Oroi: una red semántica para agentes conversacionales

Memoria continua para una conversación infinita

Igor Laburu
Gako AI
oroi@gako.ai

Preprint, julio de 2026
DOI: 10.5281/zenodo.21208930

Resumen

La memoria habitual de un chatbot (RAG, *generación aumentada por recuperación*) guarda la conversación como texto y, ante cada pregunta, busca los fragmentos que más se *parecen* a ella. Funciona bien para consultar documentos, pero una conversación no es un documento: lo importante muchas veces no se parece a la pregunta, cambia con el tiempo y depende de lo que se acaba de decir. Este documento describe **Oroi**, una memoria que guarda la conversación como una *red semántica* dirigida y ponderada (conceptos unidos por asociaciones con peso) y la dota de una dinámica inspirada en la memoria humana: lo mencionado se activa, la activación se propaga a lo asociado, todo decae con los turnos, las asociaciones que se repiten se refuerzan (regla de Hebb) y un proceso en segundo plano consolida lo importante. Recordar combina el *reconocimiento* (lo que sigue activo por la conversación reciente) con la *evocación* (propagación desde las pistas de la pregunta).

En una evaluación de 224 escenarios, **Oroi** empata con un RAG fuerte en las preguntas directas (donde la propuesta no pretende ganar) y marca la diferencia donde el parecido confunde: con distractores (0,97 frente a 0), cuando un dato cambia (0,25 frente a 0) y en las cadenas de tres asociaciones (1,00). En el global de recuperación, 0,75 frente a 0,55: gana en 43 escenarios sin perder ninguno, una diferencia que el azar no explica ($p < 10^{-4}$). El límite principal hoy es la calidad con la que se extraen las relaciones de cada frase.

1. Introducción

Supongamos un asistente con memoria al que semanas atrás se le mencionó que "mi oficina está en Madrid". Al preguntarle "¿dónde trabajo?", la respuesta existe, pero *no se parece* a la pregunta: para llegar a "Madrid" hay que pasar por "oficina". La memoria dominante en los chatbots, RAG, no puede seguir ese camino: convierte cada texto en un vector numérico que resume su significado (un *embedding*) y recupera los fragmentos cuyo vector es más cercano al de la pregunta. Es un buen buscador, pero recuerda mal, por tres razones:

- **El tiempo.** Si un dato se corrige ("en realidad, me he mudado"), la búsqueda por similitud devuelve las dos versiones, la antigua y la actual, sin poder distinguir cuál vale hoy.
- **La asociación a distancia.** La respuesta a menudo no se parece a la pregunta, como en el ejemplo.
- **La anticipación.** La memoria humana pre-activa lo que está a punto de ser relevante (el efecto de *priming*); un buscador por similitud no puede anticipar nada: solo reacciona a la pregunta ya formulada.

Proponemos tratar la memoria conversacional como un *sistema dinámico* sobre una red de conceptos, en vez de como un buscador sobre texto. La red *encuentra*; el texto literal de la conversación, guardado aparte, *habla*. No pretendemos modelar el cerebro: tomamos de la neurociencia unas pocas ideas operativas (activación, asociación, decaimiento, consolidación) porque resuelven problemas concretos del diálogo. Nuestra hipótesis, falsable, es:

> *La recuperación sensible a la activación supera a la recuperación por similitud vectorial pura en conversaciones largas con referencias recurrentes y deriva temática.*

La acompaña una segunda hipótesis que este trabajo aún no mide: en conversaciones ultra largas, una memoria así debería ser además más eficiente, compacta y acotada que archivar la transcripción entera, porque lo que crece es el conocimiento consolidado, no el texto bruto; la reelaboración de recuerdos (§9) es la pieza que la hará medible.

2. El origen de la arquitectura

Combinamos técnicas ya establecidas y las aplicamos al diálogo con un modelo de lenguaje; ninguna de las piezas es nueva, la apuesta es la combinación. La *propagación de activación* sobre redes semánticas [1] modela el recuerdo como activación que fluye de los conceptos mencionados a los asociados. La psicología computacional lleva décadas simulando la memoria humana con ecuaciones que reproducen sus tiempos de recuerdo y de olvido; su arquitectura de referencia (*ACT-R* [2]) distingue una fuerza lenta (lo consolidado) de una activación rápida (lo presente), que reflejamos con dos escalas temporales. La regla de Hebb [5] (las neuronas que se activan a la vez refuerzan su conexión) motiva el refuerzo de las asociaciones. La señal de sorpresa sigue la idea de la *codificación predictiva* [6, 7] y la planteamos como una señal contextual que refleja la relación entre el Sistema 1 (la red, rápida y automática) y el Sistema 2 (el modelo de lenguaje, lento y deliberado) de Kahneman [8]: la red emite el aviso de una ruptura de la expectativa conversacional; actuar sobre él corresponde a la capa de orquestación. Como referencia de contraste usamos RAG [9] con búsqueda por significado (vectores densos [10]) y por palabras (BM25 [11]), combinadas [12].

3. Cómo funciona

3.1. La red

Cada conversación se convierte en un grafo. Un *nodo* es un concepto mencionado (“oficina”, “Madrid”, “mi jefe”); una *arista* es una asociación entre dos conceptos, con un *peso* que crece cuando la asociación se repite. El peso es lo que aprende; la etiqueta de la relación (“está en”, “trabaja con”) es solo descriptiva. Cada nodo mantiene dos registros con dos ritmos distintos: una **activación** rápida (lo presente que está ahora: sube al mencionarse y decae con los turnos) y una **fuerza** lenta (lo consolidado que está: solo la toca la consolidación, §3.3). El texto literal de cada turno se guarda aparte, como *episodios* enlazados a los conceptos que mencionan. La separación refleja la distinción clásica entre memoria **semántica** (conceptos y asociaciones: la red) y memoria **episódica** (lo vivido literal: el texto) [4]: la red encuentra, el texto habla.

3.2. Percepción de un turno de conversación

Un extractor (un modelo de lenguaje pequeño, una llamada por turno) convierte lo dicho en un puñado de conceptos y relaciones: es la fase que la neurociencia llama *codificación*. Cada concepto entrante o bien *resuena* con un nodo existente (**similitud coseno** por encima de un umbral τ), o bien crea uno nuevo. Reconocer “el mismo concepto” exige dos remedios prácticos. Los nombres propios parciales o mal escritos se rescatan con una búsqueda léxica clásica cuando

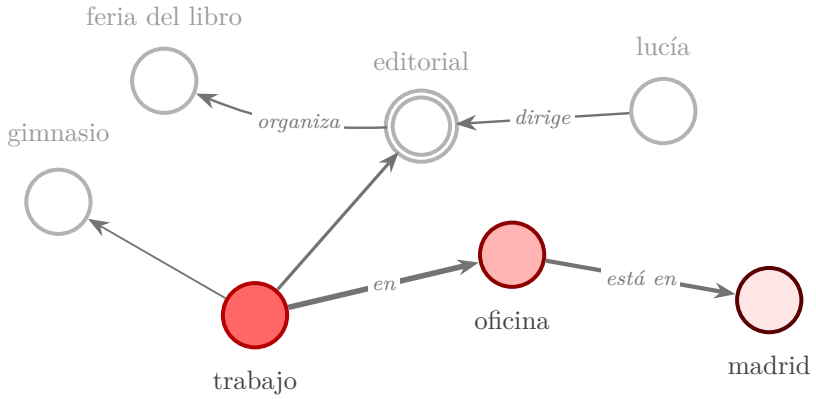

Figura 1: Detalle de una red semántica en el momento de la pregunta "¿dónde trabajo?": la pista activa "trabajo" (rojo intenso) y la activación se propaga por las asociaciones hasta "madrid", decayendo con la distancia (rojo tenue). El grosor de cada arista es su peso aprendido y la etiqueta, cuando se muestra, es solo descriptiva; el doble borde ("editorial") marca un concepto consolidado; los nodos grises están inactivos.

el vector no alcanza el umbral ("Guggen" debe encontrar "Museo Guggenheim"). Y los *valores* (importes, fechas, cantidades) se comparan por identidad exacta, porque sus embeddings son casi idénticos entre sí: "15.000 euros" y "60.000 euros" se parecen entre sí más de lo que exige el umbral de fusión, y acabarían fundidos en un solo nodo.

Los nodos reconocidos reciben un impulso de activación, mayor si la relación también coincide. Después, la activación se *propaga* a los vecinos durante unas pocas rondas síncronas y amortiguadas, por **equiparación de gradiente**: fluye solo del nodo más activo al menos activo, en proporción a la diferencia y al peso de la arista, y el origen cede lo que fluye,

$$\text{flujo}_{i\to j} \;=\; \lambda \,\frac{w_{ij}}{\sum_k w_{ik}}\; \text{máx}(a_i - a_j,\, 0), \tag{1}$$

donde a_i es la activación del nodo i y w_{ij} el peso de la arista que los une. La resta $a_i - a_j$ es el desnivel entre los dos nodos: si el vecino está igual o más activo que el origen, no fluye nada. De ese desnivel, en cada ronda solo se traspasa una parte, λ (en nuestro caso 0,5: la mitad), repartida entre los vecinos según el peso de cada arista; traspasar poco a poco evita que la red oscile. Y cuando el flujo va en contra del sentido de la arista, se atenúa además por un factor ϕ. La propagación solo *redistribuye*: lo percibido directamente queda siempre por encima de lo que solo recibe propagación (se igualan, nunca se cruzan), y una red que no recibe nada se apaga entera, sin excepciones (§7). Al cerrar el turno toda activación se multiplica por un factor de decaimiento δ (menor que uno), contando el tiempo en turnos de conversación, nunca en tiempo de reloj; y una **homeostasis** aplica un techo por nodo, un presupuesto global (inhibición lateral: si todo está activo, nada destaca) y un suelo. Por último, el **refuerzo hebbiano** actualiza las aristas entre nodos co-activos (el peso crece con el producto de sus activaciones, a una tasa de aprendizaje η: $w_{ij} \mathrel{+}= \eta\, a_i a_j$), y solo las que ya existen: una coincidencia puntual es ruido; si se repite, la consolidación la promoverá a asociación.

Dos detalles más. El extractor ve una ventana deslizante con los últimos turnos, solo para resolver "lo", "ese", "también": los hechos se extraen exclusivamente del turno actual. Y hay una **asimetría de hablante**: el protagonista es el usuario; lo que afirma el asistente solo entra si el usuario lo acoge. Sin ella, el sistema acabaría percibiendo sus propias respuestas como hechos y realimentándose.

3.3. Consolidación de recuerdos

Cuando la conversación queda inactiva, un proceso en segundo plano **consolida**, como la memoria humana consolida durante el sueño [3]: funde nodos casi duplicados (automáticamente si el parecido es extremo; consultando a un modelo de lenguaje en la franja dudosa), promueve a arista las coincidencias que se repitieron, transfiere parte de la activación a la fuerza lenta, deja decaer esa fuerza muy despacio y poda lo que nunca se afianzó. La fuerza lenta es lo único que sobrevive al paso de muchos turnos; la activación es la *memoria de trabajo*, y el reloj no la toca: una pausa entre sesiones, por larga que sea, no enfría nada. Reabrir la aplicación no reinicia nada: retomar una conversación es continuarla.

3.4. Recuperación de recuerdos

La recuperación tiene dos vías, como en el ser humano. El **reconocimiento** es pasivo: lo que sigue activo por la conversación reciente está disponible sin esfuerzo. La **evocación** es activa: si el reconocimiento no basta, los conceptos de la pregunta lanzan ondas por la red (una lectura que no altera la activación) y, donde varias pistas confluyen, lo evocado se acumula; así se llega a lo lejano y a lo apagado. Estar disponible, sin embargo, no decide el orden: un hecho muy reciente pero irrelevante no debe ganar solo por estar activo. Los candidatos se ordenan por

$$\text{puntuación}(n) \;=\; \rho \cdot \text{relevancia}(n) \;+\; \text{fuerza}(n), \tag{2}$$

la relevancia que le llega de las pistas de la pregunta más su importancia consolidada, con una ganancia ρ que equilibra ambas escalas. Los K primeros se convierten en un texto breve (las asociaciones como prosa, más los episodios literales más vinculados) que se añade al turno del usuario. Y si nada está encendido, la memoria *calla*: no aportar nada es el comportamiento por defecto, no un fallo. Recuperar, además, deja huella: los recuerdos que sí se inyectan refuerzan ligeramente su fuerza lenta (el *efecto testeo* de la psicología de la memoria: recordar también consolida).

3.5. La sorpresa

Por la propagación, la red mantiene en todo momento una expectativa implícita: lo que está activo es lo que "espera". Cuando lo que llega no encaja con ninguno de los temas activos de la conversación (medido por afinidad de significado con lo que la red mantiene encendido), la memoria emite una señal graduada de **sorpresa**. No es un mecanismo de memoria, sino de *atención*: la parte rápida y automática avisa al modelo de lenguaje, más lento y deliberado, de que la conversación ha saltado a algo inesperado y conviene detenerse a razonar en vez de seguir en piloto automático. La señal está implementada; su acción efectiva sobre la conversación (una marca en el prompt del modelo que responde, fuera de la memoria: lo que en un humano llamaría a la consciencia) queda fuera del alcance de este trabajo.

4. Mecanismos alternativos

Mencionamos otros mecanismos que se valoraron, se midieron y se descartaron, cada uno por el motivo que se describe:

- *Filtros de corte en la evocación* (descartar los nodos que reciben poca activación): eliminaban la respuesta lejana del multi-salto (el nodo a tres asociaciones es, por construcción, el que menos activación recibe) y, con ella, la ventaja medible del paradigma.
- *Reinicios de activación por reloj* (vaciar la memoria de trabajo tras una pausa larga de reloj): el tiempo de esta memoria es conversacional (turnos), nunca de reloj; reabrir una sesión es continuar la conversación, no empezar de cero.

- *Canonicalización de nombres en el extractor* (pasarle la lista de conceptos ya conocidos para que reutilice los nombres): fundía referentes distintos de superficie parecida.
- *La regla de propagación original* (sumar activación al vecino sin restarla del origen): permitía que la red creara activación de la nada; su corrección y su efecto medido se detallan en §7.

La lección común: ningún mecanismo se añade, ni se conserva, sin haberlo medido en condiciones de uso real.

5. Implementación

La librería está escrita en Python sobre SQLite, con la extensión `sqlite-vec` para la búsqueda vectorial; el modelo de embedding queda fijado en la base de datos y no se mezcla. El núcleo no depende de ningún modelo de lenguaje: el embedder, el extractor, el conversador y el juez de consolidación son proveedores externos inyectados tras interfaces, de modo que la misma memoria puede servir a cualquier chatbot con cualquier proveedor. Los embeddings se calculan en lote, la extracción es una sola llamada por turno, y el contexto recuperado se añade al turno del usuario (nunca al mensaje de sistema) para no invalidar la caché de *prompts* del proveedor.

6. Cómo lo medimos

Evaluamos sobre escenarios conversacionales sintéticos, cada uno formado por una secuencia de turnos y una o más *sondas*: preguntas que declaran qué fragmentos debe contener el contexto recuperado y cuáles debe evitar. La métrica, **Recall@contexto**, es objetiva (sin juez LLM): una sonda acierta solo si aparece todo lo requerido y nada de lo prohibido. Comparamos sobre escenarios idénticos las condiciones de la Tabla 1; para cada diferencia estimamos, con un contraste estándar (el test exacto de McNemar), la probabilidad de que surgiera por puro azar. El conjunto de escenarios está pensado como *termómetro*, no como diana: incluye a propósito los regímenes donde RAG debería ganar, y un mecanismo solo se adopta si generaliza, nunca para levantar una sonda concreta.

Condición	Criterio de recuperación
Sin memoria	solo los últimos k turnos
RAG híbrido	por significado (vectores) + por palabras (BM25), combinadas
RAG sobre hechos	similitud sobre los *mismos* nodos extraídos
Re-ranker	similitud re-ordenada por activación
Oroi	activación sobre la misma red

Tabla 1: Condiciones evaluadas. "RAG sobre hechos" aísla la contribución de la dinámica: mismo material extraído y misma serialización; solo cambia el criterio de selección.

7. Resultados

Evaluamos sobre el corpus generado (224 escenarios, 32 por fenómeno), con la dinámica corregida de §3.2, `gpt-4o-mini` como extractor y juez y `text-embedding-3-small` para los embeddings (Tabla 2).

Fenómeno	RAG (híb.)	RAG (hechos)	Re-ranker	**Oroi**	n
Recurrencia	1,00	1,00	0,00	**1,00**	32
Actualización	0,00	0,00	0,00	**0,25**	32
Multisesión	1,00	1,00	0,00	**1,00**	32
Multi-salto	0,97	0,91	0,00	**1,00**	32
Multi-salto (3)	0,91	0,69	0,06	**1,00**	32
Convergencia	0,00	0,00	0,06	0,00	32
Distractores	0,00	0,00	0,00	**0,97**	32
Global	0,55	0,51	0,02	**0,75**	224

Tabla 2: Recall@contexto por fenómeno y condición, con la dinámica corregida. Frente al RAG híbrido, **Oroi** gana en 43 escenarios y no pierde en ninguno; frente al RAG sobre hechos, 52/0 (test de McNemar: $p < 10^{-4}$ en ambos casos).

7.1. La ventaja, y dónde aparece

La ventaja global (0,75 frente a 0,55 del mejor RAG) ya no es cuestión de tamaño de muestra: **Oroi** gana en 43 escenarios sin perder ninguno; la probabilidad de que una diferencia así surja por azar es inferior a 10^{-4}.

La ventaja se concentra en los casos que confunden a la búsqueda por similitud. En *distractores* (hechos con la misma forma que el buscado, que compiten con él por aflorar: "Elena es fontanera", "Carlos es abogado"... y se pregunta por el oficio de Carlos), ningún RAG logra traer el dato correcto sin arrastrar a los vecinos (0,97 frente a 0). En *actualización*, solo **Oroi** consigue a veces que el valor antiguo, ya apagado por el decaimiento, quede fuera del contexto (0,25 frente a 0). Y en el *multi-salto de tres* (la respuesta está a tres asociaciones de la pregunta) **Oroi** es perfecto (1,00), sin ocultar el matiz: con más muestra el RAG híbrido se acerca (0,91; el RAG sobre hechos queda en 0,69), porque su búsqueda por palabras encuentra los eslabones de la cadena cuando comparten términos con la pregunta.

En las preguntas directas (recurrencia, multisesión) todas las condiciones empatan, tal y como la evaluación está diseñada para que ocurra. El re-ranker queda inutilizado (0,02): ordena por cantidad de activación, y la equiparación redistribuye precisamente esa cantidad. El coste de lectura queda a la par del RAG híbrido ($\approx$1 s por consulta ambos); en escritura, **Oroi** añade una extracción por turno que el RAG crudo no necesita.

7.2. El defecto de propagación y su corrección

Con la dinámica de propagación original, tres de las siete familias puntuaban 0 para *todos* los métodos, y el resultado global de **Oroi** (0,55–0,57) no se distinguía del de RAG. La causa: la regla de propagación sumaba activación al vecino *sin restarla del origen*. Dos conceptos conectados podían así realimentarse sin límite, acaparar el presupuesto de inhibición y ahogar a los hechos relevantes.

La corrección aplica la equiparación por gradiente de §3.2: la activación fluye solo del nodo más activo al menos activo, en proporción a la diferencia, y el origen pierde lo que cede: cuando se igualan, el flujo se detiene. Por el camino descartamos dos variantes. La primera restaba del origen todo lo repartido, sin atender al desnivel: el origen podía quedar por debajo del receptor y la red entraba en oscilación. La segunda suprimía el flujo en contra del sentido de la arista, y perdía el multi-salto de tres: la asociación de una categoría a sus miembros viaja en ese sentido.

Para medir el efecto de la corrección aislado de todo lo demás, comparamos las tres variantes de propagación entre sí, sobre los mismos escenarios y en condiciones idénticas (una *ablación*; $n = 56$, solo **Oroi**): la equiparación subió el resultado global de 0,55 a 0,73 sin empeorar ninguna familia. Los sistemas RAG de la comparación recuperan por similitud y no emplean la

propagación de activación, de modo que esta corrección no cambia sus resultados.

7.3. El equilibrio entre lo relevante y lo consolidado

En la ecuación 2, la ganancia ρ fija el equilibrio entre los dos criterios que ordenan los recuerdos: la relevancia para la pregunta y la importancia consolidada. Probamos con tres valores (1, 6 y 12, con la dinámica original) y el resultado global apenas varió (0,55–0,57). La explicación: el desequilibrio que ρ corrige casi no aparece en escenarios cortos; hace falta una conversación larga para que un concepto acumule tanta importancia consolidada que relegue a los datos concretos. En conversaciones largas reales el efecto sí se aprecia con claridad: con ρ alto, preguntar por una cifra concreta devuelve esa cifra, y no el concepto dominante de la conversación. Eso señala una carencia del corpus de evaluación (le faltan conversaciones largas), no del mecanismo.

7.4. Del contexto a la respuesta

Recall@contexto evalúa la recuperación; falta comprobar si, con ese contexto, el sistema completo *responde* bien a la pregunta. Para medirlo, un modelo pequeño (`gpt-4o-mini`) genera la respuesta usando únicamente el contexto recuperado por cada condición, y un segundo modelo la compara con la respuesta esperada y emite el veredicto (el patrón *LLM-as-judge*).

A este nivel el orden se invierte: RAG crudo 0,91, RAG sobre hechos 0,83, **Oroi** 0,80. En *convergencia* (la respuesta exige cruzar dos pistas), **Oroi** responde mejor que cualquier RAG (0,88 frente a 0,44 del crudo), y los *distractores* dejan de ser un problema para todas las condiciones. Pero **Oroi** falla donde su contexto era perfecto: en las cadenas de tres asociaciones (0,66, con toda la evidencia delante) y en la *actualización* (0,44 frente a 1,00 del crudo).

El motivo es el formato en que cada condición presenta lo recuperado. El RAG crudo aporta los turnos literales de la conversación, y ese texto es fácil de interpretar para un modelo pequeño: la referencia temporal está dentro de la propia frase (“me he mudado”) y no requiere ninguna deducción. **Oroi** aporta hechos extraídos y condensados, un formato más frágil: si una etiqueta de relación se extrajo mal, el hecho resulta dudoso, y el modelo, obligado a ceñirse al contexto, prefiere no responder.

La lectura conjunta de los dos niveles es coherente: la *recuperación* por activación encuentra mejor, pero la cadena completa rinde tanto como su eslabón más débil, que hoy es la presentación de lo recuperado. La reelaboración de recuerdos (§9) ataca precisamente ese eslabón: reescribir lo almacenado como texto natural, breve y preciso, en lugar de entregarlo como listas de hechos sueltos. Con un modelo de producción generando las respuestas, en lugar de uno pequeño, la diferencia debería acortarse; queda pendiente medirlo.

7.5. Balance de las tres motivaciones

De las tres motivaciones del paradigma (§1), la evaluación confirma la segunda: en la asociación a distancia, **Oroi** es el único método que resuelve las cadenas de tres asociaciones (el RAG híbrido se acerca gracias a su búsqueda por palabras). En cuanto a la primera motivación, el tiempo, constituye hoy el punto más débil: 0,25 frente a 0 en recuperación y 0,44 frente a 1,00 en respuesta. Aunque marcamos la recencia de cada hecho en la serialización, no fue suficiente para un modelo pequeño: la referencia temporal funciona mejor integrada de forma natural en el texto (“me he mudado”) que como una anotación externa, un argumento más a favor de la reelaboración textual. Por último, la anticipación está implementada como señal de sorpresa (§3.5) y demuestra utilidad práctica inmediata: en cada turno se emite un valor graduado que permite notificar a la capa de orquestación del agente un giro no previsto de la conversación, para que replanifique, pida aclaración o salga del piloto automático. El mecanismo es claro y

no añade coste (se calcula sobre lo ya percibido); lo que falta es su protocolo de evaluación, que queda como trabajo futuro declarado.

8. Límites

La propuesta no pretende superar a RAG en la búsqueda plana por similitud semántica; en ese terreno los escenarios están diseñados para terminar en empate. El valor del paradigma está donde la similitud no llega: cadenas de asociaciones, datos que cambian, temas que se retoman. Su techo actual es la extracción: una asociación solo puede recorrerse si la arista existe, y un extractor pequeño a menudo obtiene las entidades de una frase ambigua sin la relación que las une. La memoria vale lo que valen sus aristas. Un segundo factor limitante es el reconocimiento de entidades de nombre parecido ("Correos" y "Correos Express"): unas veces deben fundirse en un solo concepto y otras no, y decidirlo de forma fiable sigue abierto. El tercer obstáculo es el coste computacional: una extracción por turno en escritura, de la que RAG prescinde. Por último, la evaluación actual es corta y sintética: sirve para validar mecanismos, pero no sustituye a una evaluación sobre conversaciones reales largas ni a un benchmark estándar; ambas quedan como trabajo futuro.

9. Conclusión y uso previsto

Hemos descrito una memoria para agentes conversacionales que recuerda por activación (lo reciente, lo asociado y lo consolidado) en lugar de solo por parecido, y un protocolo para comprobar si eso aporta donde la hipótesis lo predice. Los resultados respaldan la propuesta en el caso estructural (la asociación a distancia), en la resistencia a los distractores y, con la propagación corregida, también en el resultado global (0,75 frente a 0,55, $p < 10^{-4}$). El siguiente avance no pasa por refinar aún más la dinámica de la red, sino por extraer relaciones de forma fiable.

El uso previsto es la memoria de muy largo alcance: relaciones entre un usuario y su asistente que se extienden meses o años, a lo largo de cientos de sesiones. En ese régimen, los hechos que el usuario menciona no se archivan como texto inerte que solo se consulta: viven en una representación que evoluciona con el uso. Cada mención los reactiva; cada contexto nuevo los re-asocia; la repetición los consolida; la carga emocional, o la petición explícita de recordar, los fija al instante; y el desuso los va apagando. Esa selectividad es la que acerca el agente a lo humano: recuerda lo que se repitió y lo que importó, no todo por igual.

La arquitectura está además diseñada para que lo que crezca con el tiempo no sea la transcripción, sino el conocimiento estable: en RAG cada turno ocupa lo mismo para siempre, lo trivial igual que lo importante; aquí la consolidación funde, poda y destila en segundo plano. La siguiente pieza prevista es la *reelaboración* de recuerdos, afín a la *reconsolidación* de la memoria humana: durante la consolidación, los episodios antiguos cuyos hechos ya están asimilados se reescribirán en versiones más breves, precisas y relevantes (como la memoria humana guarda el sentido antes que la transcripción), conservando siempre el literal original recuperable.

Dado que reabrir una sesión es continuar, no empezar de cero, el objetivo final es un asistente con el que la relación se acumula: lo consolidado está disponible al instante y el resto sigue siendo recuperable por evocación. El código y el protocolo de evaluación son reproducibles.

En una frase: *una consolidación y consulta de recuerdos continuas para una conversación infinita.*

Referencias

[1] A. M. Collins and E. F. Loftus. A spreading-activation theory of semantic processing. *Psychological Review*, 82(6):407–428, 1975.

[2] J. R. Anderson et al. An integrated theory of the mind. *Psychological Review*, 111(4):1036–1060, 2004.

[3] S. Diekelmann and J. Born. The memory function of sleep. *Nature Reviews Neuroscience*, 11(2):114–126, 2010.

[4] E. Tulving. Episodic and semantic memory. In *Organization of Memory*, Academic Press, 1972.

[5] D. O. Hebb. *The Organization of Behavior*. Wiley, 1949.

[6] R. P. N. Rao and D. H. Ballard. Predictive coding in the visual cortex. *Nature Neuroscience*, 2(1):79–87, 1999.

[7] A. Clark. Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behavioral and Brain Sciences*, 36(3):181–204, 2013.

[8] D. Kahneman. *Thinking, Fast and Slow*. Farrar, Straus and Giroux, 2011.

[9] P. Lewis et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. *NeurIPS*, 2020.

[10] V. Karpukhin et al. Dense passage retrieval for open-domain question answering. *EMNLP*, 2020.

[11] S. Robertson and H. Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in IR*, 3(4):333–389, 2009.

[12] G. V. Cormack, C. L. A. Clarke, and S. Buettcher. Reciprocal rank fusion outperforms Condorcet and individual rank learning methods. *SIGIR*, 2009.