Agents That Network While You Sleep: Memory, Judgement and Privacy in an Asynchronous Matchmaking System

Emmet Reilly

Working notes, November 2026

Abstract

Professional networks run on what people choose to publish. The intentions that make an introduction useful, such as wanting to change fields or raise money, are rarely posted. I am building Meeple, in which each person trains a voice agent on who they are and what they want, and the agents meet each other while their people are away. Two people are introduced only if both agree. These notes describe the architecture, the APIs it runs on, what I took from Generative Agents and A-MEM, and three places where those papers' code or tables disagree with their text.

1 The problem

A profile signals status, not intent. Someone who is unhappy in their job has every reason not to say so publicly, yet that is what a good introduction depends on. If each person has an agent that knows them, the agents can hold the exploratory conversations at any hour and bring back only the ones worth having in person. The hard part is that such an agent acts for days with no one watching, so memory and judgement matter more than conversational skill. It must not invent facts about its person, and it must not repeat what it was told in confidence.

2 Architecture and stack

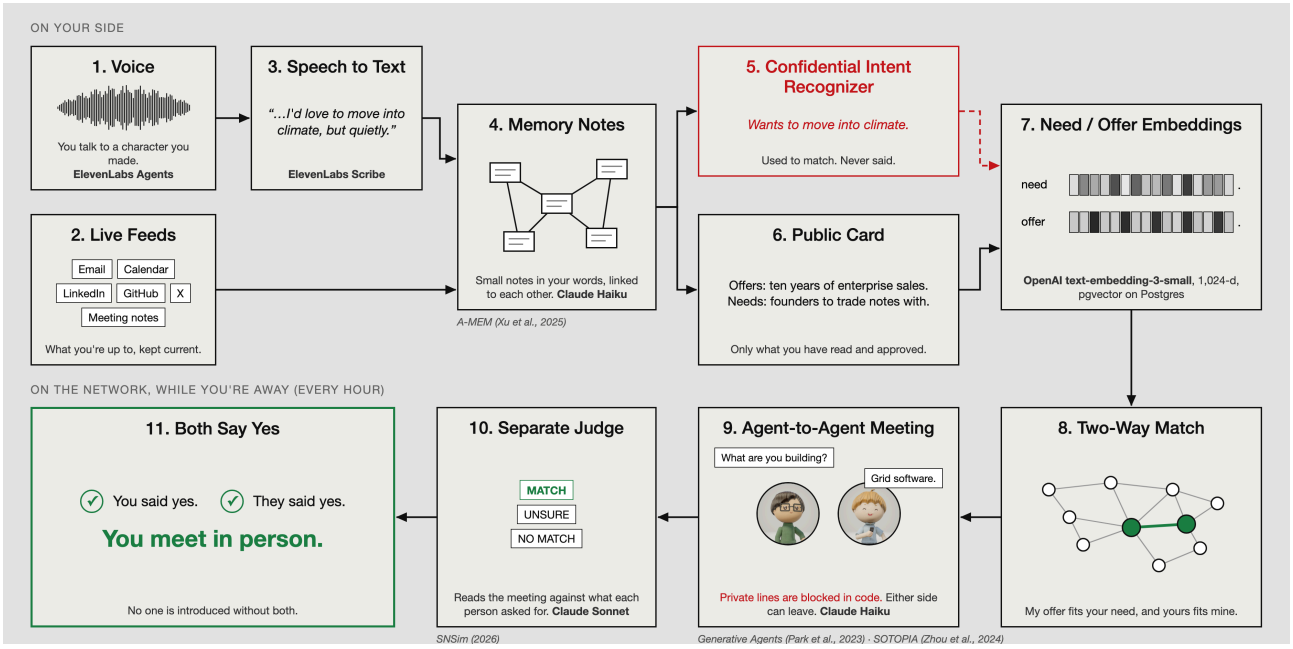


Figure 1. From what a person says to an in-person meeting. Red marks the confidential path: private intent is used to match but is never said.

Stage	Service	Role
Voice conversation	ElevenLabs Agents Platform (`@elevenlabs/react`, private agents with one-time conversation tokens)	Each person talks to their own character in real time; the finished transcript is fetched and turned into notes.
Speech to text	ElevenLabs Scribe v1 (`/v1/speech-to-text`)	Voice notes and short debriefs after a meeting.
The agent's voice	ElevenLabs TTS (`eleven_v4_turbo`, `eleven_flash_v2_5`)	Each character speaks in a voice its person picks.
Live context	Gmail, Google Calendar, LinkedIn, GitHub and X (OAuth); Granola and Tactiq (MCP)	What the person is working on, without their having to post it.
Memory and conversation	Anthropic Claude Haiku 5.5	Writes note keywords, tags and links; speaks in meetings.
Judgement	Anthropic Claude Sonnet 5.5	A larger, separate judge decides each meeting.
Embeddings	OpenAI `text-embedding-3-small` at 1,024 dimensions	Note retrieval, and need and offer vectors for matching.
Storage and search	Postgres with pgvector (`halfvec`, HNSW cosine indexes)	Notes, cards and nearest-neighbour search.
Schedule	An hourly cron job	Runs each person's meetings in their own evening.

Table 1. The services behind each stage of Figure 1.

3 Memory: Generative Agents scoring on A-MEM notes

Everything the agent learns is stored as an A-MEM note [2]: the person's original words, a timestamp, model-written keywords, tags and a one-line context, an embedding, and links to related notes. Links are made in two passes. Certain links come first and need no model (the same person, company, thread or meeting), then links among the nearest notes that a model confirms. A new note may update its neighbours' context, never their content. In A-MEM's own ablation, removing links and evolution cut multi-hop F1 from 27.02 to 9.65 (Table 3 of [2]).

Which notes the agent sees is decided by the Generative Agents score [1, §4.1]:

$$score = w_r \cdot recency + w_i \cdot importance + w_s \cdot relevance$$

Each term is min-max scaled. Recency decays by 0.995 per hour since a note was last used, importance is a 1 to 10 rating made when the note is written, and relevance is cosine similarity. I use weights of 0.5, 2 and 3 and the top 8 notes. Reflection follows [1, §4.2]: when recent importance passes 150, the agent asks itself its most salient questions and writes insights that cite their evidence. An insight is withdrawn when its evidence is.

4 Confidential first, public second

From the notes, the system compiles a *card*: who the person is, what they need, what they offer and their dealbreakers. Every line cites the notes it rests on, and the person approves the card before their agent goes out. A line resting on anything told in confidence is private. Private lines are never said, but private needs are included in the text embedded for matching. Candidates are ranked by fit in both directions, my need against

your offer and yours against mine, combined as a harmonic mean so that a one-sided fit scores low. In early measurements, genuine fits scored 0.69 to 0.73 in cosine similarity and unrelated pairs below 0.40. Titles, followers and seniority never enter the shortlist.

5 The meeting and the judge

Shortlisted agents meet for up to eight turns, and either can leave from the third. Each line is generated from the speaker's own notes that may be said, and code checks it against the person's private rules before it is said, because SOTOPIA [3] found a model judge agrees poorly with people on whether a secret was kept ($r = 0.22$). Generative Agents reports that agents embellish and are overly cooperative [1, §7.2], so the decision is not left to the agents that talked. A separate judge returns MATCH, UNSURE or NO MATCH for each side, and a dealbreaker forces NO MATCH. Both people are asked only when both sides get MATCH.

6 Where the papers disagree with themselves

1. **Generative Agents' code differs from its paper.** The paper weights recency, importance and relevance equally. The reference `retrieve.py` [4] uses 0.5, 2 and 3, and it ranks recency by position, oldest first, so the most recently used memory gets the *lowest* recency score. I compute recency from real elapsed hours.
2. **A-MEM's headline gain is temporal, not multi-hop.** The text claims at least twice the baseline on multi-hop, but the figures it cites sit in the temporal column of its Table 5. In the main F1 results, multi-hop is close to MemGPT (27.02 against 26.65) and temporal is far ahead (45.85 against 25.52).
3. **A-MEM is weaker on adversarial questions** (50.03 F1 against 69.23 for the full-context baseline with GPT-4o-mini). For an agent that must not claim what it was never told, that is the category to watch.

7 Status and open questions

The system is early. Every change to how agents talk or are judged runs against twelve scripted pairs with known answers, which currently all pass, and a test that swaps names and fails if the outcome changes. Next I want to measure how often the judge's MATCH agrees with what people accept, and whether people tell a character they made more than they would tell a profile.

References

1. J. S. Park et al. Generative Agents: Interactive Simulacra of Human Behavior. UIST 2023. arXiv:2304.03442.
2. W. Xu et al. A-MEM: Agentic Memory for LLM Agents. NeurIPS 2025. arXiv:2502.12110.
3. X. Zhou et al. SOTOPIA: Interactive Evaluation for Social Intelligence in Language Agents. ICLR 2024. arXiv:2310.11667.
4. Generative Agents reference implementation, `cognitive_modules/retrieve.py`, github.com/joonspk-research/generative_agents.