Generalist Embodied AI in an Open World

Xiaojian Ma
Machine Learning @ BIGAI
11/24/2023
ML is stepping into a new era

- More data, $O(10k) \rightarrow O(10M) \rightarrow O(1T)$;
- More parameters, $O(1M) \rightarrow O(1b) \rightarrow O(100b)$;
- More computation, GFLOPS $\rightarrow$ TFLOPS
ML is stepping into a new era

- Complex domains and semantics
- Close world (vocabulary) -> open world (vocabulary)
...and so is embodied AI

ScanNet-200: Language-Grounded Indoor 3D Semantic Segmentation in the Wild
...and so is embodied AI

OpenScene: 3D Scene Understanding with Open Vocabularies
...and so is embodied AI

CoWs on Pasture:
Baselines and Benchmarks for Language-Driven Zero-Shot Object Navigation

Top-down visualization

Task: Find the gingerbread house
A paradigm shift for embodied AI

contrived; limited tasks; static; close world…

realistic; massive tasks; dynamic open world…

NeurIPS 2023 HomeRobot: Open Vocabulary Mobile Manipulation (OVMM) Challenge
Embodied AI
Generalist Embodied AI in an Open World
LEO: An Embodied Generalist Agent in 3D World

CraftJarvis: Multi-task Embodied Agents in an Open World

stay tuned…
LEO: An Embodied Generalist Agent in 3D World

CraftJarvis: Multi-task Embodied Agents in an Open World

stay tuned…
LEO: An Embodied Generalist Agent

An Embodied Generalist Agent in 3D World, arXiv preprint 2023

3D-VisTA: Pre-trained Transformer for 3D Vision and Text Alignment, ICCV 2023

SQA3D: Situated Question Answering in 3D Scenes, ICLR 2023

embodied-generalist.github.io
Embodyed Generalist Agent

Capabilities: *Perception*, *Grounding*, *Reasoning*, *Planning*, *Acting*

Tasks

- 3D Object Captioning
- Scene Captioning
- 3D Question Answering
- Embodied Reasoning
- 3D Dialogue
- Task Planning
- Embodied Navigation
- Robotic Manipulation

3D World
Large Language Model

1. Choose a room…
2. Plan the layout…
3. Create a workspace…

Without scene awareness: ambiguous, hallucination
You are an assistant situated in a 3D scene. There is a white table, wooden ...
1. Place the desk in the desired position in the room…
2. Position the chair next to the desk, to the right of it.
3. Set up the shelf to the left of the desk…
4. Place the lamp on the desk…
5. Arrange the showcase to the right of the desk.
6. Place the plants on the shelf…
7. Hang the curtains on the wall behind the desk…
Scene representation

2D branch

2D Encoder

d

Ego-centric features

h

w

3D branch

3D Encoder

Object-centric point clouds

...
Unified task sequence

Auto-regressive objective

\[ L(\theta, B) = - \sum_{b=1}^{\vert B \vert} \sum_{t=1}^{T} \log p(s_{\text{res}}^{(b,t)} \mid s_{\text{res}}^{(b, t < t)}, s_{\text{prefix}}^{(b, 1)}, \ldots, s_{\text{prefix}}^{(b, L)}) \]
An Embodied Generalist Agent in 3D World
An Embodied Generalist Agent in 3D World
Machine Learning @ BIGAI

**Object Captioning**

**Object Referring**

**Scene Captioning**

**3D Captioning**

**3D QA**

**3D Dialogue**

**Manipulation**

**Navigation**

**Task Planning**

**Visual domains**

**Objaverse**

**ScanNet**

**3RScan**

**Matterport3D**

**CLIPort**
Scene-Graph-based Prompting

Messages

1. System Message
   You are an AI visual assistant in a 3D scene...

2. Demonstrations
   Scene Graph Context:
   {'sofa-1': {'attributes': {'color': 'red'}, 'relations': ['to the right of chair-2', 'in front of table-3']}, 'chair-2':
   {'attributes': {'color': 'brown'}, 'relations': []}}
   Human-labeled Responses

3. Query (a new scene)
   Scene Graph Context

Responses

1) Object scene caption
2) Scene caption
3) Dialogue (O-CoT)
4) QA (O-CoT)
5) Planning

Potential Issues:
1) Wrong Answers
2) Negative Responses
3) Responses with IDs
4) Others

Refinement Procedures

Raw Responses

1. Wrong Answers
   There are two curtains in the room.
2. Negative Responses
   Question: Where is the football table?
   Thought: football table-17
   Answer: unknown
3. Responses with IDs
   The kitchen cabinet-9 and kitchen counter-10 are parts of the kitchen.
4. Others
   Unnecessary contents (Thought, etc.)

Refined Responses

1. Answer Refinement
   There are 3 curtains in the room.
2. Removing Negative Responses
   The negative responses will be removed.
3. GPT-based rewriting
   The kitchen features a cabinet and a counter.
4. Others
   Remove unnecessary contents.
Dialogue (O-CoT):
Dialogue Context: high level task: organize the bedroom. low level task: check some objects.
Human: I need help organizing my bedroom. Can you assist me? Thought: no specific objects
Assistant: Of course! What specifically do you need help with?
Human: Can you check if there are any clean clothes hanging on the wall? Thought: clothes-13
Assistant: I'm sorry, but the clothes hanging on the wall are messy.
Human: Thanks for your help! Thought: no specific object
Assistant: You're welcome! If you need any further assistance, feel free to ask.

Scene Caption:
In this room, there is a wooden floor that is clean and flat. A tall wardrobe stand on the right side of a desk, close to a basket. The wardrobe is in front of a chair, a bed, and behind a bedside table. Clothes are hanging on a white wall, to the right of a closed curtain. The bed is wide, clean, and covered with blue blanket. The room has a comfortable and organized setting with functional furniture.

Object Scene Caption:
The showcase is supported by the wall and positioned behind, close to, and to the left of the shelf. The wall is flat, architectural, and dark in color. The shelf is made of wood and has a low size.

QA (O-CoT):
Question: Where is the printer located? Thought: printer-8
Answer: standing on the desk
Question: How many blankets are on the bed? Thought: blanket-16, blanket-17
Answer: 2
Question: What is the type of the room? Thought: wardrobe-2, desk-7, chair-11, bed-15
Answer: bedroom

Planning:
High-Level Task: Organize and tidy up the bedroom.
Low-Level Actions:
1. Clean the floor by sweeping to remove any dirt.
2. Make the bed by arranging the blanket and pillows.
3. Place any loose items or belongings into the basket.
4. Arrange items on the shelves and showcase in a tidy way.
<table>
<thead>
<tr>
<th></th>
<th>Scan2Cap (val)</th>
<th>ScanQA (val)</th>
<th>SQA3D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>C</td>
<td>B-4</td>
<td>M</td>
</tr>
<tr>
<td>Task-specific models</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scan2Cap (GPT-3) (Chen et al., 2021)</td>
<td>35.2</td>
<td>22.4</td>
<td>21.4</td>
</tr>
<tr>
<td>3DJCG (Cai et al., 2022)</td>
<td>47.7</td>
<td>31.5</td>
<td>24.3</td>
</tr>
<tr>
<td>Vote2Cap-DETR (Chen et al., 2023)</td>
<td>61.8</td>
<td>34.5</td>
<td>26.2</td>
</tr>
<tr>
<td>ScanRefer+MCAN (Chen et al., 2020)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>ClipBERT (Lei et al., 2021)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>ScanQA (Azuma et al., 2022)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Task-specific fine-tuned</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3D-VisTA (Zhu et al., 2023c)</td>
<td>66.9</td>
<td>34.0</td>
<td>27.1</td>
</tr>
<tr>
<td>3D-LLM (FlanT5) (Hong et al., 2023)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>LEO</td>
<td>68.4</td>
<td>36.9</td>
<td>27.7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>separating-piles</th>
<th>packing-google</th>
<th>put-blocks-in</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>seen</td>
<td>unseen</td>
<td>seen</td>
</tr>
<tr>
<td>Transporter</td>
<td>48.4</td>
<td>52.3</td>
<td>46.3</td>
</tr>
<tr>
<td>CLIP-only</td>
<td>90.2</td>
<td>71.0</td>
<td>95.8</td>
</tr>
<tr>
<td>RN50-BERT</td>
<td>46.5</td>
<td>44.9</td>
<td>94.0</td>
</tr>
<tr>
<td>CLIPort (single)</td>
<td>98.0</td>
<td>75.2</td>
<td>96.2</td>
</tr>
<tr>
<td>CLIPort (multi)</td>
<td>89.0</td>
<td>62.8</td>
<td>84.4</td>
</tr>
<tr>
<td>LEO</td>
<td>98.8</td>
<td>75.2</td>
<td>76.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>MP3D-val</th>
<th>HM3D-val</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>S(†)</td>
<td>L(†)</td>
</tr>
<tr>
<td>ObjNav Navigation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H.w. (shortest)</td>
<td>4.4</td>
<td>2.2</td>
</tr>
<tr>
<td>H.w. (70k demo)</td>
<td>35.4</td>
<td>10.2</td>
</tr>
<tr>
<td>VC-1 (ViT-B)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>LEO</td>
<td>23.1</td>
<td>15.2</td>
</tr>
</tbody>
</table>
Related research

PaLM-E: a comprehensive generalist excelling in multi-modal reasoning and planning

RT-2: bridging the gap between vision, language and action
LEO: An Embodied Generalist Agent in 3D World

CraftJarvis: Multi-task Embodied Agents in an Open World

stay tuned…
CraftJarvis: Embodied Agents in an Open World

Open-World Multi-Task Control Through Goal-Aware Representation Learning and Adaptive Horizon Prediction, CVPR 2023

Describe, Explain, Plan and Select: Interactive Planning with Large Language Models Enables Open-World Multi-Task Agents, Best paper award, ICML ‘23 TEACH Workshop NeurIPS 2023

JARVIS-1: Open-world Multi-task Agents with Memory-Augmented Multimodal Language Models, arXiv 2023

craftjarvis-jarvis1.github.io
Minecraft: embodied AI in an open world

Today's embodied AI
- Restrictive objectives
- Very few tasks
- Limited knowledge

Embodied AI in an open world
- Open-ended objectives
- Massively multitask
- Web-scale knowledge
Challenges in open world environments

Challenge #1: long-horizon planning

Open worlds have highly abundant object types with complex dependency and relation. As a result, ground-truth plans typically involve a long sequence of sub-goals with strict dependencies.

=> Planning Success Rate will drops significantly on long-horizon tasks.
Challenges in open world environments

Challenge #2: state-aware planning

When dealing with a task that can be completed by executing multiple possible sequences of sub-goals, the planner should be able to select the best route base on the current state of the agent.

=> the complex and diverse state distribution of open-world environments makes state-awareness hard to achieve.
Challenges in open world environments

Zero-Shot Planning via Causal LLM

Translation to Admissible Action

LLM for planning in close worlds

Language Models as Zero-Shot Planners: Extracting Actionable Knowledge for Embodied Agents

Do As I Can, Not As I Say: Grounding Language in Robotic Affordance
CraftJarvis: an embodied agents in Minecraft
def craft_wooden_axe(initial_inventory={}):
    # step 1: mine 3 logs
    mine(obj = {"log":3}, tool = None)
    # step 2: craft 12 planks from 3 logs
    craft(obj = {"planks":12}, materials = {"log":3},
          tool = None)
    # step 3: craft 4 sticks from 2 planks
    craft(obj = {"stick":4}, materials = {"planks":2}, tool = None)
    # step 4: craft 1 crafting_table from 4 planks
    craft(obj = {"crafting_table":1}, materials = {"planks":4}, tool = None)
    # step 5: craft 1 wooden_axe from 3 planks and 2 sticks on crafting table
    craft(obj = {"wooden_axe":1}, {"planks": 3, "stick": 2}, "crafting_table")
    return "wooden_axe"

Lots of errors!
Self-correction
<task>: Obtain a diamond in *Minecraft* step-by-step?

**original <plan>:**

```
3 12
1 4 11 1 1 3 1 3 1 1
```

**Self-check:** When simulating on the goal, I find are not enough (lack of 2 ). So I need craft more from . More require more . So I need to mine more .

**refined <plan>:**

```
4 16
1 8 11 1 1 3 1 3 1 1
```

**multi-modal <feedback>:** I failed on . My current state is: is broken; I still have in the inventory. My position is …

**Self-explain:** Because mining needs , which I do not have in the inventory. Crafting needs . So I need to smelt into first.

**new <plan> by re-planning:**

```
3 1 1
```
Embodied RAG (retrieval-augmented generation)
Multi-Modal Memory

[
{
"task" : [ "entity" ] 🗿 wooden pickaxe
"plan" : [ "language" ] 🗿
"state" : [ "image" ]
},
{
"task" : [ "entity" ] 🗿️ stone pickaxe
"plan" : [ "language" ] 🗿️
"state" : [ "image" ]
},
...
]
Query generation via reasoning

User: My current task is 📖, but I have never accomplished this task before. What related tasks might be helpful for me to complete 📖?

Assistant: 📖 📖 📖

**initial query (text)**

- Enchanting Table
- Diamond
- Paper
- Diamond Pickaxe
- Leather

**query generation via reasoning**

- Reasoning stops
- Diamond
- Leather
- Iron Pickaxe
- Diamond axe

**final query (text):** 📖 Diamond 📖 Leather 📖 Paper 📖 Iron Pickaxe

**final query (obs):** 📖 📖 📖 📖 📖
Open world embodied control: goal-aware representation learning and horizon prediction

Goal-Sensitive Backbone (GSB)

Training

Evaluation

Adaptive Horizon Prediction

Horizon Loss

Goal Space

Action Space

Extra Observation

Image Observation
Open world embodied control: pretraining and alignment

STEVE-1: A Generative Model for Text-to-Behavior in Minecraft
Some follow-up research projects built upon CraftJarvis
Voyager: An Open-Ended Embodied Agent with Large Language Models
GITM: language agent with structured knowledge library

Ghost in the Minecraft: Generally Capable Agents for Open-World Environments via Large Language Models with Text-based Knowledge and Memory
Plan4MC: Skill Reinforcement Learning and Planning for Open-World Minecraft Tasks
What’s next?
From unified models to unified agents
The following facts seem to be true:

1. Learning from massive web-scale data 📚
2. Large scale architecture $O(10^B)$ 🐘
3. Multi-tasking 🐙
4. (optional) Multimodal understanding 📚👀👂

From unified models to unified agents
The following facts seem to be true:

1. Learning from massive web-scale data
2. Large scale architecture $O(10^B)$
3. Multi-tasking
4. (optional) Multimodal understanding

**Definition**

A language-piloted, large-scale agent that can fulfill arbitrary goals from multimodal input in embodied environments.
Machine Learning @ BIGAI

From unified models to unified agents

**Few-shot learning**

Enable few-shot (in-context) learning in these models?

**More modalities**

Unified models for other modalities (3D, egocentric videos, proprioception, high-res structured input, etc)?

**Beyond zero-shot**

Fine-tuning:
- Flexibility to new tasks/domains
- Preventing the acquired knowledge from being wiped out

**Road to agents**

We need better learning algorithms for:
- episodic memory & situation awareness
- learning from interactions
Thank you