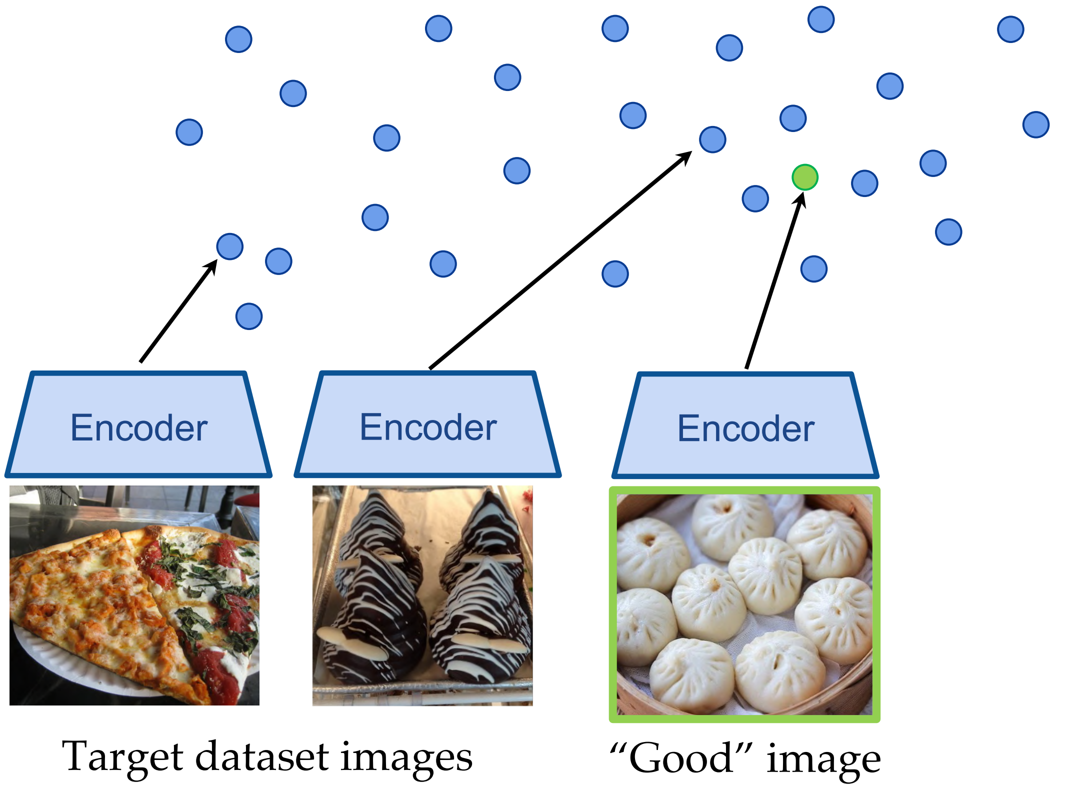
Iteration 0:



Target dataset images

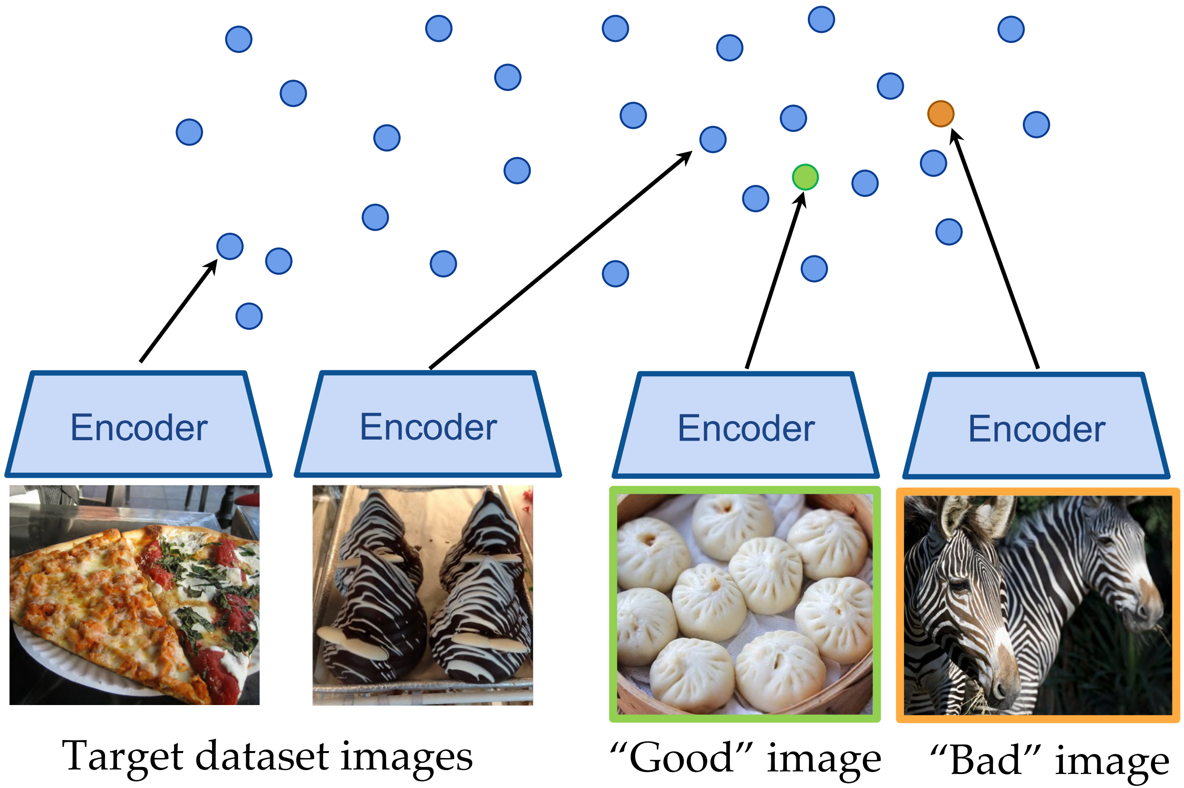
Embedding space (and image reward) improves over time

Iteration 0:



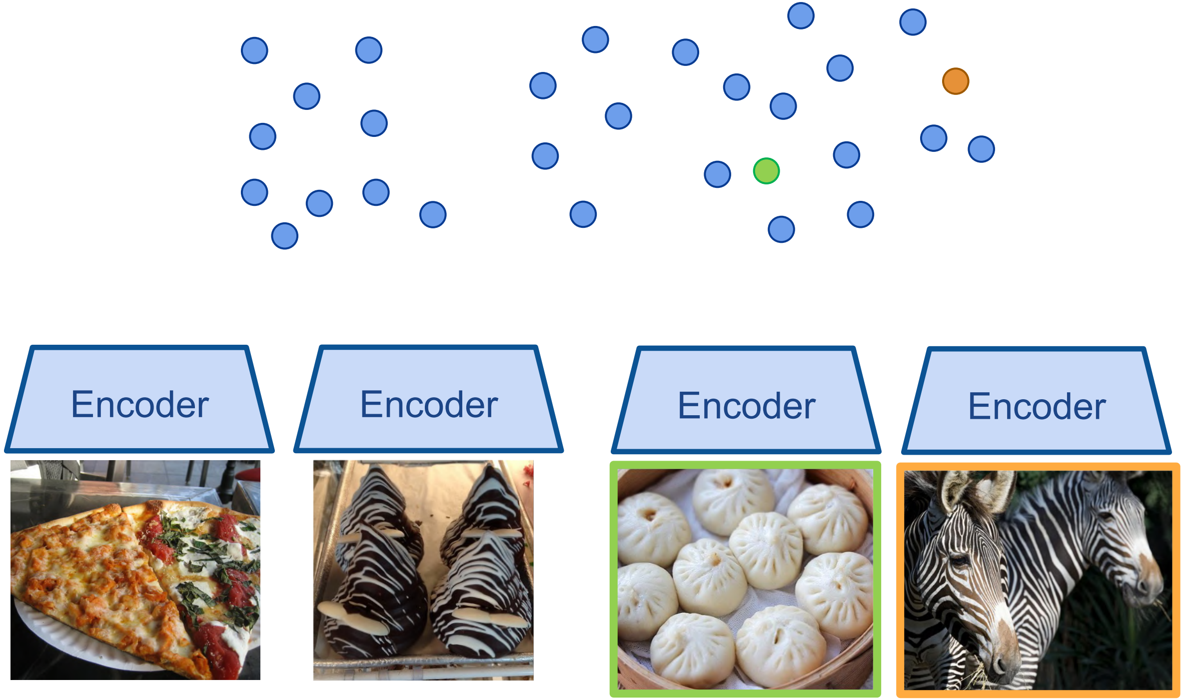
Embedding space (and image reward) improves over time

Iteration 0:



Embedding space (and image reward) improves over time

Iteration 5:



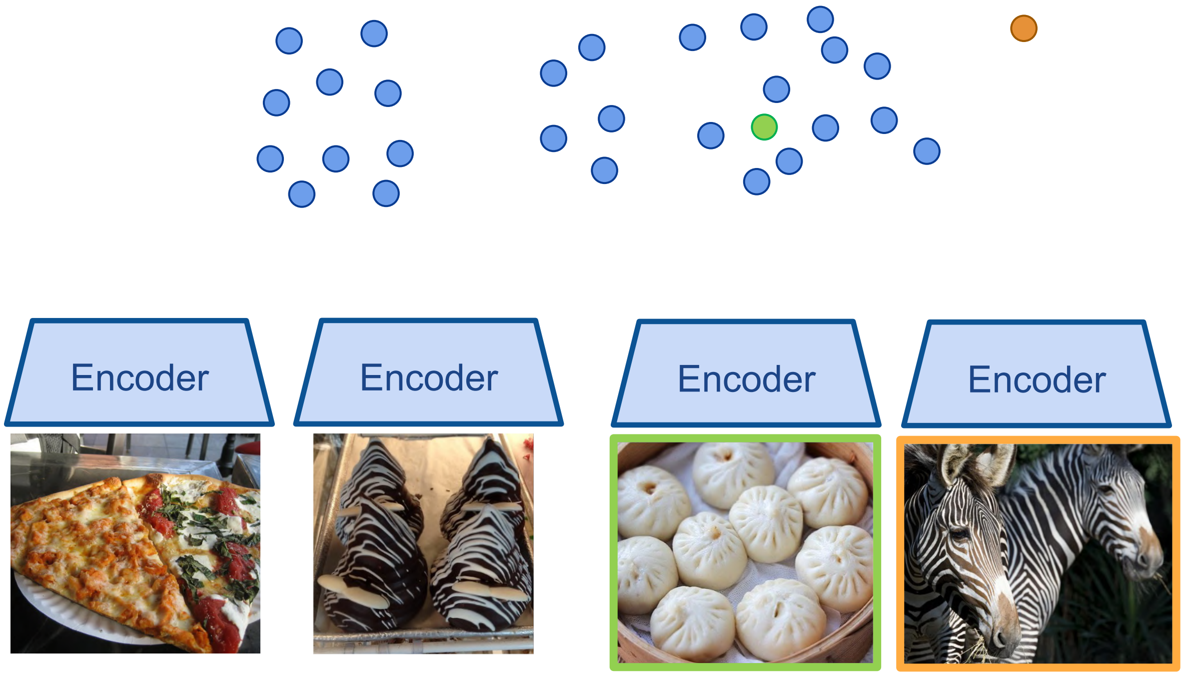
Target dataset images

“Good” image

“Bad” image

Embedding space (and image reward) improves over time

Iteration 10:



Target dataset images

“Good” image

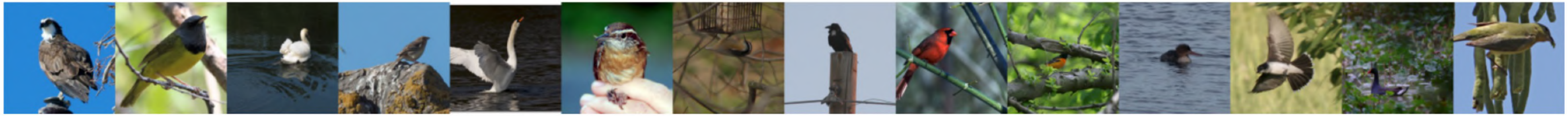
“Bad” image

Results

Self-supervised exploration progressively finds relevant data

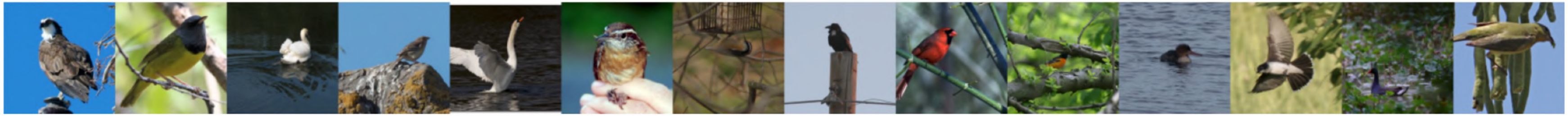
Self-supervised exploration progressively finds relevant data

Target dataset: Birdsnap



Self-supervised exploration progressively finds relevant data

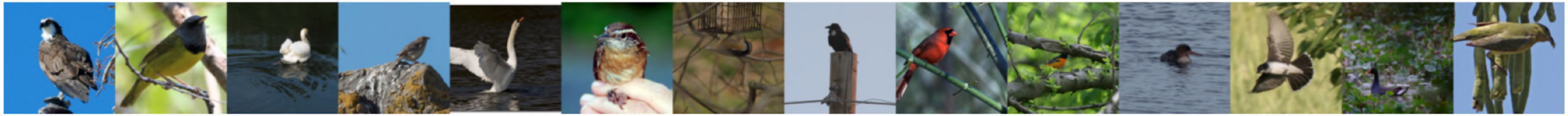
Target dataset: Birdsnap



Iteration 0

Self-supervised exploration progressively finds relevant data

Target dataset: Birdsnap

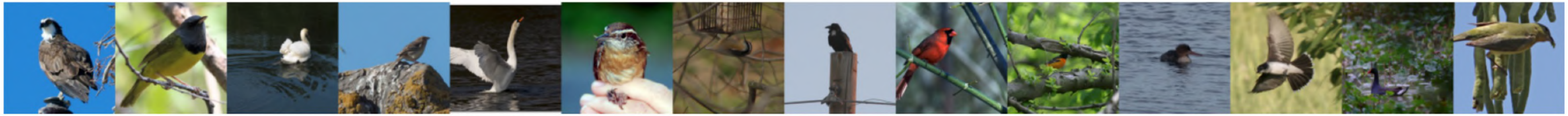


Iteration 0

Iteration 1

Self-supervised exploration progressively finds relevant data

Target dataset: Birdsnap



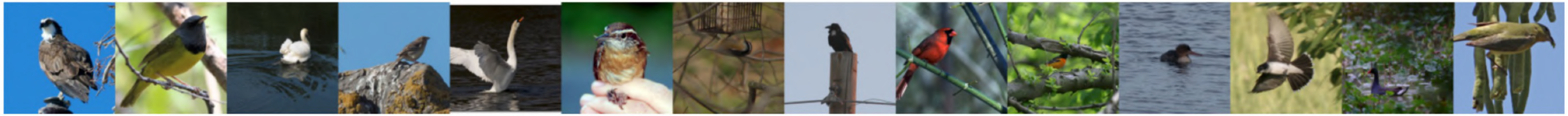
Iteration 0

Iteration 1

Iteration 3

Self-supervised exploration progressively finds relevant data

Target dataset: Birdsnap

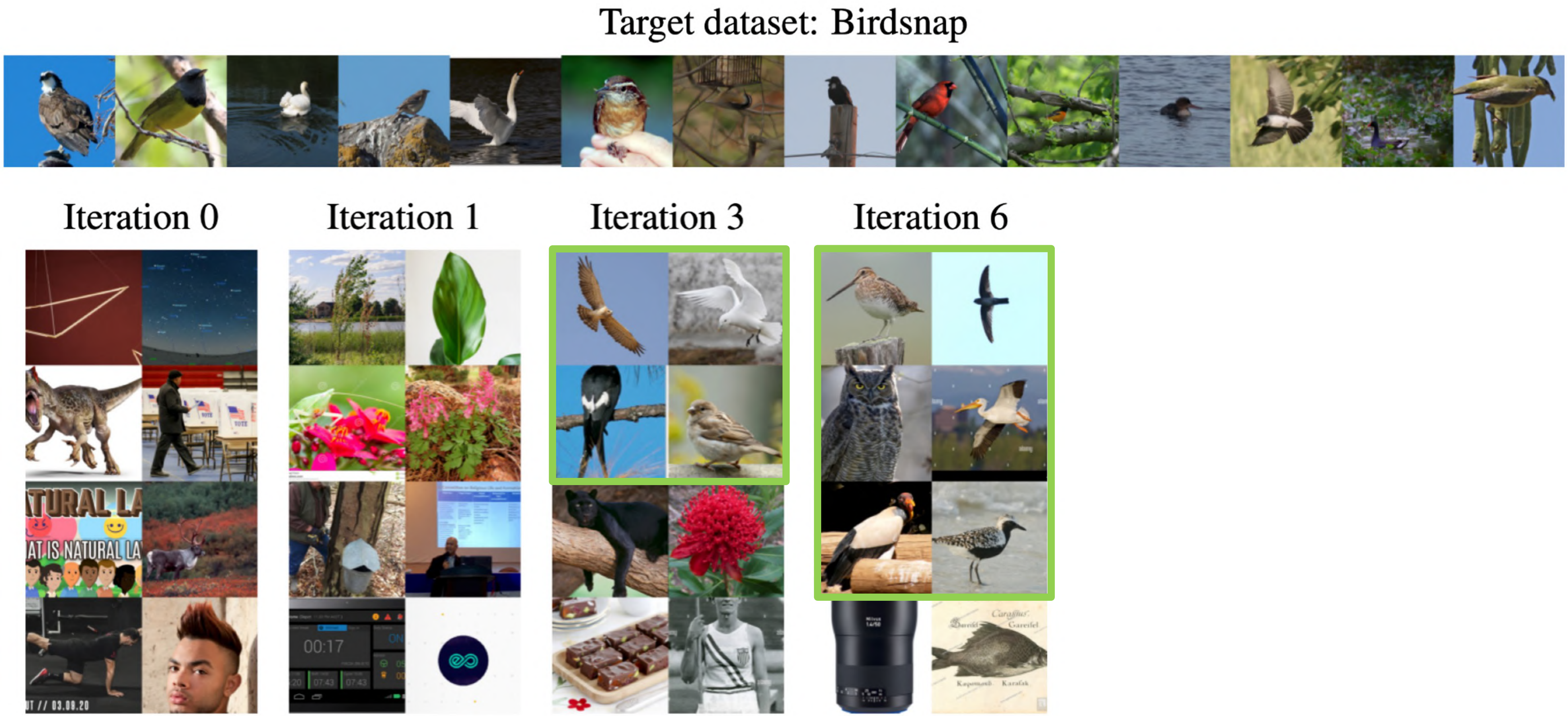


Iteration 0

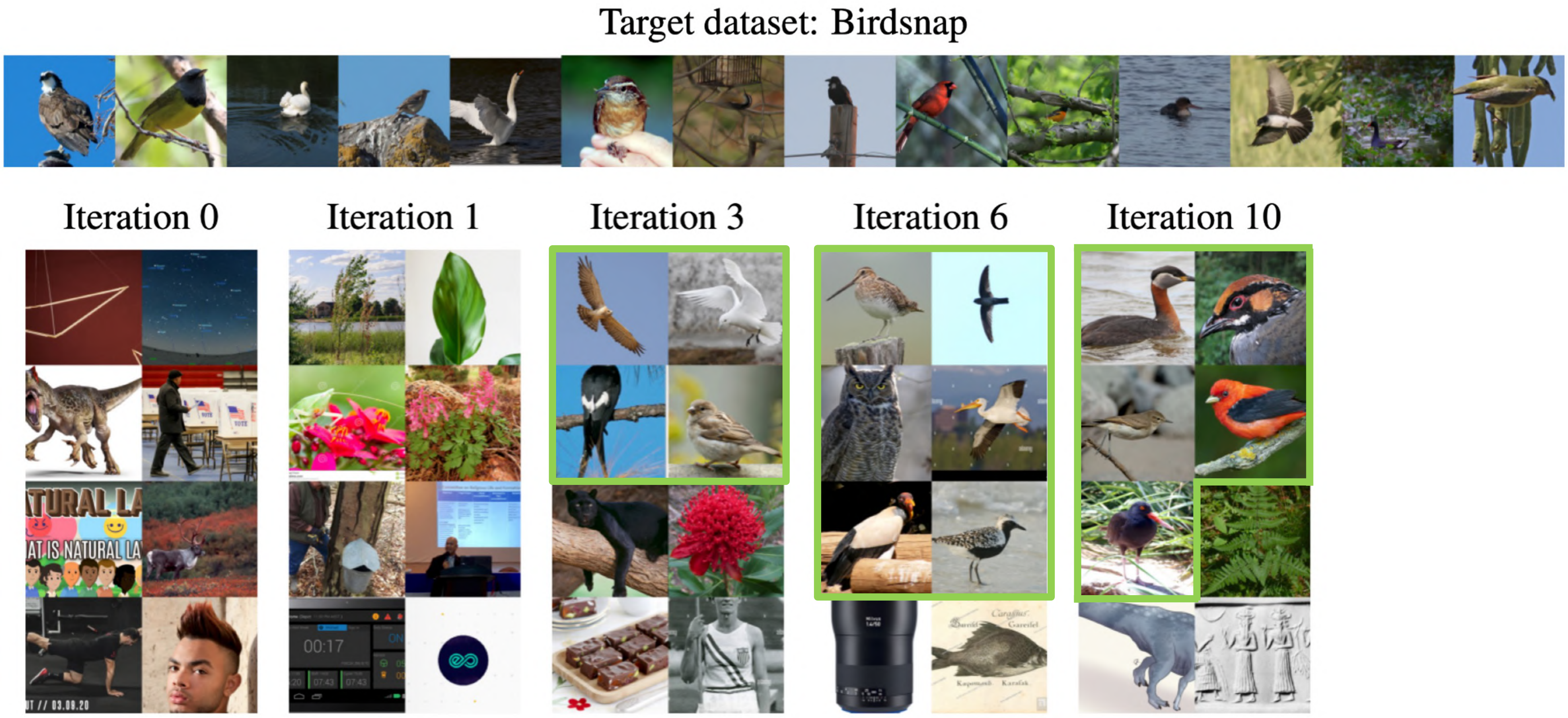
Iteration 1

Iteration 3

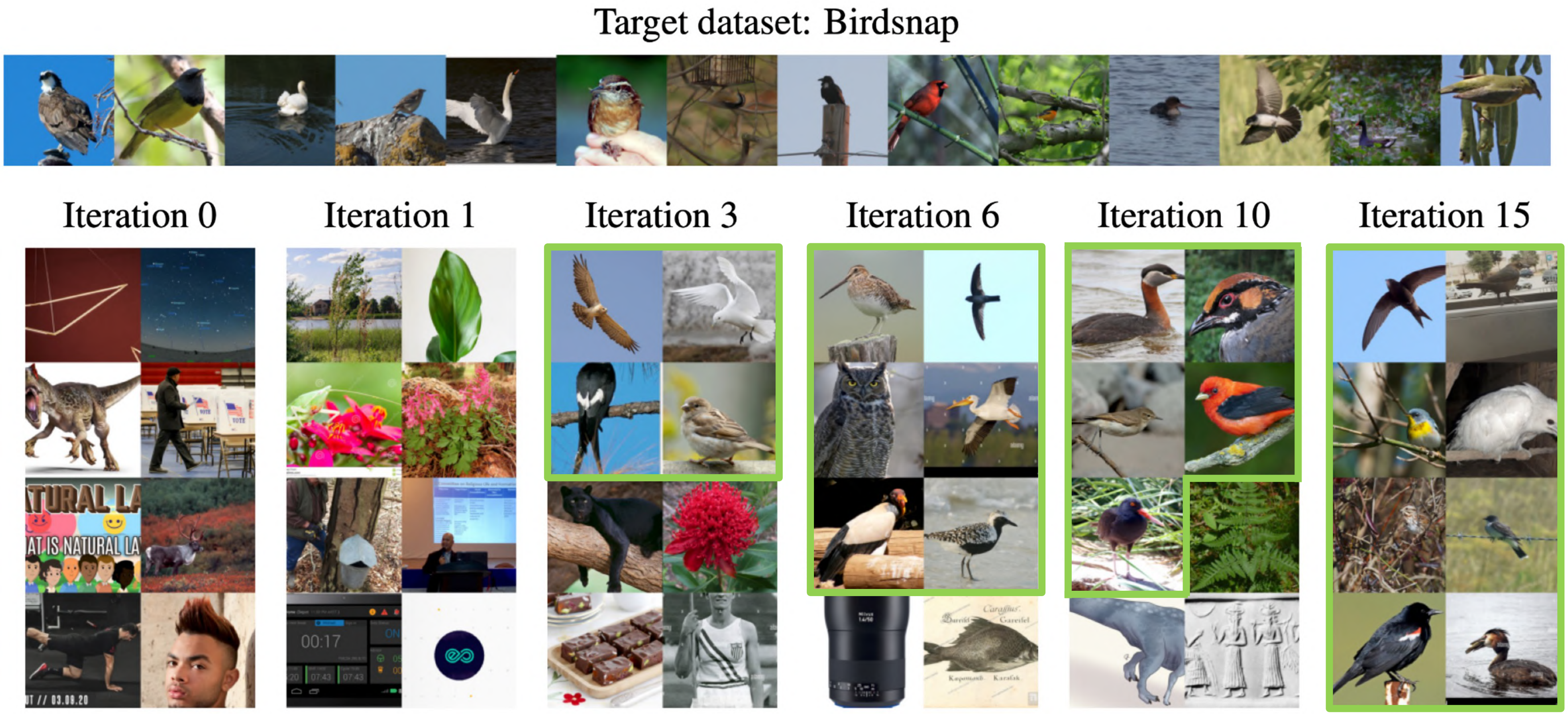
Self-supervised exploration progressively finds relevant data



Self-supervised exploration progressively finds relevant data



Self-supervised exploration progressively finds relevant data



Internet Explorer outperforms fixed datasets

Model	Birdsnap	Flowers	Food	Pets	VOC2007	Images	GPU-hours
Fixed dataset, self-supervised							
MoCo-v3 (ImageNet + target)	39.9	94.6	78.3	85.3	58.0†	1.2M	84

Table 1. **Improved representation quality (linear probe accuracy) with Internet Explorer.**

Internet Explorer outperforms fixed datasets

Model	Birdsnap	Flowers	Food	Pets	VOC2007	Images	GPU-hours
Fixed dataset, self-supervised							
MoCo-v3 (ImageNet + target)	39.9	94.6	78.3	85.3	$58.0^{\dagger}$	1.2M	84
Exploring the Internet							
Random exploration	39.6 (−0.3)	95.3 (+0.7)	77.0 (−1.3)	85.6 (+0.3)	70.2 (+12.2)	2.2M	124

Table 1. **Improved representation quality (linear probe accuracy) with Internet Explorer.**

Internet Explorer outperforms fixed datasets

Model	Birdsnap	Flowers	Food	Pets	VOC2007	Images	GPU-hours
Fixed dataset, self-supervised							
MoCo-v3 (ImageNet + target)	39.9	94.6	78.3	85.3	58.0[†]	1.2M	84
Exploring the Internet							
Random exploration	39.6 (−0.3)	95.3 (+0.7)	77.0 (−1.3)	85.6 (+0.3)	70.2 (+12.2)	2.2M	124
Search labels only	47.1 (+7.2)	96.3 (+1.7)	80.9 (+2.6)	85.7 (+0.4)	61.8 (+3.8)	2.2M	124

Table 1. **Improved representation quality (linear probe accuracy) with Internet Explorer.**

Internet Explorer outperforms fixed datasets

Model	Birdsnap	Flowers	Food	Pets	VOC2007	Images	GPU-hours
Fixed dataset, self-supervised							
MoCo-v3 (ImageNet + target)	39.9	94.6	78.3	85.3	58.0†	1.2M	84
Exploring the Internet							
Random exploration	39.6 (−0.3)	95.3 (+0.7)	77.0 (−1.3)	85.6 (+0.3)	70.2 (+12.2)	2.2M	124
Search labels only	47.1 (+7.2)	96.3 (+1.7)	80.9 (+2.6)	85.7 (+0.4)	61.8 (+3.8)	2.2M	124
Ours++ (no label set)	54.4 (+14.5)	98.4 (+3.8)	82.2 (+3.9)	89.6 (+4.3)	80.1 (+**22.1**)	2.2M	124

Table 1. **Improved representation quality (linear probe accuracy) with Internet Explorer.**

Internet Explorer outperforms fixed datasets

Model	Birdsnap	Flowers	Food	Pets	VOC2007	Images	GPU-hours
Fixed dataset, self-supervised							
MoCo-v3 (ImageNet + target)	39.9	94.6	78.3	85.3	58.0†	1.2M	84
Exploring the Internet							
Random exploration	39.6 (−0.3)	95.3 (+0.7)	77.0 (−1.3)	85.6 (+0.3)	70.2 (+12.2)	2.2M	124
Search labels only	47.1 (+7.2)	96.3 (+1.7)	80.9 (+2.6)	85.7 (+0.4)	61.8 (+3.8)	2.2M	124
Ours++ (no label set)	54.4 (+14.5)	98.4 (+3.8)	82.2 (+3.9)	89.6 (+4.3)	80.1 (+**22.1**)	2.2M	124
Ours++ (with label set)	**62.8** (+**22.9**)	**99.1** (+**4.5**)	84.6 (+**6.3**)	**90.8** (+**5.5**)	79.6 (+21.6)	2.2M	124

Table 1. **Improved representation quality (linear probe accuracy) with Internet Explorer.**

Internet Explorer outperforms fixed datasets

Model	Birdsnap	Flowers	Food	Pets	VOC2007	Images	GPU-hours
Fixed dataset, self-supervised							
MoCo-v3 (ImageNet + target)	39.9	94.6	78.3	85.3	58.0†	1.2M	84
Exploring the Internet							
Random exploration	39.6 (−0.3)	95.3 (+0.7)	77.0 (−1.3)	85.6 (+0.3)	70.2 (+12.2)	2.2M	124
Search labels only	47.1 (+7.2)	96.3 (+1.7)	80.9 (+2.6)	85.7 (+0.4)	61.8 (+3.8)	2.2M	124
Ours++ (no label set)	54.4 (+14.5)	98.4 (+3.8)	82.2 (+3.9)	89.6 (+4.3)	80.1 (**+22.1**)	2.2M	124
Ours++ (with label set)	**62.8** (**+22.9**)	**99.1** (**+4.5**)	84.6 (**+6.3**)	**90.8** (**+5.5**)	79.6 (+21.6)	2.2M	124

+40 hrs
on 1 GPU

Table 1. **Improved representation quality (linear probe accuracy) with Internet Explorer.**

Internet Explorer outperforms fixed datasets

Model	Birdsnap	Flowers	Food	Pets	VOC2007	Images	GPU-hours
Fixed dataset, self-supervised							
MoCo-v3 (ImageNet + target)	39.9	94.6	78.3	85.3	58.0†	1.2M	84
Exploring the Internet							
Random exploration	39.6 (−0.3)	95.3 (+0.7)	77.0 (−1.3)	85.6 (+0.3)	70.2 (+12.2)	2.2M	124
Search labels only	47.1 (+7.2)	96.3 (+1.7)	80.9 (+2.6)	85.7 (+0.4)	61.8 (+3.8)	2.2M	124
Ours++ (no label set)	54.4 (+14.5)	98.4 (+3.8)	82.2 (+3.9)	89.6 (+4.3)	80.1 (+**22.1**)	2.2M	124
Ours++ (with label set)	**62.8** (+**22.9**)	**99.1** (+**4.5**)	84.6 (+**6.3**)	**90.8** (+**5.5**)	79.6 (+21.6)	2.2M	124
Fixed dataset, language supervision							
CLIP (**oracle & 2x params**)	57.1	96.0	**86.4**	88.4	**86.7**	400M	4000

Table 1. **Improved representation quality (linear probe accuracy) with Internet Explorer.**

+40 hrs
on 1 GPU

Internet Explorer outperforms fixed datasets

Model	Birdsnap	Flowers	Food	Pets	VOC2007	Images	GPU-hours
Fixed dataset, self-supervised							
MoCo-v3 (ImageNet + target)	39.9	94.6	78.3	85.3	58.0†	1.2M	84
Exploring the Internet							
Random exploration	39.6 (−0.3)	95.3 (+0.7)	77.0 (−1.3)	85.6 (+0.3)	70.2 (+12.2)	2.2M	124
Search labels only	47.1 (+7.2)	96.3 (+1.7)	80.9 (+2.6)	85.7 (+0.4)	61.8 (+3.8)	2.2M	124
Ours++ (no label set)	54.4 (+14.5)	98.4 (+3.8)	82.2 (+3.9)	89.6 (+4.3)	80.1 (**+22.1**)	2.2M	124
Ours++ (with label set)	**62.8** (**+22.9**)	**99.1** (**+4.5**)	84.6 (**+6.3**)	**90.8** (**+5.5**)	79.6 (+21.6)	2.2M	124
Fixed dataset, language supervision							
CLIP (**oracle & 2x params**)	57.1	96.0	**86.4**	88.4	**86.7**	400M	4000

+40 hrs
on 1 GPU

32x time,
180x data

Table 1. **Improved representation quality (linear probe accuracy) with Internet Explorer.**

Are we just finding the test images online?

Are we just finding the test images online?

Are we just finding the test images online?

	Birdsnap	Flowers	Food	Pets	VOC2007	Images Downloaded
Target test set size	1849	6142	25246	3663	4952	—
No exploration						
Target training set overlap	1 (0.05%)	5 (0.01%)	34 (0.13%)	21 (0.57%)	0 (0.00%)	—

Are we just finding the test images online?

	Birdsnap	Flowers	Food	Pets	VOC2007	Images Downloaded
Target test set size	1849	6142	25246	3663	4952	–
No exploration						
Target training set overlap	1 (0.05%)	5 (0.01%)	34 (0.13%)	21 (0.57%)	0 (0.00%)	–
Internet Explorer						
Ours++ (no label set)	28 (+1.46%)	11 (+0.01%)	35 (+0.00%)	26 (+0.14%)	1 (+0.02%)	$\approx 10^6$

Are we just finding the test images online?

	Birdsnap	Flowers	Food	Pets	VOC2007	Images Downloaded
Target test set size	1849	6142	25246	3663	4952	–
No exploration						
Target training set overlap	1 (0.05%)	5 (0.01%)	34 (0.13%)	21 (0.57%)	0 (0.00%)	–
Internet Explorer						
Ours++ (no label set)	28 (+1.46%)	11 (+0.01%)	35 (+0.00%)	26 (+0.14%)	1 (+0.02%)	$\approx 10^6$
Ours++ (with label set)	57 (+3.03%)	27 (+0.36%)	35 (+0.00%)	43 (+0.60%)	1 (+0.02%)	$\approx 10^6$

But we are finding very relevant images…

Oxford-IIIT Pets

Test Img. Ranked Nearest Neighbors in Downloaded Images

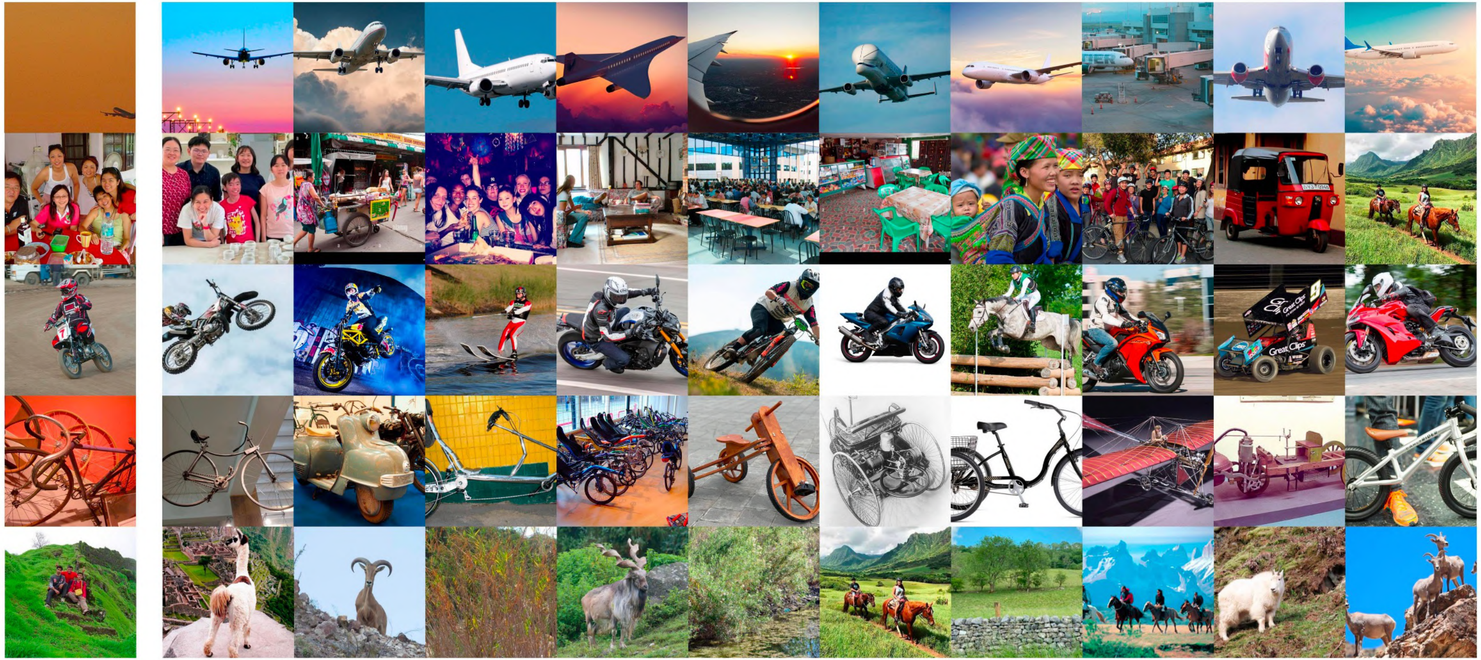
Food101

Test Img. Ranked Nearest Neighbors in Downloaded Images

Oxford Flowers 102

Test Img. Ranked Nearest Neighbors in Downloaded Images

VOC2007



Test Img. Ranked Nearest Neighbors in Downloaded Images

Internet Explorer is robust to choice of search engine

Internet Explorer is robust to choice of search engine

Google

Internet Explorer is robust to choice of search engine

Google Bing flickr . . .

Internet Explorer is robust to choice of search engine

Google Bing flickr ...

Q: do we rely on fancy tricks in modern search engines?

Internet Explorer is robust to choice of search engine

Google Bing flickr ...

Q: do we rely on fancy tricks in modern search engines?

What if we could create our *own* search engine using <u>just text</u>?

Show me: sunflower

LAION-5B

Large-scale Artificial Intelligence Open Network

Similar trends

Model	Flowers	Food	Pets

Table 1. **Linear probe accuracy with other search engines**. Internet Curiosity improves its performance using any search engine, including Flickr and our custom text-only LAION search engine.

Similar trends

Model	Flowers	Food	Pets
	Google	Google	Google

Table 1. **Linear probe accuracy with other search engines**. Internet Curiosity improves its performance using any search engine, including Flickr and our custom text-only LAION search engine.

Similar trends

Model	Flowers	Food	Pets
	Google	Google	Google
Fixed dataset			
MoCo-v3 (IN)	83.2	70.5	79.6
MoCo-v3 (IN + target)	94.6	78.3	85.3

Table 1. **Linear probe accuracy with other search engines**. Internet Curiosity improves its performance using any search engine, including Flickr and our custom text-only LAION search engine.

Similar trends

Model	Flowers	Food	Pets
	Google	Google	Google
Fixed dataset			
MoCo-v3 (IN)	83.2	70.5	79.6
MoCo-v3 (IN + target)	94.6	78.3	85.3
Undirected search			
Random exploration	95.3	77.0	85.6

Table 1. **Linear probe accuracy with other search engines**. Internet Curiosity improves its performance using any search engine, including Flickr and our custom text-only LAION search engine.

Similar trends

Model	Flowers	Food	Pets
	Google	Google	Google
Fixed dataset			
MoCo-v3 (IN)	83.2	70.5	79.6
MoCo-v3 (IN + target)	94.6	78.3	85.3
Undirected search			
Random exploration	95.3	77.0	85.6
Internet Explorer			
Ours++ (no label set)	98.4	81.2	87.3
Ours++ (with label set)	**99.1**	**83.8**	**90.8**

Table 1. **Linear probe accuracy with other search engines**. Internet Curiosity improves its performance using any search engine, including Flickr and our custom text-only LAION search engine.

Similar trends

Model	Flowers		Food		Pets	
	Google	Flickr	Google	Flickr	Google	Flickr
Fixed dataset						
MoCo-v3 (IN)	83.2	83.2	70.5	70.5	79.6	79.6
MoCo-v3 (IN + target)	94.6	94.6	78.3	78.3	85.3	85.3
Undirected search						
Random exploration	95.3	95.2	77.0	80.0	85.6	84.4
Internet Explorer						
Ours++ (no label set)	98.4	98.1	81.2	80.3	87.3	88.4
Ours++ (with label set)	**99.1**	**99.0**	**83.8**	**81.9**	**90.8**	**89.1**

Table 1. **Linear probe accuracy with other search engines**. Internet Curiosity improves its performance using any search engine, including Flickr and our custom text-only LAION search engine.

Similar trends

Model	Flowers			Food			Pets		
	Google	Flickr	LAION	Google	Flickr	LAION	Google	Flickr	LAION
Fixed dataset									
MoCo-v3 (IN)	83.2	83.2	83.2	70.5	70.5	70.5	79.6	79.6	79.6
MoCo-v3 (IN + target)	94.6	94.6	94.6	78.3	78.3	78.3	85.3	85.3	85.3
Undirected search									
Random exploration	95.3	95.2	94.8	77.0	80.0	80.2	85.6	84.4	85.1
Internet Explorer									
Ours++ (no label set)	98.4	98.1	94.6	81.2	80.3	80.9	87.3	88.4	85.9
Ours++ (with label set)	**99.1**	**99.0**	**95.8**	**83.8**	**81.9**	**81.0**	**90.8**	**89.1**	**86.7**

Table 1. **Linear probe accuracy with other search engines**. Internet Curiosity improves its performance using any search engine, including Flickr and our custom text-only LAION search engine.

What's next on the open web?

What’s next on the open web?

- Scale to larger / more diverse datasets like ImageNet

What’s next on the open web?

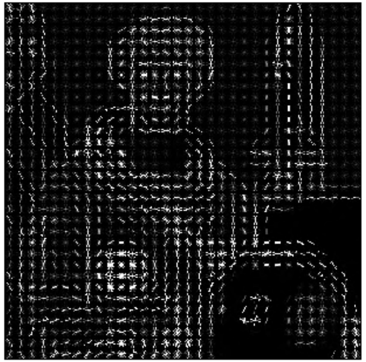
- Scale to larger / more diverse datasets like ImageNet
- Apply to more challenging vision tasks, videos, and robotics

What’s next on the open web?

- Scale to larger / more diverse datasets like ImageNet
- Apply to more challenging vision tasks, videos, and robotics
- Finetune a CLIP model online using captions + search terms!

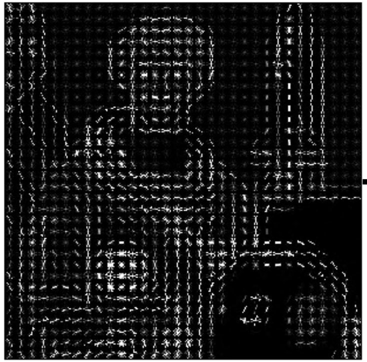
Deep Learning

Deep Learning



Handcrafted features

Deep Learning

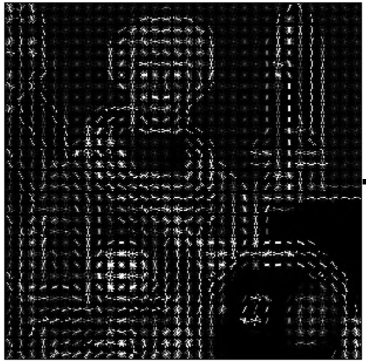


Handcrafted features

Model learns features

Deep Learning

Internet Explorer

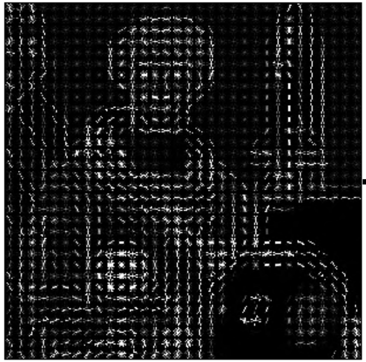


Handcrafted features

Model learns features

Deep Learning

Internet Explorer

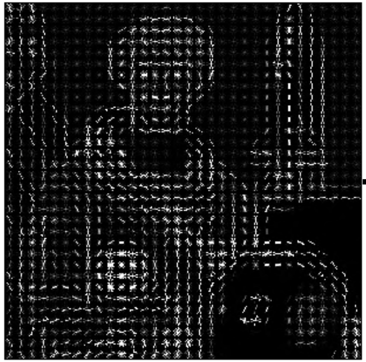


Handcrafted features

Model learns features

Handcrafted dataset

Deep Learning



Handcrafted features

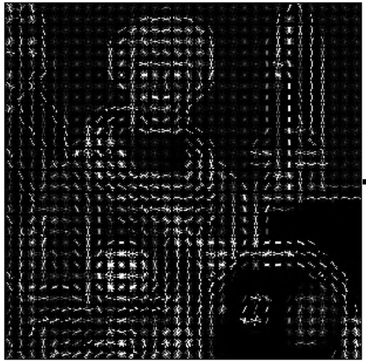
Model learns features

Internet Explorer

Handcrafted dataset

Model learns to craft its own dataset

Deep Learning



Handcrafted features

Model learns features

Internet Explorer

Handcrafted dataset

Model learns to craft its own dataset

http://internet-explorer-ssl.github.io

Questions?

Your Diffusion Model is Secretly a Zero-Shot Classifier

NEW

Alexander C. Li   Mihir Prabhudesai   Shivam Duggal   Ellis Brown   Deepak Pathak

Carnegie Mellon University

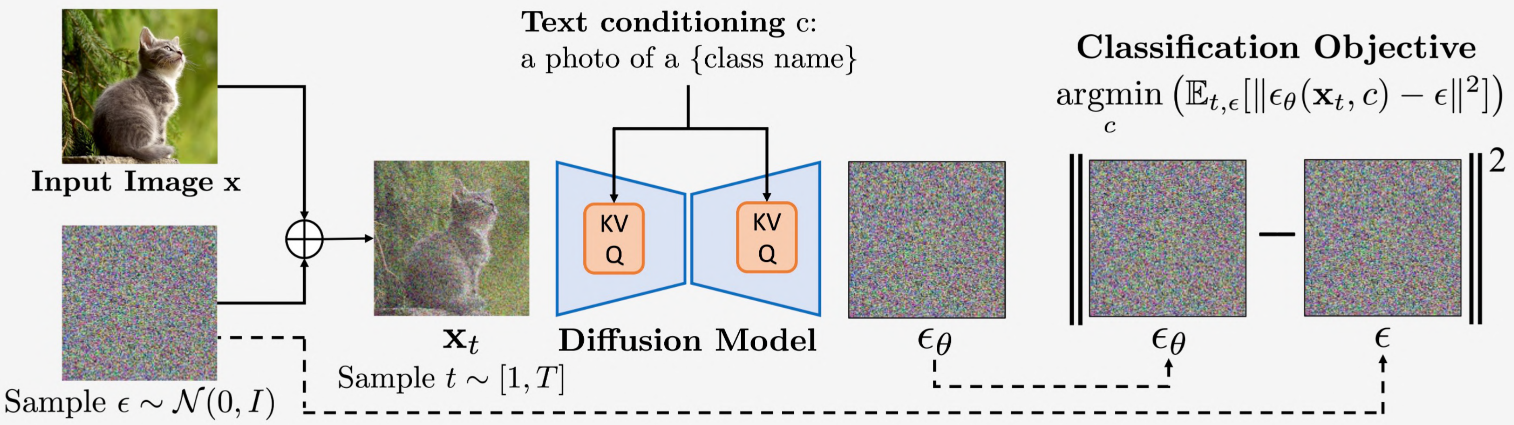
Your Diffusion Model is Secretly a Zero-Shot Classifier

NEW

Alexander C. Li   Mihir Prabhudesai   Shivam Duggal   Ellis Brown   Deepak Pathak

Carnegie Mellon University

"Diffusion Classifier"



Bayes' Rule + Generative Model → Classification!

$$p_\theta(\mathbf{c}_i \mid \mathbf{x}) = \frac{p(\mathbf{c}_i)\ p_\theta(\mathbf{x} \mid \mathbf{c}_i)}{\sum_j p(\mathbf{c}_j)\ p_\theta(\mathbf{x} \mid \mathbf{c}_j)}$$

Bayes' Rule + Generative Model → Classification!

$$p_\theta(\mathbf{c}_i \mid \mathbf{x}) = \frac{p(\mathbf{c}_i)\ p_\theta(\mathbf{x} \mid \mathbf{c}_i)}{\sum_j p(\mathbf{c}_j)\ p_\theta(\mathbf{x} \mid \mathbf{c}_j)}$$

$$p(\mathbf{c}_i) = \frac{1}{N}$$

We use a uniform label distribution and a simple approximate ELBO to get:

$$\text{ELBO} \approx -\mathbb{E}_{t,\epsilon}[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, \mathbf{c}_i)\|^2]$$

Bayes' Rule + Generative Model → Classification!

$$p_\theta(\mathbf{c}_i \mid \mathbf{x}) = \frac{p(\mathbf{c}_i)\ p_\theta(\mathbf{x} \mid \mathbf{c}_i)}{\sum_j p(\mathbf{c}_j)\ p_\theta(\mathbf{x} \mid \mathbf{c}_j)}$$

$$p(\mathbf{c}_i) = \frac{1}{N}$$

We use a uniform label distribution and a simple approximate ELBO to get:

$$\text{ELBO} \approx -\mathbb{E}_{t,\epsilon}[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, \mathbf{c}_i)\|^2]$$

$$p_\theta(\mathbf{c}_i \mid \mathbf{x}) \approx \frac{\exp\{-\mathbb{E}_{t,\epsilon}[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, \mathbf{c}_i)\|^2]\}}{\sum_j \exp\{-\mathbb{E}_{t,\epsilon}[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, \mathbf{c}_j)\|^2]\}}$$

Diffusion Classifier – **OOD Generalization**

	Zero-shot?	Food101	CIFAR10	FGVC	Oxford Pets	Flowers102	STL10	ImageNet	ObjectNet
Synthetic SD Data	✓	12.6	35.3	9.4	31.3	22.1	38.0	18.9	5.2
SD Features	✗	73.0	84.0	**35.2**	75.9	**70.0**	87.2	56.6	10.2
Diffusion Classifier (ours)	✓	**77.9**	**87.1**	24.3	**86.2**	59.4	**95.3**	**58.9**	**38.3**
CLIP ResNet-50	✓	81.1	75.6	19.3	85.4	65.9	94.3	58.2	40.0
OpenCLIP ViT-H/14	✓	92.7	97.3	42.3	94.6	79.9	98.3	76.8	69.2

Using **Stable Diffusion** as an image-text model

Diffusion Classifier – **OOD Generalization**

	Zero-shot?	Food101	CIFAR10	FGVC	Oxford Pets	Flowers102	STL10	ImageNet	ObjectNet
Synthetic SD Data	✓	12.6	35.3	9.4	31.3	22.1	38.0	18.9	5.2
SD Features	✗	73.0	84.0	35.2	75.9	70.0	87.2	56.6	10.2
Diffusion Classifier (ours)	✓	**77.9**	**87.1**	24.3	**86.2**	59.4	**95.3**	**58.9**	**38.3**
CLIP ResNet-50	✓	81.1	75.6	19.3	85.4	65.9	94.3	58.2	40.0
OpenCLIP ViT-H/14	✓	92.7	97.3	42.3	94.6	79.9	98.3	76.8	69.2

Using **Stable Diffusion** as an image-text model

Method	**ID**	**OOD**		
	IN	IN-v2	IN-A	ObjectNet
ResNet-18	74.1	57.3	15.0	26.6
ResNet-34	78.1	59.8	10.5	31.6
ResNet-50	79.7	61.6	9.8	35.6
ResNet-101	82.2	63.2	19.5	38.2
ViT-L/32	79.0	61.6	26.3	29.9
ViT-L/16	81.0	66.6	25.6	36.7
ViT-B/16	83.4	66.6	30.1	37.8
Diffusion Classifier	78.9	62.1	22.6	32.3

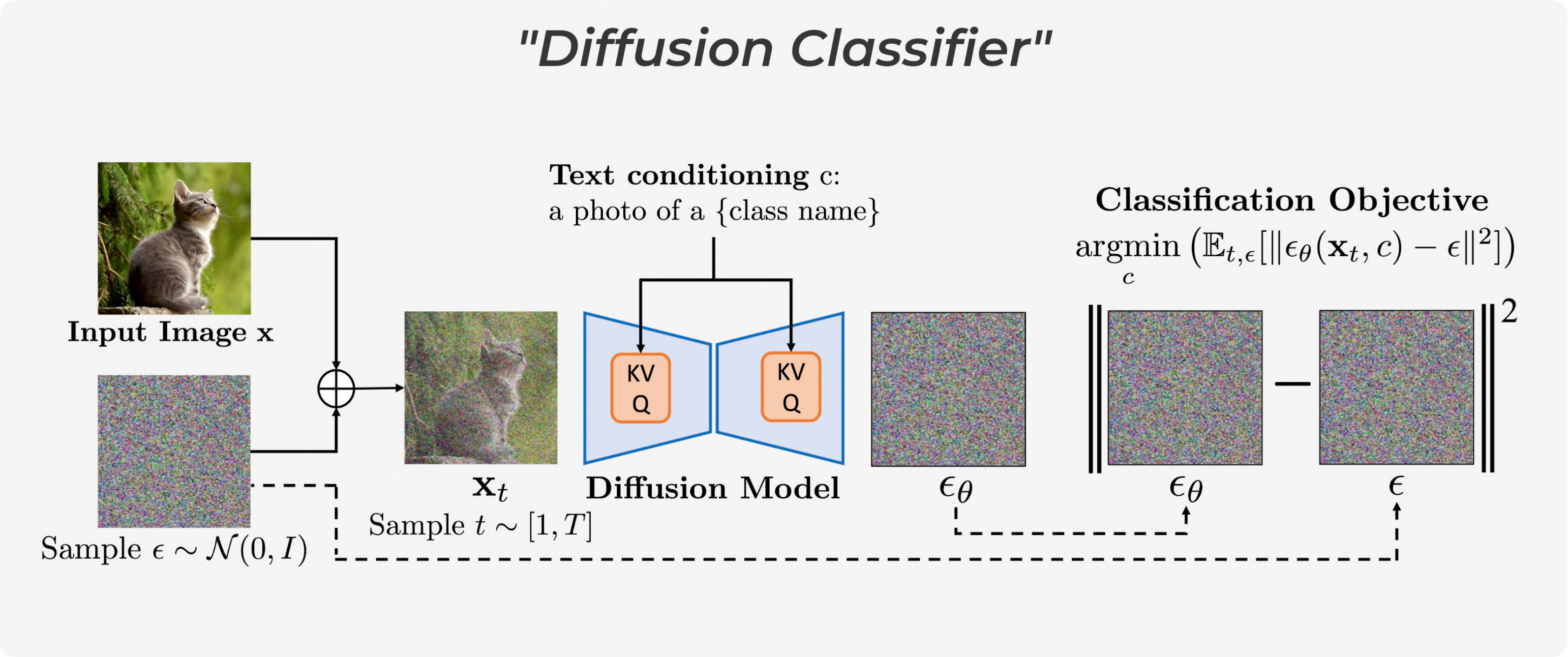
Table 3. **Diffusion Classifier performs well ID and OOD.**

Using **Diffusion Transformers (DiT)** as a class-conditioned diffusion model

Peebles & Xie. *Scalable Diffusion Models with Transformers* **(DiT)**
Rombach et al. *High-Resolution Image Synthesis with Latent Diffusion Models* **(Stable Diffusion)**

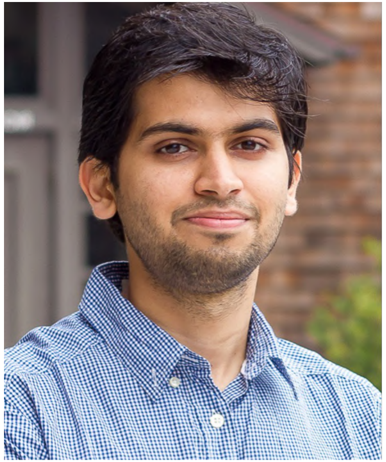
Diffusion Classifier – **Compositional Reasoning**

Thrush et al. Winoground: Probing Vision and Language Models for Visio-Linguistic Compositionality



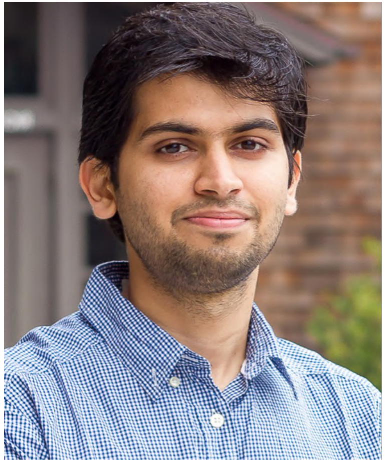
https://diffusion-classifier.github.io/

Acknowledgements



Deepak Pathak
(advisor)

Thank you, Thesis Committee!

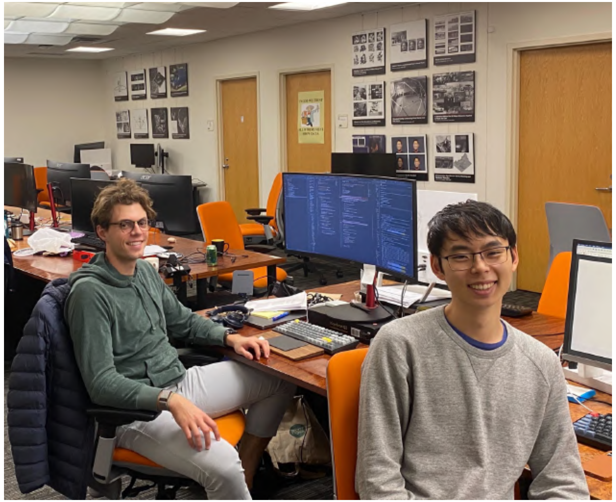


Deepak Pathak
(advisor)

Deva Ramanan

Alyosha Efros

Thank you, LEAP Lab!



Thank you, MSCS folks!

Thanks to all my friends in Smith Hall, NSH, and around CMU for a great 2 years!

Thanks to my family for your support throughout!