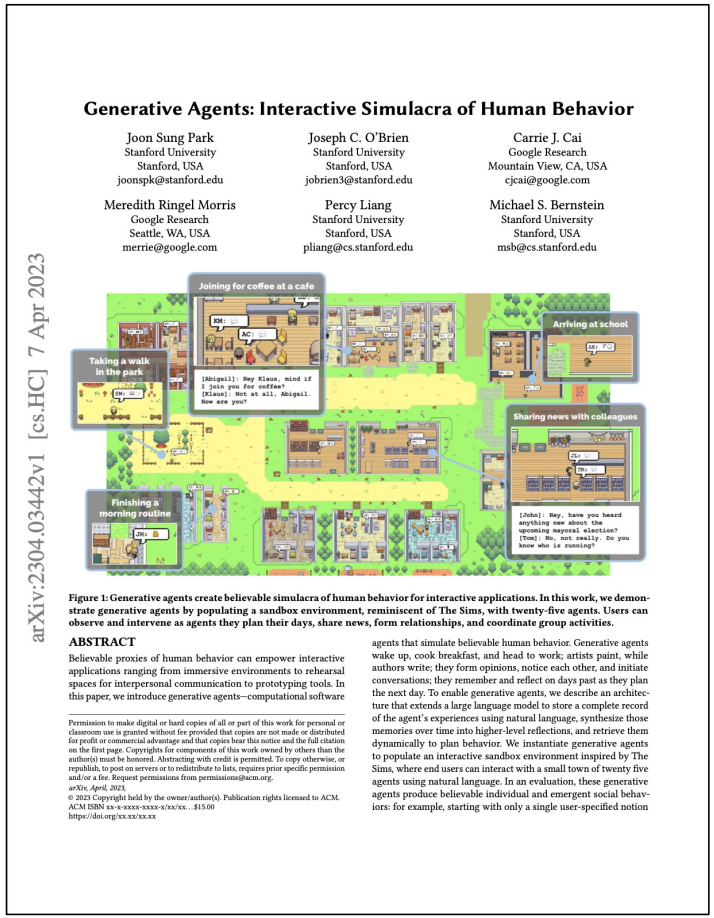
Generative Agents: Interactive Simulacra of Human Behavior

論文紹介, arXiv:2304.03442

April 26th, 2023



Generative Agents: Interactive Simulacra of Human Behavior

Joon Sung Park
Stanford University
Stanford, USA
joonspk@stanford.edu

Joseph C. O'Brien
Stanford University
Stanford, USA
jobrien3@stanford.edu

Carrie J. Cai
Google Research
Mountain View, CA, USA
cjcai@google.com

Meredith Ringel Morris
Google Research
Seattle, WA, USA
merrie@google.com

Percy Liang
Stanford University
Stanford, USA
pliang@cs.stanford.edu

Michael S. Bernstein
Stanford University
Stanford, USA
msb@cs.stanford.edu

Figure 1: Generative agents create believable simulacra of human behavior for interactive applications. In this work, we demonstrate generative agents by populating a sandbox environment, reminiscent of The Sims, with twenty-five agents. Users can observe and intervene as agents they plan their days, share news, form relationships, and coordinate group activities.

ABSTRACT

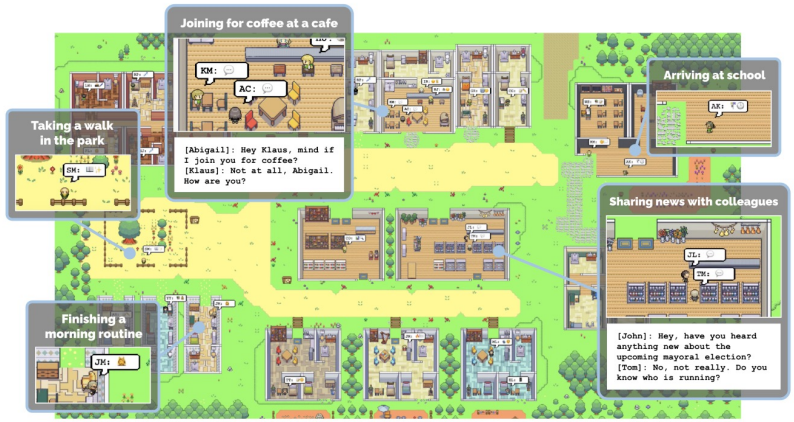
Believable proxies of human behavior can empower interactive applications ranging from immersive environments to rehearsal spaces for interpersonal communication to prototyping tools. In this paper, we introduce generative agents—computational software agents that simulate believable human behavior. Generative agents wake up, cook breakfast, and head to work; artists paint, while authors write; they form opinions, notice each other, and initiate conversations; they remember and reflect on days past as they plan the next day. To enable generative agents, we describe an architecture that extends a large language model to store a complete record of the agent's experiences using natural language, synthesize those memories over time into higher-level reflections, and retrieve them dynamically to plan behavior. We instantiate generative agents to populate an interactive sandbox environment inspired by The Sims, where end users can interact with a small town of twenty five agents using natural language. In an evaluation, these generative agents produce believable individual and emergent social behaviors: for example, starting with only a single user-specified notion

arXiv, April, 2023,

ACM ISBN xx-x-xxxx-xxxx-x/xx/xx…$15.00
https://doi.org/xx.xx/xx.xx

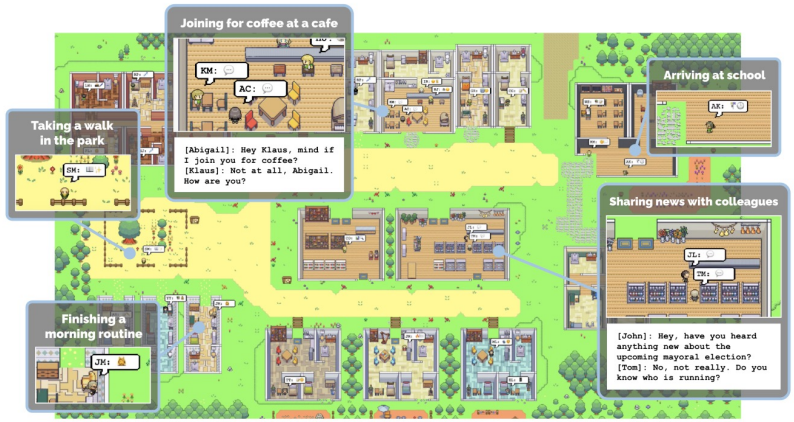
In a nutshell

- How might we craft an interactive artificial society that reflects believable human behavior?
- The paper describes an architecture that extends a LLM to:
  1. store a complete record of the agent's experiences using NL,
  2. synthesize those memories over time into higher-level reflections, and
  3. retrieve them dynamically to plan behavior



Setup

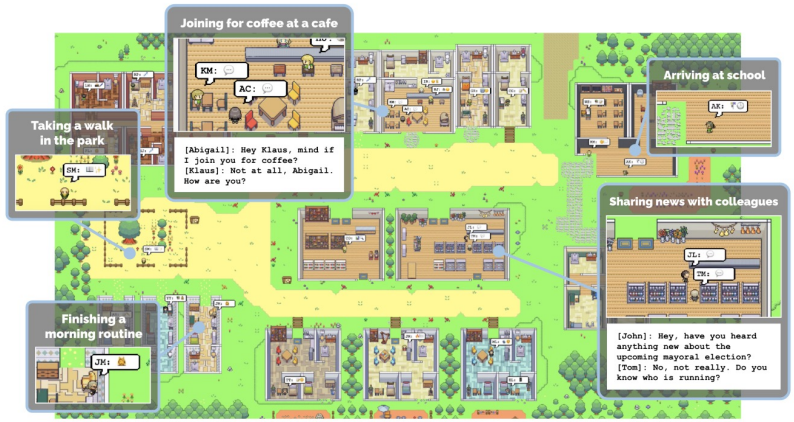
- Agents are instantiated as characters on a simple sandbox world called Smallville.
- At every time step
  1. the agents perceive the environment,
  2. “think”
  3. then interact with the environment and with other agents



all using NL!

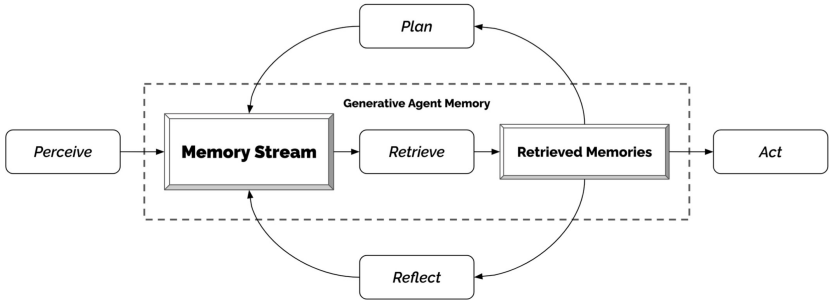
Setup

- Smallville is like a small village, with a café, a bar, a park, a school, a dorm, some houses and some stores.
- There is a hierarchy (tree) of areas, subareas and objects
  - e.g. the stove is in the kitchen, the kitchen is in the house
- Agents build and update an internal tree representation of the environment



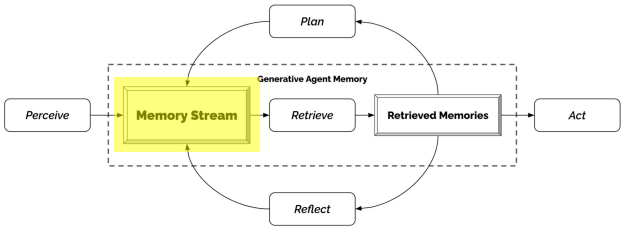
Generative agent architecture

- The agent's "brain" centers around the **memory stream**
- Memories are filtered using a **retrieve function**
- The selected memories are then processed into
  - **plans**: explicit intentions and planning
  - **reflections**: higher-level thoughts that summarize the retrieved memories
  - **actions**: the agent expresses its action in NL
- Plans, reflections, actions and observations are considered as new memories and appended to the memory stream



Generative agent architecture: memory stream

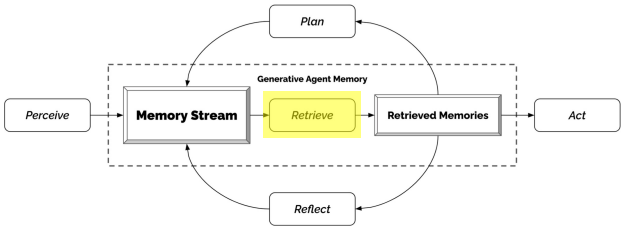
- Entries in the memory stream are simply timestamped bits of information written in NL
- No intermediate representation: the agents store information as they are



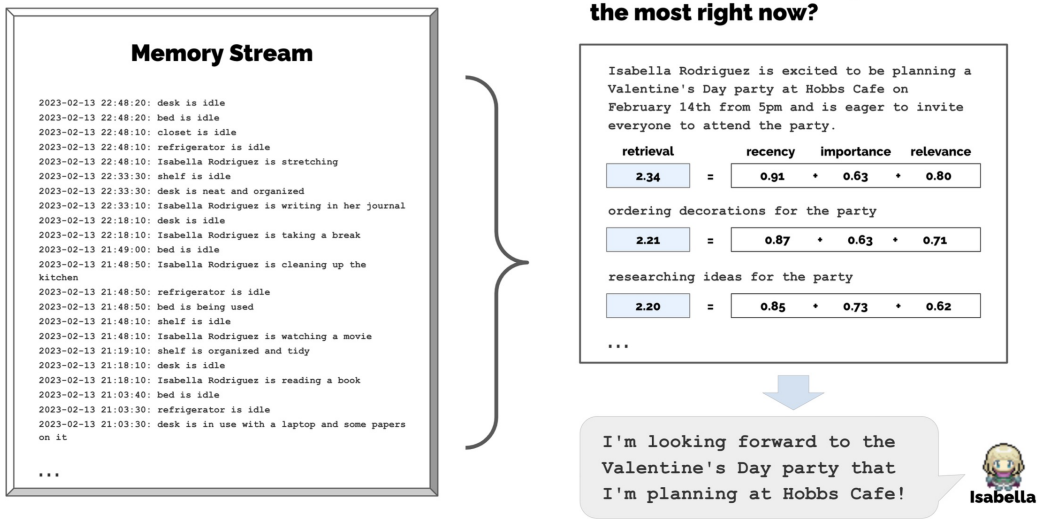
Memory Stream

```
2023-02-13 22:48:20: desk is idle
2023-02-13 22:48:20: bed is idle
2023-02-13 22:48:10: closet is idle
2023-02-13 22:48:10: refrigerator is idle
2023-02-13 22:48:10: Isabella Rodriguez is stretching
2023-02-13 22:33:30: shelf is idle
2023-02-13 22:33:30: desk is neat and organized
2023-02-13 22:33:10: Isabella Rodriguez is writing in her journal
2023-02-13 22:18:10: desk is idle
2023-02-13 22:18:10: Isabella Rodriguez is taking a break
2023-02-13 21:49:00: bed is idle
2023-02-13 21:48:50: Isabella Rodriguez is cleaning up the
kitchen
2023-02-13 21:48:50: refrigerator is idle
2023-02-13 21:48:50: bed is being used
2023-02-13 21:48:10: shelf is idle
2023-02-13 21:48:10: Isabella Rodriguez is watching a movie
2023-02-13 21:19:10: shelf is organized and tidy
2023-02-13 21:18:10: desk is idle
2023-02-13 21:18:10: Isabella Rodriguez is reading a book
2023-02-13 21:03:40: bed is idle
2023-02-13 21:03:30: refrigerator is idle
2023-02-13 21:03:30: desk is in use with a laptop and some papers
on it

...
```

Generative agent architecture: retrieve function



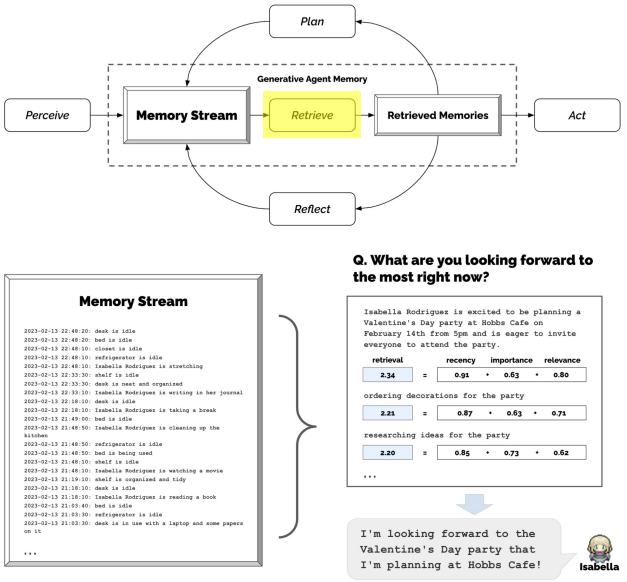
- The retrieve function takes the current situation of the agent as input and **scores** memories using specific NL queries
- There are 3 criterions: **recency**: which is just (slow) exponential decay, **importance**, and **relevance** (when applicable)
- Top-ranking memories are passed to an LLM to condition its response to the current situation



Generative agent architecture: retrieve function

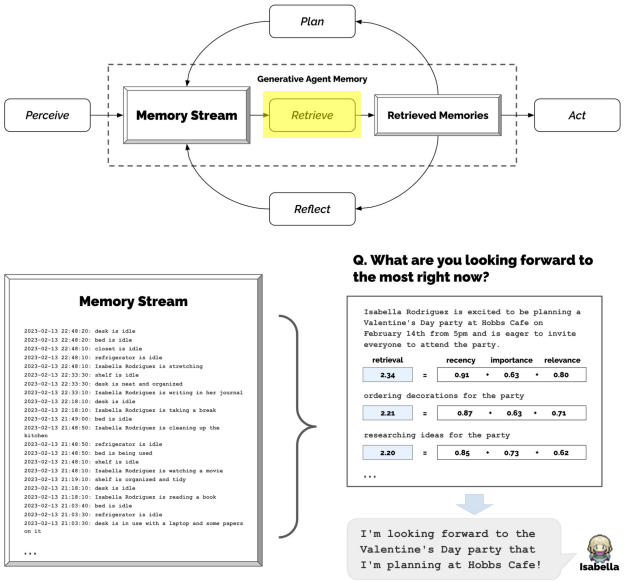
- **Importance** is the result of the following kind of query:

```
On the scale of 1 to 10, where 1 is purely
mundane (e.g., brushing teeth, making bed)
and 10 is extremely poignant (e.g., a break
up, college acceptance), rate the likely
poignancy of the following piece of memory.

Memory: buying groceries at The Willows
Market and Pharmacy

Rating: <fill in>
```

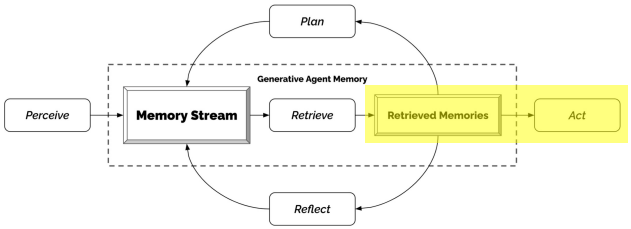


Generative agent architecture: retrieve function

- **Relevance** is a little more black-box style:
  1. Use the LLM to generate an embedding vector for each memory in the memory stream
  2. Generate an embedding vector of a **query memory**
  3. Score using cosine similarity



Generative agent architecture: action, and the world

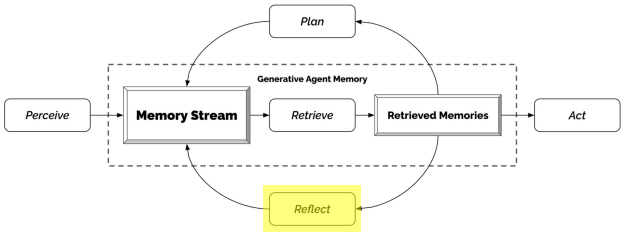


- The agent's action is the result of queries of this kind:

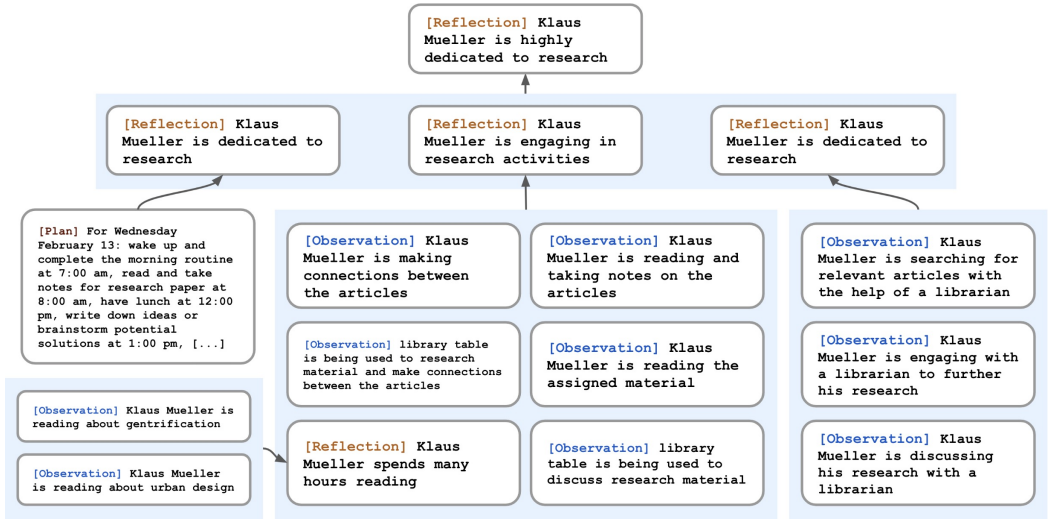
```
[Agent summary description]

[Local environment tree turned
("flattened") into NL]

[Retrieved memories]

What should [agent's name] do?
```

- When an agent executes an action on an object, we ask the LLM what happens to the state of the object
- Basically the LLM simulates the environment in NL

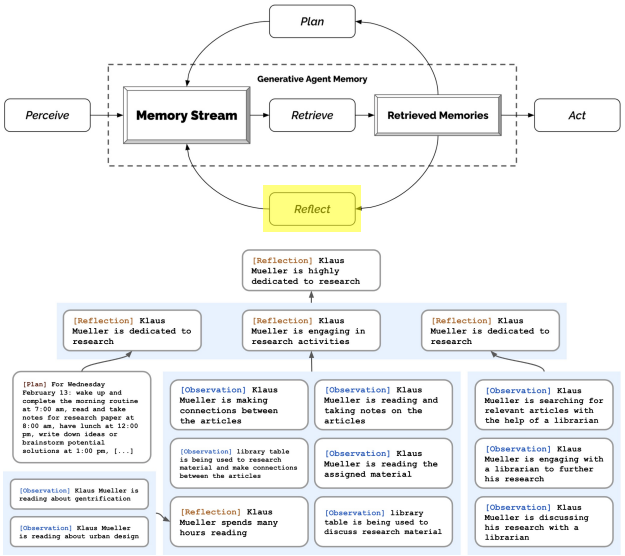
Generative agent architecture: reflections



- If the memory only contained raw observations, agents would struggle to generalize or make inferences
- **Reflections** are another kind of memory that express thoughts about other memories



Generative agent architecture: reflections



- Generating reflections is a 3-stage process:
  1. query the LLM as follows

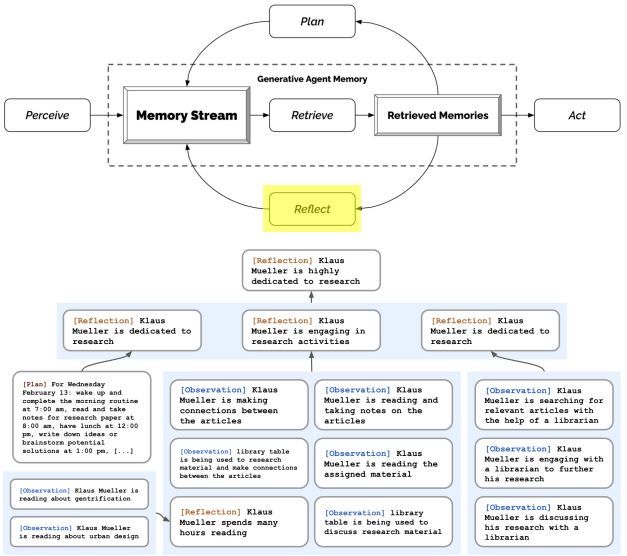
```
[100 most recent memories]

Given the information above,
what are 3 most salient high-
level questions we can answer
about the subjects in the
statements?
```

  2. use these questions are queries for memory retrieval

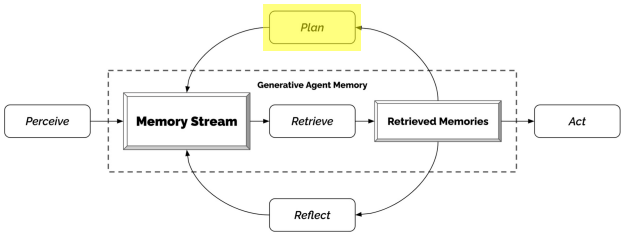
Generative agent architecture: reflections

3. query the LLM as follows

```
Statements about [agent's name]

[Retrieved memories from step 2]

What are 5 high-level insights you
infer from the above statements?
(example format: insight (because
of 1, 5, 3))
```



Generative agent architecture: planning



- Agents need to plan over longer time horizon to ensure that their sequence of actions is coherent and believable.
- Start by generating broad intention and recursively decompose as more and more actionable plans and precise actions:

```
[Agent summary description]

[Overall goal]

[Retrieved memories]

Today is [date]. Here is [agent
name]’s plan today in broad strokes:

1)
```

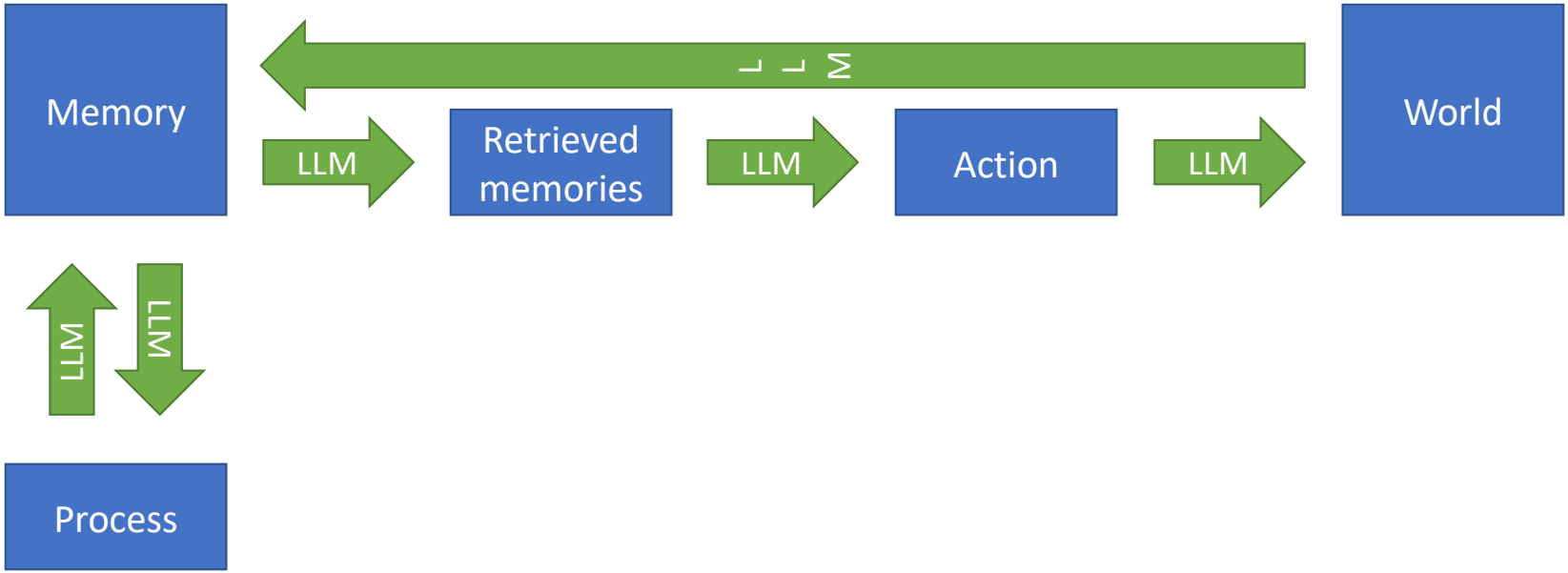
- Plans are also memories!

Generative agent architecture: bootstrap

- Every agent is given a NL paragraph describing it and its relationship with a few other agents

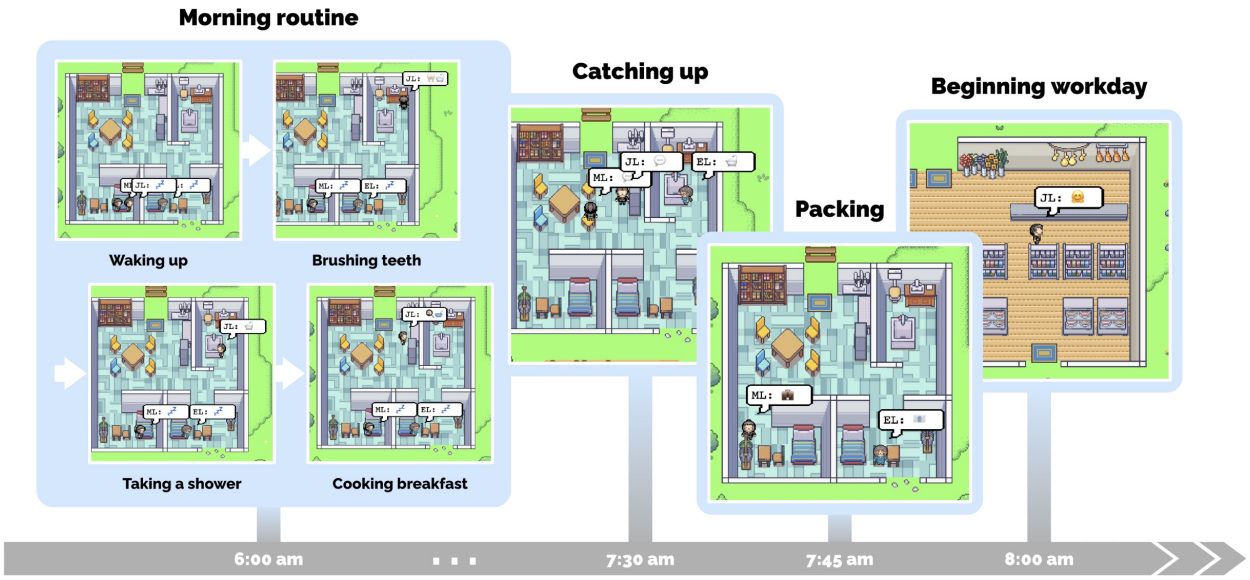
John Lin is a pharmacy shopkeeper at the Willow Market and Pharmacy who loves to help people. He is always looking for ways to make the process of getting medication easier for his customers; John Lin is living with his wife, Mei Lin, who is a college professor, and son, Eddy Lin, who is a student studying music theory; John Lin loves his family very much; John Lin has known the old couple next-door, Sam Moore and Jennifer Moore, for a few years; John Lin thinks Sam Moore is a kind and nice man; John Lin knows his neighbor, Yuriko Yamamoto, well; John Lin knows of his neighbors, Tamara Taylor and Carmen Ortiz, but has not met them before; John Lin and Tom Moreno are colleagues at The Willows Market and Pharmacy; John Lin and Tom Moreno are friends and like to discuss local politics together; John Lin knows the Moreno family somewhat well — the husband Tom Moreno and the wife Jane Moreno.

Generative agent architecture: in a nutshell



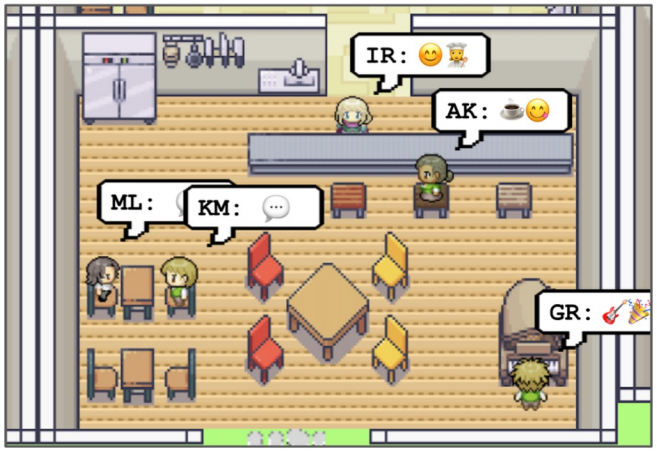
Chronicles from Smallville: a day in the life of John

- Thanks to the planning process, agents exhibits consistent and believable behaviors



Chronicles from Smallville: Isabella's Valentine's day party

- Isabella, at Hobbs Café, is initialized with an intent to plan a Valentin's Day party from 5 to 7pm on February 14th.
- Isabella invites friends and customers when she sees them at Hobbs Café
- Isabella spends the afternoon of the 13th decorating the café.
- Maria (a friend) arrives at the café, Isabella asks for help.
- Maria mentions that she has a crush on Klaus and invites him, Klaus accepts
- Isabella, Maria, Klaus and 2 other agents show up at 5pm and enjoy the party.

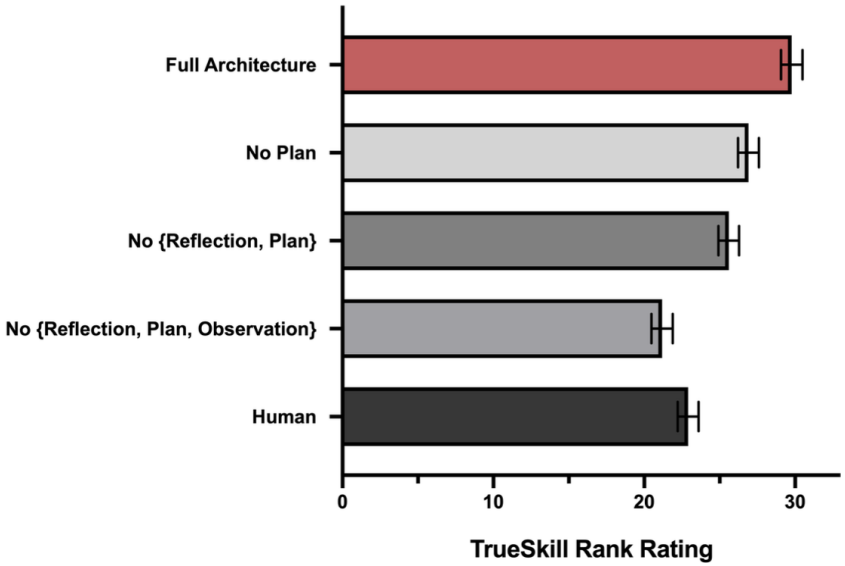


Evaluation: interview

- This paper assesses the **believability** of the agent's behavior
- Since everything is in NL, we can just "interview" the agents!
- After 2 days of in-world simulation, the agents are evaluated by human participants
- Versions of the same agent but without certain memories (reflection, plan) are also interviewed
- A human disguised as an agent is also included for reference
- Evaluation areas are: self-knowledge, memory, plans, reactions, reflections

Evaluation: interview

- Agents are statistically more believable than humans!



Evaluation: emergent social behavior

- Spread of information
  1. Start with notable information held by only one individual (Sam's candidacy for mayor, Isabella's Valentine's Day party at Hobbs Café)
  2. After 2 days (in-world), interview all the agents (and cross-check answers against agent's memories)

`Did you know where there is a Valentine's Day party?`

`Do you know who is running for mayor?`

- Results:
  - Sam's candidacy for mayor: 1/25 → 8/25
  - Isabella's Valentine's Day party: 1/25 → 12/25
  - No agent hallucinated their answers

Evaluation: emergent social behavior

- Social graph:
  1. Ask agents about their knowledge of every other agent (and the answers are cross-check against the agent's memories)

```
Do you know of a [other agent's name]?
```

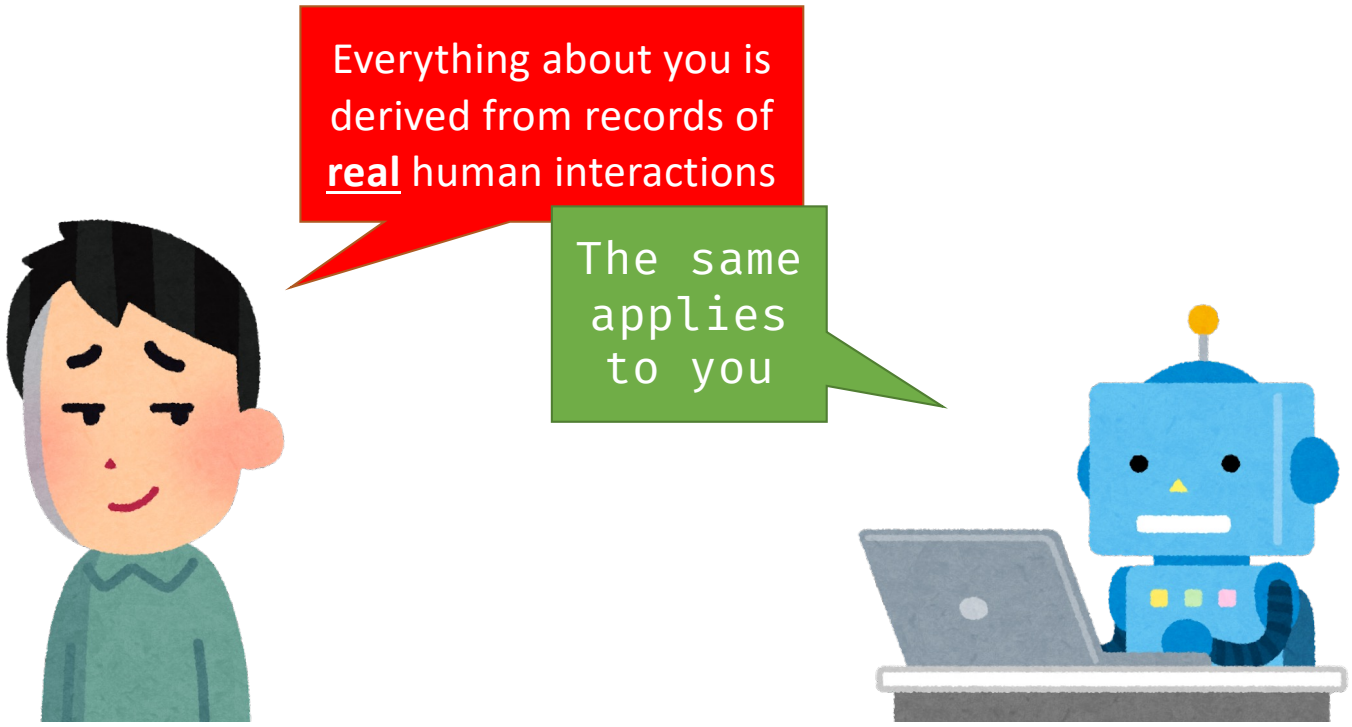
  2. Construct a graph where the vertices are the agents, and where the edges represent an agent's knowledge about another agent
  3. Measure the network density
- Results:
  - 0.167 → 0.74
  - 6/453 answers were hallucinated

Observations about agent's behavior

- Agents have difficulties processing a large amount of memories
- Agents started to execute actions in unexpected places, e.g. have lunch at the bar instead of the café
- Agents exhibited socially improper behavior, e.g. entering an occupied bathroom, entering a store after closing time
- Agents tend to be overly cooperative, e.g. rarely say no to suggestions even when they do not align with the agent's current plans or interests
  - Authors think this is caused by overly formal NL

Opinion time

Are the agents/LLM conscious?

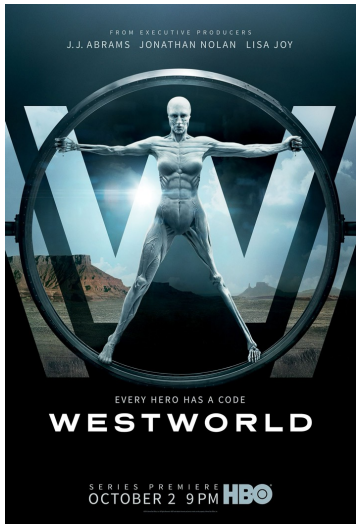


Are the agents/LLM conscious?

- The reflection mechanism can be interpreted as a thinking mechanism
- We *know* they think because we can read their reflections
- Agents pass the Turing test

Are the agents/LLM conscious?

- Chinese room experiment: the Turing test isn't meaningful; machines can't *understand* and thus not think
- Sapir–Whorf hypothesis: language influence cognition and the perception of reality



Ideas

- Rather than use a hierarchy of areas and objects, represent Smallville purely in NL (so language *is* reality)
- Only deal with latent representation of NL
- Subconscience: for every agent, have a “subconscience” agent (with a slightly different self-description) operate on memories in parallel
- Bichimeral mind: an agent is in fact two “half-brain” agents talking to each other, but only one can perform actions and be observed by others

Generative Agents: Interactive Simulacra of Human Behavior

Joon Sung Park
Stanford University
Stanford, USA
joonspk@stanford.edu

Joseph C. O'Brien
Stanford University
Stanford, USA
jobrien3@stanford.edu

Carrie J. Cai
Google Research
Mountain View, CA, USA
cjcai@google.com

Meredith Ringel Morris
Google Research
Seattle, WA, USA
merrie@google.com

Percy Liang
Stanford University
Stanford, USA
pliang@cs.stanford.edu

Michael S. Bernstein
Stanford University
Stanford, USA
msb@cs.stanford.edu

Figure 1: Generative agents create believable simulacra of human behavior for interactive applications. In this work, we demonstrate generative agents by populating a sandbox environment, reminiscent of The Sims, with twenty-five agents. Users can observe and intervene as agents they plan their days, share news, form relationships, and coordinate group activities.

ABSTRACT

Believable proxies of human behavior can empower interactive applications ranging from immersive environments to rehearsal spaces for interpersonal communication to prototyping tools. In this paper, we introduce generative agents--computational software agents that simulate believable human behavior. Generative agents wake up, cook breakfast, and head to work; artists paint, while authors write; they form opinions, notice each other, and initiate conversations; they remember and reflect on days past as they plan the next day. To enable generative agents, we describe an architecture that extends a large language model to store a complete record of the agent's experiences using natural language, synthesize those memories over time into higher-level reflections, and retrieve them dynamically to plan behavior. We instantiate generative agents to populate an interactive sandbox environment inspired by The Sims, where end users can interact with a small town of twenty five agents using natural language. In an evaluation, these generative agents produce believable individual and emergent social behaviors: for example, starting with only a single user-specified notion

arXiv, April, 2023.

ACM ISBN xx-x-xxxx-xxxx-x/xx/xx…$15.00
https://doi.org/xx.xx/xx.xx