Grounding Language Models to Real-World Environments

Yu Su
The Ohio State University

THE OHIO STATE UNIVERSITY

NLP

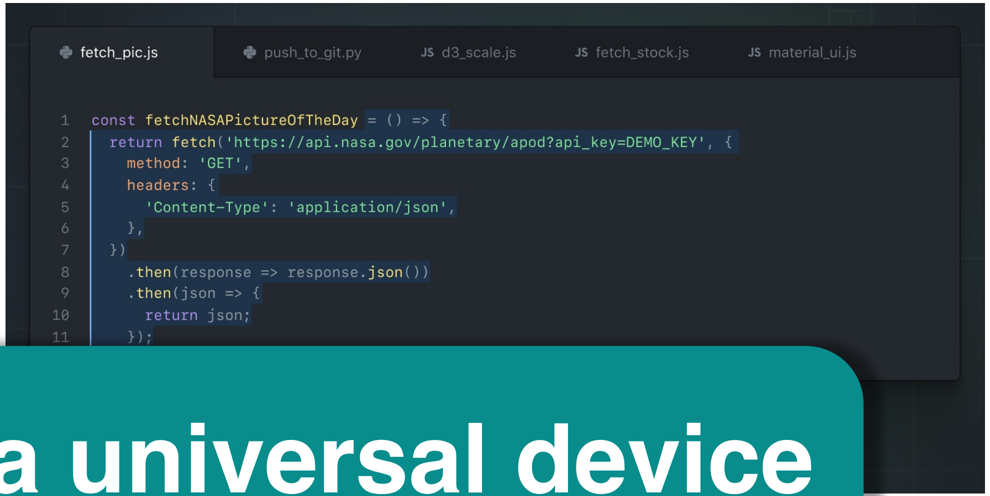
Slides credit to my amazing student Yu Gu

Language Models Nail Everything?

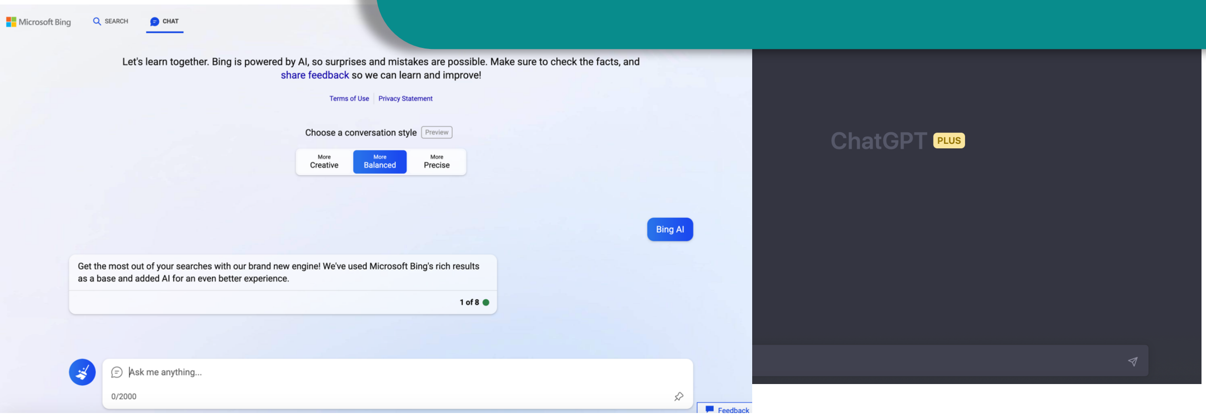
GLUE

SuperGLUE

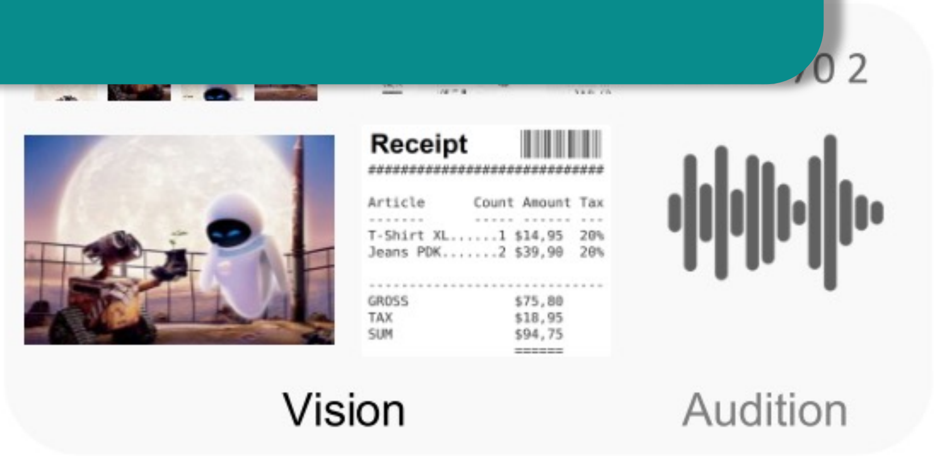
Languag



Natural language as a universal device for human-machine interaction?



Dialogue



Multimodal Applications

Language Models: What’s Missing?

SU Book a meeting with Tommy tomorrow at 3 pm

I'm sorry, but I am a text-based AI assistant and don't have the ability to book meetings. I'm not connected to the internet, so I don't have access to your calendar or any other information that would be necessary to book a meeting. I can provide general information and answer questions, but I'm not able to perform tasks like booking meetings. Is there something else I can help you with?

Cannot take actions in the real world

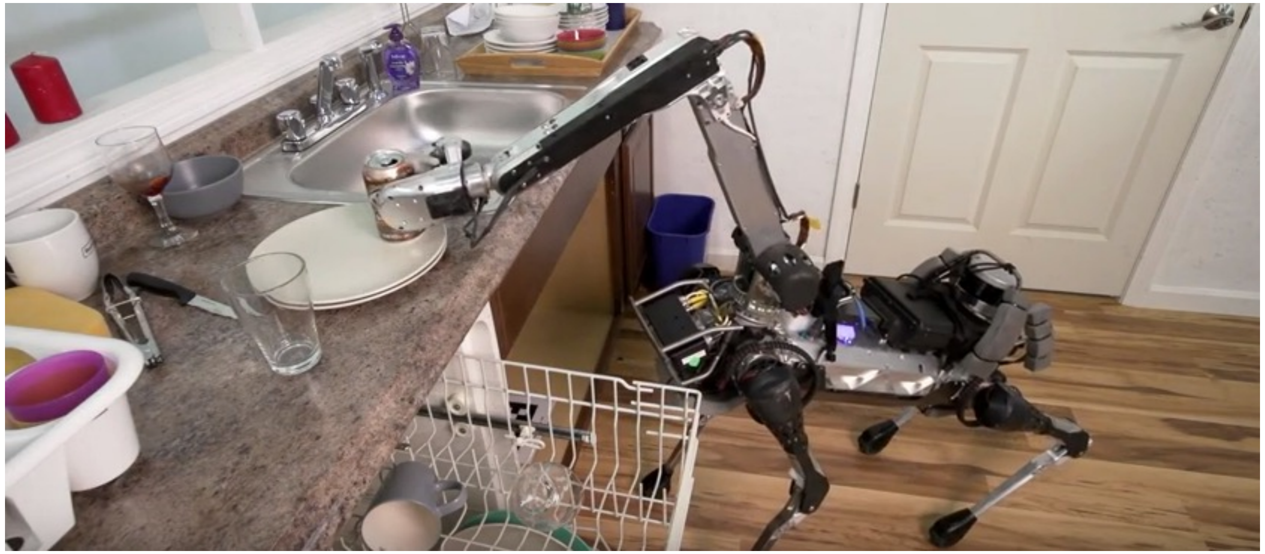
SU Marc Bulger had the most yards rushing in what season?

As a text-based AI assistant, I don't have access to specific information about individuals or events. I don't have the ability to browse the internet or access other sources of information, so I can't provide you with specific answers to questions like this. I'm trained to provide general information and answer questions to the best of my ability based on the information I have been trained on, but I don't have the ability to look up specific facts or provide detailed information about specific individuals or events. Is there something else I can help you with?

Cannot handle more specific information needs

Grounded Language Understanding: What and Why?



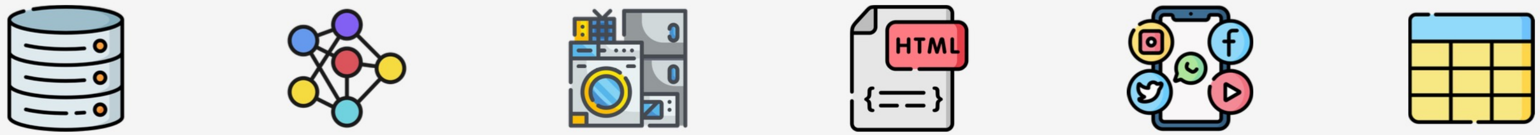


Grounded Language Understanding: Formal Definition

Given a natural language utterance u and a target environment E

$$\pi: (u, E) \rightarrow p, \; s.t. \; [\![u]\!]_E = [\![p]\!]_E$$

Where p is a plan/program in a formal language, and $[\![\cdot]\!]_E$ is the denotation



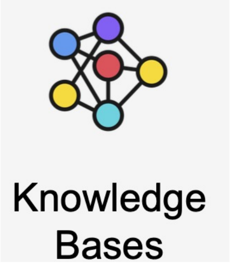
Databases | Knowledge Bases | Physical World | Web Pages | Apps | Tables

Grounded Language Understanding: Formal Definition

Given a natural language utterance u and a target environment E

$$\pi: (u, E) \rightarrow p, \ s.t. \ [\![u]\!]_E = [\![p]\!]_E$$

Where p is a plan/program in a formal language, and $[\![\cdot]\!]_E$ is the denotation



u: *What is the latest released computer emulator developed in Java?*

p: (ARGMAX (AND ComputerEmulator
(JOIN LanguagesUsed Java))
LatestReleaseDate)

Grounded Language Understanding: Formal Definition

Given a natural language utterance u and a target environment E

$$\pi: (u, E) \rightarrow p, \ s.t. \ [\![u]\!]_E = [\![p]\!]_E$$

Where p is a plan/program in a formal language, and $[\![\cdot]\!]_E$ is the denotation

Physical World

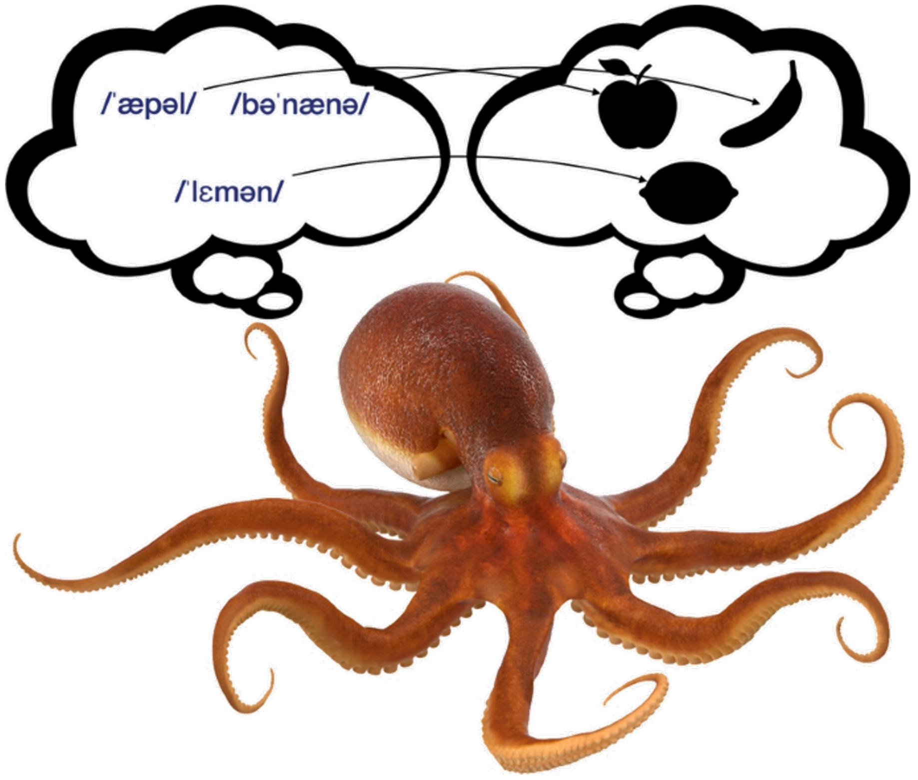
u: *Bring me a cup of coffee*

p: [turn left, move forward, pick up cup, turn around, move forward, …, put cup in coffee maker, toggle coffee maker, …]

The Symbol Grounding Problem

Language models are mostly trained with textual corpora

- **BERT:** Wikipedia (2.5B words) + BookCorpus (800M words)
- **T5:** C4 (two orders of magnitude larger)
- **GPT-3:** 45TB text data + others



Key challenge: How to ground textual symbols to different environments/formal languages

Pangu: A Unified Framework for Grounded Language Understanding

Yu Gu, Xiang Deng, Yu Su

The Ohio State University

THE OHIO STATE UNIVERSITY

NLP


QUIZ
TIME!

Q1 Find the right program over a KB

Question: Who has ever coached an ice hockey team in Canada?

Program:

A. (AND cricket.cricket_coach (JOIN cricket.cricket_team.coach_inv (JOIN sports.sports_team.location Canada)))

B. (AND ice_hockey.hockey_coach (JOIN ice_hockey.hockey_team.coach_inv (JOIN sports.sports_team.location Canada)))

C. (AND ice_hockey.hockey_team (JOIN sports.sports_team.location Canada))

Q2 Write the corresponding KB program

Question: What's the classification of the M10 engine?

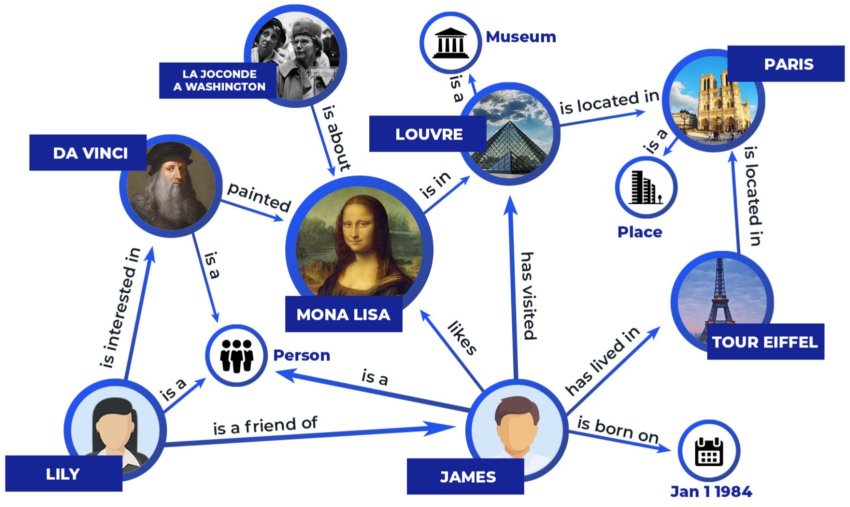
Program:

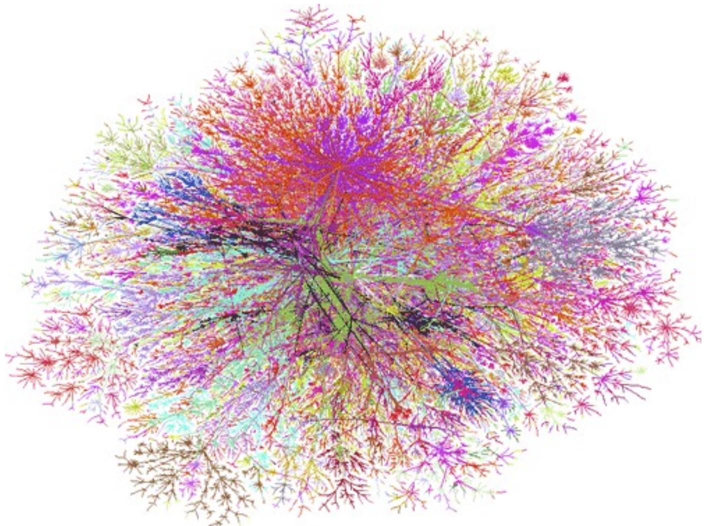
(AND automotive.engine_type (JOIN automotive.engine_type.used_in M10))

Why is Q2 harder?

1 You need to learn the grammar

2 You need to know the environment specifics





Key Message

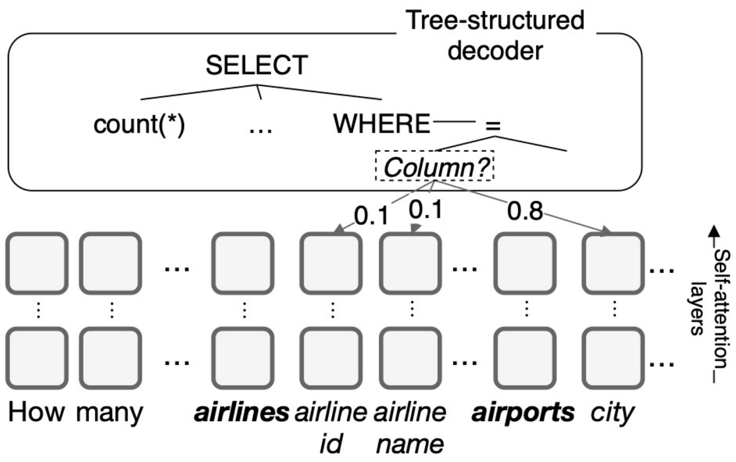
Directly generating plans (programs) may not be the optimal way of using LMs for grounded language understanding

Pangu:

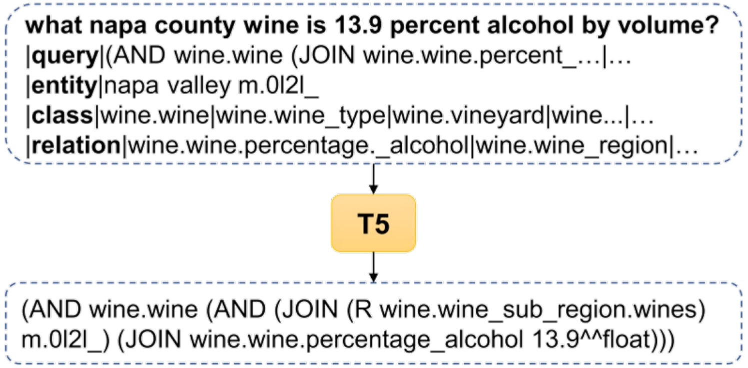
A unified framework that models grounded language understanding as a discrimination task

The Status Quo

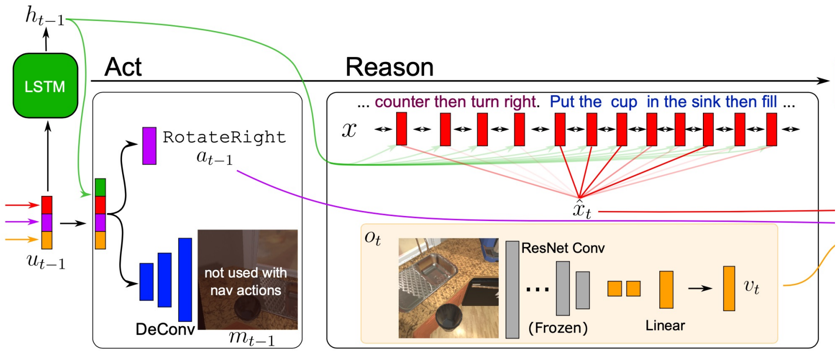
Autoregressive generation with Seq2Seq LMs



Text-to-SQL Parsing
(Wang et al. 2020)



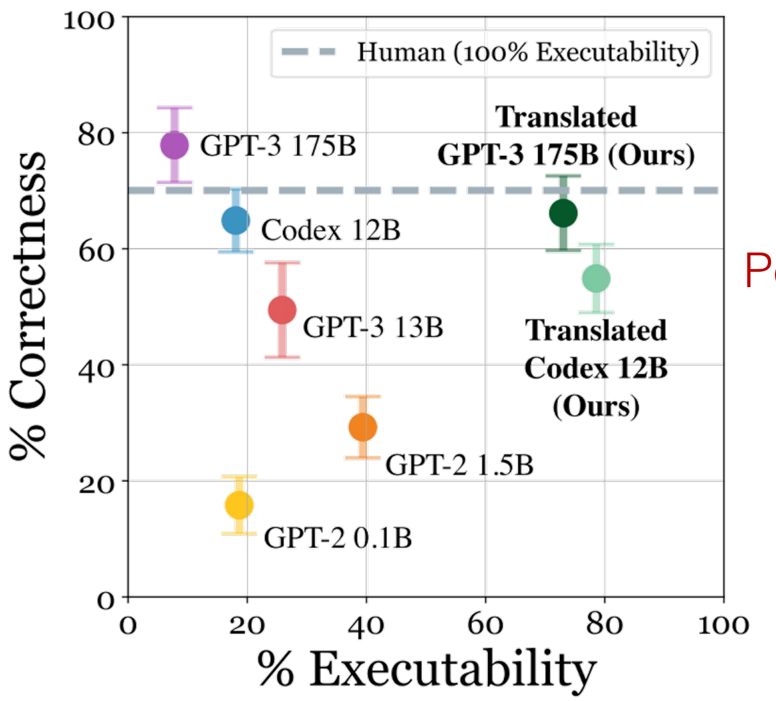
KBQA
(Shu et al. 2022)



Embodied AI
(Shridhar et al. 2019)

The Status Quo

Autoregressive generation can produce invalid plans



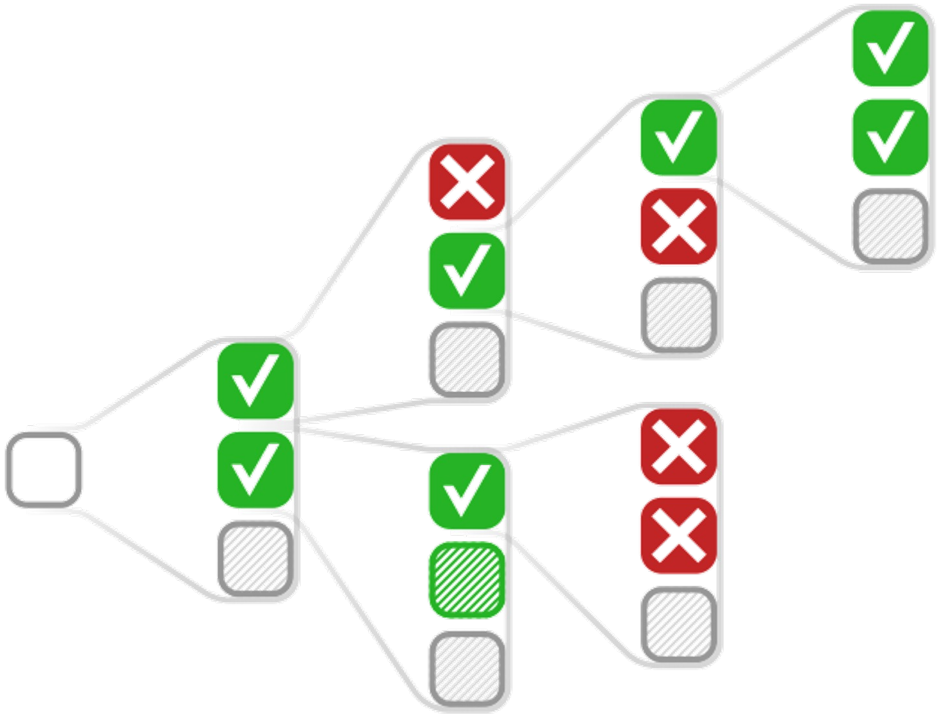
Percentage of executable plans for embodied AI (Huang et al. 2022)

	GrailQA		WebQSP	
	EXEC	VALID	EXEC	VALID
Top-1	99.7	88.1	98.7	91.1
Top-3	99.7	89.4	99.5	94.5
Top-5	99.7	89.8	99.5	94.6
Top-10	99.7	90.4	99.5	95.4

Percentage of executable and valid programs for KBQA (Ye et al. 2021)

The Status Quo

A possible fix: constrained decoding



Example Decoding Rules

- The first token must be ‘(’
- The token after ‘(’ can be ‘AND’, ‘JOIN’, ‘ARGMAX’ ..
- …

Picard (Scholak et al. 2021)

The Status Quo

Constrained decoding can be shortsighted and hard to control

Question: Neil Diamond composed what TV song?

Gold: (JOIN Composer Neil_Diamond) (AND TV_Song #0)

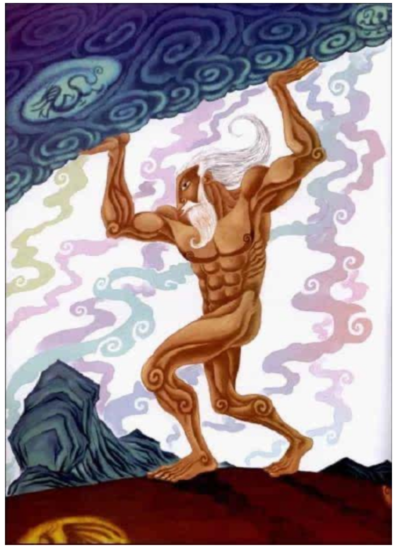
6 steps later

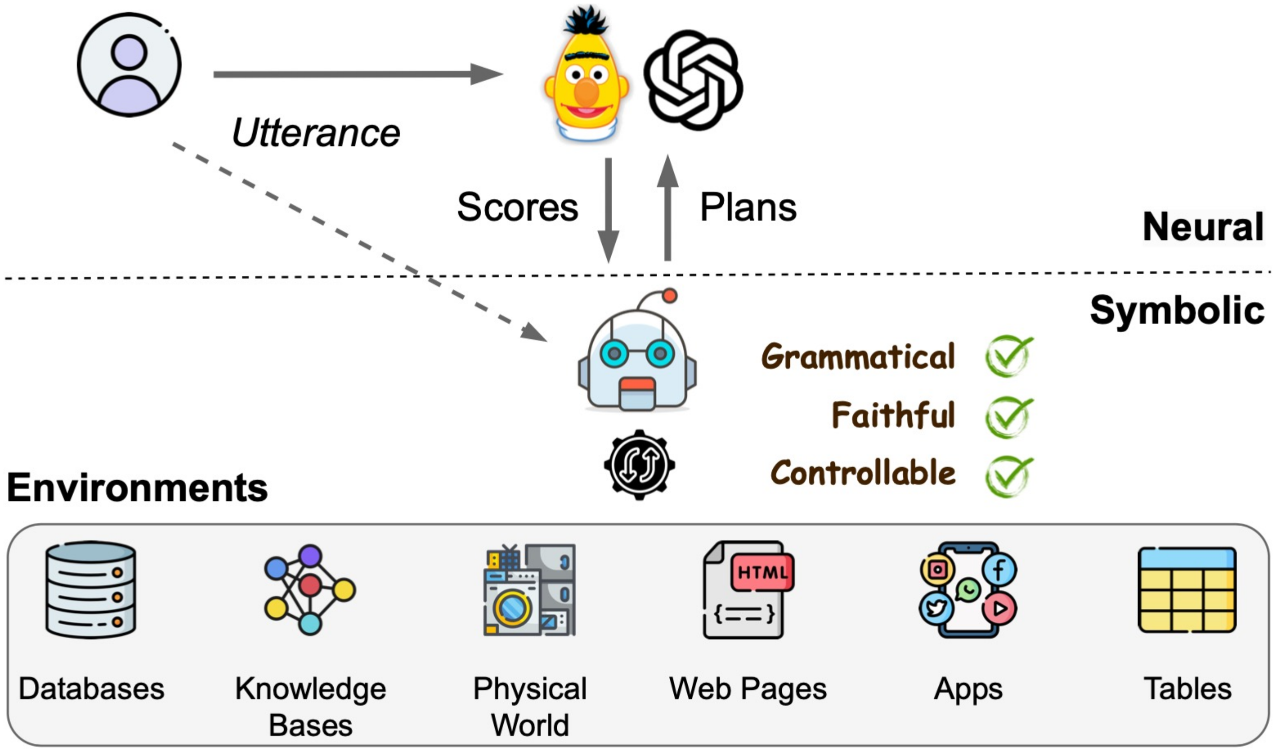
Predicted: (JOIN Composer Neil_Diamond) (**JOIN** Song #0) (AND Recording #1)

Our Proposal: Pangu Framework

Goals:

- Allow LMs to focus on discrimination
- Generic for different tasks





A symbolic agent searches the environment to propose valid candidate plans, while a neural LM scores the plans to guide the search process

Key Assumptions

1. A complex plan can be expanded from smaller sub-plans incrementally
2. Valid action space at each step is much smaller compared with decoding

Our Proposal: Framework

Algorithm 1: PANGU

```
Input: utterance q, initial plans P_0, environment E
t ← 1;
while True do
    /* AGENT PROPOSES PLANS */
    C_t ← Candidate-Plans(P_{t−1}, E)
    /* LM SCORES AND PRUNES PLANS */
    P_t ← Top-K(q, C_t)
    if Check-Termination() = True then
        return top-scored plan
    t ← t + 1
```

Initialization of search

Enumerate candidate plans from the environment

Rank candidate plans using LMs

Repeat until the termination condition is met

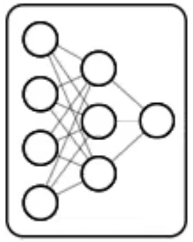
Our Proposal: Instantiation

Testbed:

- KBQA

LMs:

- BERT
- T5
- Codex



New SoTA on KBQA

Prior Art	78.7
Pangu w/ BERT-base	79.9
Pangu w/ T5-base	79.9
Pangu w/ T5-3B	**81.7**

F1 on GrailQA (i.i.d. + non-i.i.d., ~45K training examples)

Prior Art	34.3
Pangu w/ BERT-base	52.0
Pangu w/ T5-base	53.3
Pangu w/ T5-3B	**62.2**

F1 on GraphQuestions (non-i.i.d., ~2K training examples)

Prior Art	78.8
Pangu w/ BERT-base	77.9
Pangu w/ T5-base	77.3
Pangu w/ T5-3B	**79.6**

F1 on WebQSP (i.i.d., ~3K training examples)

Findings:

1. Particularly strong performance for non-i.i.d. generalization
2. Stable gain from increased model size

In-Context Learning with LLMs

Prior Art	**78.7**
Codex 10-shot	48.9
Codex 100-shot	53.3
Codex 1000-shot	56.4

F1 on GrailQA (i.i.d. + non-i.i.d., ~45K training examples)

Prior Art	34.3
Codex 10-shot	42.8
Codex 100-shot	43.3
Codex 1000-shot	**44.3**

F1 on GraphQuestions (non-i.i.d., ~2K training examples)

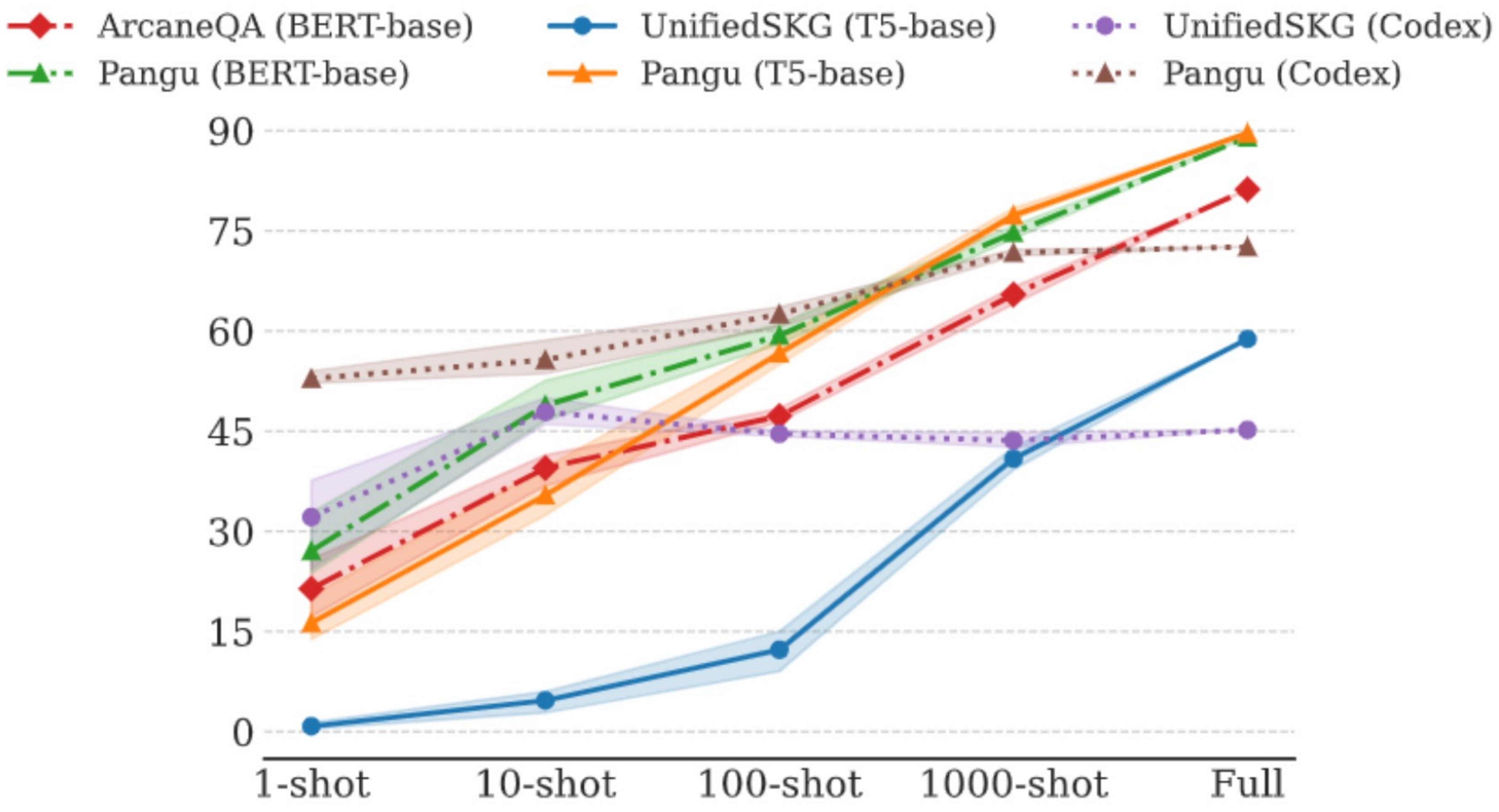
Prior Art	**78.8**
Codex 10-shot	45.9
Codex 100-shot	54.5
Codex 1000-shot	68.3

F1 on WebQSP (i.i.d., ~3K training examples)

Findings:

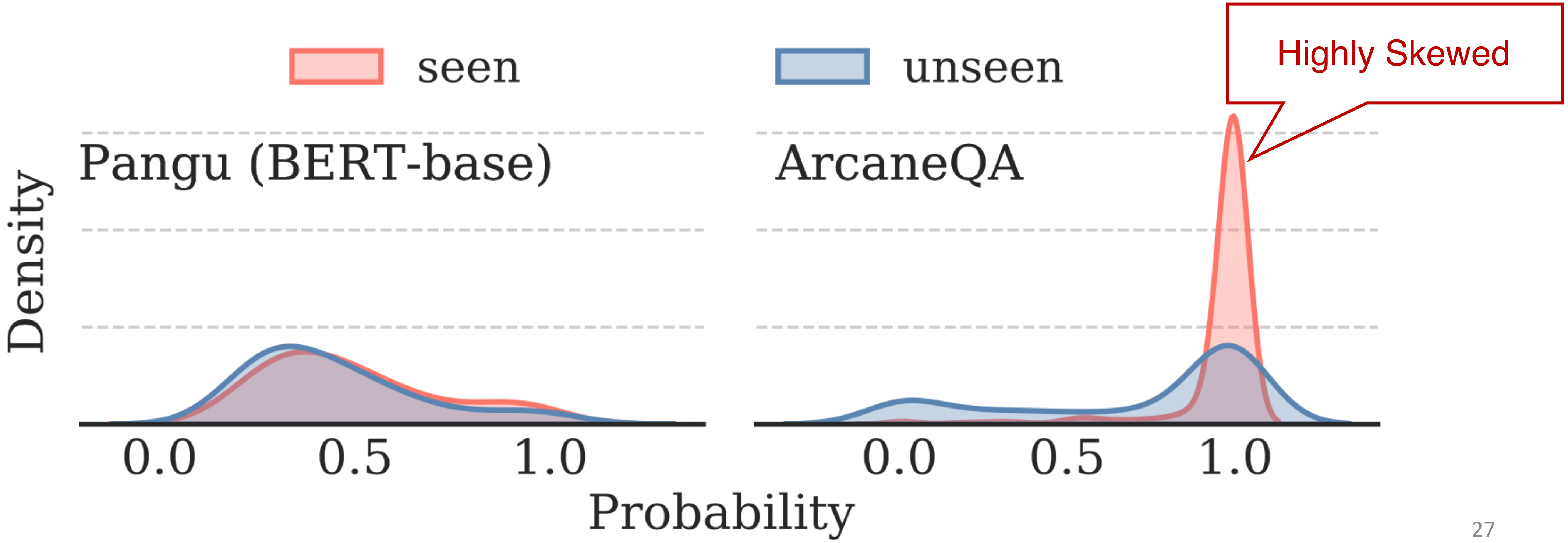
1. SoTA performance on GraphQ with only 10 training examples
2. Marginal gain from more training data for non-i.i.d.

Pangu Improves Sample Efficiency



Pangu vs. Constrained Decoding

Autoregressive models tend to overfit seen structures during training



LLM-Planner: Few-Shot Grounded Planning for Embodied Agents with Large Language Models

Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M. Sadler, Wei-Lun Chao, Yu Su

The Ohio State University

DEVCOM Army Research Laboratory

NLP

Language-driven Embodied Agents

- Embodied agents follow language instructions to complete tasks in a physical environment
- Long-horizon tasks: 50+ steps
- Diverse tasks and environments
- Can LLMs help?

Embodied Agent Planning with LLMs?

Instruction: “make me a cup of coffee”

LLM?

Low-level Plan: [turn left, move forward, pick up cup, turn around, move forward, …, put cup in coffee maker, …]

Embodied Agent Planning with LLMs?

Instruction: “make me a cup of coffee”

⬇ LLM-Planner

High-level Plan: [navigation cup, pick up cup, navigation coffee machine, …]

⬇ Low-level planner

Low-level Plan: [turn left, move forward, pick up cup, turn around, move forward, …, put cup in coffee maker, …]

Dynamic Grounded Planning

Instruction: “make me a cup of coffee”

LLM-Planner

High-level Plan: [navigation cup, pick up cup, navigation coffee machine, …]

Low-level planner

Low-level Plan: [Turn left, move forward, pick up cup, turn around, move forward, …, put cup in coffee maker, …]

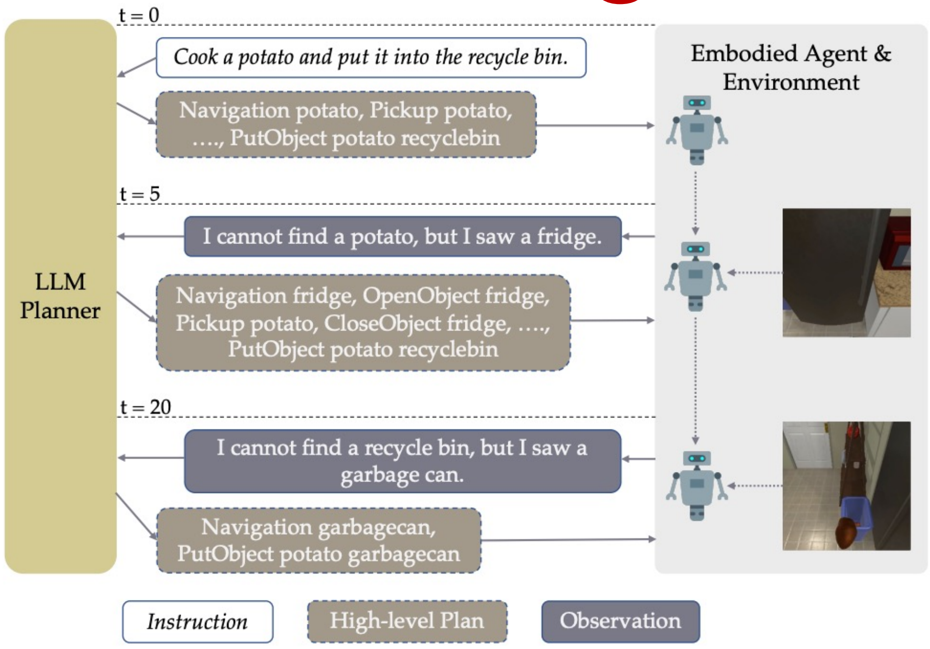
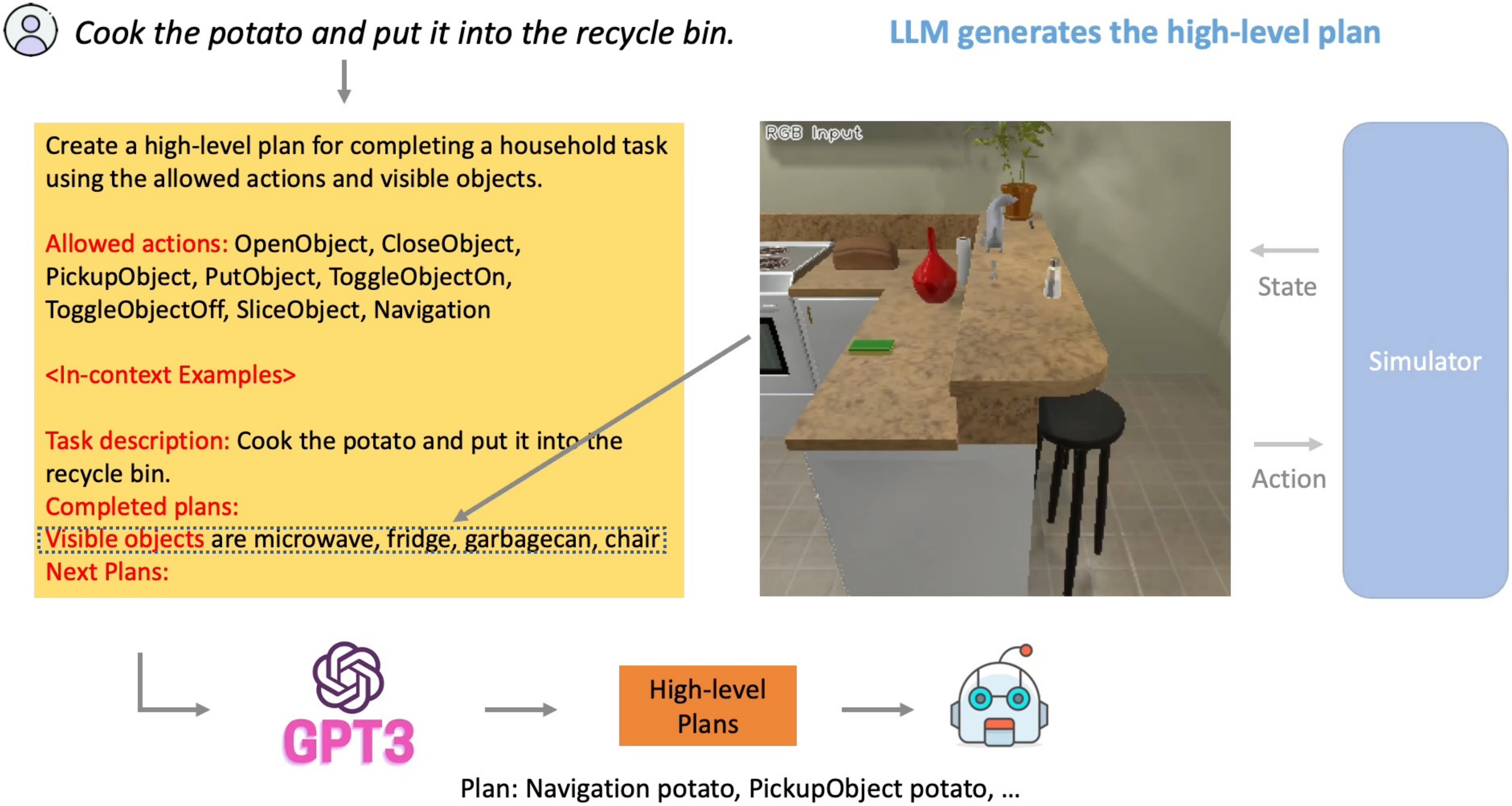


Figure 1. An illustration of LLM-Planner for high-level planning. After receiving the natural language instruction ($t = 0$), LLM-Planner first generates a high-level plan by prompting a large language model (*e.g.*, GPT-3). When the embodied agent gets stuck during the execution of the current plan ($t = 5$ and 20), LLM-Planner re-plans based on observations from the environment to generate a more grounded plan, which may help the agent get unstuck. The commonsense knowledge in the LLM (*e.g.*, food is often stored in a fridge) allows it to produce plausible high-level plans and re-plan based on new information from the environment.


Cook the potato and put it into the recycle bin.
LLM generates the high-level plan
Create a high-level plan for completing a household task using the allowed actions and visible objects.
Allowed actions: OpenObject, CloseObject, PickupObject, PutObject, ToggleObjectOn, ToggleObjectOff, SliceObject, Navigation
<In-context Examples>
Task description: Cook the potato and put it into the recycle bin.
Completed plans:
Visible objects are microwave, fridge, garbagecan, chair
Next Plans:
RGB Input
State
Action
Simulator
GPT3
High-level Plans
Plan: Navigation potato, PickupObject potato, ...

Evaluation on ALFRED

- LLM-Planner achieves competitive performance with only **100** training examples
- Existing methods can barely complete any task under the same low-data setting

Model	SR	GC	HLP ACC
Full-data setting: 21,023 (instruction, trajectory) pairs			
E.T. [27]	8.57	18.56	–
HiTUT [40]	13.87	20.31	–
M-TRACK [36]	16.29	22.60	–
FILM [26]	27.80	**38.52**	–
LEBP [18]	**28.30**	36.79	–
Few-shot setting: 100 (instruction, high-level plan) pairs			
HLSM [3]	0.61	3.72	0.00
FILM [26]	0.20	6.71	0.00
SayCan [1]	9.88	22.54	37.57
LLM-Planner (Static) + HLSM	15.83	20.99	43.24
LLM-Planner + HLSM	**16.42**	**23.37**	**46.59 – 68.31**

SR: Success Rate, GC: Goal Completion Rate, HLP ACC: High-level Planning Accuracy

What's the journey ahead of us?

- Is NLP dead?
- Absolutely not. It's the most exciting time for NLP ever!
- However, instead of *natural language processing*, perhaps we should focus on *natural language programming* next

Natural Language Programming

When is my flight to Seattle?

How long will it take to get to the airport?

Book a Uber 1.5 hours before that.

Any good Chinese restaurants close to my hotel?

Tomorrow at 5:00 pm.

It will take 20 minutes according to Google Maps.

Sure. Booked an Uber for 3:30 pm tomorrow to the Columbus airport.

According to Yelp, Haidilao has 4.5 stars and is 2-min walk from Hyatt.

Foundation Model

Thanks &