# DSC190: Machine Learning with Few Labels

## Self-Supervised Learning

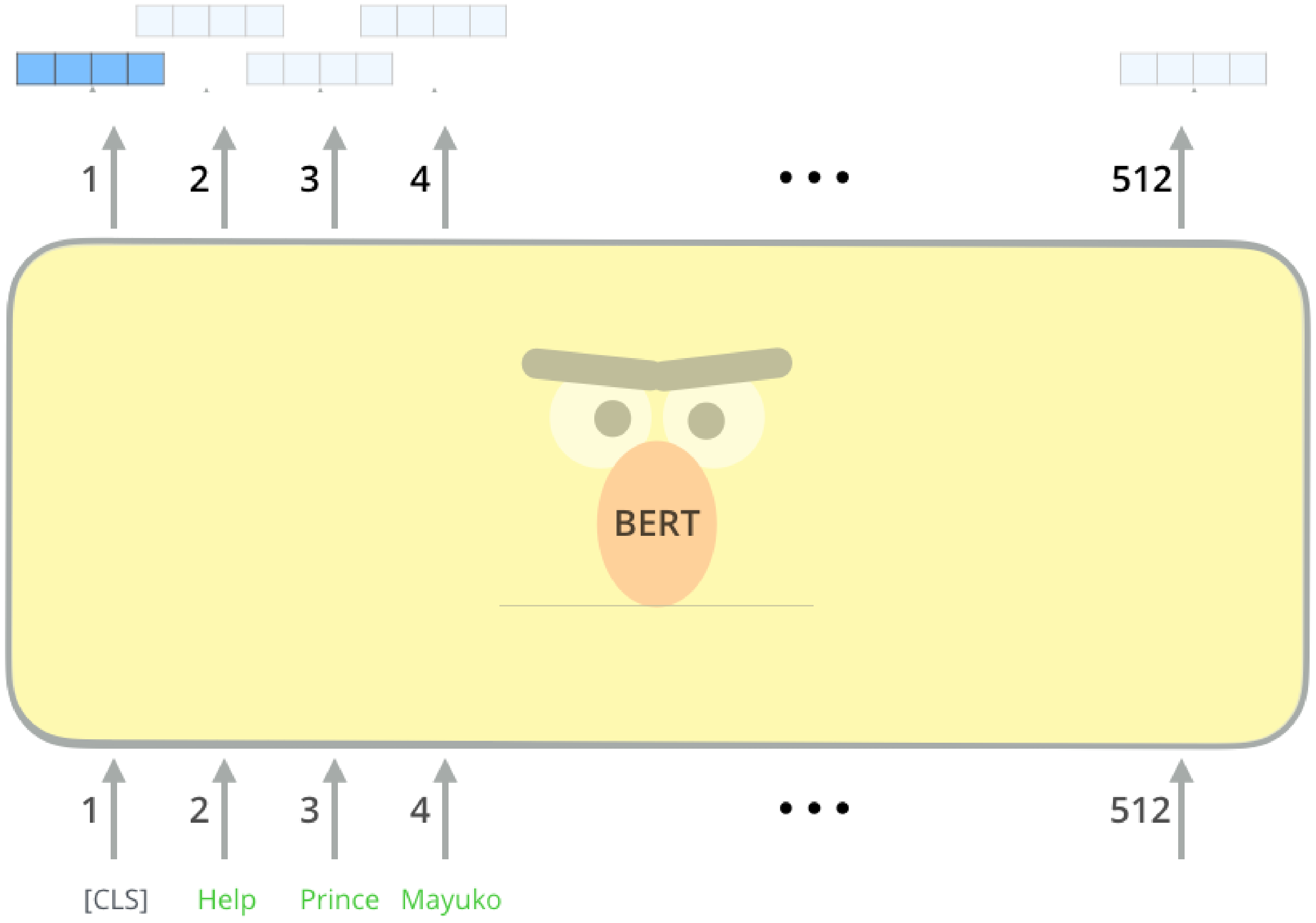
**Zhiting Hu**

Lecture 6, October 11, 2024

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

# BERT

- BERT: A bidirectional model to extract contextual word embedding



# BERT: Pre-training Procedure

- Dataset:
  - Wikipedia (2.5B words) + a collection of free ebooks (800M words)

# BERT: Pre-training Procedure

- Dataset:
  - Wikipedia (2.5B words) + a collection of free ebooks (800M words)

- Training: **masked language model** (masked LM)
  - Masks some percent of words from the input and has to reconstruct those words from context

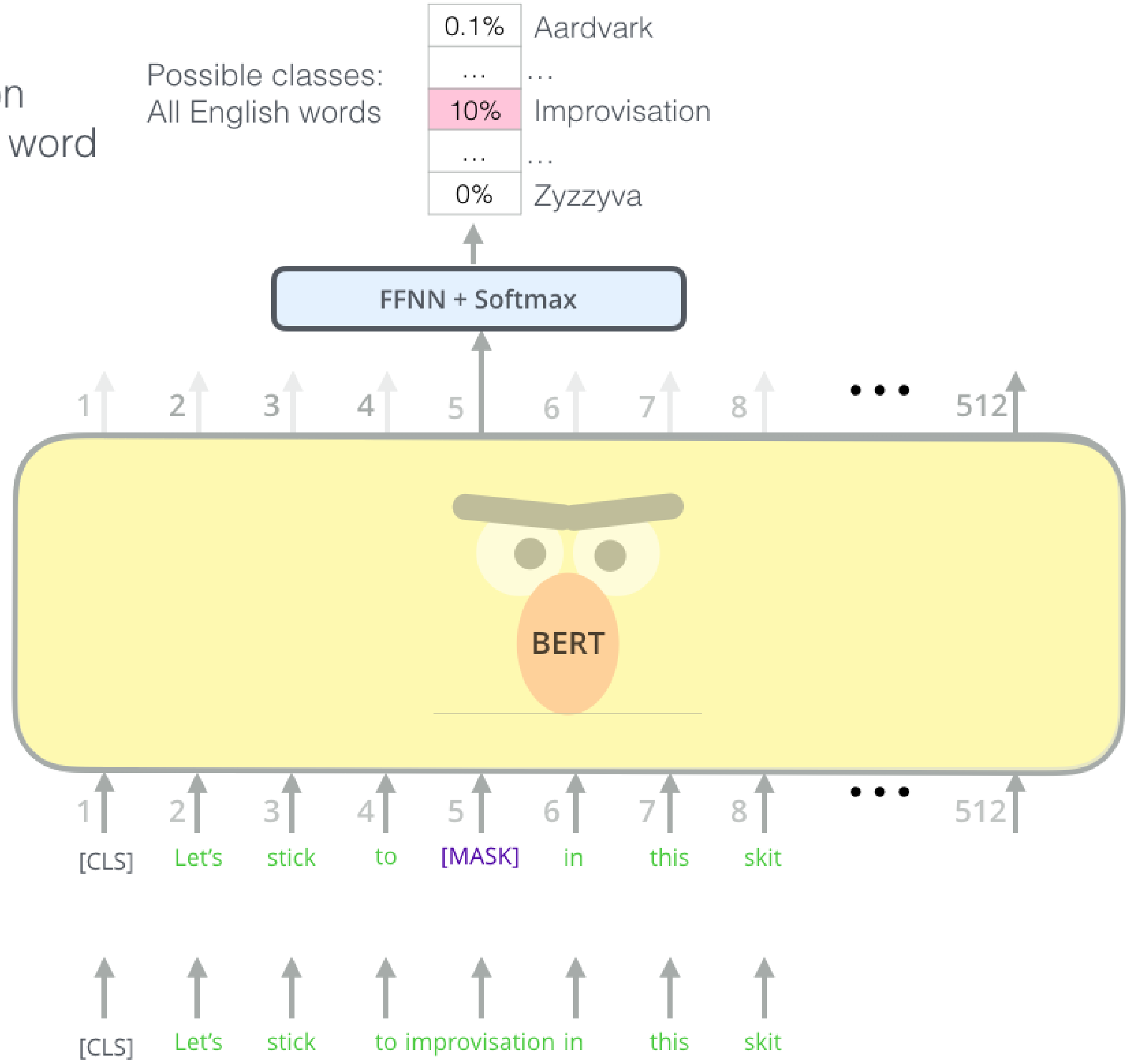
# BERT: Pre-training Procedure

- Masked LM

Use the output of the masked word's position to predict the masked word

Randomly mask 15% of tokens

Input

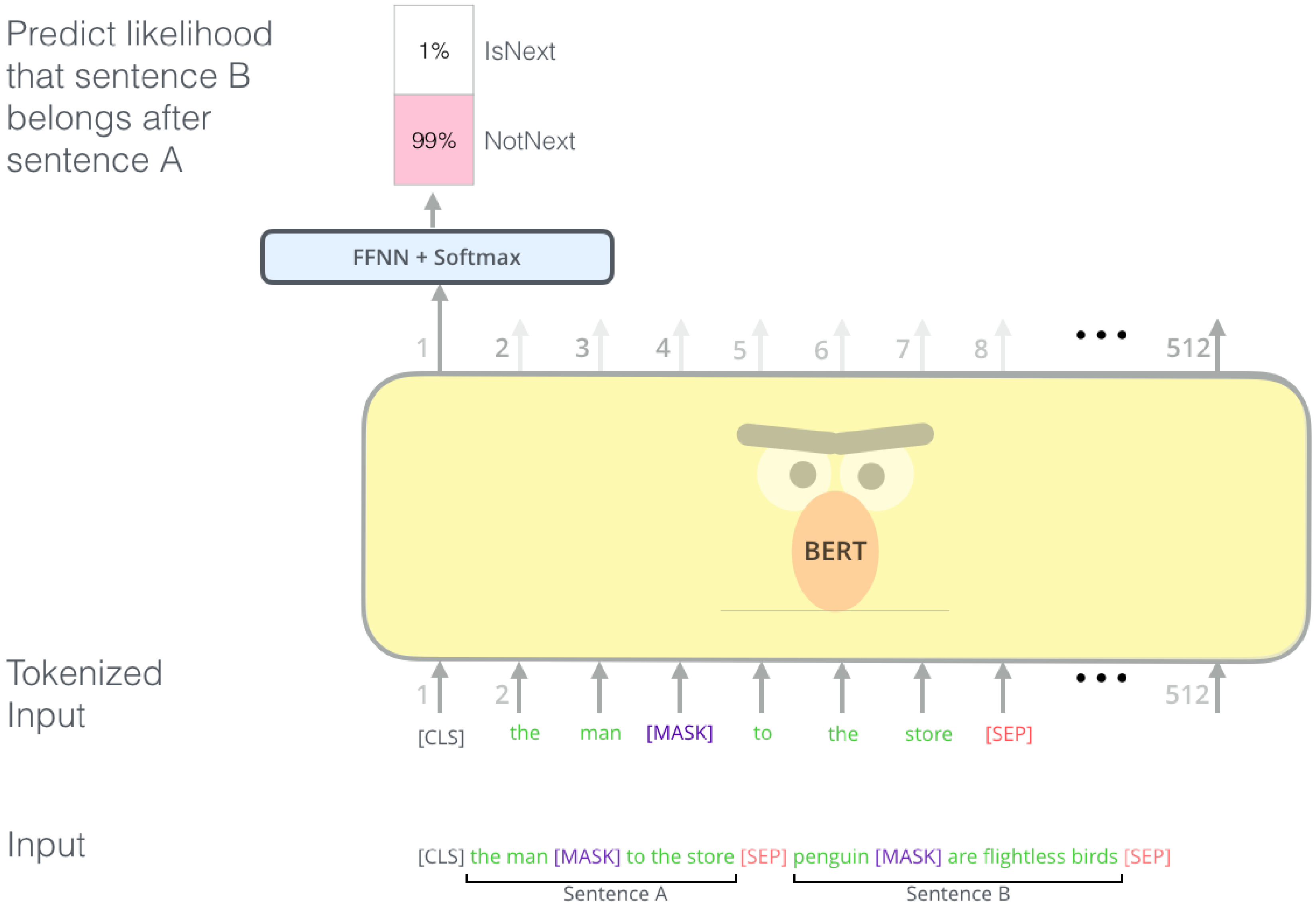


# BERT: Pre-training Procedure

- Dataset:
  - Wikipedia (2.5B words) + a collection of free ebooks (800M words)
- Training procedure
  - **masked language model** (masked LM)
    - Masks some percent of words from the input and has to reconstruct those words from context
  - **Two-sentence task**
    - To understand relationships between sentences
    - Concatenate two sentences A and B and predict whether B actually comes after A in the original text

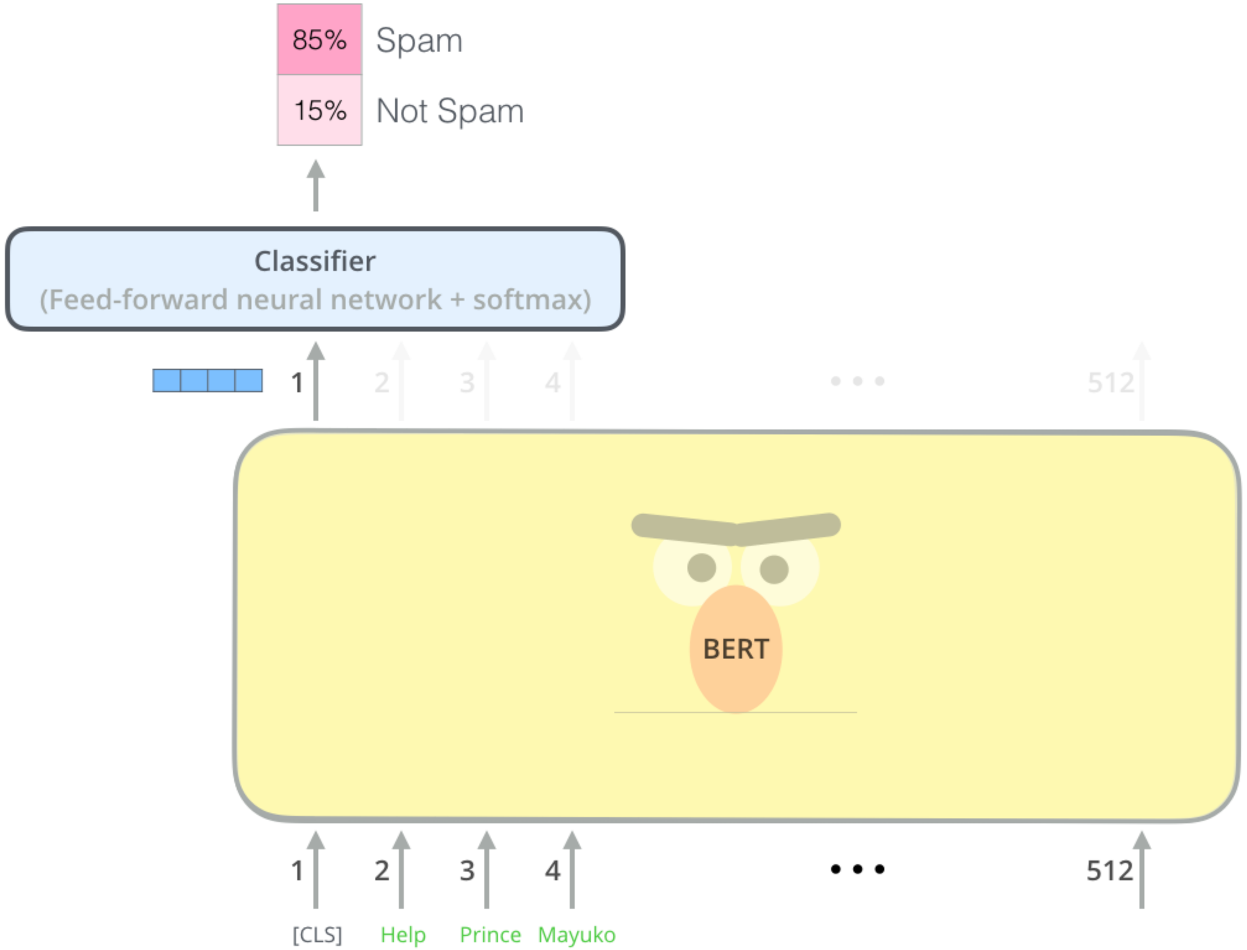
# BERT: Pre-training Procedure

- Two sentence task

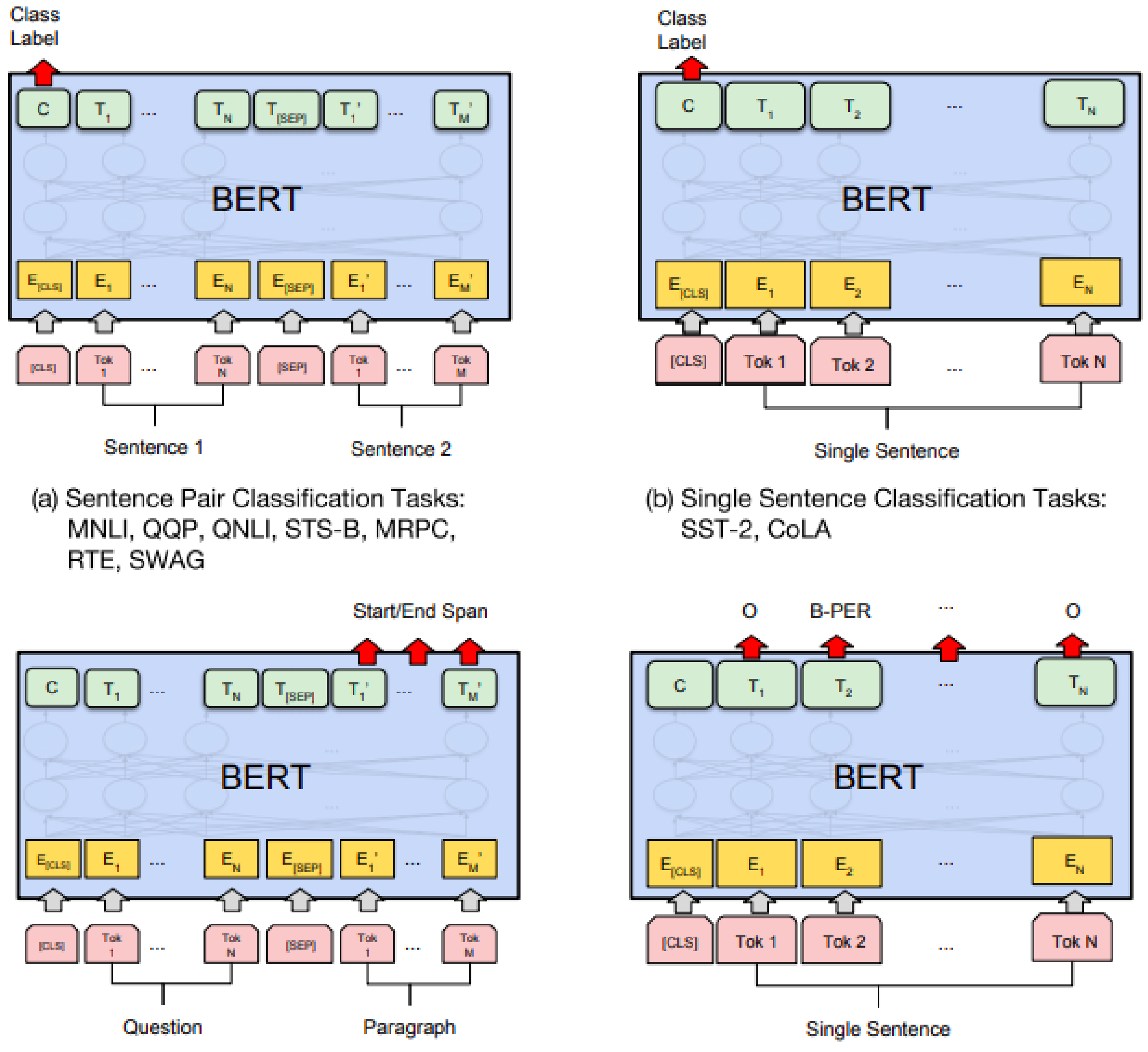


# BERT: Downstream Fine-tuning

- Use BERT for sentence classification



# BERT: Downstream Fine-tuning




(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

(b) Single Sentence Classification Tasks: SST-2, CoLA

(c) Question Answering Tasks: SQuAD v1.1

(d) Single Sentence Tagging Tasks: CoNLL-2003 NER

# BERT Results

- Huge improvements over SOTA on 12 NLP task

| System | MNLI-(m/mm) | QQP | QNLI | SST-2 | CoLA | STS-B | MRPC | RTE | **Average** |
|---|---|---|---|---|---|---|---|---|---|
| | 392k | 363k | 108k | 67k | 8.5k | 5.7k | 3.5k | 2.5k | - |
| Pre-OpenAI SOTA | 80.6/80.1 | 66.1 | 82.3 | 93.2 | 35.0 | 81.0 | 86.0 | 61.7 | 74.0 |
| BiLSTM+ELMo+Attn | 76.4/76.1 | 64.8 | 79.9 | 90.4 | 36.0 | 73.3 | 84.9 | 56.8 | 71.0 |
| OpenAI GPT | 82.1/81.4 | 70.3 | 88.1 | 91.3 | 45.4 | 80.0 | 82.3 | 56.0 | 75.2 |
| $\text{BERT}_{\text{BASE}}$ | 84.6/83.4 | 71.2 | 90.1 | 93.5 | 52.1 | 85.8 | 88.9 | 66.4 | 79.6 |
| $\text{BERT}_{\text{LARGE}}$ | **86.7/85.9** | **72.1** | **91.1** | **94.9** | **60.5** | **86.5** | **89.3** | **70.1** | **81.9** |

Table 1: GLUE Test results, scored by the GLUE evaluation server. The number below each task denotes the number of training examples. The "Average" column is slightly different than the official GLUE score, since we exclude the problematic WNLI set. OpenAI GPT = (L=12, H=768, A=12); $\text{BERT}_{\text{BASE}}$ = (L=12, H=768, A=12); $\text{BERT}_{\text{LARGE}}$ = (L=24, H=1024, A=16). BERT and OpenAI GPT are single-model, single task. All results obtained from `https://gluebenchmark.com/leaderboard` and `https://blog.openai.com/language-unsupervised/`.

# SSL from Images, EX (I): masked autoencoder (MAE)

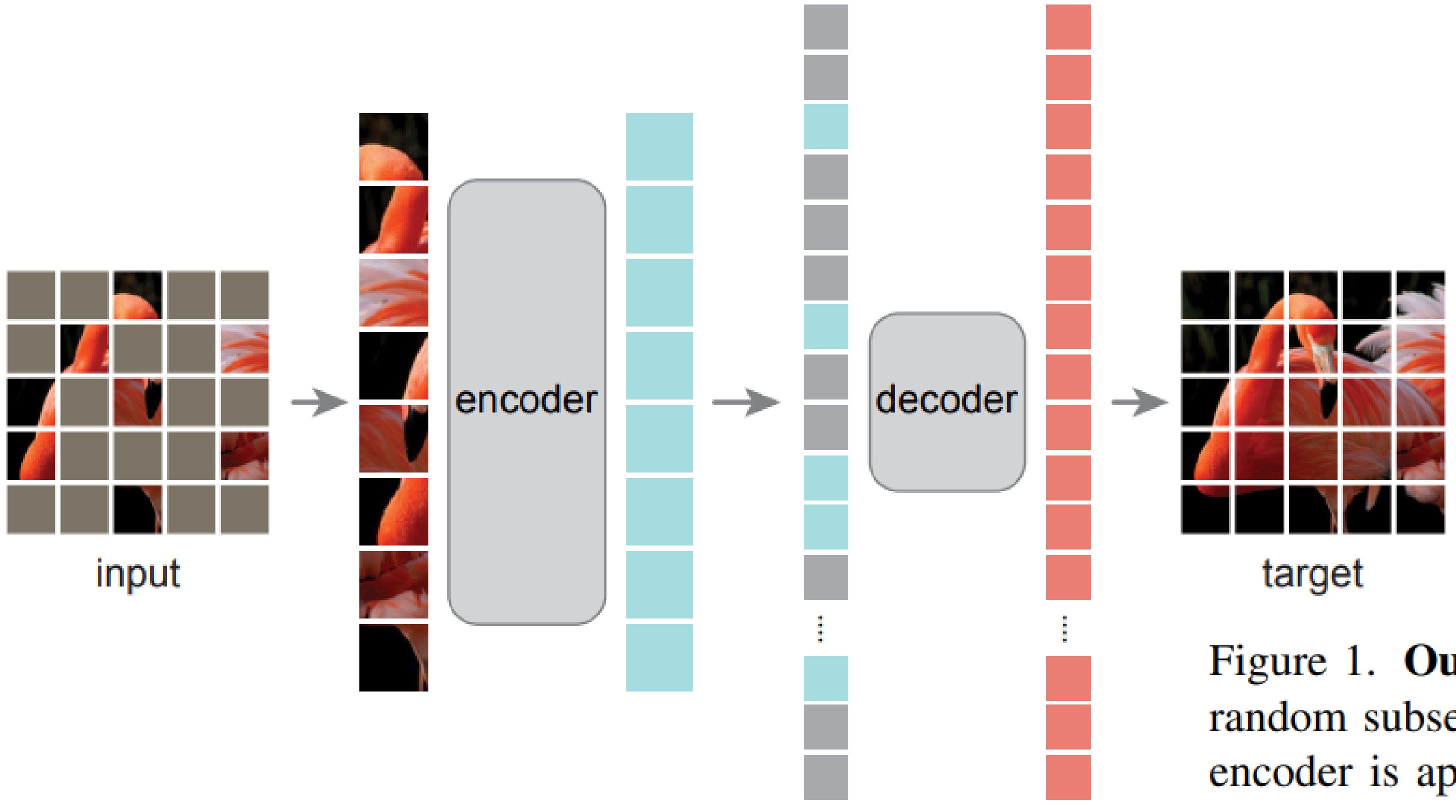
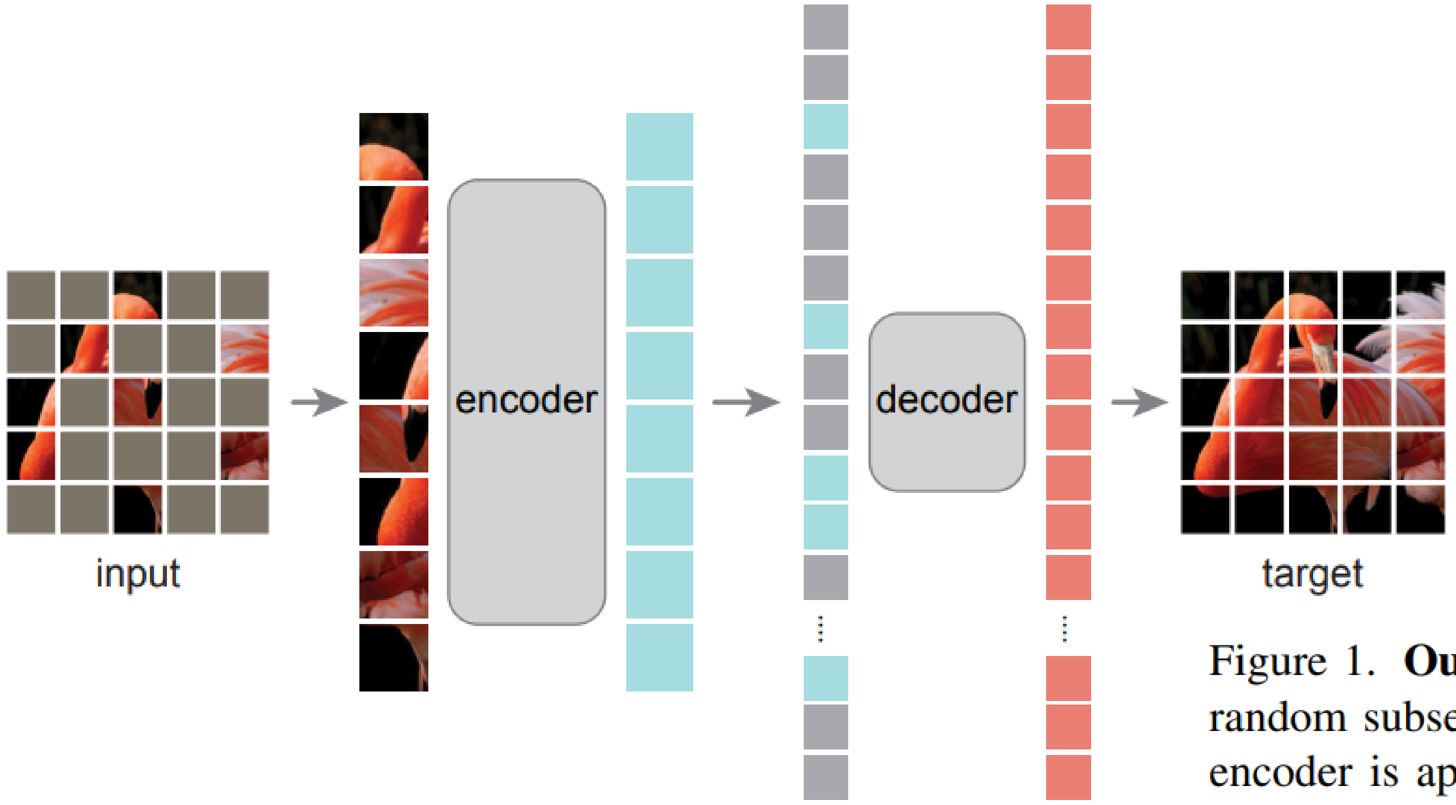



Figure 1. **Our MAE architecture**. During pre-training, a large random subset of image patches (*e.g.*, 75%) is masked out. The encoder is applied to the small subset of *visible patches*. Mask tokens are introduced *after* the encoder, and the full set of encoded patches and mask tokens is processed by a small decoder that reconstructs the original image in pixels. After pre-training, the decoder is discarded and the encoder is applied to uncorrupted images (full sets of patches) for recognition tasks.

[He et al., 2021: Masked Autoencoders Are Scalable Vision Learners]

# SSL from Images, EX (I): masked autoencoder (MAE)



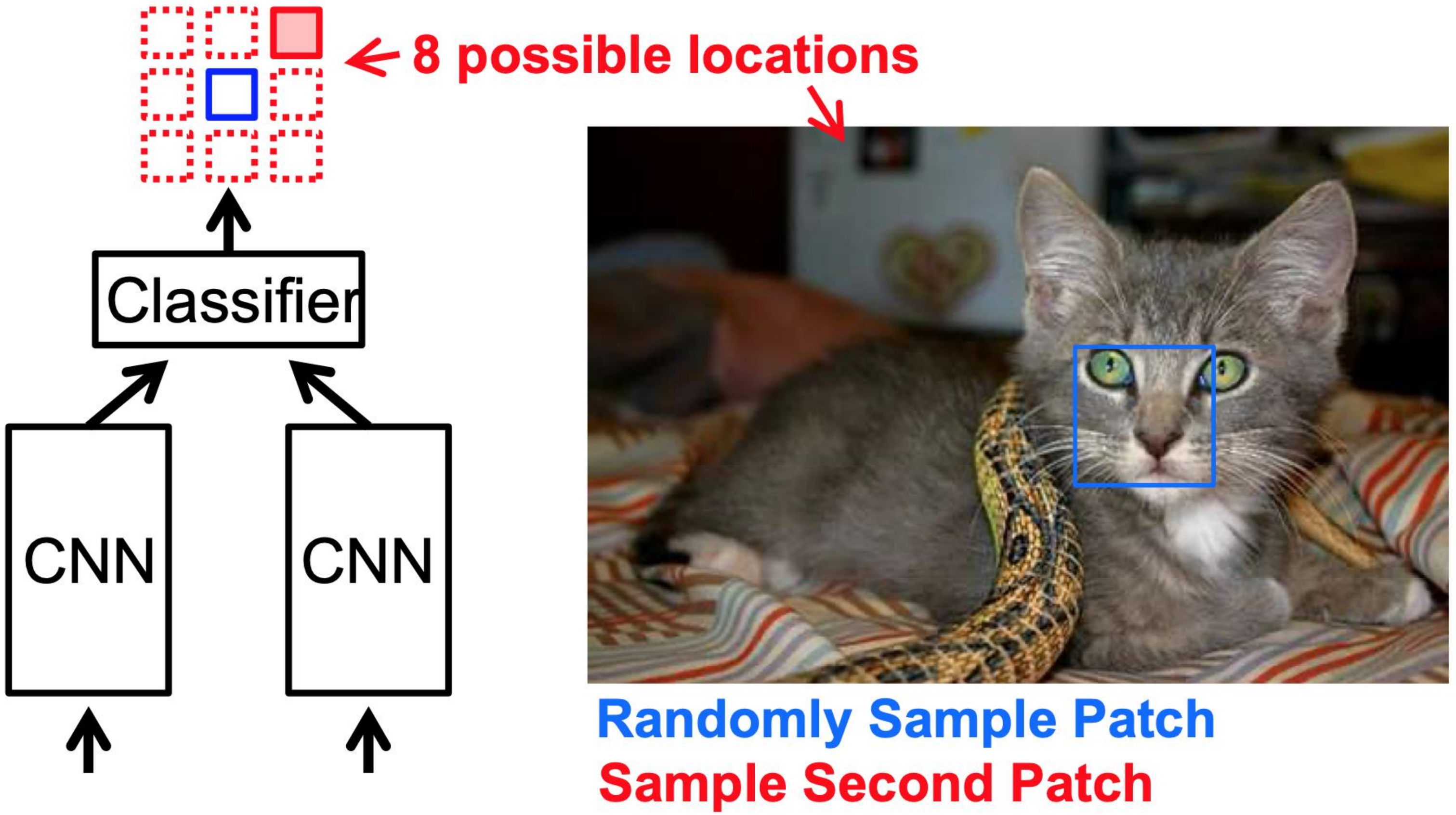

**Question:** Why is this (75%) much larger than the mask rate in BERT (15%)?

Figure 1. **Our MAE architecture**. During pre-training, a large random subset of image patches (*e.g.*, 75%) is masked out. The encoder is applied to the small subset of *visible patches*. Mask tokens are introduced *after* the encoder, and the full set of encoded patches and mask tokens is processed by a small decoder that reconstructs the original image in pixels. After pre-training, the decoder is discarded and the encoder is applied to uncorrupted images (full sets of patches) for recognition tasks.

[He et al., 2021: Masked Autoencoders Are Scalable Vision Learners]

# SSL from Images, EX (II): relative positioning

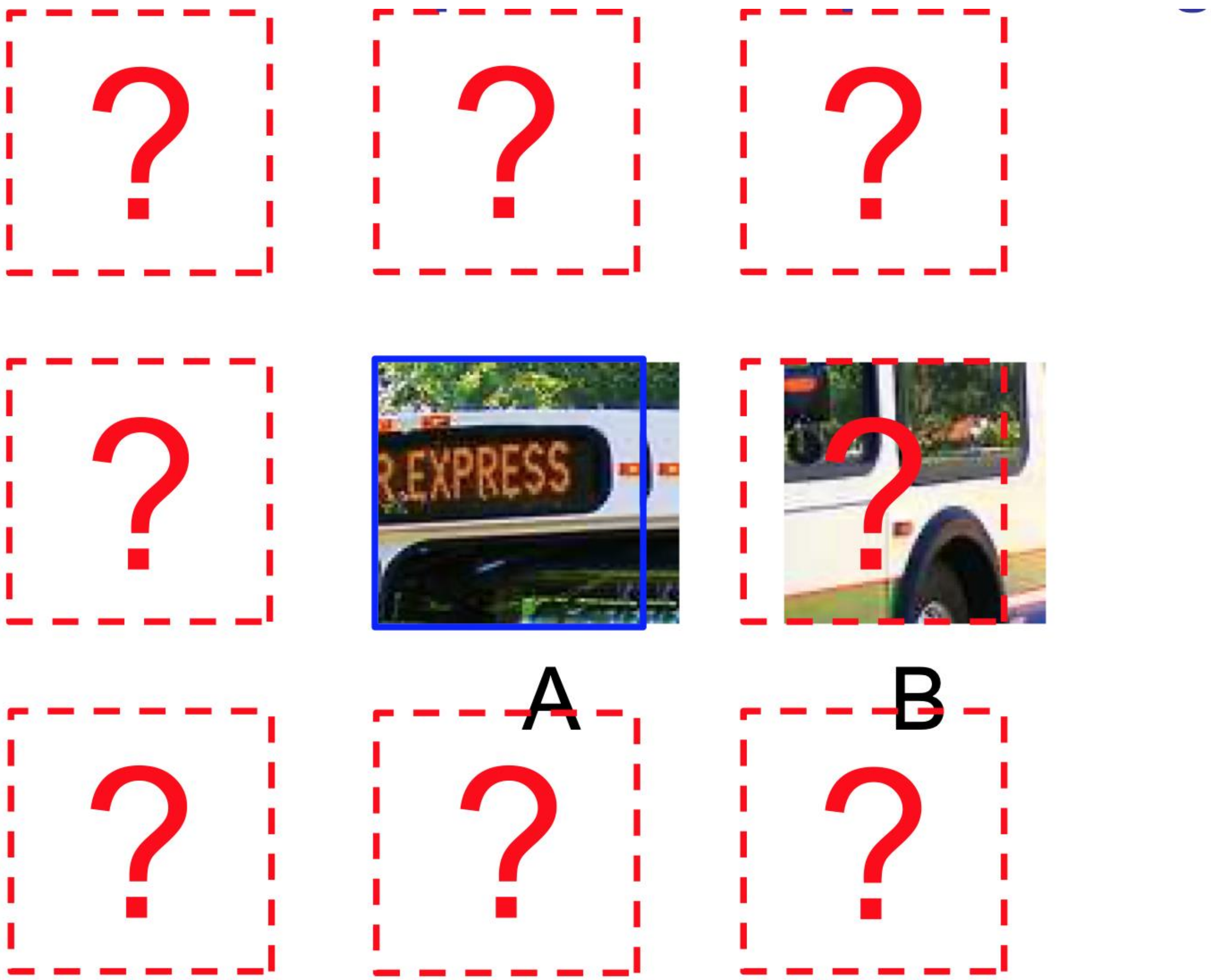
Train network to predict relative position of two regions in the same image




Unsupervised visual representation learning by context prediction, Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

[Courtesy: Zisserman "Self-supervised Learning"]

# SSL from Images, EX (II): relative positioning




Unsupervised visual representation learning by context prediction,
Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

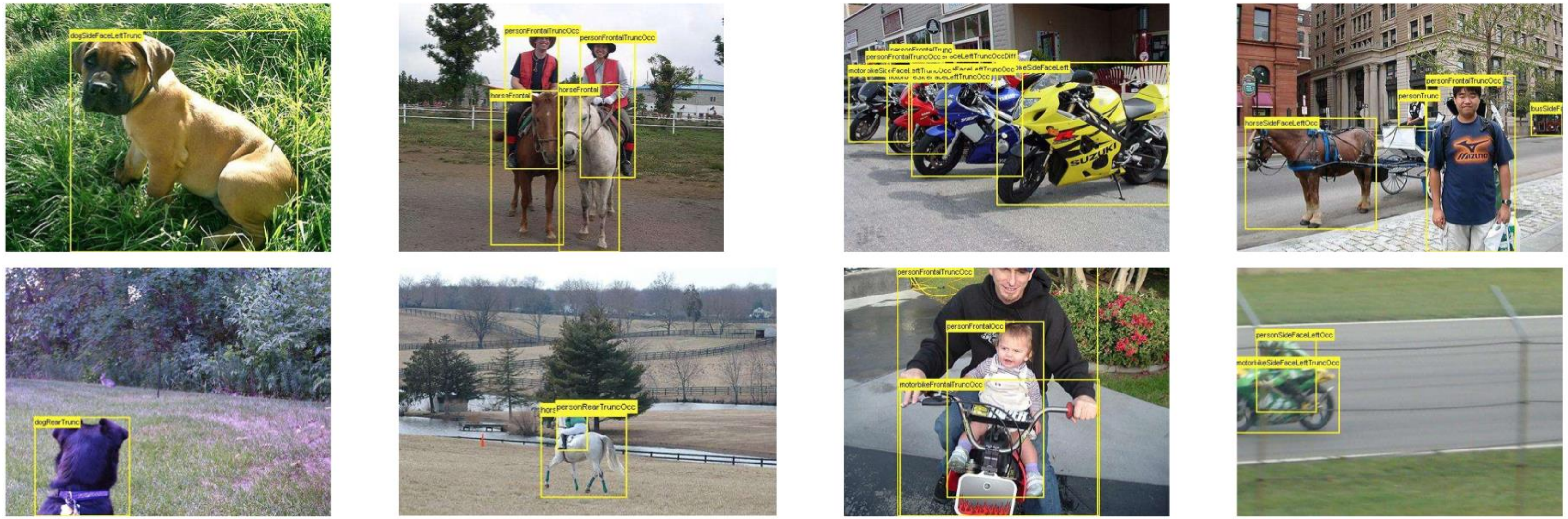
[Courtesy: Zisserman "Self-supervised Learning"]

# SSL from Images, EX (II): relative positioning

## Evaluation: PASCAL VOC Detection

- 20 object classes (car, bicycle, person, horse …)
- Predict the bounding boxes of all objects of a given class in an image (if any)

Dog Horse Motorbike Person



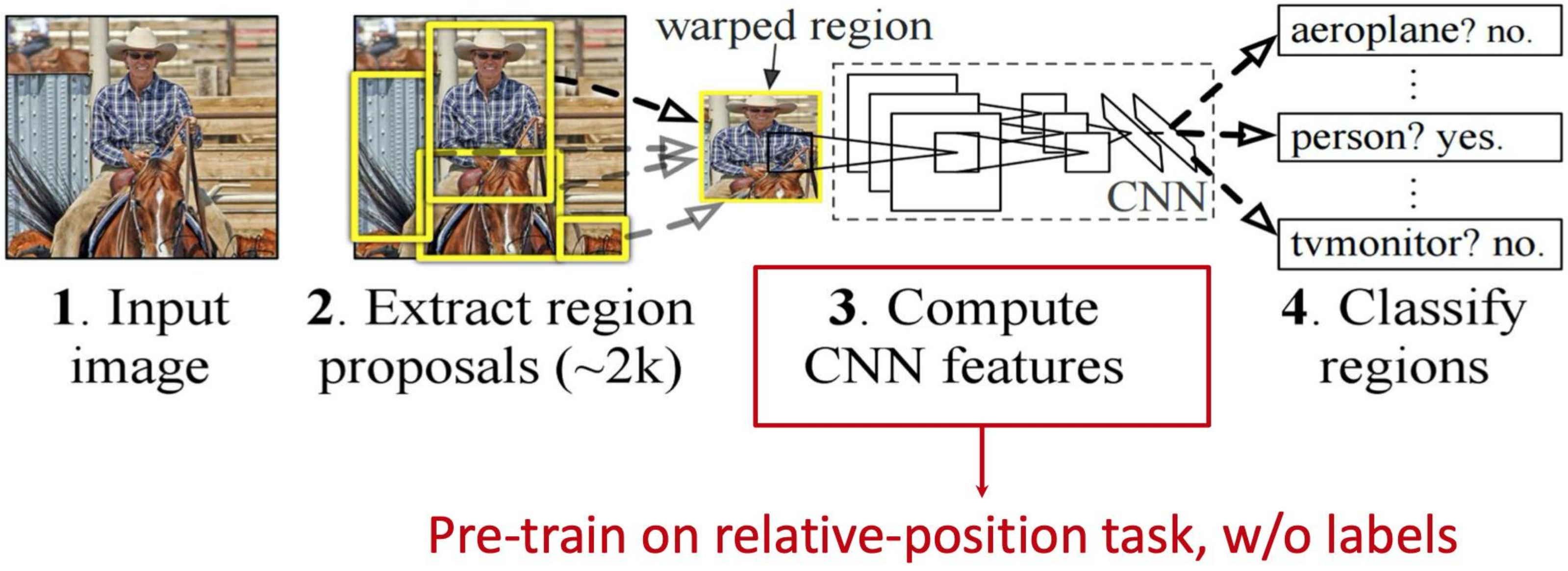
[Courtesy: Zisserman “Self-supervised Learning”]

# SSL from Images, EX (II): relative positioning

## Evaluation: PASCAL VOC Detection

- Pre-train CNN using self-supervision (no labels)
- Train CNN for detection in R-CNN object category detection pipeline

R-CNN

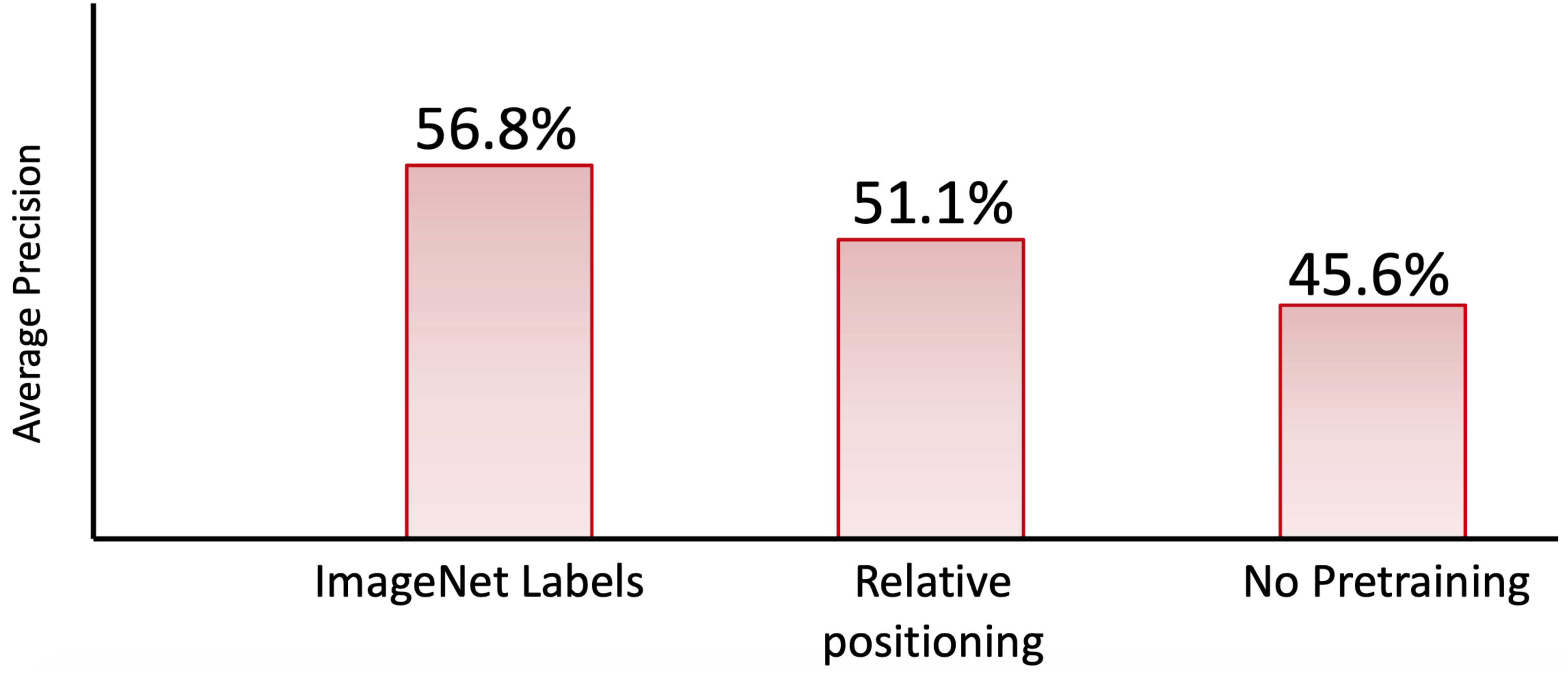



[Girshick et al. 2014]

[Courtesy: Zisserman “Self-supervised Learning”]

# SSL from Images, EX (II): relative positioning

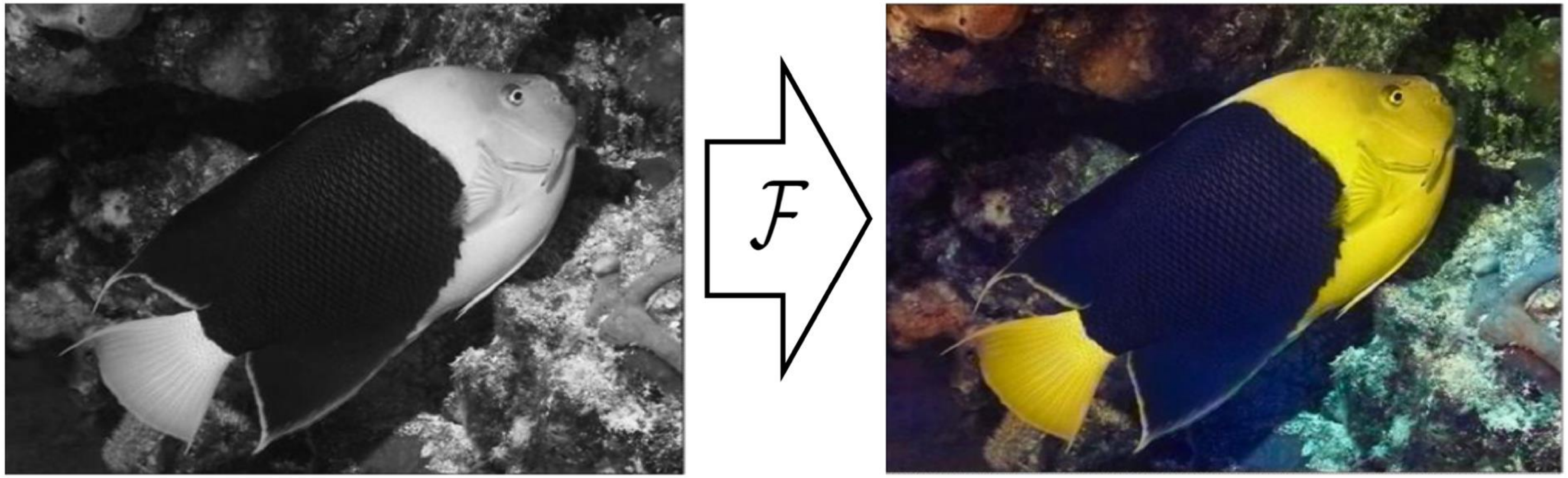
## Evaluation: PASCAL VOC Detection




[Courtesy: Zisserman "Self-supervised Learning"]

# SSL from Images, EX (III): colorization

Train network to predict pixel colour from a monochrome input

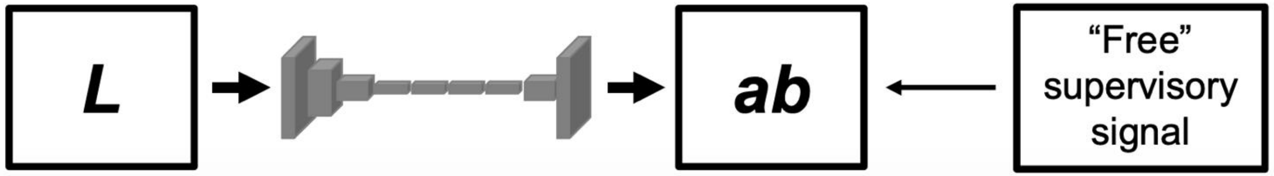



Grayscale image: *L* channel

$$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$$

Concatenate (L,ab)

$$(\mathbf{X}, \widehat{\mathbf{Y}})$$

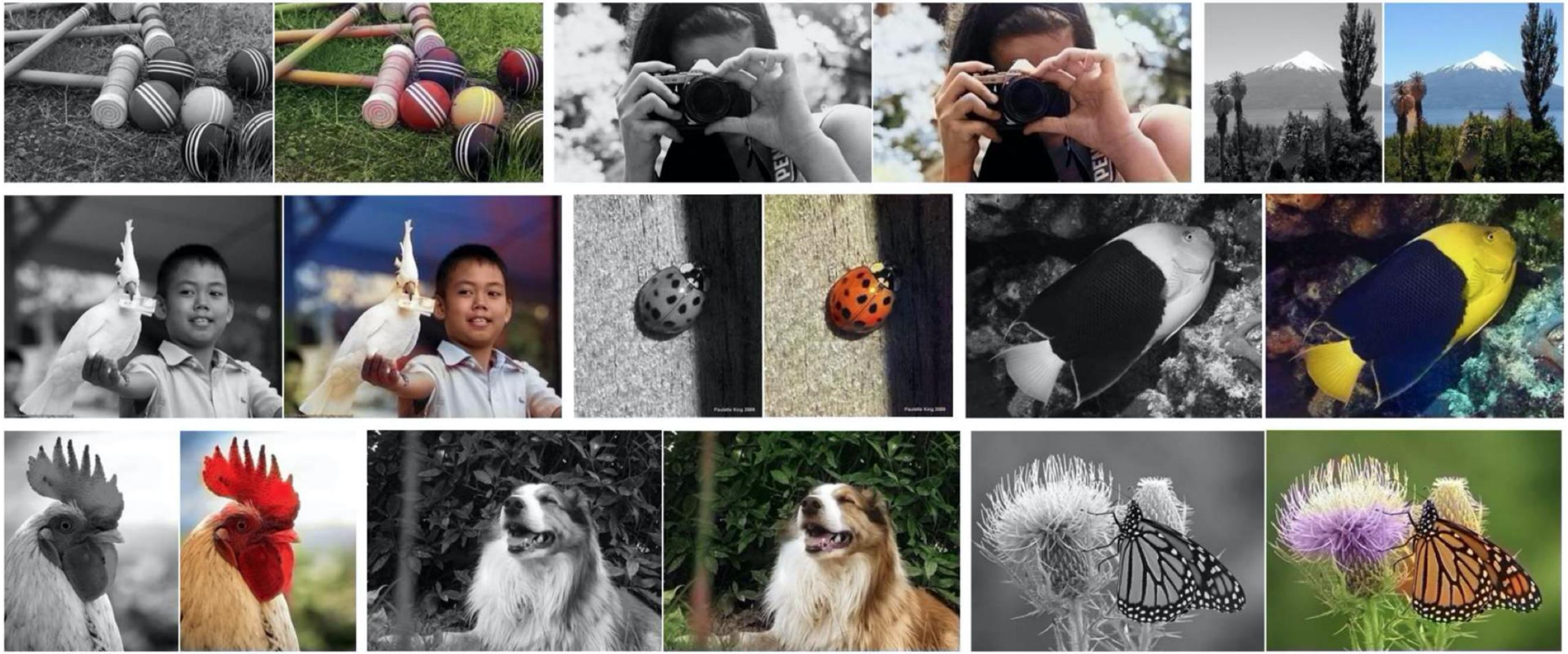



[Courtesy: Zisserman “Self-supervised Learning”]

Colorful Image Colorization, Zhang et al., ECCV 2016

# SSL from Images, EX (III): colorization

Train network to predict pixel colour from a monochrome input



[Courtesy: Zisserman "Self-supervised Learning"]

Colorful Image Colorization, Zhang et al., ECCV 2016

## SSL from Images, EX (IV): exemplar networks

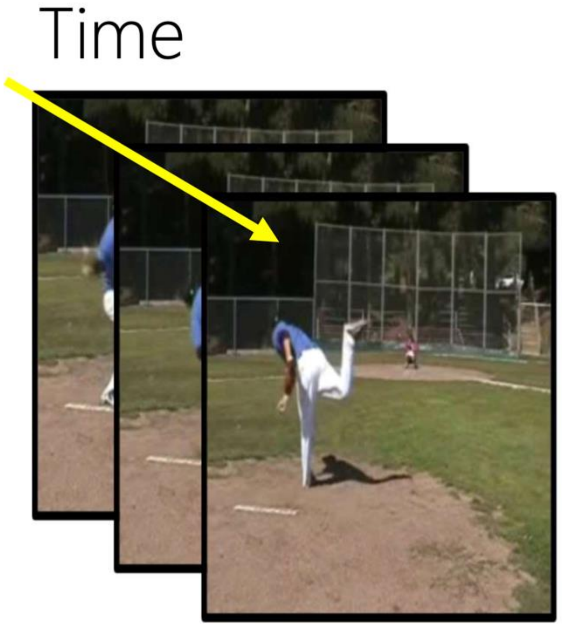
- Exemplar Networks (Dosovitskiy et al., 2014)
- Perturb/distort image patches, e.g. by cropping and affine transformations
- Train to classify these exemplars as same class

[Courtesy: Zisserman "Self-supervised Learning"]

# SSL from Videos

Three example tasks:

- Video sequence order
  - Sequential Verification: Is this a valid sequence?




"Sequence" of data

[Courtesy: Zisserman "Self-supervised Learning"]

Wei et al., 2018 Arrow of Time

# SSL from Videos

Three example tasks:

- Video sequence order
    - Sequential Verification: Is this a valid sequence?
- Video direction
    - Predict if video playing forwards or backwards

[Courtesy: Zisserman “Self-supervised Learning”]

Wei et al., 2018 Arrow of Time

# SSL from Videos

Three example tasks:

- Video sequence order
  - Sequential Verification: Is this a valid sequence?
- Video direction
  - Predict if video playing forwards or backwards
- Video tracking
  - Given a color video, colorize all frames of a gray scale version using a reference frame

[Courtesy: Zisserman “Self-supervised Learning”]

Vondrick et al., 2018

# Key Takeaways

- Self supervision learning
  - Predicting any part of the observations given any available information
  - The prediction task forces models to learn semantic representations
  - Massive/unlimited data supervisions
- SSL for text:
  - Language models: next word prediction
  - BERT text representations: masked language model (MLM)
- SSL for images/videos:
  - Various ways of defining the prediction task

# Enhancing LLM Training

## Limitation I:

# LLMs Lack World and Agent Knowledge

As we discussed before:

Emily found a desk and placed the cell phone on top of it. *[Irrelevant Actions]*, … putting the lime down next to the cell phone. *[Irrelevant Actions]* She finally put an apple on the desk. How many items are there on the desk?

GPT4

There are two items.

*(correct answer: three)*

Does this person need help?

GPT-4V

… I can't determine the actual need for help …

## Limitation I:

# LLMs Lack World and Agent Knowledge

As we discussed before:

**Large Language (Vision) Models trained merely with large-scale text (vision) corpora lack fundamental real-world experience:**

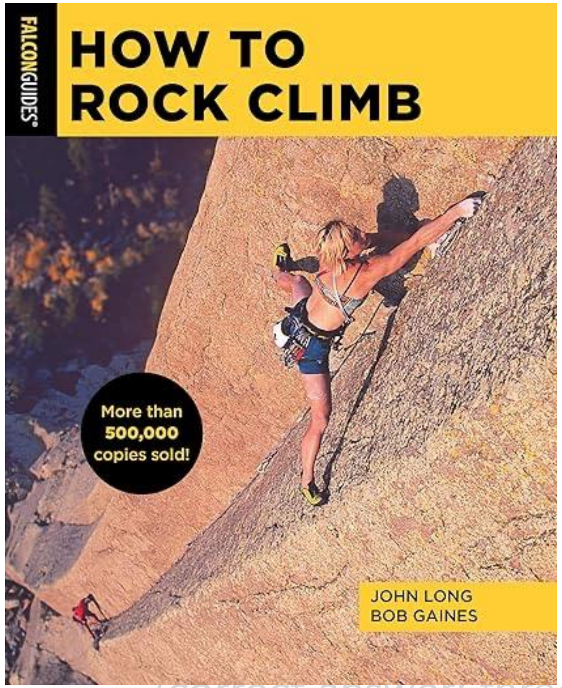
- tracking and interacting with objects
- understanding real-world physics and spatiotemporal relationships
- sensing and tracking the world states
- recognizing other agents' behaviors

## Limitation I:

# LLMs Lack World and Agent Knowledge

As we discussed before:

**Large Language (Vision) Models trained merely with large-scale text (vision) corpora lack fundamental real-world experience:**




?

nships

(correct answer: three)

Limitation I:

# LLMs Lack World and Agent Knowledge

As we discussed before:

**Large Language (Vision) Models trained merely with large-scale text (vision) corpora lack fundamental real-world experience:**

Need richer learning mechanisms!

- Embodied experiences
- Social learning

ships


(correct answer: three)

## Limitation II:

# Inefficiency of the language modality

- Language is often not the most efficient medium to describe all information during reasoning
- Other modalities (e.g., images/videos) can be more efficient

## Limitation II:

# Inefficiency of the language modality

- Language is often not the most efficient medium to describe all

ages/


In auto-driving: describe the street scene

- Vehicles' locations & movements


Pour liquid into a glass without spilling

- Viscosity & volume of the fluid
- shape & position of the container

# Limitation II:

## Inefficiency of the language modality

- Language is often not the most efficient medium to describe all information during reasoning
- Other modalities (e.g., images/videos) can be more efficient

Need **multi-modal** capabilities for world and agent modeling!

In auto-driving: describe street scene

- Vehicles' locations & movements

Pour liquid into a glass without spilling

- Viscosity & volume of the fluid
- shape & position of the container

# Outline: Enhancing the Backend Beyond LMs

- Richer learning mechanisms
  - Learning with Embodied Experiences
  - Social Learning
- Multi-modal capabilities
- Latent-space reasoning
- Agent models with external augmentations (e.g., tools)

# Outline: Enhancing the Backend Beyond LMs

- Richer learning mechanisms
  - **Learning with Embodied Experiences**
    - Where to get experiences
    - How to get experiences
    - How to learn with the experiences
  - Social Learning

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

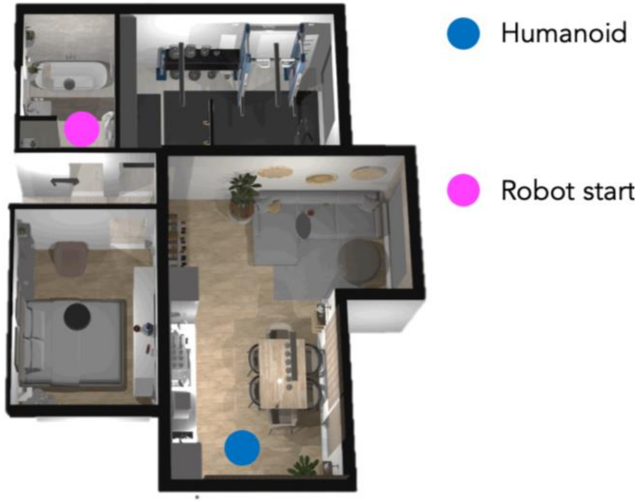
# Learning from Embodied Experiences

- Embodied simulators

Everyday household activities

Virtual Home

Habitat 3.0



# Learning from Embodied Experiences

- Embodied simulators

### Touchdown

navigating in urban scenes

*Orient yourself so that the umbrellas are to the right. Go straight and take a right at the first intersection. At the next intersection there should be an old-fashioned store to the left. There is also a dinosaur mural to the right. Touchdown is on the back of the dinosaur.*

### Minecraft

exploring a 3D infinite world and conducting rich tasks

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Embodied simulators

### Touchdown

navigating in urban scenes

*Orient yourself so that the umbrellas are to the right. Go straight and take a right at the first intersection. At the next intersection there should be an old-fashioned store to the left. There is also a dinosaur mural to the right. Touchdown is on the back of the dinosaur.*

### Minecraft

exploring a 3D infinite world and conducting rich tasks

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Embodied simulators

## Touchdown

navigating in urban scenes

*Orient yourself so that the umbrellas are to the right. Go straight and take a right at the first intersection. At the next intersection there should be an old-fashioned store to the left. There is also a dinosaur mural to the right. Touchdown is on the back of the dinosaur.*

## Minecraft

exploring a 3D infinite world and conducting rich tasks


[Wang et al., 2023]

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Embodied simulators

## Touchdown

navigating in urban scenes

*Orient yourself so that the umbrellas are to the right. Go straight and take a right at the first intersection. At the next intersection there should be an old-fashioned store to the left. There is also a dinosaur mural to the right. Touchdown is on the back of the dinosaur.*

## Minecraft

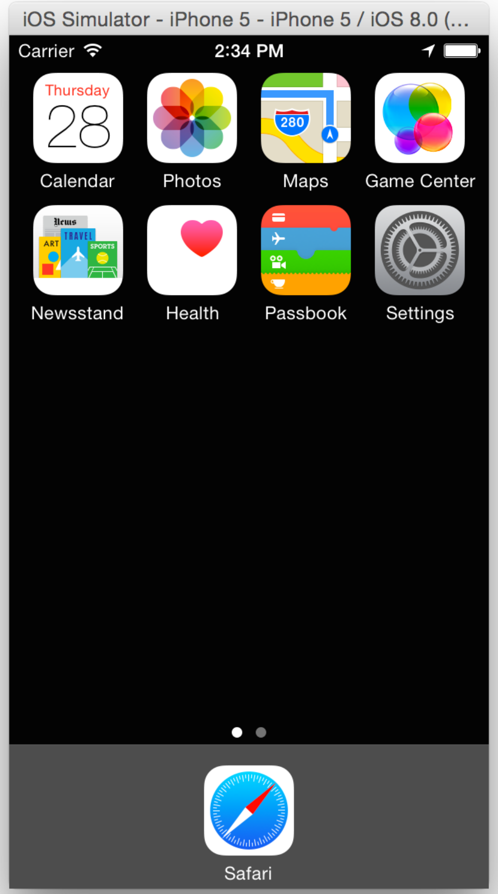
exploring a 3D infinite world and conducting rich tasks


[Wang et al., 2023]

# Learning from Embodied Experiences

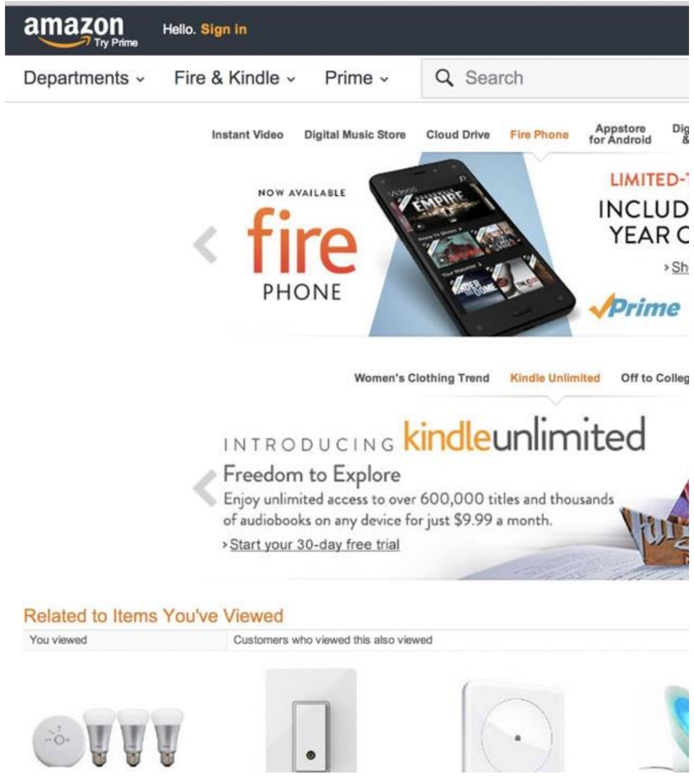
- Other simulators

## OS




## Simulated websites

(shopping, navigating, search)



(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

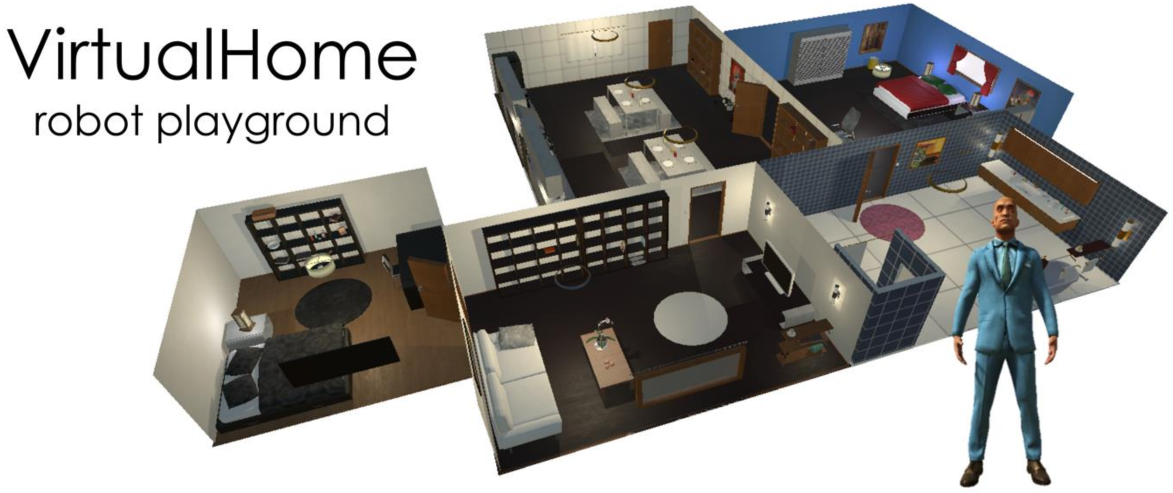
# Learning from Embodied Experiences

- Goal-oriented
  - Collecting experiences by completing a given task

Goal: Work on computer
Description: Turn on your computer and sit in front of it. Type on the keyboard, grab the mouse to scroll.

Goal: Make coffee
Description: Go to the kitchen and swith on the coffee machine. Wait until it’s done and pour the coffee into a cup.

Goal: Read a book
Description: Sit down in recliner. Pick up a novel off of coffee table. Open novel to last read page. Read.




[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

# Learning from Embodied Experiences

- Goal-oriented
  - Collecting experiences by completing a given task

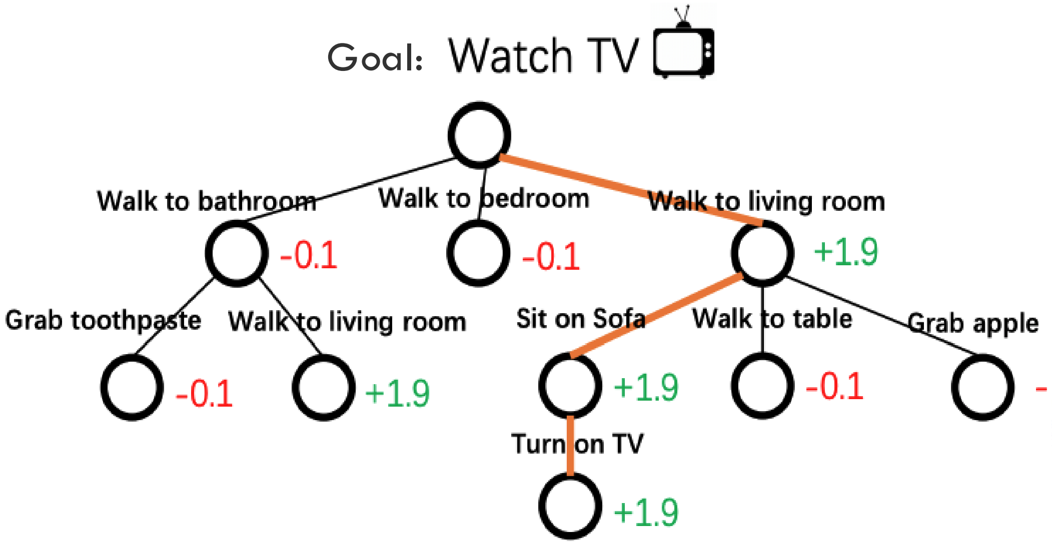

[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Goal-oriented
  - Collecting experiences by completing a given task

Goal-Oriented Planning




Monte Carlo Tree Search (MCTS)

[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

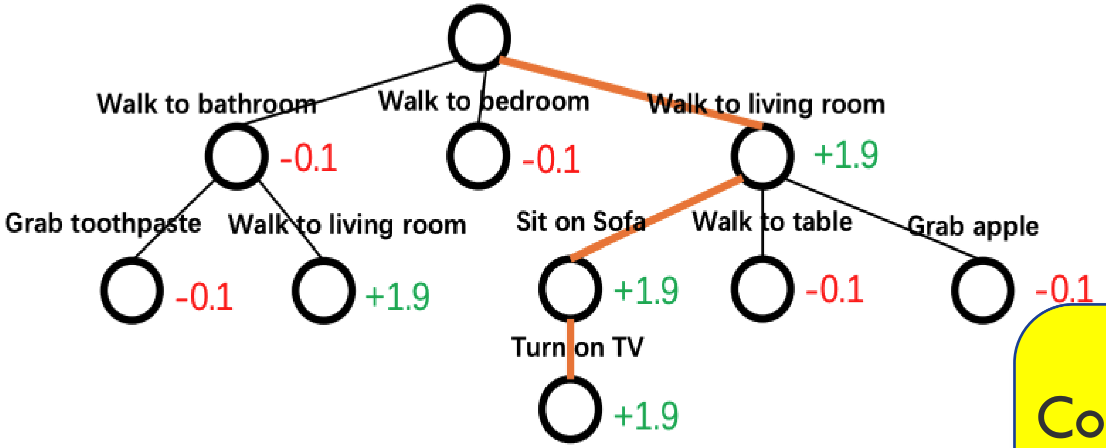
(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Goal-oriented
  - Collecting experiences by completing a given task

Goal-Oriented Planning

Goal: Watch TV




Monte Carlo Tree Search (MCTS)

Convert experiences into training data (question answering)

Question:
How to watch TV? TV and sofa is in living room…

**Answer:**
**Walk to living room. Sit on sofa. Turn on TV.**

Plan Generation

Question:
Given a plan: Walk to living room. Sit on sofa. Turn on TV. What is the task?

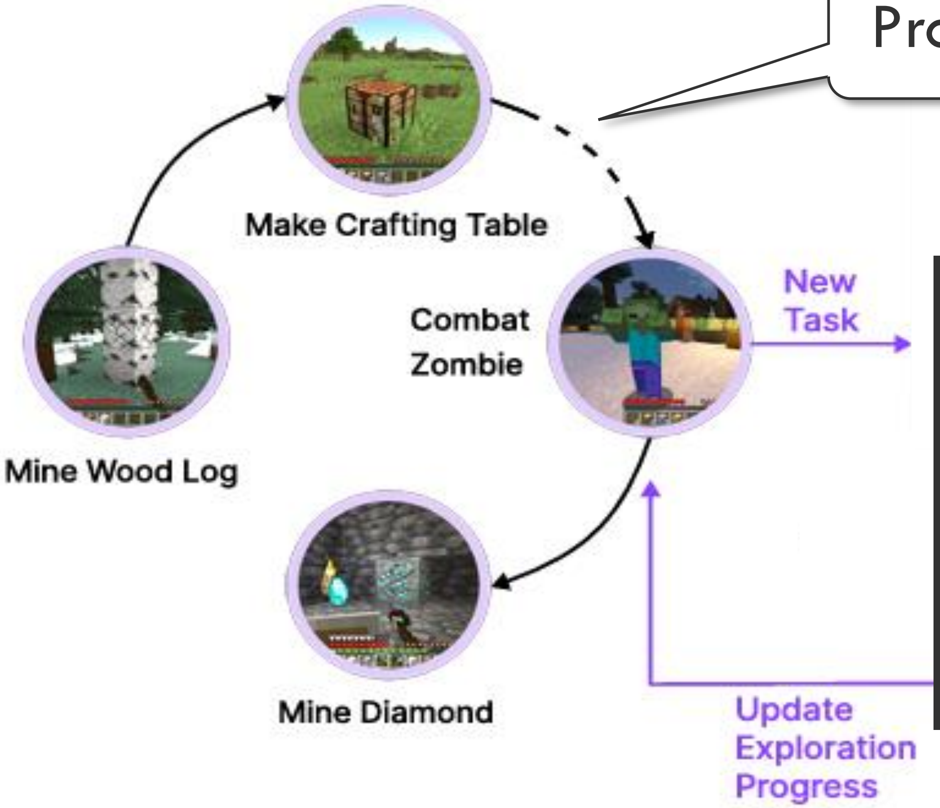
**Answer: Watch TV.**

Activity Recognition

# Learning from Embodied Experiences

- Auto curriculum
  - Proposing new tasks automatically

Prompt GPT-4 to generate new tasks




- Collect experiences by completing the task
- Learn with the experiences

[Wang et al., 2023. Voyager: An Open-Ended Embodied Agent with Large Language Models]

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Auto curriculum
  - Proposing new tasks automatically

Prompt GPT-4 to generate new tasks

Examples:

**Inventory (5/36)**: {'oak_planks': 3, 'stick': 4, 'crafting_table': 1, 'stone': 3, 'wooden_pickaxe': 1}

→ GPT-4 →

**Reasoning**: Since you have a wooden pickaxe and some stones, it would be beneficial to upgrade your pickaxe to a stone pickaxe for better efficiency.
**Task**: Craft 1 stone pickaxe.

**Biome**: river
**Inventory (4/36)**: {'sugar_cane': 3, 'paper': 3, 'fishing_rod': 1, 'wooden_pickaxe': 1}

→ GPT-4 →

**Reasoning**: Since you have a fishing rod in your inventory and you are near a river biome, it would be a good opportunity to catch some fish for food and experience.
**Task**: Catch 1 fish.

**Nearby entities**: pig, cat, villager
**Health**: 12/20
**Hunger**: 0/20

→ GPT-4 →

**Reasoning**: Your hunger is at 0, which means you need to find food to replenish your hunger. Since there are pigs nearby, you can kill one to obtain raw porkchops.
**Task**: Kill 1 pig.

[Wang et al., 2023. Voyager: An Open-Ended Embodied Agent with Large Language Models]

# Learning from Embodied Experiences

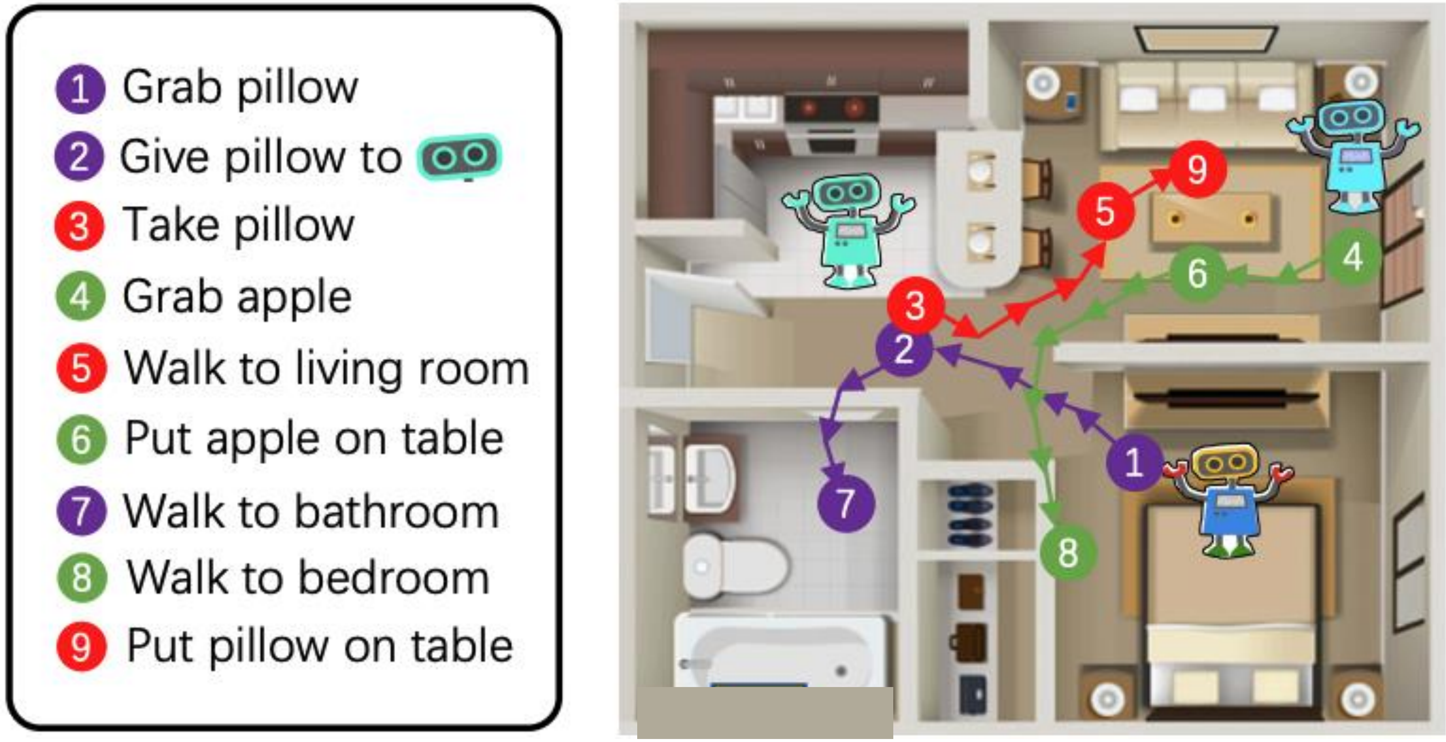
- Random Exploration

Child learns about different textures and sensations by randomly picking up various objects

# Learning from Embodied Experiences

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

- Random Exploration



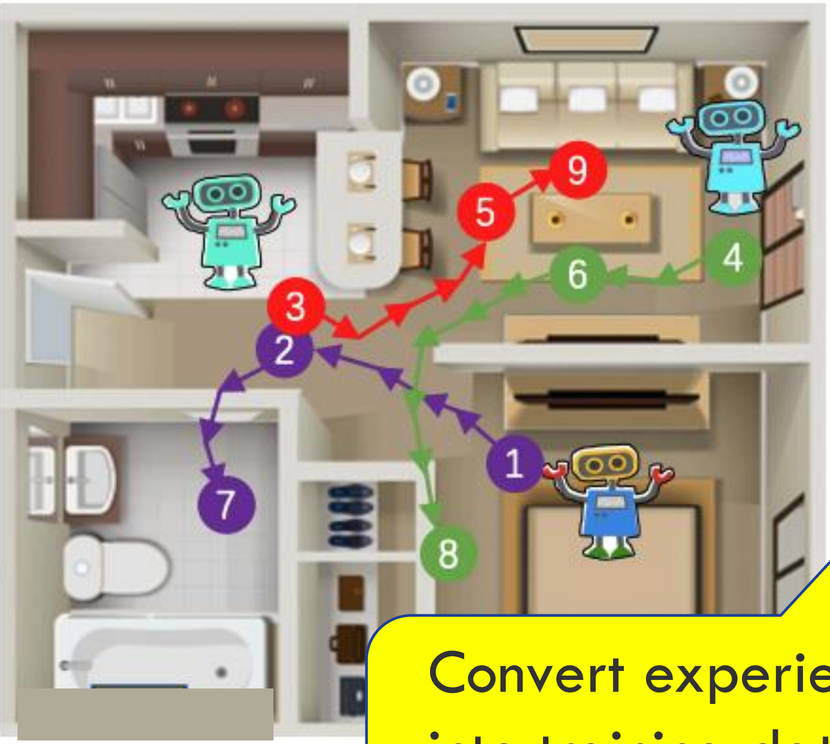

[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

# Learning from Embodied Experiences

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

- Random Exploration

1. Grab pillow
2. Give pillow to
3. Take pillow
4. Grab apple
5. Walk to living room
6. Put apple on table
7. Walk to bathroom
8. Walk to bedroom
9. Put pillow on table



Convert experiences into training data (question answering)

Question:
Tom grabbed pillow. Tom gave pillow to ··· How many objects are on the table?

**Answer:**
**Two. They are pillow and apple.**

Counting

Question:
Tom grabbed pillow. Tom walked to kitchen ··· What is the order of rooms where pillow appears?

**Answer:**
**Bedroom, kitchen, living room**

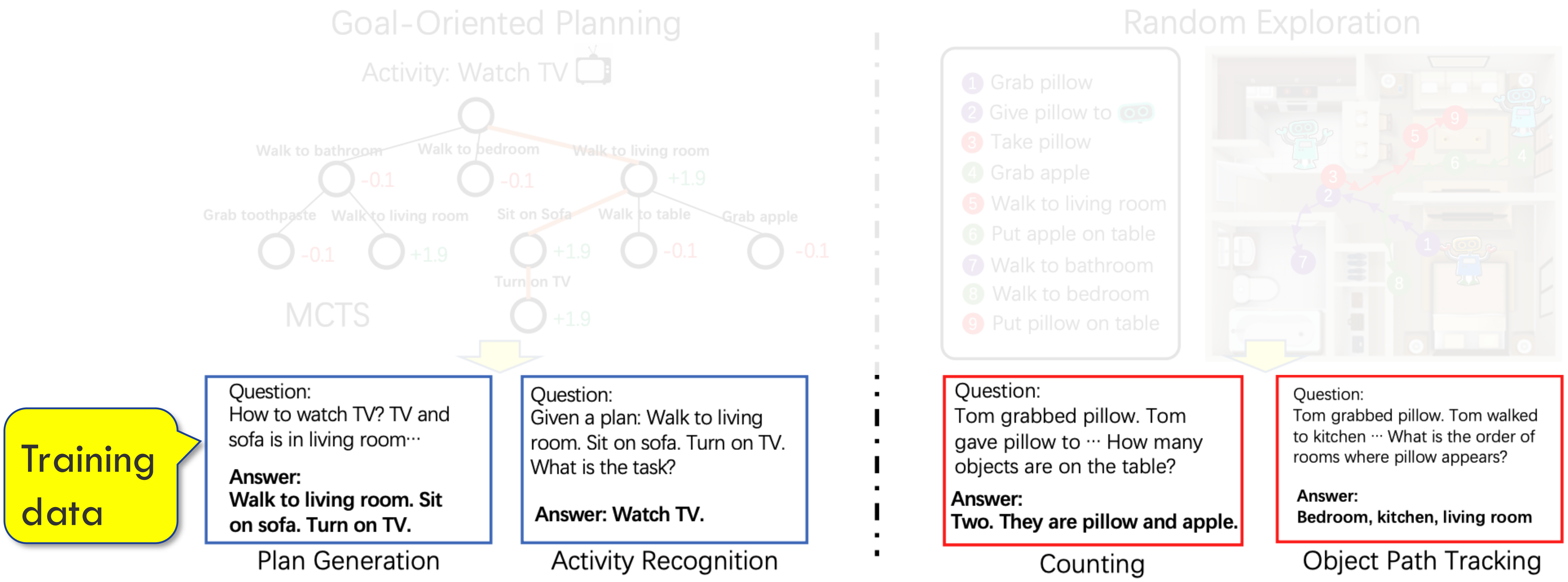
Object Path Tracking

[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

# Learning from Embodied Experiences

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

- Finetuning LMs with the experiences

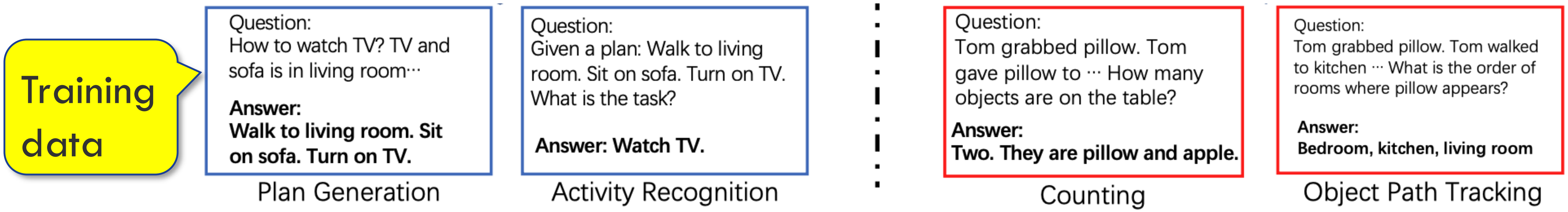



[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Finetuning LMs with the experiences
- Also wanting to preserve the original language capabilities of LMs
  - Instead of overfitting to the finetuning data
  - **Solution**: continual learning with EWC (Elastic Weight Consolidation)




[Kirkpatrick et al., 2017. Overcoming catastrophic forgetting in neural networks]

[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Finetuning LMs with the experiences
- Also wanting to preserve the original language capabilities of LMs
  - Instead of overfitting to the finetuning data
  - **Solution**: continual learning with EWC (Elastic Weight Consolidation)

$$F_{i,i} = \frac{1}{N} \sum_{j=1}^{N} \left( \frac{\partial \mathcal{L}_U^{(j)}}{\partial \theta_{U,i}^*} \right)^2$$

Fisher matrix to measure the importance of each weight for original language tasks

$$\mathcal{L}(\theta) = \mathcal{L}_V(\theta) + \lambda \sum_i F_{i,i} (\theta_i - \theta_{U,i}^*)^2$$

[Kirkpatrick et al., 2017. Overcoming catastrophic forgetting in neural networks]

[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

# Learning from Embodied Experiences

- Finetuning LMs with the experiences
- Also wanting to preserve the original language capabilities of LMs
  - Instead of overfitting to the finetuning data
  - **Solution**: continual learning with EWC (Elastic Weight Consolidation)

Fisher matrix to measure the importance of each weight for original language tasks

$$F_{i,i} = \frac{1}{N}\sum_{j=1}^{N}\left(\frac{\partial \mathcal{L}_U^{(j)}}{\partial \theta^*_{U,i}}\right)^2$$

Conventional finetuning objective

$$\mathcal{L}(\theta) = \mathcal{L}_V(\theta) + \lambda \sum_i F_{i,i}(\theta_i - \theta^*_{U,i})^2$$

Regularizor to preserve important weights

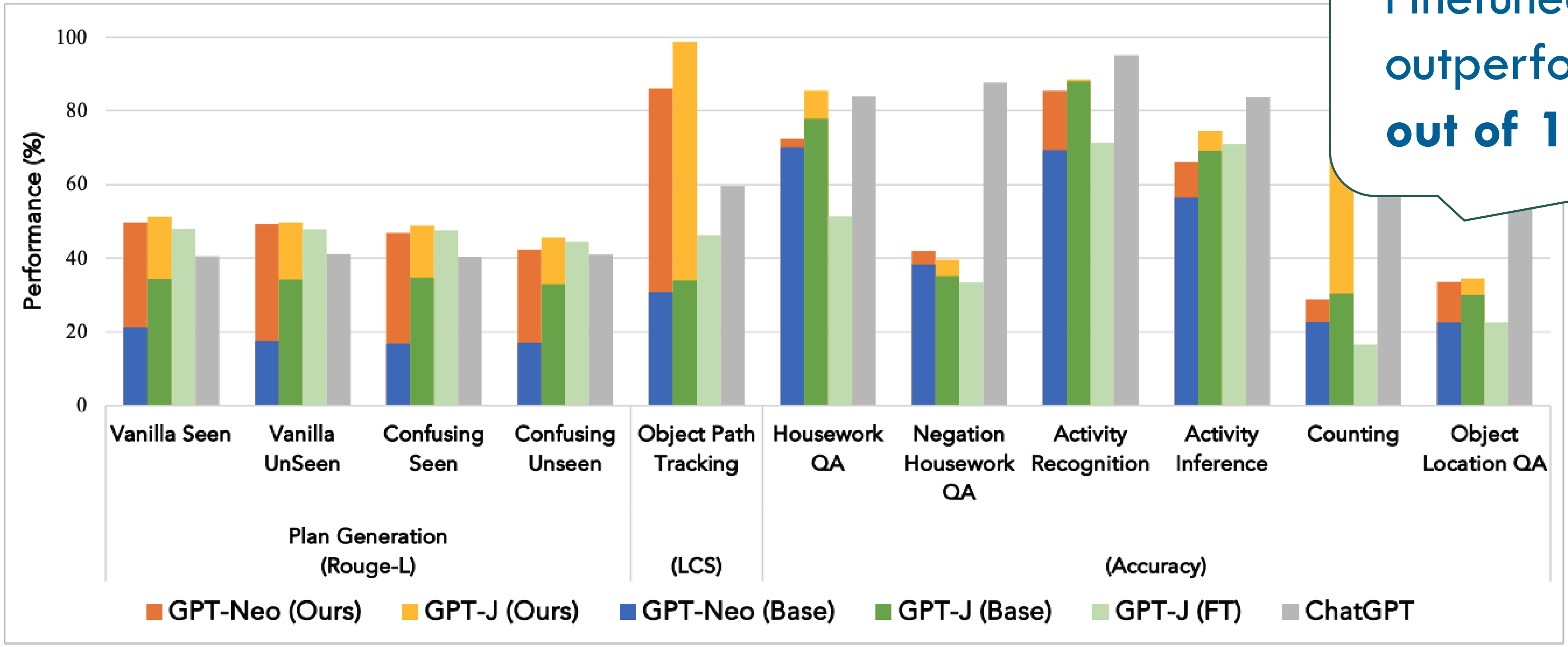
[Kirkpatrick et al., 2017. Overcoming catastrophic forgetting in neural networks]

[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Finetuning LMs with the experiences




Finetuned GPT-J-6B outperforms ChatGPT on **7 out of 11** tasks

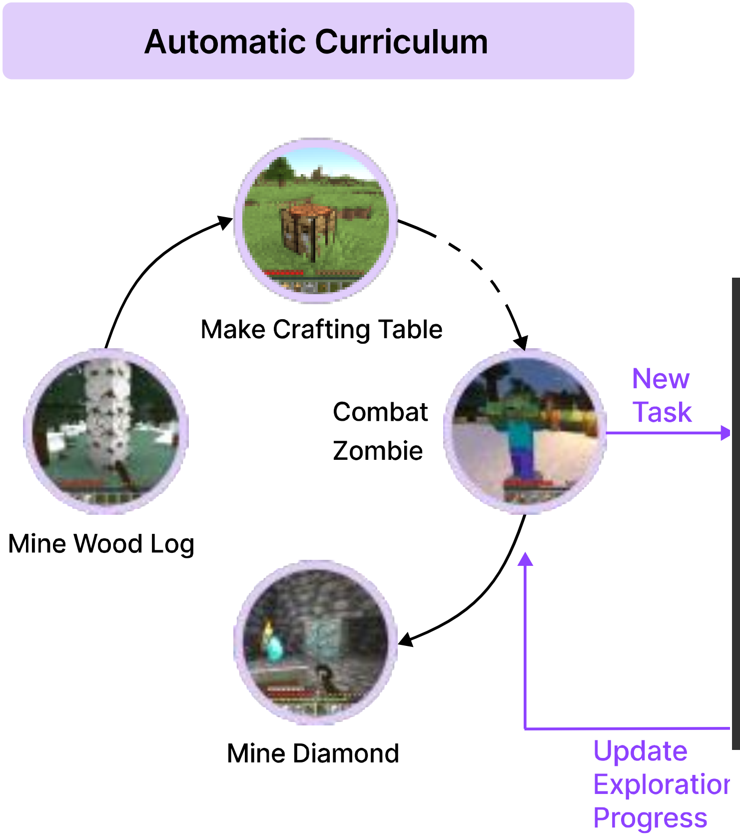

[Kirkpatrick et al., 2017. Overcoming catastrophic forgetting in neural networks]

[Xiang et al., 2023. Language Models Meet World Models: Embodied Experiences Enhance Language Models]

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

# Learning from Embodied Experiences

- Updating external memory
  - Instead of changing LM parameters




- Collect experiences by completing the task
- Learn with the experiences

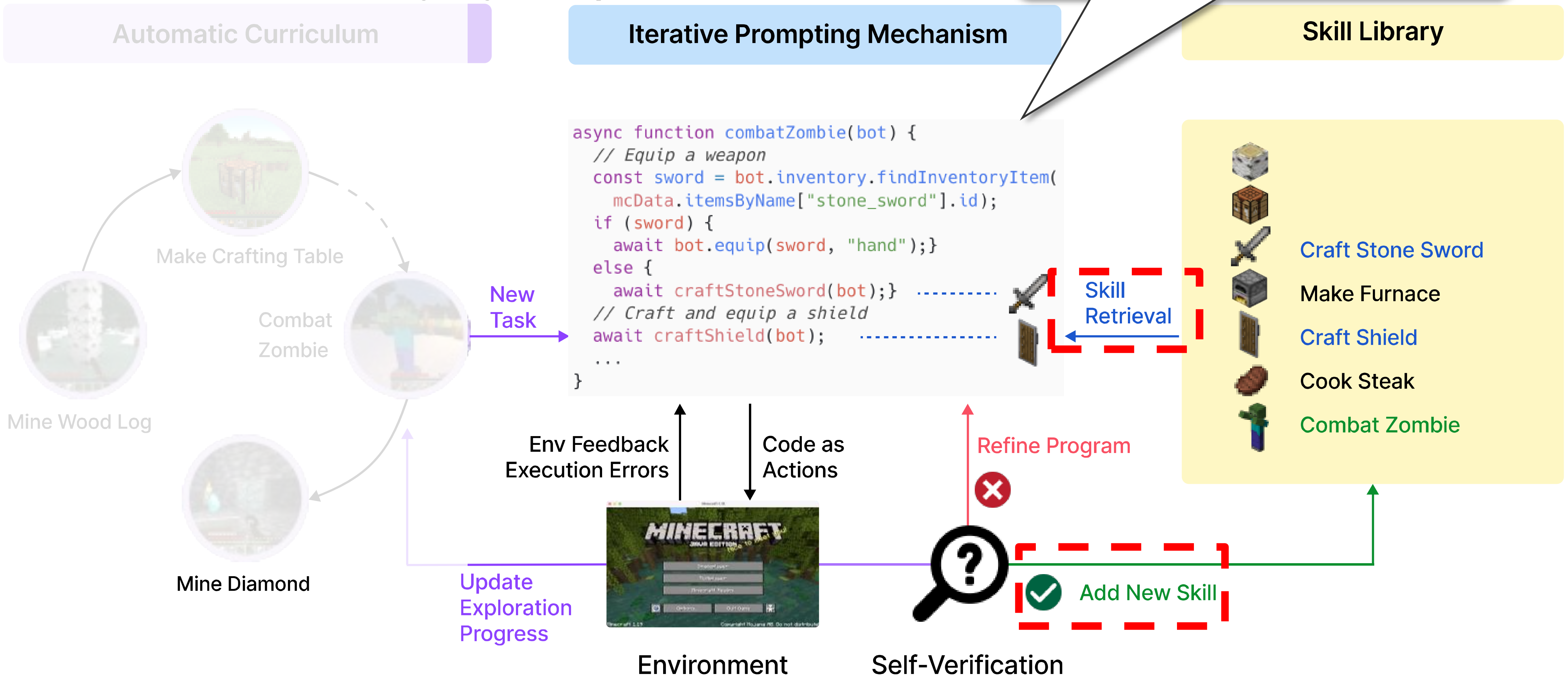
[Wang et al., 2023. Voyager: An Open-Ended Embodied Agent with Large Language Models]

# Learning from Embodied Experiences

(1) Where to get experiences
(2) How to get experiences
(3) How to learn w/ experiences

- Updating external memory
  - Instead of changing LM parameters

Skill represented as code




[V

## Summary: Learning with Embodied Experiences

- Where to get experiences
  - Simulators (embodied env., OS, simulated websites, …)
- How to get experiences
  - Goal-oriented planning
  - Auto-curriculum
  - Random exploration
- How to learn with the experiences
  - Finetuning LMs while preserving original language capabilities: continual learning
  - Updating external memory

# Questions?