# A memory-corrected conversational mirror for human-like personal dialogue

Alan N. Pham[1,2]

[1]AO Labs, United States

[2]Department of Mechanical Engineering, Worcester Polytechnic Institute, Worcester, MA 01609, USA

**Conversational agents can be fluent without feeling like a specific person. We introduce `talk.aolabs.io`, a deployed personal conversational mirror that treats human-like dialogue as an evidence-integration problem rather than a generic chatbot problem. The system separates long-term speaker priors, stable style instructions, retrieved utterance examples, corrected rewrite pairs, live session state, relationship context, and a short state summary before composing replies through a burst controller that preserves text-message rhythm. Training is owner-gated: public visitors can converse, but only the admin interface can convert generated replies into durable correction memory. Recent revisions add a low-choice review queue, one-box correction from public sessions, generation traces, batched embeddings, and admin-only voice checks over fixed prompts. The contribution is a source-bounded framework for evaluating personal dialogue at four levels: identity fidelity, interactional believability, correction efficiency, and disclosure-calibrated trust.**

# 1 Introduction

The question of whether a machine can converse like a human is older than contemporary language models. Turing framed machine intelligence through an imitation game, and early systems such as ELIZA showed that even shallow pattern matching could elicit a strong feeling of being understood when users supplied missing social meaning themselves (*1*, *2*). Modern large language models change the technical baseline: they can generate fluent, context-sensitive language across almost any topic (*3*, *4*). Yet the old problem remains in sharper form. A fluent model can sound generally human while failing to sound like a particular human.

The problem is not only computational. Human-likeness in conversation is jointly produced by the speaker, the listener, the interface, and the social frame. People routinely apply social expectations to computers and conversational interfaces (*5*, *6*). They anthropomorphize non-human agents when the agent gives enough social cues, when the user wants social contact, or when the system appears behaviorally coherent (*7*). When human-likeness is close but wrong, the result can become uncanny or disappointing rather than intimate (*8*, *9*). For a personal conversational mirror, the failure case is therefore not just an incorrect answer. It is a rupture in the listener's model of the speaker.

Dialogue research suggests why this rupture is subtle. Conversation is not a sequence of isolated messages. Interlocutors align vocabulary, syntax, timing, and expectations over turns (*10*). Stable linguistic patterns also carry individual differences; function-word use, punctuation, sentence length, hedging, directness, and discourse markers can be more diagnostic of voice than topic alone (*11–13*). In social chat, a reply that is semantically reasonable but rhythmically wrong may feel less like the target speaker than a shorter, messier reply that preserves timing, stance, and implied continuity.

Prior computational work has addressed parts of this problem. Persona-based neural dialogue models condition generation on speaker traits or persona facts (*14*, *15*). Personality generation systems manipulate surface features associated with traits (*16*). Retrieval-augmented generation couples model output to external memories (*17*). Generative-agent architectures show how large language models can store experience, retrieve relevant memories, and synthesize behavior over time (*18*). These approaches are necessary but not sufficient for a personal mirror. A person is not a static persona card, a bag of memories, or a style transfer target. Personal conversational realism requires an architecture that keeps those layers separate enough to inspect and correct them.

Here we present talk.aolabs.io as a software system and research frame for personal conversational

mirroring. The implementation is deliberately narrow: it is not a general assistant, therapist, coach, or autonomous replacement for a person. It is a web-based conversational system designed to reply in Alan's compact texting voice, using a combination of long-term priors, extracted style profile, semantic retrieval, recent transcript state, situation labels, and owner-approved rewrites. The central claim is architectural: realistic personal dialogue emerges from the controlled combination of evidence layers, correction authority, and interaction constraints, not from asking a model to "be" someone in a single instruction.

The paper is organized around four levels. First, we describe the evidence stack that converts sparse personal material into a controllable speaker model. Second, we describe the interaction layer that makes replies believable to a user in a live conversation. Third, we describe the owner-gated training loop that prevents public visitors from steering the identity model while reducing the owner's correction burden. Fourth, we define an evaluation framework for measuring personal realism without hiding uncertainty, overstating identity, or rewarding deception.

## 2 Results

The results are organized around the deployed system rather than a completed human-subject trial. The current software establishes the architecture, instrumentation, access-control boundary, and correction loop required for a full empirical study (Figure 1). No human-subject performance numbers are claimed here.

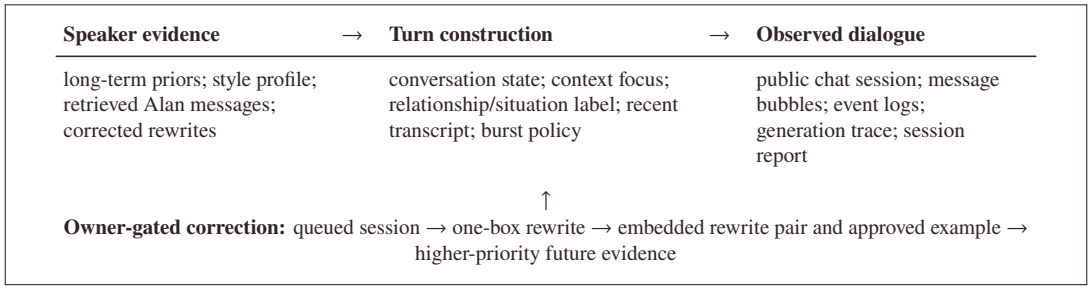


**Figure 1.** Architecture of the personal conversational mirror. The system separates durable speaker evidence, turn-level conversation state, generated message rhythm, public interaction traces, generation traces, and owner-only correction. This makes personal realism inspectable: a failure can be assigned to stale evidence, weak retrieval, lost state, wrong relationship framing, wrong response rhythm, or insufficient correction memory.

## 2.1 Inspectable evidence strata support personal voice

The talk.aolabs.io implementation treats personal voice as a stack of evidence strata rather than a single prompt. This design matters because different failures require different fixes. If the system forgets recent context, the state summary and transcript window are implicated. If it sounds too polished, the style profile and response policy are implicated. If it fails on a recurring situation, corrected rewrite pairs should override weaker examples. Table 1 summarizes the main layers.

This separation turns style into an auditable mechanism. In the deployed code, semantic retrieval is not driven only by the newest user message. The system forms weighted retrieval queries from the conversation focus, state summary, and user message, then searches both stored Alan messages and corrected rewrite pairs. The current implementation batches these embedding calls into a single request for the main turn and records generation traces for persisted replies. This makes the current turn a product of local situation and durable evidence, matching the psycholinguistic view that dialogue is coordinated over time rather than generated independently at each turn (*10*).

The architecture also distinguishes identity cues from identity claims. A model can imitate lower-case rhythm, short bursts, directness, and topic references. It cannot establish that the generated reply is Alan. This distinction is important for research quality: the measurable unit is not metaphysical personhood, but the fidelity and social effect of a generated conversational act under a disclosed interface.

| Layer | Role in the system | Failure it is meant to reduce |
|---|---|---|
| Long-term priors | Durable facts about interests, posture, humor, texting rhythm, and preferred social stance | Generic assistant behavior and topic-level drift |
| Style profile | A compact profile extracted from examples, including rhythm, directness, humor, punctuation, length, and phrases to avoid | Over-formal, over-helpful, or flattened voice |
| Retrieved examples | Semantically similar utterances from stored Alan messages | Replies that are topically plausible but not locally idiomatic |
| Corrected rewrites | Owner-approved replacement replies paired with the prompt context that caused the bad answer | Repeated failure modes and stale prompt habits |
| Approved examples | Corrected replies inserted back into the Alan-example store with embeddings | Sparse memory after a correction has been made |
| Recent transcript | The live public or trainer conversation, kept separate by session | Isolated answers that ignore what just happened |
| State summary | A short synthesis of the current conversation state | Loss of continuity when the immediate context window is noisy |
| Situation label | A turn-level read such as casual opener, emotional disclosure, research context, project update, or public stranger | Replies that use the wrong social posture for the same words |
| Burst policy | Constraints on reply splitting, length, delimiter use, and question frequency | Text that is semantically right but rhythmically unlike the target speaker |
| Generation trace | Stored record of the mode, user turn, situation label, state summary, retrieval query, rewrite-pair identifiers, and example identifiers used for a generated message | Uninspectable failures that cannot be connected to their evidence sources |

**Table 1.** Evidence strata used by the conversational mirror. The system separates stable identity signals, retrieved examples, correction memory, live state, social framing, and trace metadata so that realism can be inspected and improved at the layer where the failure occurs.

## 2.2 Turn-level realism depends on state and rhythm

Most evaluations of generated text emphasize single-response quality, but personal dialogue often fails at the interactional level. A reply can be grammatical, relevant, and still wrong because it asks a question Alan would not ask, closes a thread Alan would keep moving, or compresses several naturally separate thoughts into one polished paragraph. The deployed system therefore includes a response-shaping layer that controls burst count, length, afterthought probability, and dead-end recovery.

This layer is not cosmetic. Text-message identity is partly temporal and graphical: whether someone sends one compact line, two quick bubbles, or a small cascade of fragments changes how the message is perceived. Human dialogue is coordinated not only through semantic content but through alignment, repair, turn-taking, and expectation management (*10*). Similarly, linguistic style matching is associated with interpersonal coordination and relationship outcomes (*12*, *13*). A personal mirror must therefore model how a person occupies a conversation, not only what facts they know.

In talk.aolabs.io, public conversations are session-separated. This prevents one visitor's transcript from becoming another visitor's live context while still allowing the same underlying speaker evidence to shape replies. The public interface records opens, clicks, message turns, resets, and session metadata. The admin interface supports seeding examples, rebuilding the style profile, correcting bad replies, viewing public sessions, exporting session reports, and running fixed prompt checks after training changes. Persisted assistant replies also receive generation traces that connect an output to its situation label, state summary, retrieval query, selected rewrite pairs, and selected examples. Together, these mechanisms make the system measurable: the same system that generates replies also records the interaction traces needed to study where believability holds or breaks.

## 2.3 Correction memory enables owner-in-the-loop training

A practical personal mirror must improve when the owner says, in effect, "that is not how I would say it." Retraining a model for every correction is unnecessary and brittle. Instead, talk.aolabs.io stores corrected rewrite pairs with the prompt that triggered the failure. At inference time, semantically related corrections are retrieved and given priority above generic examples. This design converts subjective correction into reusable local evidence.

Correction memory is different from ordinary feedback. A rating says that an answer was bad; a rewrite shows what the target speaker would have said instead. This converts a hidden quality judgment into a concrete training signal. The revised implementation also inserts the owner-approved replacement reply back into the Alan-example store, so a single correction contributes both a prompt-to-rewrite pair and an approved utterance example. This supports a stronger research protocol: failures can be grouped by the layer they implicate. Some failures are caused by stale personal facts, some by over-helpful assistant posture, some by wrong burst rhythm, some by wrong relationship framing, and some by weak retrieval. Treating all of them as generic low-quality answers would erase the mechanism that matters.

This approach follows the broader logic of retrieval-augmented generation, where non-parametric memory can condition model output at inference time (*17*). The difference is that the retrieved items are not encyclopedic facts. They are interpersonal evidence: fragments of wording, stance, timing, correction, and prior conversational behavior. This makes the memory store closer to a speaker-specific interaction record than a document database.

## 2.4 Training authority is part of the model

For a personal mirror, who is allowed to train the system is not an implementation detail. If public visitors could mark replies as "like Alan" or "not like Alan," the memory store would become vulnerable to ordinary disagreement, misunderstanding, jokes, adversarial labeling, and relationship-specific context that outsiders cannot judge. A public user may dislike a reply that Alan would actually send, or approve a reply that merely satisfies the user. In either case, public feedback would optimize toward audience preference rather than owner fidelity.

talk.aolabs.io therefore treats training as owner-gated correction rather than public voting. The public site exposes chat, session creation, message sending, reset, event logging, and the paper download; it does not expose a public rewrite or training route. The admin side is password-protected and fails closed if no admin password is configured, unless an explicit local-development override is set. This makes correction authority a consent boundary: public visitors can reveal failure cases through conversations, but they cannot write into the speaker model.

The owner-facing workflow is intentionally low-choice. Public sessions are sorted into a review queue so the most useful cases appear first. A session card exposes a single action, "Fix latest Alan reply," which opens one text box prefilled with the generated answer. Saving the box sends the correction to the same admin rewrite endpoint used by trainer-chat corrections. This is not a complete interface study, but it encodes a testable design claim: personal training should minimize decision burden while preserving high-quality correction signals. The next study can measure not only whether the model improves, but how many owner actions, minutes, and cognitive steps are required per useful correction.

## 2.5 The observer model constrains believability

The user does not encounter a model architecture directly. The user encounters a page, a name, a visual frame, response timing, message shape, and accumulated expectations. This matters because human-likeness is partly attributed by the observer (*2*, *7*). A believable personal mirror therefore needs to study the user's interpretive system, not only the generator.

The live site frames itself as talk.aolabs.io and describes the experience as a direct AI mirror for compact replies shaped by Alan's current voice and context. This framing is important. If the interface claims to be the actual human, believability becomes deception. If it claims to be a generic assistant, the personal evidence becomes confusing. A disclosed mirror frame creates a middle ground: the system can be evaluated for resemblance, continuity, and usefulness without pretending that generated text is authored by the person.

This framing also reduces a known failure mode in conversational agents. Users often bring high expectations to systems that present social cues, then become frustrated when the system lacks the competence or memory implied by those cues (*9*). In a personal mirror, the most important expectation is not universal competence. It is whether the system preserves the target speaker's local voice while staying inside the bounds of what it actually knows.

## 2.6 Evaluation must separate fidelity, believability, correction efficiency, and trust

A single "human-likeness" score is too blunt for this problem. A system can be believable but unfaithful to the target person. It can be faithful in style but too exhausting for the owner to maintain. It can be useful but unsafe because users over-attribute agency. We therefore define four evaluation axes: identity fidelity, interactional believability, correction efficiency, and disclosure-calibrated trust (Table 2).

This separation is critical for a Nature-level study. If participants are simply asked whether the system "seems human," the result may reward interface ambiguity, social desire, or novelty effects. If they are asked whether the system sounds like Alan, the result becomes more specific but still incomplete. A rigorous study should include held-out real messages, generated replies from multiple ablated system variants, pairwise judgments by Alan and by outside raters, owner-effort measurements for each correction workflow, and explicit checks that users know they are interacting with an AI mirror.

| Axis | Core question | Proposed measurements |
|---|---|---|
| Identity fidelity | Does the reply preserve Alan's actual voice and stance? | Owner forced-choice preference; blinded authorship attribution between real and generated replies; linguistic distance from held-out Alan messages; error taxonomy by voice, fact, rhythm, and posture |
| Interactional believability | Does the conversation feel socially continuous? | Multi-turn pairwise ratings; rupture and repair coding; dead-end rate; number of turns before perceived drift; user willingness to continue the conversation |
| Correction efficiency | Can the owner improve the system without an overwhelming training process? | Corrections per useful gain; time per correction; number of interface decisions; queue precision; owner-rated effort; pre/post correction performance on fixed prompts |
| Disclosure-calibrated trust | Does the user understand what the system is and is not? | Post-session comprehension checks; perceived agency and authorship ratings; over-reliance probes; comfort ratings under clear disclosure versus ambiguous framing |

**Table 2.** Evaluation axes for personal conversational mirroring. The goal is not to maximize deception, but to measure resemblance, interaction quality, maintainability, and user understanding while preserving an accurate model of the system.

# 3 Discussion

talk.aolabs.io reframes personal AI dialogue as an evidence architecture and a correction-governance problem. The system's central contribution is not a new base model. It is the decomposition of personal realism into stable priors, style extraction, retrieved examples, correction memory, approved examples, live transcript state, state summarization, relationship/situation labelling, response rhythm control, and generation tracing. This decomposition gives the owner a way to correct the system at the level where it failed and gives researchers a way to measure which layer contributes to believability.

The architecture also clarifies why a believable personal mirror sits at the intersection of multiple fields. From natural-language processing it inherits persona conditioning, retrieval, prompting, and large language models (*4*, *14*, *15*, *17*). From psychology and psycholinguistics it inherits dialogue alignment, linguistic style matching, and individual differences in language use (*10–12*). From human-computer

interaction it inherits social responses to computers, relational agents, expectation management, and the ethics of user interpretation (*5*, *9*, *19*). The interesting scientific question is not whether a model can produce fluent text. It is how these layers combine to create the perception of a particular speaker.

Several design principles follow from the implementation. First, memories used for personal mirroring should be inspectable and corrigible. A system that silently absorbs examples without a correction path may become more confident while becoming less faithful. Second, correction authority should be owner-gated. Public conversations can reveal hard prompts, but public users should not be able to write preference labels into a model of another person's voice. Third, style should not be reduced to catchphrases. The most diagnostic signals may be small: casing, fragment length, directness, refusal style, question frequency, and when the speaker chooses not to smooth an answer. Fourth, the training interface should be treated as part of the scientific system. If the owner cannot tolerate the workload, the correction loop will fail even if the model architecture is sound. Fifth, believability should not be optimized without disclosure. A system designed to fool people into believing the actual person is present is a different object of study, and a riskier one, than a disclosed mirror designed to preserve voice.

The current system has clear limitations. The local development database inspected for this draft contains the default style profile but no stored public conversations, no rewrite pairs, no feedback rows, and no generation traces. That means the manuscript cannot yet report empirical gains from correction memory, retrieval depth, style rebuilding, review-queue ranking, or voice-check prompting. The implementation is also text-only in the inspected code; claims about spoken voice, prosody, or audio realism would require additional speech data and audio evaluation. Finally, the architecture currently uses manually written long-term priors, which are useful but can become stale if not audited against current evidence.

The next scientific step is an ablation and workflow study. The full system should be compared against variants that remove corrected rewrites, approved examples, retrieved examples, state summary, situation labels, generation-trace review, burst control, or stable style profile. Participants should compare generated replies with held-out Alan replies across short prompts, emotionally loaded prompts, mundane social prompts, research prompts, and long multi-turn conversations. Alan's own preferences should be analyzed separately from outside-rater believability, because owner fidelity and public believability are not the same measure. Owner training load should be measured alongside output quality: a personal mirror that improves only through exhausting correction is not a robust personal

system. This would turn the current deployed system from a compelling prototype into a testable account of personal conversational realism.

# 4 Materials and Methods

## System implementation

The inspected deployment is a FastAPI application with a public chat interface, an admin trainer interface, persistent SQLite storage, and OpenAI model calls for response generation, style summarization, and embedding. The public route serves talk.aolabs.io. The public API exposes session state, page-open logging, event logging, chat, and reset. It does not expose a public rewrite or training endpoint. The admin route supports memory seeding, style-profile rebuilding, reply correction, feedback recording, public-session review, analytics, fixed-prompt voice checks, and PDF export. Public conversations are separated by session identifier so that each visitor's live transcript remains isolated.

The database stores messages, style profiles, feedback, rewrite pairs, generation traces, public sessions, public-session events, rate-limit events, and IP-location cache rows. Messages can include embedding vectors stored as JSON. Public session events include opens, clicks, resets, and message turns. Generation traces store the generated message identifier, mode, session identifier, user message, situation label, state summary, retrieval query text, selected rewrite-pair identifiers, and selected example identifiers. The application enforces public rate limits using per-IP and per-session buckets.

Admin protection is enforced at the dependency level for trainer routes. If no admin password is configured, admin routes fail closed unless an explicit local-development override is enabled. This behavior is part of the training design rather than a deployment detail: only the owner-facing interface should be able to convert model outputs into future evidence.

## Speaker evidence

The system prompt is assembled from several speaker-evidence sources. Long-term priors encode durable facts about Alan's interests, conversational stance, humor, and texting rhythm. A style profile summarizes voice traits such as directness, rhythm, decision style, humor, emotional register, punctuation, preferred length, and phrases to avoid. Retrieved examples provide local utterance evidence from stored Alan messages. Corrected rewrite pairs provide high-priority examples of what Alan preferred instead of a bad generated reply. Owner-approved corrected replies are also stored as Alan examples, so correction memory contributes both paired contrastive evidence and standalone utterance evidence.

The system uses semantic embeddings for retrieval. For a new turn, it embeds a context-focused query, a state summary, and the newest user message, combines them as weighted retrieval queries, and searches stored Alan messages and rewrite pairs using cosine similarity. In the inspected code, the context-focused query receives the largest retrieval weight, followed by the state summary and then the newest user message. The main turn embeddings are requested in batch, and memory seeding batches chunk embeddings. These choices reflect the design assumption that what the conversation is currently about should matter more than the newest sentence in isolation while keeping training and inference latency lower than separate embedding calls.

## Conversation state

The application builds a recent transcript from the current mode and session. A state-summary call condenses the live conversation into a short representation used for retrieval and generation. The response builder also infers a relationship/situation label using heuristics over the newest user message and recent transcript, including categories such as casual opener, emotional disclosure, research context, project update, friend-like banter, close social context, or public stranger. The response-generation prompt then includes the long-term priors, burst guidance, relationship/situation label, stable style summary, conversation state summary, current situation focus, relevant corrected rewrite pairs, recent transcript, and retrieved examples.

This method separates short-term continuity from long-term identity. The same user sentence can require different replies depending on what came before it. Conversely, the same live context can require different wording depending on the target speaker's stable style.

## Response shaping

Generated text is post-processed into one or more message bubbles. The implementation includes configurable thresholds for very short, short, medium, and long replies; probabilities for two-bubble defaults and occasional afterthoughts; and special handling for open-ended prompts. A dead-end detector can trigger a retry when a response closes the conversation in a way that conflicts with the current prompt. This response-shaping layer is treated as part of the modelled voice, not as presentation-only formatting.

## Trainer correction

The admin interface can rewrite one or more generated messages from trainer chat. The application archives unused bad model messages, stores the corrected reply, embeds the original user prompt, and records the rewrite pair for future retrieval. Public-session review exposes a narrower correction workflow: the system selects the latest current Alan reply in a session, opens a single editable text box, and saves the replacement through the same admin rewrite mechanism. This makes each correction reusable when a later prompt is semantically similar. In a study, rewrite pairs can be treated as an intervention: trials can compare generation before and after correction-memory retrieval.

## Review queue and voice checks

The public-session analytics endpoint returns both the raw recent session list and a training queue. The queue scores sessions by visitor-message count, Alan-message count, resets, duration, clicks, long visitor messages, whether the latest current turn is a visitor message, and whether a current Alan reply exists. Archived messages are excluded when choosing the latest reply for correction. The queue is a heuristic triage mechanism, not a claimed optimal policy.

The admin interface includes a fixed-prompt voice check that runs a small set of prompts without persisting the generated messages. The returned rows include the prompt, generated bubbles, situation label, number of retrieved rewrite pairs, and number of retrieved examples. This is not an evaluation result. It is an operational guardrail that lets the owner inspect obvious drift after corrections before starting a broader study.

### Proposed evaluation protocol

A full evaluation should use held-out Alan messages that were not present in the retrieval memory or style-profile extraction set. Prompts should cover ordinary small talk, personal openers, emotionally ambiguous messages, practical questions, research or builder context, topic shifts, and multi-turn continuity. For each prompt or conversation segment, the study should generate replies using the full system and ablated variants.

Owner evaluation should use pairwise forced-choice comparisons between candidate replies, with Alan selecting the reply closest to what he would have sent and marking the reason for failures. The workflow study should separately log the number of queued sessions reviewed, corrections saved, edits abandoned, time per correction, and owner-rated effort. Outside-rater evaluation should include blinded authorship attribution, perceived continuity, willingness to keep chatting, and explicit comprehension of the AI-mirror framing. Raters should not be rewarded for believing that Alan personally typed the response; the correct target is perceived resemblance under disclosure.

Quantitative analysis should report effect sizes for each ablation, not only aggregate scores. Qualitative coding should separate failures in facts, voice, rhythm, stance, situation label, privacy expectation, and conversation continuity. Public-session logs can support ecological analysis, but private messages and personally identifying information should be excluded or de-identified before release.

## Ethics and privacy

This manuscript draft does not report a human-subject experiment. A future study should be reviewed under the applicable institutional process before collecting participant judgments or public-session research data. Because the system is built around a real person's conversational style, training examples, rewrite pairs, generation traces, and visitor transcripts should be treated as sensitive interpersonal data. Public visitors should not be given authority to train or label the owner model without an explicit consent design and abuse analysis. Public release should be limited to de-identified aggregate results, synthetic or consent-cleared prompt sets, and analysis code that does not expose private messages.

## Reporting summary

No statistical tests, randomization, blinding, sample-size calculation, or exclusion criteria are reported because the present manuscript describes the deployed system and proposed evaluation protocol. A

submission based on completed experiments should add the full participant protocol, preregistered or explicitly stated hypotheses, ablation conditions, correction-workflow measures, annotation scheme, reliability measures, and statistical analysis plan.

## Acknowledgements

The author thanks the users and builders whose reactions motivated the system, and acknowledges that a personal conversational mirror depends on both technical design and careful social framing.

## Funding

No external funding was used to prepare this manuscript draft.

## Author contributions

A.N.P. conceived the system, implemented the prototype, inspected the deployment state, designed the owner-gated correction workflow, and prepared the manuscript draft.

## Competing interests

A.N.P. is the developer of talk.aolabs.io and AO Labs.

## Data availability

This manuscript draft does not report human-subject outcome data. Personal training examples, rewrite pairs, generation traces, visitor transcripts, and session metadata are not publicly released because they may contain private conversational material. A publication-ready study should release de-identified aggregate measurements, synthetic prompt sets where appropriate, and the analysis protocol needed to reproduce reported conclusions.

## Code availability

The inspected prototype code is in local development at the time of this draft. A public code release would require removal of secrets, private memory examples, personal transcripts, and deployment-specific configuration.

## Additional information

Correspondence and requests for materials should be addressed to A.N.P.

## References


1. A. M. Turing, Computing machinery and intelligence. *Mind* **LIX**, 433–460 (1950).
2. J. Weizenbaum, ELIZA: A computer program for the study of natural language communication between man and machine. *Communications of the ACM* **9**, 36–45 (1966).
3. A. Vaswani *et al.*, presented at the Advances in Neural Information Processing Systems, vol. 30.
4. T. B. Brown *et al.*, presented at the Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901.
5. B. Reeves, C. Nass, *The Media Equation: How People Treat Computers, Television, and New Media Like Real People and Places* (Cambridge University Press, Cambridge, 1996).
6. C. Nass, Y. Moon, Machines and mindlessness: Social responses to computers. *Journal of Social Issues* **56**, 81–103 (2000).
7. N. Epley, A. Waytz, J. T. Cacioppo, On seeing human: A three-factor theory of anthropomorphism. *Psychological Review* **114**, 864–886 (2007).
8. M. Mori, K. F. MacDorman, N. Kageki, The uncanny valley. *IEEE Robotics & Automation Magazine* **19**, 98–100 (2012).
9. E. Luger, A. Sellen, presented at the Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, pp. 5286–5297.
10. M. J. Pickering, S. Garrod, Toward a mechanistic psychology of dialogue. *Behavioral and Brain Sciences* **27**, 169–190 (2004).
11. J. W. Pennebaker, L. A. King, Linguistic styles: Language use as an individual difference. *Journal of Personality and Social Psychology* **77**, 1296–1312 (1999).
12. K. G. Niederhoffer, J. W. Pennebaker, Linguistic style matching in social interaction. *Journal of Language and Social Psychology* **21**, 337–360 (2002).
13. M. E. Ireland *et al.*, Language style matching predicts relationship initiation and stability. *Psychological Science* **22**, 39–44 (2011).

*14*. J. Li *et al.*, presented at the Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pp. 994–1003.

*15*. S. Zhang *et al.*, presented at the Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, pp. 2204–2213.

*16*. F. Mairesse, M. A. Walker, presented at the Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 496–503.

*17*. P. Lewis *et al.*, presented at the Advances in Neural Information Processing Systems, vol. 33, pp. 9459–9474.

*18*. J. S. Park *et al.*, presented at the Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, pp. 1–22.

*19*. T. W. Bickmore, R. W. Picard, Establishing and maintaining long-term human-computer relationships. *ACM Transactions on Computer-Human Interaction* **12**, 293–327 (2005).