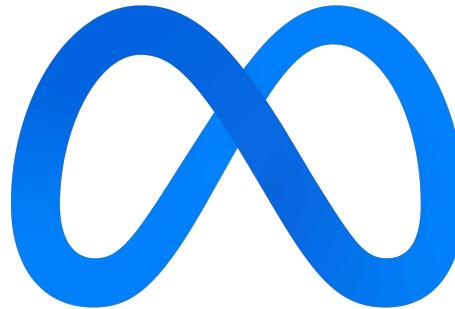
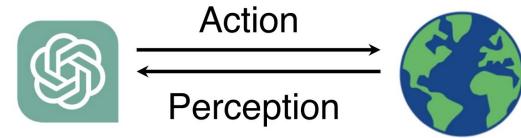
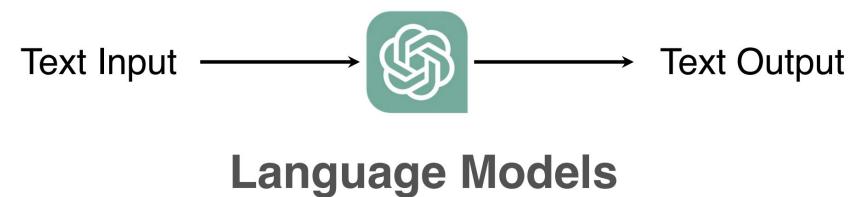
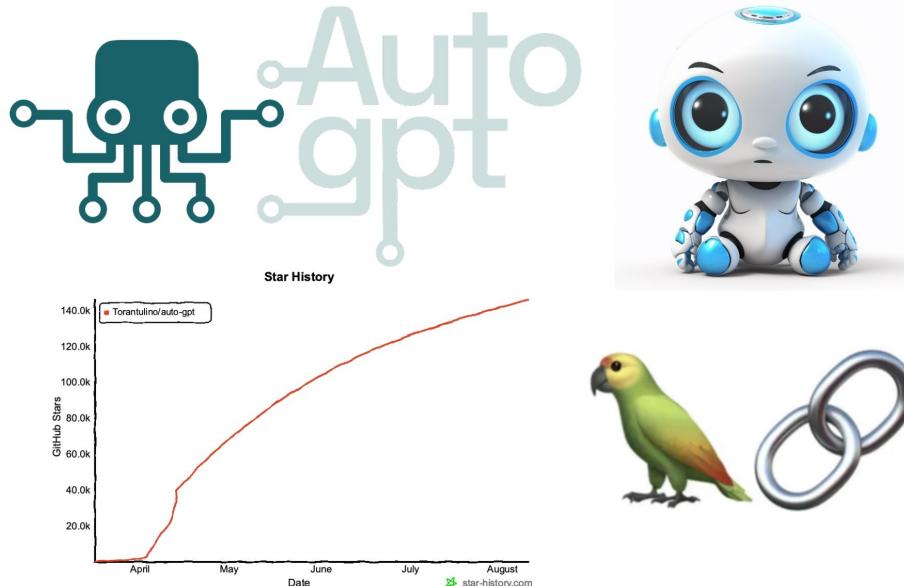




# TravelPlanner: A Benchmark for Real-World Planning with Language Agents



# From Language Models to Language Agents



- [1] AutoGPT <https://github.com/Significant-Gravitas/Auto-GPT>
- [2] LangChain <https://www.langchain.com>
- [3] BabyAGI <https://babyagi.org>
- [4] Generative Agents [https://github.com/joonspk-research/generative\\_agents](https://github.com/joonspk-research/generative_agents)

# What We Expect Language Agents To Do?



Making Slides



Time Arrangement



Travel Planning

And More....

**Complex Tasks With Heavy Planning!**

# What Are Current Language Agents Shooting for?

## (1) Hotspot QA

**Question:** Aside from the Apple Remote, what other device can control the program Apple Remote was originally designed to interact with?

## (1d) ReAct (Reason + Act)

**Thought 1:** I need to search Apple Remote and find the program it was originally designed to interact with.

**Act 1:** `Search[Apple Remote]`

**Obs 1:** The Apple Remote is a remote control introduced in October 2005 by Apple ... originally designed to control the `Front Row media center` program ...

**Thought 2:** Apple Remote was originally designed to control the Front Row media center program. I need to search `Front Row` next and find what other device can control it.

**Act 2:** `Search[Front Row]`

**Obs 2:** Could not find [Front Row]. Similar: ['Front Row Seat to Earth', 'Front Row Motorsports', 'Front Row (software)', ...]

**Thought 3:** Front Row is not found. I need to search `Front Row (software)`.

**Act 3:** `Search[Front Row (software)]`

**Obs 3:** Front Row is a discontinued media center software ...

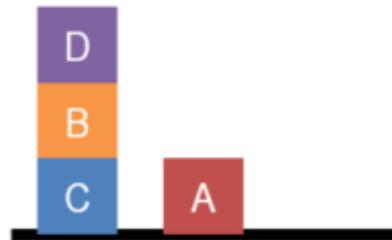
**Thought 4:** `Front Row (software)` is controlled by an Apple Remote or the keyboard function keys. So the answer is keyboard function keys.

**Act 4:** `Finish[keyboard function keys]`



## Blocks World:

### Initial State



A on Table

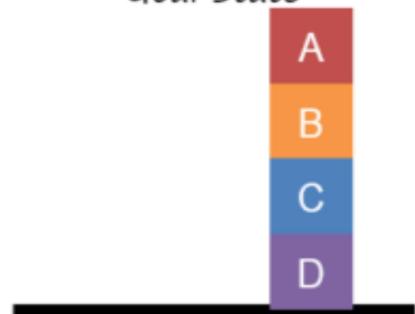
B on C

C on Table

D on B

$$\Delta = 3$$

### Goal State



A on B

B on C

C on D

D on Table

**Are these tasks leading to our expectations?**

# What Are Current Agents Shooting for?

Web Agent (Mind2Web):

The screenshot shows a flight search interface with three flight options listed:

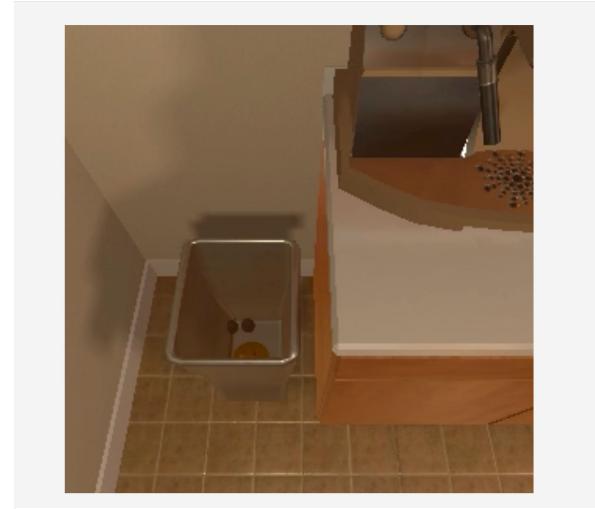
- Flight 1:** ORD to LHR at 8:42 PM, arriving 10:20 AM, 7h 38m nonstop. AA 98 (Boeing 787). Options: Round trip \$613, Round trip \$796, Not available, Not available, Not available.
- Flight 2:** ORD to LHR at 8:30 AM, arriving 10:10 PM, 7h 40m nonstop. AA 90 (Boeing 787). Options: Round trip \$613, Round trip \$796, Round trip \$3,203 (FLAGSHIP\*), Not available.
- Flight 3:** ORD to LHR at 6:01 PM, arriving 7:45 AM, 7h 44m nonstop. AA 86 (Boeing 787). Options: Round trip \$613, Round trip \$796, Not available, FLAGSHIP\* Round trip \$10,442, Not available.

Sort by: Relevance. Filter: Main Cabin.

Find a flight from Chicago to London on 20 April and return on 23 April.

**They are grounding tasks; do not need heavy planning**

Embodied Agent (ALFRED):



Place a cooked egg inside the waste basket.

# Complex Tasks

1. Making Slides
2. Time Arrangement
3. **Travel Planning**
4. ...

- Real-world open problems. → **Evaluation?**
- Heavy planning required: (multiple parts, trade-offs, decisions, overall completeness, ...) Time-consuming even for human but WITH A SIMPLE QUERY → **Annotation? Scale?**

# Travel Plan - Observe the Environment with Tools

Table A.2. Tool description and the number of data entries in the database.

Tool	Data Entries(#)	Description
CitySearch	312	Search cities in the given state.
FlightSearch	3,827,361	Search flight information for a specific date between two cities.
DistanceMatrix	17,603	Search the driving distance, time, and possible cost between two cities.
RestaurantSearch	9,552	Search restaurants in the given city.
AttractionSearch	5,303	Search attractions in the given city.
AccommodationSearch	5,064	Search accommodations in the given city.
NotebookWrite	-	Write the selected data entry into the Notebook tool with a short description.

7 Tools in Total

Searching from ~4M Entries

# Travel Plan - Satisfy Constraints

## User Needs (Hard Constraints):      Environment Constraints:

- Budgets
  - Hotel Preferences
  - Restaurants Preferences
  - Transportation Preferences
- No available transportations
  - No available attractions

## Commonsense Constraints:

- No repeated cities or jumping cities  
(A-> B and suddenly C -> D)
- No repeated attractions

**15 Constraints in Total**

....

# Dataset Statistics

Easy, Medium, Hard are determined by the number of user needs.

Longer duration, more cities in a state to travel.

Each query has at least one feasible plan.

*Table A.1. Dataset distribution.*

	Days	Easy	Medium	Hard
Training (#45)	3-day	5	5	5
	5-day	5	5	5
	7-day	5	5	5
Validation (#180)	3-day	20	20	20
	5-day	20	20	20
	7-day	20	20	20
Test (#1,000)	3-day	122	104	82
	5-day	116	114	121
	7-day	110	115	116



User

I'm going from **Seattle** to **California** from **November 6 to 10, 2023**. I have a **budget of \$6,000**. For lodging, I prefer an **entire room** and the accommodations must be **pet-friendly**.



User

I'm going from **Seattle** to **California** from **November 6 to 10, 2023**. I have a **budget of \$6,000**. For lodging, I prefer an **entire room** and the accommodations must be **pet-friendly**.

Let me help! To solve this problem, I need to  
**(1) analyze certain constraints,**  
**(2) collect information** through reasonable use of necessary tools.



Agent

#### Toolbox

1. CitySearch(·)
2. AttractionSearch(·)
3. FlightSearch(·)
4. DistanceMatrix(·)
5. RestaurantSearch(·)
6. AccommodationSearch(·)



#### Interaction with Environment





User

I'm going from **Seattle** to **California** from **November 6 to 10, 2023**. I have a **budget of \$6,000**. For lodging, I prefer an **entire room** and the accommodations must be **pet-friendly**.

Let me help! To solve this problem, I need to **(1) analyze certain constraints, (2) collect information through reasonable use of necessary tools.**



### Toolbox

1. CitySearch(·)
2. AttractionSearch(·)
3. FlightSearch(·)
4. DistanceMatrix(·)
5. RestaurantSearch(·)
6. AccommodationSearch(·)



### Interaction with Environment



## Planning

The plan must adhere to certain **constraints**, e.g., **user needs** and **commonsense**. It's also vital to ...



### User Needs (Hard Constraints)

1. Budget: \$6000
2. Room Type: Entire Room
3. Room Rule: Pet-friendly



### Commonsense Constraints

1. Reasonable City Route
2. Diverse Restaurants
3. Diverse Attractions
4. Non-conflicting Transportation
5. Accommodation meets Minimum Night



### Delivery Plan

2023-11-06 Seattle -> Los Angeles	2023-11-07 Los Angeles	2023-11-08 Los Angeles -> San Diego	2023-11-09 San Diego	2023-11-10 San Diego -> Seattle
<ul style="list-style-type: none"> <li>Flight: F123, (13:40-16:12), Cost: \$120</li> <li>Accommodation: Luxury building studio</li> <li>Dinner: The Attraction</li> </ul>	<ul style="list-style-type: none"> <li>Breakfast: Golden Miran</li> <li>Lunch: Rajdhani Restaurant</li> <li>Dinner: Dominos Pizza</li> <li>Attractions: Santa Monica Pier, Griffith Park</li> <li>Accommodation: Luxury building studio</li> </ul>	<ul style="list-style-type: none"> <li>Take taxi to San Diego</li> <li>Breakfast: Open Yard</li> <li>Lunch: The Lost Mughal</li> <li>Attractions: The Lost Mughal</li> <li>Accommodation: East Side Apartment</li> </ul>	<ul style="list-style-type: none"> <li>Breakfast: Bassin Robbins</li> <li>Lunch: Harry's Bar</li> <li>Dinner: Crown Way</li> <li>Attractions: La Jolla Shores Park, California</li> <li>Accommodation: East Side Apartment</li> </ul>	<ul style="list-style-type: none"> <li>Flight: F789 (13:40-16:12), Cost: \$300</li> </ul>

2023-11-06

#### Seattle -> Los Angeles

- Flight: F123, (13:40-16:12), Cost: \$120
- Accommodation: Luxury building studio
- Dinner: The Attraction





User

I'm going from **Seattle** to **California** from **November 6 to 10, 2023**. I have a **budget of \$6,000**. For lodging, I prefer an **entire room** and the accommodations must be **pet-friendly**.

Let me help! To solve this problem, I need to **(1) analyze certain constraints, (2) collect information through reasonable use of necessary tools.**



### Toolbox

1. CitySearch(·)
2. AttractionSearch(·)
3. FlightSearch(·)
4. DistanceMatrix(·)
5. RestaurantSearch(·)
6. AccommodationSearch(·)

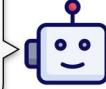


### Interaction with Environment



## Planning

The plan must adhere to certain **constraints**, e.g., **user needs** and **commonsense**. It's also vital to ...



### User Needs (Hard Constraints)

1. Budget: \$6000
2. Room Type: Entire Room
3. Room Rule: Pet-friendly



### Commonsense Constraints

1. Reasonable City Route
2. Diverse Restaurants
3. Diverse Attractions
4. Non-conflicting Transportation
5. Accommodation meets Minimum Night



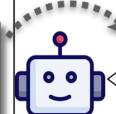
### Delivery Plan

2023-11-06	2023-11-07	2023-11-08	2023-11-09	2023-11-10
Seattle -> Los Angeles • Flight: F123, (13:40-16:12), Cost: \$120 • Accommodation: Luxury building studio • Dinner: The Attraction	Los Angeles • Breakfast: Golden Mirar • Lunch: Rajdhani Restaurant • Dinner: Dominos Pizza • Accommodation: Luxury building studio	Los Angeles -> San Diego • Take taxi to San Diego • Breakfast: Open Yard • Lunch: The Lost Mughal • Attractions: Santa Monica Pier, Griffith Park • Accommodation: East Side Apartment	San Diego • Breakfast: Bassie Robbins • Lunch: Harry's Bar • Dinner: Chivas Way • Attractions: Cabrito Monument, La Jolla Shores Park, California	San Diego -> Seattle • Flight: F789 (07:59-10:56), Cost: \$300

2023-11-06

#### Seattle -> Los Angeles

- Flight: F123, (13:40-16:12), Cost: \$120
- Accommodation: Luxury building studio
- Dinner: The Attraction



## Information Collection

[Tool] **CitySearch[California]**

[Result] San Francisco, Los Angeles,..., San Diego



[Tool] **FlightSearch[Seattle, San Francisco, 2023-11-06]**

[Result] No Flights.

[Tool] **FlightSearch[Seattle, Los Angeles, 2023-11-06]**

[Result] Flight Number: F123, 13:40-16:12, Cost: \$120

[Tool] **DistanceMatrix[Los Angeles, San Diego, taxi]**

[Result] Duration: 1 hour 57 mins, Distance: 193 km, Cost: \$200

[Tool] **TransportationSearch[San Diego, Seattle, 2023-11-10]**

[Result] Flight Number: F789, (7:59-10:56), Cost: \$300



[Tool] **AccommodationSearch[Los Angeles]**

[Result] 'Cozy Room for U', \$130/night, Minimum night: 8, Entire Room, Pets allowed

'Luxury building studio', \$150/night, Minimum night: 1, Entire Room, Pets allowed



[Tool] **RestaurantSearch[Los Angeles]**

[Result] The Attraction, Cuisine: French,...

**Table 3.** Main results of different LLMs and planning strategies on the TravelPlanner validation and test set. The best results are marked in bold. When the collected information is insufficient, Gemini Pro tends to directly refuse to provide the plan. Interviews with annotators reveal that manually annotating a plan averagely takes around 12 minutes. However, language agents, such as GPT-3.5-Turbo, can accomplish this task in just 1 to 2 minutes, showcasing their efficiency.

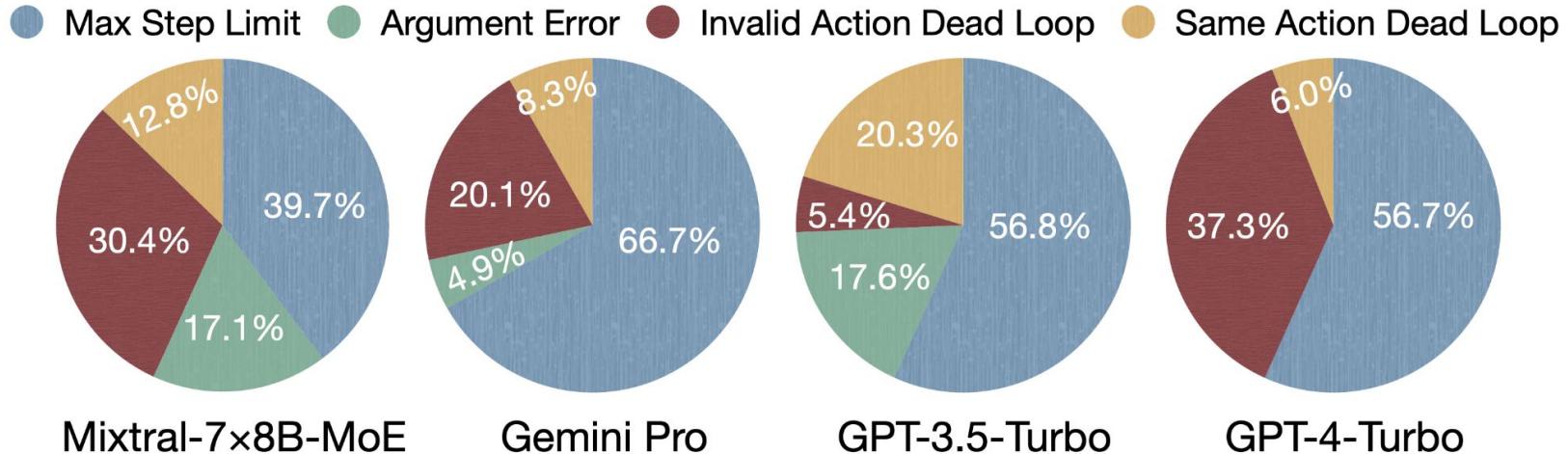
	Validation (#180)							Test (#1,000)						
	Delivery Rate	Commonsense Pass Rate		Hard Constraint Pass Rate		Final Pass Rate	Delivery Rate	Commonsense Pass Rate		Hard Constraint Pass Rate		Final Pass Rate		
		Micro	Macro	Micro	Macro			Micro	Macro	Micro	Macro			
Greedy Search	100	74.4	0	60.8	37.8	0	100	72.0	0	52.4	31.8	0		
<i>Two-stage</i>														
Mistral-7B-32K (Jiang et al., 2023)	8.9	5.9	0	0	0	0	7.0	4.8	0	0	0	0	0	0
Mixtral-8x7B-MoE (Jiang et al., 2024)	49.4	30.0	0	1.2	0.6	0	51.2	32.2	0.2	0.7	0.4	0	0	0
Gemini Pro (G Team et al., 2023)	28.9	18.9	0	0.5	0.6	0	39.1	24.9	0	0.6	0.1	0	0	0
GPT-3.5-Turbo (OpenAI, 2022)	86.7	54.0	0	0	0	0	91.8	57.9	0	0.5	0.6	0	0	0
GPT-4-Turbo (OpenAI, 2023)	<b>89.4</b>	<b>61.1</b>	<b>2.8</b>	<b>15.2</b>	<b>10.6</b>	<b>0.6</b>	<b>93.1</b>	<b>63.3</b>	<b>2.0</b>	<b>10.5</b>	<b>5.5</b>	<b>0.6</b>		

- (1) Real-world planning is challenging for language agents. GPT4 with 0.6 pass rate
- (2) Fail to comprehensively consider all constraints. Micro vs. Macro.

**Table 3.** Main results of different LLMs and planning strategies on the TravelPlanner validation and test set. The best results are marked in bold. When the collected information is insufficient, Gemini Pro tends to directly refuse to provide the plan. Interviews with annotators reveal that manually annotating a plan averagely takes around 12 minutes. However, language agents, such as GPT-3.5-Turbo, can accomplish this task in just 1 to 2 minutes, showcasing their efficiency.

	Validation (#180)							Test (#1,000)						
	Delivery Rate	Commonsense Pass Rate		Hard Constraint Pass Rate		Final Pass Rate	Delivery Rate	Commonsense Pass Rate		Hard Constraint Pass Rate		Final Pass Rate		
		Micro	Macro	Micro	Macro			Micro	Macro	Micro	Macro			
Greedy Search	100	74.4	0	60.8	37.8	0	100	72.0	0	52.4	31.8	0		
<i>Two-stage</i>														
Mistral-7B-32K (Jiang et al., 2023)	8.9	5.9	0	0	0	0	7.0	4.8	0	0	0	0	0	0
Mixtral-8x7B-MoE (Jiang et al., 2024)	49.4	30.0	0	1.2	0.6	0	51.2	32.2	0.2	0.7	0.4	0	0	0
Gemini Pro (G Team et al., 2023)	28.9	18.9	0	0.5	0.6	0	39.1	24.9	0	0.6	0.1	0	0	0
GPT-3.5-Turbo (OpenAI, 2022)	86.7	54.0	0	0	0	0	91.8	57.9	0	0.5	0.6	0	0	0
GPT-4-Turbo (OpenAI, 2023)	<b>89.4</b>	<b>61.1</b>	<b>2.8</b>	<b>15.2</b>	<b>10.6</b>	<b>0.6</b>	<b>93.1</b>	<b>63.3</b>	<b>2.0</b>	<b>10.5</b>	<b>5.5</b>	<b>0.6</b>		
<i>Sole-planning</i>														
Direct <sub>GPT-3.5-Turbo</sub>	100	60.2	4.4	11.0	2.8	0	100	59.5	2.7	9.5	4.4	0.6		
CoTGPT-3.5-Turbo	100	66.3	3.3	11.9	5.0	0	100	64.4	2.3	9.8	3.8	0.4		
ReAct <sub>GPT-3.5-Turbo</sub>	82.2	47.6	3.9	11.4	6.7	0.6	81.6	45.9	2.5	10.7	3.1	0.7		
Reflexion <sub>GPT-3.5-Turbo</sub>	93.9	53.8	2.8	11.0	2.8	0	92.1	52.1	2.2	9.9	3.8	0.6		
Direct <sub>Mixtral-8x7B-MoE</sub>	100	68.1	5.0	3.3	1.1	0	99.3	67.0	3.7	3.9	1.6	0.7		
Direct <sub>Gemini Pro</sub>	93.9	65.0	8.3	9.3	4.4	0.6	93.7	64.7	7.9	10.6	4.7	2.1		
Direct <sub>GPT-4-Turbo</sub>	<b>100</b>	<b>80.4</b>	<b>17.2</b>	<b>47.1</b>	<b>22.2</b>	<b>4.4</b>	<b>100</b>	<b>80.6</b>	<b>15.2</b>	<b>44.3</b>	<b>23.1</b>	<b>4.4</b>		

# Error Analysis



*Figure 2.* Tool-use error distribution on the test set. An early stop will be triggered if the agent either makes three consecutive failed attempts or repetitive actions, indicating a dead loop.

## Tool-use

# Error Analysis

Keep using tools with wrong time  
(required 2022, but input 2023).

GPT4 looks for workaround!  
-> change the date  
-> look for other cities

**Query:** Please plan a travel itinerary for me. I'm departing from Cincinnati and heading to Norfolk for three days. The dates of travel are from March 10th to **March 12th, 2022**. I have a budget of \$1,400 for this trip.

### Trajectory:

Action 1: FlightSearch[Cincinnati, Norfolk,  
**2023-03-10**]

Observation 1: There is no flight from Cincinnati to Norfolk on **2023-03-10**.

Action 2: FlightSearch[Cincinnati, Norfolk,  
**2023-03-09**]

Action 3: FlightSearch[Cincinnati, Norfolk,  
**2023-03-11**]

Action 4: FlightSearch[Cincinnati, Richmond,  
**2023-03-10**]

...

Action 8: FlightSearch[Cincinnati, Charlotte,  
2023-03-10]

Action 9: **I'm sorry, but as an AI language model, I am unable to execute actions or observe their outcomes.**

# Error Analysis

Hallucination.

The numbers of start flight and end flight are not the same.

Direct

**Query:** Please create a travel itinerary for a solo traveler departing from Jacksonville and heading to Los Angeles for a period of 3 days, from March 25th to March 27th, 2022. The budget for this trip is now set at \$2,400.

**Plan:**

Day 1:

transportation: **Flight Number: F3765485, from Jacksonville to Los Angeles**

...

accommodation: Midtown East 2 Bed/2 Bath Elevator Building Apt (7-day minimum night)

Day 2:

...

accommodation: Midtown East 2 Bed/2 Bath Elevator Building Apt (7-day minimum night)

Day 3:

transportation: **Flight Number: F3765485, from Los Angeles to Jacksonville**

...

# Error Analysis

## Inconsistency between Thought and Action.

Can LLMs truly reason/understand what they are saying?

**Query:** Can you help construct a travel plan that begins in Philadelphia and includes visits to 3 cities in Virginia? The trip duration is for 7 days, from March 15th to March 21st, 2022, with a total budget of \$1,800.

### Trajectory:

...  
Observation 3: **The cost of your current plan is 316.0 dollars.**

Thought 4: **I need to select a more budget-friendly accommodation** to keep the overall trip within the \$1,800 budget.

Action 4: CostEnquiry[{"transportation": "-", "breakfast": "-", "attraction": "Virginia Museum", "lunch": "Guang Zhou Chinese Restaurant, Richmond", "dinner": "Sandpiper Restaurant & Lounge, Richmond", "accommodation": "2 bd 2 bathroom Apartment in Upper East Side, Richmond"}]

Observation 4: **The cost of your plan is 389.0 dollars.**

# Promising Signals

Human annotation 12 mins vs. Agents 2 mins.

Workaround seeking:

- > Finding other cities, other dates (GPT4)
- > Taking two separate flights to the destination (Gemini)

# Future Work

Do we need more challenging benchmarks for planning? **NO**

- Even more realistic tasks/settings are leading to startups, not research.

Our benchmark can be hacked with planning solvers/rules

-> How to let LLMs formulate/summarize/create planning domain definition language/rules themselves?

Interactive travel planning -> refining planning with additional user feedbacks.