

ESCUELA DE CIENCIAS DE LA COMPUTACIÓN

Ingeniería en Sistemas de Información

Chatbot híbrido quantum-inspired con Small Language Models locales para una inteligencia artificial conversacional sostenible

Trabajo de Integración Curricular presentado como requisito previo a la obtención del título de Ingeniero en Sistemas de Información

LEONARDO GERMAN VEINTIMILLA

Director: Ing. Pablo Andrade

Quito, Ecuador

Septiembre 2026

Certificación de autoría

El autor, LEONARDO GERMAN VEINTIMILLA, declaro bajo juramento que el trabajo aquí descrito es de su autoría; que no ha sido presentado anteriormente para ningún grado o calificación profesional y que se ha consultado la bibliografía detallada.

El autor ceden sus derechos de propiedad intelectual a la Universidad Internacional del Ecuador (UIDE), para que sea publicado y divulgado en internet, según lo establecido en la Ley de Propiedad Intelectual, su reglamento y demás disposiciones legales.

D. M. Quito,

_____

Firma

Leonardo German Veintimilla

Autorización de Derechos de Propiedad Intelectual

El autor, LEONARDO GERMAN VEINTIMILLA, en calidad de autor del trabajo titulado «Chatbot híbrido quantum-inspired con Small Language Models locales para una inteligencia artificial conversacional sostenible», autoriza a la Universidad Internacional del Ecuador (UIDE) para hacer uso de todos los contenidos que le pertenecen o de parte de los que contiene esta obra, con fines estrictamente académicos o de investigación. Los derechos que como autor le corresponden son los establecidos en los artículos 5, 6, 8, 19 y demás pertinentes de la Ley de Propiedad Intelectual y su Reglamento en Ecuador.

D. M. Quito, Septiembre 2026

Firma

Leonardo German Veintimilla

Aprobación de dirección y coordinación del programa

Quien suscribe, Ing. Pablo Andrade, Director del presente trabajo de titulación, declara que LEONARDO GERMAN VEINTIMILLA**,** son los autores exclusivos de la presente investigación y que esta es original, auténtica y personal de ellos.

Firma	Firma
Director del trabajo de titulación	Coordinador/a del programa

DEDICATORIA

Dedico este trabajo, en primer lugar, a mi esposa y a mi hijo, quienes son la razón más importante de mi esfuerzo diario. Su amor, comprensión, paciencia y apoyo incondicional me han dado la fortaleza necesaria para continuar avanzando incluso en los momentos más difíciles. Cada logro que alcanzo tiene como propósito brindarles un mejor futuro y devolverles una parte de todo lo que me inspiran cada día.

A mi madre, por ser ejemplo de trabajo, sacrificio y perseverancia, y por enseñarme con sus acciones el verdadero valor de la responsabilidad y la honestidad.

A mis hermanas y sobrinos, quienes han sido una fuente constante de cariño, motivación y alegría a lo largo de este camino.

Y de manera muy especial, a mi padre, quien hoy me acompaña desde el cielo. Aunque ya no está físicamente entre nosotros, su ejemplo, sus enseñanzas y los valores que sembró en mí continúan guiando cada paso de mi vida. Este logro también es suyo y lo llevo con profundo amor y gratitud en mi corazón.

AGRADECIMIENTOS

Expreso mi más sincero agradecimiento a Dios por brindarme salud, fortaleza y la oportunidad de culminar esta importante etapa académica.

A la Universidad Internacional del Ecuador y a todos los docentes que formaron parte de mi proceso de aprendizaje, por compartir sus conocimientos, experiencias y orientación profesional, contribuyendo de manera significativa a mi crecimiento académico y personal.

A mi director de proyecto y a las personas que aportaron con sus observaciones, recomendaciones y acompañamiento durante el desarrollo de este trabajo de integración curricular.

A mi esposa y a mi hijo, por su paciencia, comprensión y apoyo incondicional durante los momentos de estudio, investigación y desarrollo de este proyecto. Su confianza en mí fue una motivación constante para seguir adelante.

A mi madre, hermanas y sobrinos, por su cariño, respaldo y palabras de aliento en cada etapa de este proceso.

Finalmente, agradezco a la memoria de mi padre, cuya enseñanza, ejemplo de vida y

valores continúan siendo una inspiración permanente para alcanzar mis metas y superar cada desafío.

RESUMEN

La adopción masiva de asistentes conversacionales basados en Large Language Models (LLM) ha generado capacidades notables, pero también efectos negativos crecientes: un alto consumo energético y una huella de carbono considerable en el entrenamiento y la inferencia, una dependencia de hardware especializado (unidades de procesamiento gráfico) y de servicios en la nube que eleva los costos de operación, y riesgos de privacidad derivados del procesamiento externo de los datos del usuario. Frente a esta problemática, el presente trabajo desarrolla un chatbot conversacional en español que opera íntegramente sobre hardware doméstico, sin unidad de procesamiento gráfico ni conexión a servicios de nube, integrando técnicas clásicas de aprendizaje automático con un Small Language Model (SLM) ejecutado de forma local.

El sistema se estructura en tres capas: un clasificador de intención basado en TF-IDF y Regresión Logística; un motor de contexto conversacional quantum-inspired que representa el diálogo como un vector semántico de doce dimensiones, actualizado mediante un promedio móvil exponencial y una matriz de acoplamiento; y una capa de generación que invoca al modelo qwen2.5:1.5b mediante Ollama. El motor de contexto extrae métricas de dispersión (entropía), continuidad (similitud coseno) y dimensión dominante, que se emplean para calibrar dinámicamente la estrategia de respuesta del modelo de lenguaje.

La evaluación experimental comparó el sistema frente a una línea base —el mismo modelo sin directiva de contexto— sobre un corpus de 57 consultas y 80 turnos conversacionales, con 160 generaciones ejecutadas en hardware doméstico sin unidad de procesamiento gráfico. Se midió la coherencia conversacional mediante rúbrica humana aplicada por jueces ciegos a la condición, la latencia de respuesta y la huella de carbono mediante estimación instrumentada con CodeCarbon, empleando el factor de emisión oficial de la red eléctrica ecuatoriana (0,1616 $kgCO_2eq/kWh$). Los resultados muestran que el sistema alcanza una coherencia media de 3,93 sobre 5 en términos absolutos, pero que la diferencia frente a la línea base (3,91) no resulta estadísticamente distinguible de cero (Wilcoxon $W = 1255,5$; $p = 0,8974$; $d_z = 0,04$; IC 95 % [−0,10; 0,14]). No se observó sobrecarga de latencia ni de emisiones atribuible al motor de contexto. La huella estimada fue de 0,010537 gCO_2eq por turno, con un total de 1,7142 gCO_2eq para la campaña completa. Un estudio de

sensibilidad factorial sobre veinticinco configuraciones de los hiperparámetros del motor no mostró fragilidad paramétrica en el rango explorado, sobre métricas internas del motor; el criterio de estabilidad preregistrado se cumplió, aunque resultó poco exigente por diseño. El trabajo aporta una arquitectura reproducible y de bajo costo para la inteligencia artificial conversacional local, con una huella operacional de inferencia cuantificada, junto con evidencia empírica de que la orientación del modelo por directiva textual no produce, por sí sola, una mejora medible de la coherencia percibida.

Palabras clave:

Chatbot híbrido; Small Language Models; Green AI; huella de carbono; procesamiento de lenguaje natural; motor de contexto quantum-inspired; inteligencia artificial sostenible.

TABLA DE CONTENIDOS

Certificación de autoría 2

Autorización de Derechos de Propiedad Intelectual 3

Aprobación de dirección y coordinación del programa 4

DEDICATORIA 5

AGRADECIMIENTOS 6

RESUMEN 8

LISTA DE TABLAS 14

LISTA DE FIGURAS 15

CAPÍTULO 1 17

INTRODUCCIÓN 17

1.1 Definición del proyecto 17

1.2 Justificación e importancia del trabajo de investigación 17

1.3 Alcance 18

1.4 Objetivos 19

1.4.1 Objetivo general 19

1.4.2 Objetivos específicos 19

CAPÍTULO 2 22

FUNDAMENTACIÓN TEÓRICA 22

2.1 Estado del Arte 22

2.1.1 Agentes conversacionales: evolución y tendencias 22

2.1.2 Green AI y huella ambiental de los modelos de lenguaje 22

2.1.3 Small Language Models como alternativa eficiente 23

2.1.4 Sistemas neuro-simbólicos 24

2.1.5 Representaciones vectoriales inspiradas en el formalismo cuántico 24

2.1.6 Síntesis y brecha identificada 25
2.2 Marco Teórico 28
2.2.1 Procesamiento de Lenguaje Natural clásico 28
2.2.2 Motor de contexto quantum-inspired 28
2.2.3 Small Language Models y su despliegue local 29
2.2.4 Sostenibilidad computacional y medición de huella de carbono 29
CAPÍTULO 3 31
MÉTODOS Y MATERIALES 31
3.1 Desarrollo del Trabajo 31
3.1.1 Visión general del sistema 31
3.1.2 Arquitectura de software y stack tecnológico 35
3.1.3 Componente 1: clasificador de intención 37
3.1.4 Componente 2: motor de contexto conversacional 38
3.1.5 Componente 3: capa de generación 40
3.1.6 Consideraciones de diseño para la sostenibilidad 41
3.1.7 Aporte diferencial del motor de contexto frente al SLM directo 41
3.1.8 Protección de datos personales y cumplimiento normativo 43
CAPÍTULO 4 46
RESULTADOS Y DISCUSIÓN 46
4.1 Pruebas de campo y pruebas individuales por componente 46
4.1.1 Entorno y condiciones de ejecución 46
4.1.2 Composición del corpus de prueba 47
4.1.3 Prueba de la capa 1: clasificador de intención 48
4.1.4 Prueba de la capa 2: motor de contexto 52
4.1.5 Prueba de la capa 3: generación con SLM local 57
4.2 Comparativa de resultados frente al baseline 57

4.2.1 Diseño de la comparación 57
4.2.2 Latencia 58
4.2.3 Huella de carbono 60
4.2.4 Coherencia conversacional 62
4.3 Estudios de sensibilidad de hiperparámetros 67
4.3.1 Diseño y alcance del estudio 67
4.3.2 Protocolo del estudio de sensibilidad 68
4.3.3 Criterio de identificación de la región de operación 69
4.3.4 Resultados de la malla y contraste de H_3 70
4.4 Discusión según variables 73
4.4.1 Variable coherencia 73
4.4.2 Variable latencia 75
4.4.3 Variable huella de carbono 76
4.4.4 Variable representación semántica 77
4.4.5 Contraste con la literatura 78
4.4.6 Estado de las hipótesis 79
4.4.7 Limitaciones del estudio 80
4.5 Síntesis del capítulo 83
CAPÍTULO 5 85
CONCLUSIONES Y RECOMENDACIONES 85
5.1 Conclusiones 85
5.1.1 Sobre el objetivo específico 1: arquitectura de software de tres capas 85
5.1.2 Sobre el objetivo específico 2: motor de contexto de doce dimensiones 85
5.1.3 Sobre el objetivo específico 3: evaluación experimental 87
5.1.4 Sobre el objetivo específico 4: dashboard de visualización 89
5.1.5 Conclusión general 90

5.2 Recomendaciones 90
5.2.1 Recomendaciones para la extensión del estudio de sensibilidad 90
5.2.2 Recomendaciones sobre las limitaciones identificadas 91
5.2.3 Recomendaciones para trabajos futuros 92
5.3 Presupuesto ejecutado 93
5.4 Referencias bibliográficas 94
5.5 Anexos 99
ANEXO A. FORMALIZACIÓN MATEMÁTICA DEL MOTOR DE CONTEXTO 100
A.1 Vector de estado semántico 100
A.2 Actualización por promedio móvil exponencial 101
A.3 Matriz de acoplamiento semántico 101
A.4 Distribución de probabilidad y entropía de dispersión 102
A.5 Continuidad conversacional 103
A.6 Selección de la dimensión dominante 103
A.7 Selección de la estrategia de respuesta 103
A.8 Detección de quiebre temático 104
A.9 Nota sobre la terminología 104
ANEXO B. RÚBRICA DE EVALUACIÓN DE COHERENCIA CONVERSACIONAL 105
Criterio 1. Pertinencia respecto de la consulta 105
Criterio 2. Continuidad temática con los turnos previos 105
Criterio 3. Corrección del contenido 106
Criterio 4. Adecuación de la extensión 106
Criterio 5. Naturalidad del registro 106
Procesamiento de las puntuaciones 107
ANEXO C. DOCUMENTACIÓN COMPLEMENTARIA 108
C.1 Vista general del panel de control 108

C.2 Panel izquierdo — Métricas del estado semántico 12D & Green AI 110

C.3 Panel derecho — Estado del sistema 113

C.4 Panel central — Interfaz conversacional 116

C.5 Detalle — Telemetría CodeCarbon (Green AI)..... 118

LISTA DE TABLAS

Tabla 2.1. Comparación de los trabajos revisados según las dimensiones de la brecha identificada

Tabla 3.1. Dimensiones semánticas y términos de activación del motor de contexto

Tabla 3.2. Coeficientes de acoplamiento entre dimensiones semánticas

Tabla 3.3. Componentes y tecnologías de la arquitectura de software

Tabla 4.1. Distribución del corpus experimental por categoría y tipo

Tabla 4.2. Exactitud del clasificador de intención por categoría de referencia

Tabla 4.3. Precisión, exhaustividad y F1 del clasificador de intención por categoría de referencia (n = 80)

Tabla 4.4. Distribución de directivas según los rangos de entropía normalizada

Tabla 4.5. Continuidad, fidelidad y entropía media por turno conversacional

Tabla 4.6. Frecuencia de activación dominante de las dimensiones semánticas

Tabla 4.7. Estadísticos descriptivos de latencia por condición experimental

Tabla 4.8. Contrastes inferenciales de latencia entre condiciones experimentales

Tabla 4.9. Estadísticos descriptivos de emisiones por condición experimental

Tabla 4.10. Contrastes inferenciales de emisiones entre condiciones experimentales

Tabla 4.11. Comparación del consumo energético con sistemas de referencia

Tabla 4.12. Latencia y emisiones por categoría temática y condición experimental

Tabla 4.13. Longitud de respuesta y variación de latencia por categoría temática

Tabla 4.14. Entropía y distribución de directivas por categoría temática

Tabla 4.15. Continuidad media del estado ($\psi_t \cdot \psi_{t-1}$) por configuración de la malla factorial (n = 23 transiciones por celda)

Tabla 4.16. Estado de las hipótesis y síntesis de la evidencia experimental

Tabla 4.17. Estadísticos descriptivos de la puntuación agregada de coherencia por condición

Tabla 4.18. Contrastes inferenciales de la puntuación agregada de coherencia entre condiciones

Tabla 5.1. Presupuesto de recursos para el desarrollo y la experimentación

Tabla A.1. Reglas de selección de directiva según la entropía del estado

Tabla B.1. Rúbrica de pertinencia respecto de la consulta

Tabla B.2. Rúbrica de continuidad temática con el contexto previo

Tabla B.3. Rúbrica de corrección factual de la respuesta

Tabla B.4. Rúbrica de proporcionalidad de la extensión de la respuesta

Tabla B.5. Rúbrica de naturalidad y adecuación lingüística en español

LISTA DE FIGURAS

Figura 3.1. Vista general del panel de control (Lab Dashboard) en su estado inicial, con los tres paneles y la configuración nominal de hiperparámetros visible en el encabezado.

Figura 3.2. Arquitectura de tres capas del artefacto y flujo de datos por turno conversacional.

Figura 4.1. Composición del corpus experimental por categoría temática y tipo de consulta (n = 57).

Figura 4.2. Exactitud del clasificador de intención por categoría de referencia, ordenada de mayor a menor, con la exactitud global (0,600) como referencia.

Figura 4.3. Rango (mínimo–máximo) y media de la entropía normalizada dentro de cada una de las tres directivas de generación.

Figura 4.4. Frecuencia con que cada una de las doce dimensiones semánticas resultó dominante en el estado del turno (n = 80).

Figura 4.5. Latencia media (en segundos), con barras de error correspondientes a la desviación estándar, por categoría temática y condición experimental.

Figura 4.6. Emisiones medias de CO_2 equivalente por turno (gCO_2eq/turno), por categoría temática y condición experimental.

Figura 4.7. Distribución porcentual de la directiva de generación seleccionada (directa, equilibrada, exploratoria) por categoría temática.
Figura 4.8. Evolución de la continuidad media y de la entropía media a lo largo de los tres primeros turnos, restringida a los 17 diálogos multiturno de la campaña.
Figura 4.9. Mapa de sensibilidad factorial de la continuidad media del estado en el plano (α, c) sobre el subconjunto multiturno (23 transiciones por configuración). La banda de estabilidad corresponde al criterio de ±0,05 preregistrado en 4.3.3.
Figura 4.10. Dispersión de la longitud de la respuesta (en palabras) frente a la latencia total (en milisegundos), condiciones A y B, n = 160 generaciones.
Figura A.1. Secuencia de cómputo del motor de contexto en cada turno, correspondiente a las ecuaciones A.1 a A.7 del Anexo A.
Figura A.2. Grafo de la matriz de acoplamiento semántico C entre las doce dimensiones del vector de estado, con el peso de cada uno de los diez enlaces definidos en COUPLING_LINKS
Figura C.1. Vista general del Lab Dashboard mostrando los tres paneles —métricas del estado semántico, interfaz conversacional y estado del sistema— en su estado inicial, previo al primer turno.
Figura C.2. Panel de métricas del estado semántico y telemetría CodeCarbon: entropía de Shannon normalizada, fidelidad semántica, concurrencia, divergencia KL, inercia, temperatura de exploración y modalidad de directiva del turno.
Figura C.3. Panel de estado del sistema: mapa de las doce dimensiones semánticas, dimensiones dominantes, pares acoplados, salida del clasificador de intención (6 clases) e historial de turnos.
Figura C.4. Interfaz conversacional del dashboard, mostrando el intercambio con el usuario, la etiqueta de la modalidad de directiva del turno y la barra de telemetría (modelo, entropía, fidelidad, fuente de la respuesta, latencia, emisiones).
Figura C.5. Detalle de la tarjeta de telemetría CodeCarbon del panel de métricas del estado semántico, correspondiente a las variables reportadas en la sección 4.2.3.

CAPÍTULO 1

INTRODUCCIÓN

1.1 Definición del proyecto

El presente trabajo de titulación desarrolla un chatbot conversacional híbrido en español, diseñado para operar íntegramente sobre hardware doméstico, sin unidad de procesamiento gráfico (GPU) ni dependencia de servicios en la nube. El sistema integra tres componentes de software complementarios: un clasificador de intención basado en técnicas clásicas de aprendizaje automático (TF-IDF y Regresión Logística), un motor de contexto conversacional denominado quantum-inspired —por adoptar de forma metafórica, y no computacional, tres principios del formalismo cuántico (superposición, entrelazamiento y colapso) aplicados a un espacio vectorial de doce dimensiones—, y una capa de generación de lenguaje natural que invoca un Small Language Model (SLM) ejecutado localmente mediante Ollama.

El proyecto se enmarca en la corriente de investigación conocida como Green AI, que busca reducir la huella ambiental y económica de los sistemas de inteligencia artificial sin sacrificar su utilidad. A diferencia de los asistentes conversacionales comerciales, que dependen de modelos de lenguaje de gran escala alojados en centros de datos, el sistema propuesto explora la viabilidad de ofrecer una interacción contextualizada empleando exclusivamente modelos pequeños y recursos computacionales modestos.

1.2 Justificación e importancia del trabajo de investigación

La pertinencia del proyecto se sustenta en evidencia científica reciente sobre tres efectos negativos, cada vez mejor documentados, de los asistentes conversacionales basados en modelos de lenguaje de gran escala. El primero es de carácter ambiental: se ha estimado que el entrenamiento de un solo modelo grande puede emitir cientos de toneladas de dióxido de carbono equivalente (Strubell et al., 2019; Schwartz et al., 2020), y que el ciclo de vida completo de los chatbots impulsados por estos modelos incrementa de forma considerable su huella

energética (Jiang et al., 2024). El segundo efecto es económico y de acceso: la dependencia de unidades de procesamiento gráfico especializadas y de servicios de terceros en la nube impone costos de operación elevados que excluyen a estudiantes, pequeñas organizaciones e instituciones con recursos limitados, restricción especialmente relevante en contextos como el ecuatoriano. El tercer efecto concierne a la privacidad y la soberanía de los datos: al procesarse las conversaciones en servidores externos, la información del usuario queda fuera de su control, situación problemática en dominios sensibles como la salud o la educación. Esta preocupación se inscribe en un debate más amplio sobre los riesgos de los modelos de lenguaje de gran escala, que la literatura ha sistematizado en categorías que incluyen las filtraciones de información privada, la reproducción de sesgos y los costos ambientales del entrenamiento (Bender et al., 2021; Weidinger et al., 2021).

Frente a estos efectos, el trabajo se justifica en tres dimensiones. En el plano científico, aporta una arquitectura híbrida interpretable y reproducible, y una metodología para medir empíricamente la huella de carbono de un asistente conversacional ejecutado localmente, aspecto todavía poco explorado en la literatura de habla hispana. En el plano social, al operar sobre hardware modesto y sin nube, el sistema democratiza el acceso a la inteligencia artificial conversacional, extremo que se documenta de forma expresa en el apartado 3.1.8 conforme al régimen ecuatoriano de protección de datos personales. En el plano económico y ambiental, la eliminación de los costos de unidad de procesamiento gráfico y de servicios en la nube, junto con la priorización de modelos pequeños, reduce tanto el costo operativo como el consumo energético y las emisiones asociadas, en línea con los principios de Green AI y los Objetivos de Desarrollo Sostenible.

1.3 Alcance

El alcance del proyecto comprende el diseño, la implementación y la evaluación experimental de un sistema de chatbot conversacional en idioma español, con las siguientes delimitaciones:

- El sistema se ejecuta exclusivamente sobre hardware doméstico (procesador central, sin unidad de procesamiento gráfico), lo que constituye tanto su principal restricción técnica como su aporte diferenciador frente a soluciones basadas en la nube.
- La interacción del sistema se limita al idioma español; no se contempla soporte multilingüe en la presente fase del proyecto.
- La evaluación experimental se circunscribe a un corpus de conversaciones representativas construido para este trabajo, del orden de cincuenta a cien consultas, y no pretende ser exhaustiva frente a la diversidad total de posibles interacciones conversacionales.
- El desarrollo del trabajo de integración curricular se planifica para un periodo de tres meses (doce semanas), lo que acota la profundidad de los estudios de sensibilidad de hiperparámetros a los que resulten viables dentro de dicho plazo.
- El dashboard de visualización constituye un entregable de desarrollo, previsto en el objetivo específico 4, cuya función es de inspección y evidencia. No es objeto de hipótesis ni de evaluación experimental: no se mide su usabilidad ni su efecto sobre el usuario.

1.4 Objetivos

1.4.1 Objetivo general

Diseñar, implementar y evaluar experimentalmente un chatbot conversacional híbrido quantum-inspired que integre un Small Language Model local, un motor de contexto semántico de doce dimensiones y un clasificador de intención clásico, operando en español sobre hardware doméstico sin unidad de procesamiento gráfico ni dependencia de la nube, y cuantificando su coherencia conversacional, su latencia y su huella ambiental de operación mediante estimación instrumentada en tiempo de ejecución.

1.4.2 Objetivos específicos

1. Diseñar e implementar la arquitectura de software de tres capas del sistema —clasificador de intención, motor de contexto conversacional y capa de generación con Small Language Model local—, aplicando principios de modularidad y de degradación gradual del servicio (graceful degradation).
2. Construir el motor de contexto quantum-inspired que representa el diálogo como un vector semántico de doce dimensiones, actualizado turno a turno

mediante un promedio móvil exponencial y una matriz de acoplamiento de doce por doce, y que extrae métricas de dispersión (entropía), continuidad (similitud coseno) y dimensión dominante.

3. Evaluar experimentalmente el sistema frente a una línea base, midiendo coherencia conversacional, latencia de respuesta y huella de carbono mediante estimación instrumentada con CodeCarbon, y realizando estudios de sensibilidad sobre los hiperparámetros del motor.
4. Desarrollar un panel de visualización en tiempo real que exponga métricas internas del motor de contexto (entropía, similitud coseno, dimensión dominante), sirviendo como herramienta de inspección y evidencia visual.

1.5 Problema de investigación, preguntas e hipótesis

El problema que orienta la investigación se formula del modo siguiente: ¿es posible diseñar un chatbot conversacional coherente en español que opere íntegramente sobre hardware doméstico, sin unidad de procesamiento gráfico ni dependencia de la nube, integrando un Small Language Model local con un motor de contexto quantum-inspired y un clasificador clásico, y estimando su huella ambiental de operación mediante instrumentación en tiempo de ejecución?

De esta formulación se derivan cuatro preguntas de sistematización:

1. ¿Qué arquitectura de software permite combinar clasificación clásica, gestión de contexto vectorial y generación con un Small Language Model local de forma modular y robusta?

2. ¿Qué métricas internas —dispersión por entropía, continuidad por similitud coseno y dimensión dominante— resultan suficientes para guiar la estrategia de respuesta sin recurrir a modelos de mayor tamaño?

3. ¿Cómo se comporta el sistema, en coherencia y latencia, frente a una línea base al variar sus hiperparámetros?

4. ¿Cuál es la huella de carbono del sistema operando localmente, estimada con instrumentación en tiempo de ejecución, frente a las estimaciones publicadas para soluciones basadas en modelos de gran escala?

El contraste experimental se organiza en torno a tres hipótesis, evaluadas en el Capítulo 4:

H_1. La incorporación del motor de contexto mejora la coherencia conversacional respecto de la línea base, medida mediante rúbrica de cinco criterios aplicada por jueces ciegos.

H_2. El costo computacional del motor es aceptable: la sobrecarga de latencia respecto de la línea base es inferior a 500 ms por turno y la sobrecarga de emisiones es inferior a 1 gCO_2eq por turno.

H_3. Existe una región compacta y contigua del espacio de hiperparámetros (α, c) en la que el desempeño del motor se mantiene estable.

CAPÍTULO 2

FUNDAMENTACIÓN TEÓRICA

2.1 Estado del Arte

La revisión se realizó sobre Scopus, IEEE Xplore, ACM Digital Library y Google Scholar, mediante combinaciones de los términos *green AI*, *small language model*, *neuro-symbolic*, *quantum-inspired NLP*, *conversational agent* y *carbon footprint*, en inglés y español. Se aplicaron como criterios de inclusión la publicación entre 2013 y 2025, la disponibilidad de texto completo y la pertinencia directa respecto de alguna de las cinco corrientes identificadas; y como criterios de exclusión los trabajos centrados exclusivamente en computación cuántica con hardware, salvo cuando se citan como antecedente conceptual, en modelos de gran escala sin consideración de eficiencia, y las publicaciones sin revisión por pares. De los trabajos localizados se seleccionaron los que se reportan en las secciones siguientes, priorizando publicaciones en revistas y conferencias de referencia en cada corriente.

2.1.1 Agentes conversacionales: evolución y tendencias

Los agentes conversacionales han evolucionado desde los sistemas basados en reglas de la década de 1960 hasta los modelos generativos contemporáneos. Schöbel et al. (2024), en un estudio bibliométrico sobre más de cinco mil artículos publicado en Information Systems Frontiers, describen la evolución del área en cinco olas. Las cuatro primeras se derivan del corpus analizado, compuesto por artículos con fecha de corte en 2021; la quinta es una proyección de los autores sobre la dirección futura de la investigación y no un resultado del análisis bibliométrico.

2.1.2 Green AI y huella ambiental de los modelos de lenguaje

La corriente de investigación conocida como Green AI cuestiona el supuesto de que los avances en inteligencia artificial deban lograrse mediante modelos cada vez más grandes. Schwartz et al. (2020), en un trabajo influyente publicado en

Communications of the ACM, argumentan que la eficiencia computacional debe considerarse un criterio de evaluación tan relevante como la exactitud, y documentan el crecimiento exponencial del costo computacional de los modelos de lenguaje. Strubell et al. (2019) cuantifican, en la conferencia de la Association for Computational Linguistics, el consumo energético y las emisiones asociadas al entrenamiento de modelos de procesamiento de lenguaje natural, reportando que la búsqueda de arquitectura neuronal (neural architecture search) de un modelo Transformer puede generar aproximadamente 626 155 libras de CO2 equivalente —cerca de 284 toneladas métricas—, una cifra comparable a las emisiones de varios automóviles durante toda su vida útil. Más recientemente, Jiang et al. (2024), en la revista Engineering, extienden este análisis al ciclo de vida completo de chatbots comerciales impulsados por modelos de lenguaje de gran escala, evidenciando que la fase de inferencia —y no solo el entrenamiento— constituye una fuente significativa y creciente de huella de carbono a escala de despliegue masivo (Patterson et al., 2021).

2.1.3 Small Language Models como alternativa eficiente

La viabilidad de los modelos de lenguaje de pequeña escala descansa sobre un conjunto de técnicas de compresión desarrolladas durante la última década. La destilación de conocimiento permite transferir el comportamiento de un modelo grande a uno considerablemente menor mediante el entrenamiento sobre las distribuciones de salida del primero (Hinton et al., 2015); su aplicación a modelos de lenguaje preentrenados como BERT (Devlin et al., 2019) dio lugar a variantes que conservan la mayor parte del desempeño con aproximadamente el sesenta por ciento del tamaño (Sanh et al., 2019). De forma complementaria, las técnicas de cuantización reducen la precisión numérica de los parámetros y, con ello, la memoria necesaria para la inferencia; Lu et al. (2024) miden la latencia de inferencia y la huella de memoria de modelos de lenguaje pequeños ejecutados en el propio dispositivo. Sobre esta base han surgido modelos abiertos entrenados desde su origen a escala reducida, como TinyLlama (Zhang et al., 2024), que documentan la viabilidad de operar por debajo de los dos mil millones de parámetros. Revisiones

recientes sistematizan este ecosistema: Wang et al. (2025) ofrecen una revisión exhaustiva de sus técnicas, aplicaciones y formas de colaboración con los modelos de gran escala. En el extremo de mayor restricción de recursos, Scherer et al. (2024) demuestran el despliegue energéticamente eficiente de modelos de lenguaje pequeños sobre microcontroladores heterogéneos, lo que refuerza la pertinencia de la ejecución local como estrategia de sostenibilidad.

2.1.4 Sistemas neuro-simbólicos

El paradigma neuro-simbólico busca combinar la capacidad de aprendizaje de las redes neuronales con la interpretabilidad y el razonamiento explícito de los sistemas simbólicos. Garcez y Lamb (2023), en Artificial Intelligence Review, caracterizan esta corriente como la «tercera ola» de la inteligencia artificial neuro-simbólica y proponen un marco para clasificar los sistemas híbridos según el grado de integración entre sus componentes neuronales y simbólicos. Yu (2025) aplica principios afines en un sistema multiagente de codificación clínica que opera de forma local y preserva la privacidad de los datos, ofreciendo un precedente de arquitectura híbrida neuro-simbólica desplegada sin infraestructura de nube, en un dominio distinto pero conceptualmente cercano al abordado en este trabajo.

2.1.5 Representaciones vectoriales inspiradas en el formalismo cuántico

Un conjunto más reducido, pero creciente, de trabajos explora la aplicación de conceptos del formalismo cuántico —sin requerir hardware cuántico— a la representación semántica del lenguaje natural. Widdows et al. (2024), en Annals of Mathematics and Artificial Intelligence, reportan experimentos de procesamiento de lenguaje natural ejecutados en computadores cuánticos reales y en simulación: clasificación temática, modelado de bigramas y resolución de ambigüedad. Se cita aquí como antecedente conceptual del uso de vectores normalizados y de probabilidades derivadas de amplitudes para representar significado, y no como precedente del enfoque adoptado, que es íntegramente clásico. Panahi et al. (2020) presentan, en la International Conference on Learning Representations, una técnica de representación de palabras («word2ket») inspirada en el entrelazamiento

cuántico que logra representaciones vectoriales más compactas y eficientes. Estos trabajos fundamentan la decisión de adoptar, en el presente proyecto, un motor de contexto que emplea vectores normalizados, una matriz de acoplamiento y una selección de la dimensión dominante a partir de una distribución de probabilidad como analogías —explícitamente no computacionales— de la superposición, el entrelazamiento y el colapso.

A diferencia de una representación vectorial convencional —por ejemplo, un vector TF-IDF no normalizado o un embedding denso aprendido por un modelo neuronal, en la línea de word2vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014) o las representaciones a nivel de oración de Sentence-BERT (Reimers & Gurevych, 2019)—, el formalismo adoptado impone tres restricciones adicionales que aportan valor funcional. Primero, la normalización obligatoria evita que una categoría con alta frecuencia de términos domine numéricamente al resto, un problema común en representaciones de frecuencia cruda. Segundo, el operador de acoplamiento simétrico modela explícitamente las correlaciones semánticas entre categorías temáticas, en lugar de dejarlas implícitas y no auditables en la geometría de un espacio de alta dimensionalidad, como ocurre en los embeddings entrenados por modelos neuronales. Tercero, la extracción de la dimensión dominante mediante una distribución de probabilidad derivada del propio vector de estado permite una decisión interpretable sobre la directiva de generación, en lugar de depender de umbrales arbitrarios aplicados sobre valores numéricos sin una interpretación probabilística clara.

2.1.6 Síntesis y brecha identificada

La revisión de la literatura evidencia una convergencia de intereses entre la sostenibilidad computacional (Green AI), la eficiencia de los Small Language Models y los sistemas neuro-simbólicos interpretables, pero también revela una brecha: son escasos los trabajos que integren estas tres corrientes en un sistema conversacional único, evaluado empíricamente en términos de coherencia, latencia y huella de carbono estimada con instrumentación en tiempo de ejecución,

particularmente en el contexto de habla hispana y bajo la matriz energética de un país específico como Ecuador. El presente trabajo se propone abordar esta brecha.

Tabla 2.1. Comparación de los trabajos revisados según las dimensiones de la brecha identificada

Trabajo	SLM local	Contexto explícito	Quantum-inspired	Green AI	Español	CPU sin GPU	Huella medida
Schöbel et al. (2024)	—	—	—	—	—	—	—
Schwartz et al. (2020)	—	—	—	Sí	—	—	—
Strubell et al. (2019)	—	—	—	Sí	—	—	Sí
Jiang et al. (2024)	—	—	—	Sí	—	—	Sí
Sanh et al. (2019)	Sí	—	—	Parcial	—	—	—
Lu et al. (2024)	Sí	—	—	Parcial	—	Parcial	—
Zhang et al. (2024)	Sí	—	—	Parcial	—	—	—
Garcez y Lamb (2023)	—	Sí	—	—	—	—	—
Yu (2025)	Sí	Sí	—	—	—	Parcial	—
Widdows et al. (2024)	—	Parcial	Parcial	—	—	—	—
Panahi et al. (2020)	—	—	Sí	Parcial	—	—	—

Trabajo	SLM local	Contexto explícito	Quantum-inspired	Green AI	Español	CPU sin GPU	Huella medida
Presente trabajo	**Sí**	**Sí**	**Sí**	**Sí**	**Sí**	**Sí**	**Sí**

Nota. indica que el trabajo aborda la dimensión de forma explícita; «Parcial», que la aborda de manera tangencial o como consecuencia secundaria; «—», que no la aborda. La clasificación se realizó sobre los trabajos citados en las secciones 2.1.1 a 2.1.5.

La Tabla 2.1 permite precisar la brecha. Ningún trabajo revisado cubre simultáneamente las siete dimensiones. Los trabajos de Green AI cuantifican la huella pero no proponen arquitecturas conversacionales con gestión explícita de contexto; los de Small Language Models documentan la viabilidad de la escala reducida sin evaluar su impacto ambiental en operación; los neuro-simbólicos aportan interpretabilidad sin consideración de sostenibilidad; y los de representación inspirada en el formalismo cuántico permanecen en el nivel de la representación léxica sin integrarse en un sistema conversacional completo. La combinación de las siete dimensiones en un artefacto evaluado empíricamente sobre la matriz energética ecuatoriana constituye el aporte diferencial de este trabajo.

Conviene precisar qué parte del motor de contexto constituye aporte y qué parte es decisión de implementación. Un mecanismo convencional de gestión de contexto —ventana deslizante de turnos, resumen del historial o recuperación de fragmentos— conserva el contexto como texto y deja al modelo generador la tarea de interpretarlo. El motor propuesto se diferencia en tres características. Primera, representa el contexto como un estado numérico explícito, de baja dimensionalidad y con etiquetas legibles, inspeccionable turno a turno. Segunda, deriva de ese estado magnitudes cuantitativas de dispersión y de alineamiento que no existen en una ventana de texto. Tercera, convierte esas magnitudes en una decisión determinista y auditable sobre la estrategia de respuesta, con un costo de cómputo despreciable frente a la generación.

Estas tres características constituyen el aporte de la propuesta. Es un aporte de interpretabilidad y de instrumentación, no de calidad conversacional: los resultados del Capítulo 4 no evidencian una mejora de la coherencia frente a la línea base. Son, en cambio, decisiones de implementación, sustituibles sin alterar la propuesta: el número de dimensiones, el vocabulario de activación, los coeficientes de la matriz de acoplamiento, los valores de α, c y de los umbrales, el criterio de selección de la dimensión dominante y la denominación quantum-inspired, que es una analogía terminológica.

2.2 Marco Teórico

2.2.1 Procesamiento de Lenguaje Natural clásico

El clasificador de intención del sistema se apoya en dos técnicas clásicas y ampliamente validadas del procesamiento de lenguaje natural: la vectorización TF-IDF (Term Frequency–Inverse Document Frequency), que representa cada mensaje como un vector numérico ponderado según la frecuencia de sus términos y su especificidad dentro del corpus, y la Regresión Logística, un modelo de clasificación supervisada que estima la probabilidad de pertenencia de un mensaje a cada categoría temática. Ambas técnicas se implementan mediante la biblioteca scikit-learn (Pedregosa et al., 2011), un estándar de facto en la comunidad de aprendizaje automático en Python por su robustez, documentación y eficiencia computacional en tareas de esta escala.

2.2.2 Motor de contexto quantum-inspired

El componente central y original del sistema es un motor de contexto que representa el estado de la conversación como un vector semántico normalizado de doce dimensiones. Su diseño se sustenta en tres principios matemáticos: primero, la actualización del contexto mediante un promedio móvil exponencial (EWMA), que combina el vector del mensaje actual con el del turno anterior, suavizando la evolución del contexto y evitando rupturas abruptas ante mensajes breves; segundo, una matriz de acoplamiento simétrica que modela correlaciones lingüísticas entre categorías semánticas relacionadas, de forma análoga a un operador de evolución

en un espacio vectorial; y tercero, una entropía de Shannon normalizada, que cuantifica la dispersión semántica del mensaje de cada turno y orienta la estrategia de respuesta del sistema entre posturas directas, equilibradas o exploratorias. La formalización matemática completa de estos mecanismos se documenta en el Anexo A del presente trabajo.

2.2.3 Small Language Models y su despliegue local

Un Small Language Model es un modelo de lenguaje generativo con un número de parámetros considerablemente menor al de los modelos de gran escala —típicamente por debajo de los diez mil millones—, lo que permite su ejecución en hardware de consumo sin unidad de procesamiento gráfico dedicada. El sistema emplea el modelo qwen2.5:1.5b, servido localmente mediante Ollama, una herramienta que expone modelos de lenguaje a través de una interfaz local sencilla, sin requerir conexión a servicios externos. La elección de un modelo de mil quinientos millones de parámetros responde al equilibrio entre capacidad conversacional suficiente y viabilidad de ejecución en tiempo razonable sobre un procesador central convencional.

2.2.4 Sostenibilidad computacional y medición de huella de carbono

La huella de carbono del sistema se estima mediante CodeCarbon (Schmidt et al., 2021), una herramienta de código abierto que estima el consumo energético de un proceso de cómputo a partir de la utilización de procesador y memoria y lo convierte en emisiones equivalentes de dióxido de carbono a partir del factor de emisión de la red eléctrica correspondiente. El procedimiento sigue el método de estimación propuesto por Lacoste et al. (2019), que combina el consumo energético estimado con la intensidad carbónica de la red que suministra la electricidad. Para este trabajo se emplea el factor de emisión oficial de la red eléctrica ecuatoriana continental —0,1616 toneladas de CO_2 equivalente por megavatio-hora—, publicado por el Operador Nacional de Electricidad y el Ministerio de Ambiente y Energía del Ecuador y calculado bajo la metodología del Alcance 2 del GHG Protocol (Ministerio de Ambiente y Energía del Ecuador y Operador Nacional de Electricidad, 2025). La

magnitud comparativamente baja de este factor se explica porque la generación de bajo costo —hidráulica y renovable no convencional— representó el 87,11 % promedio de la generación del Sistema Nacional Interconectado en el quinquenio analizado en dicho informe.

CAPÍTULO 3

MÉTODOS Y MATERIALES

Nota metodológica sobre el estado del artefacto. *Con anterioridad a la aprobación del plan de trabajo se construyó, a título exploratorio y como prueba de concepto, una primera versión funcional del sistema, con el fin de validar la viabilidad técnica de la propuesta antes de comprometer el periodo formal de titulación a una arquitectura no probada; dicho prototipo no incluyó evaluación experimental alguna. Durante el periodo de titulación el diseño fue revisado, formalizado y documentado en su forma actual, y sometido a la evaluación experimental que se reporta en el Capítulo 4.*

3.1 Desarrollo del Trabajo

El desarrollo del artefacto sigue el ciclo de Design Science Research, paradigma de investigación que propone la construcción y evaluación rigurosa de artefactos tecnológicos como forma legítima de generación de conocimiento científico (Hevner et al., 2004). La secuencia de actividades adoptada —identificación del problema, definición de objetivos de la solución, diseño y desarrollo, demostración, evaluación y comunicación— corresponde a la metodología de seis pasos formalizada por Peffers et al. (2007). Los capítulos 3 y 4 del presente trabajo cubren, respectivamente, las fases de diseño y desarrollo, y las de demostración y evaluación.

3.1.1 Visión general del sistema

El sistema desarrollado combina tres componentes de software para producir respuestas en lenguaje natural a partir de mensajes del usuario, operando completamente en hardware doméstico, sin requerir unidad de procesamiento gráfico ni conexión a servicios externos en la nube. Desde la perspectiva del usuario, el sistema recibe un mensaje de texto en español y devuelve una respuesta también en texto; internamente, cada mensaje atraviesa tres capas de procesamiento secuencial: una capa de clasificación que identifica la categoría temática del mensaje, una capa de gestión de contexto que mantiene un registro

acumulado de los temas abordados durante la conversación, y una capa de generación que produce la respuesta final invocando un modelo de lenguaje pequeño ejecutado localmente.

Este patrón de arquitectura híbrida sigue lo que la literatura denomina sistemas neuro-simbólicos de tercera ola (Garcez & Lamb, 2023), al combinar componentes neuronales —el modelo de lenguaje— con componentes simbólicos y estructurados —las reglas de clasificación y de gestión de contexto—. El trabajo de Yu (2025) en codificación clínica, que demuestra la viabilidad de un patrón de tres capas operando sin unidad de procesamiento gráfico, sirvió como referencia directa para el diseño aquí propuesto.

Precisión sobre la calificación neuro-simbólica. La arquitectura se califica como neuro-simbólica en el sentido específico de Garcez y Lamb (2023) porque combina un único componente neuronal —el Small Language Model qwen2.5:1.5b, encargado exclusivamente de la generación de lenguaje natural— con dos componentes simbólicos explícitos: el clasificador de intención (TF-IDF y Regresión Logística, cuyos pesos son inspeccionables) y el motor de contexto vectorial, que opera mediante reglas deterministas —promedio móvil exponencial, matriz de acoplamiento fija y umbrales de entropía— y no mediante parámetros aprendidos por descenso de gradiente. El sistema no constituye, por tanto, ni una arquitectura neuronal de extremo a extremo (end-to-end) ni un sistema puramente basado en reglas; la separación explícita de responsabilidades entre ambos paradigmas es una decisión de diseño deliberada, coherente con el uso que la literatura hace del término, y no una etiqueta aplicada de forma laxa.

Tabla 3.1. Dimensiones semánticas y términos de activación del motor de contexto

#	Dimensión	Descripción funcional	Términos de activación
0	Emoción	Estados afectivos y bienestar subjetivo	feliz, triste, cansado, estrés, bien, mal, amor, miedo, alegre, ansioso
1	Lógica	Razonamiento técnico, código y tecnología	código, python, función, error, bug, api, clase, variable, sql, bucle

#	Dimensión	Descripción funcional	Términos de activación
2	Incertidumbre	Formulación de dudas, preguntas y exploración	cómo, qué, porque, cuándo, dónde, cuál, duda, pregunta, explicar
3	Tiempo	Urgencia, planificación y plazos	mañana, hoy, después, antes, urgente, rápido, plazo, plan, ahora
4	Creación	Creatividad, diseño y construcción	crear, diseñar, inventar, nuevo, idea, innovar, construir, generar
5	Análisis	Comparación, evaluación y contraste	comparar, analizar, evaluar, diferencia, mejor, peor, versus
6	Conexión	Continuidad discursiva y referencia a lo previo	entonces, además, relación, contexto, anterior, seguir
7	Profundidad	Detalle técnico, especificidad y precisión	específico, detalle, exactamente, técnico, avanzado, profundo, complejo
8	Social	Colaboración e interacción entre personas	nosotros, equipo, juntos, ayuda, colaborar, compartir, usuario, persona
9	Aprendizaje	Estudio, comprensión y desarrollo personal	aprender, estudiar, entender, saber, conocimiento, curso, tutorial
10	Decisión	Elección entre alternativas y recomendación	elegir, decidir, opción, alternativa, escoger, recomendación, consejo
11	Meta	Objetivos, propósitos y resultados esperados	objetivo, meta, lograr, conseguir, resultado, quiero, necesito

El diseño de este vocabulario explica de forma directa uno de los resultados reportados en la sección 4.1.4: la dimensión de incertidumbre resultó dominante en el 46,3 % de los turnos porque sus términos ancla son los marcadores interrogativos del español, presentes en la práctica totalidad de las consultas de un corpus construido mayoritariamente con preguntas.

Tabla 3.2. Coeficientes de acoplamiento entre dimensiones semánticas

Dimensión origen	Dimensión destino	Coeficiente	Justificación lingüística
Emoción	Social	0,65	Los estados afectivos se expresan predominantemente en contextos de interacción
Emoción	Aprendizaje	0,35	La disposición emocional condiciona los procesos de estudio
Lógica	Análisis	0,70	El razonamiento técnico invoca operaciones de comparación y evaluación
Lógica	Profundidad	0,65	Las consultas técnicas tienden a requerir especificidad
Lógica	Creación	0,50	La construcción de soluciones combina razonamiento y diseño
Incertidumbre	Aprendizaje	0,60	La formulación de dudas es el mecanismo primario del aprendizaje
Incertidumbre	Decisión	0,55	La duda precede a la elección entre alternativas
Tiempo	Meta	0,70	Los plazos se definen respecto de objetivos
Análisis	Decisión	0,75	La evaluación de alternativas antecede a la elección
Conexión	Social	0,50	La continuidad discursiva es un fenómeno de interacción

Los coeficientes no consignados en la Tabla 3.2 son nulos, y la diagonal principal es unitaria. La matriz es simétrica por construcción, de modo que el acoplamiento entre dos dimensiones opera en ambos sentidos. Este mecanismo explica el segundo resultado de la sección 4.1.4: la dimensión social alcanzó el 33,8 % de dominancia

pese a no ser activada léxicamente en esa proporción, por recibir activación indirecta desde emoción y desde conexión.

3.1.2 Arquitectura de software y stack tecnológico

El sistema se implementó en Python 3.12, por la madurez de su ecosistema de aprendizaje automático y procesamiento numérico. El código se organiza en clases independientes que se comunican mediante interfaces bien definidas, siguiendo el principio de responsabilidad única: la clase orquestadora coordina la invocación de los tres componentes funcionales, mientras que clases auxiliares especializadas se encargan, respectivamente, del clasificador de intención, de la codificación semántica del contexto, de la actualización del estado conversacional y del empaquetado de las métricas para el dashboard. Esta separación permite que, por ejemplo, la sustitución futura del clasificador por uno basado en transformers afecte a un único componente sin repercutir en el resto del sistema.

Tabla 3.3. Componentes y tecnologías de la arquitectura de software

Componente	Tecnología	Rol en el sistema
Cómputo numérico	numpy ≥ 1.24	Operaciones vectoriales y matriciales del motor de contexto
Aprendizaje automático	scikit-learn ≥ 1.3, joblib ≥ 1.3	Clasificador TF-IDF + Regresión Logística y su persistencia
Comunicación con el SLM	httpx ≥ 0.27	Cliente HTTP para la comunicación con Ollama
API / servicio	FastAPI ≥ 0.110, uvicorn ≥ 0.27, pydantic ≥ 2.0	Endpoints REST, servidor ASGI y validación de esquemas
Modelo de lenguaje	qwen2.5:1.5b vía Ollama	Generación de la respuesta en lenguaje natural, en CPU local
Visualización	Dashboard HTML/CSS/JS	Inspección en tiempo real de las métricas internas

El modelo de lenguaje se ejecuta a través de Ollama, una plataforma de código abierto que sirve modelos cuantizados localmente mediante un servidor HTTP simple. El modelo empleado, qwen2.5:1.5b, es un Small Language Model de aproximadamente mil quinientos millones de parámetros que ocupa menos de dos gigabytes de memoria y puede generar respuestas en un procesador central convencional, sin aceleración por unidad de procesamiento gráfico.

El sistema expone tres endpoints principales mediante FastAPI: uno que procesa un mensaje del usuario y devuelve la respuesta junto con las métricas internas del turno; uno que reporta el estado general del sistema; y uno que devuelve la definición de las doce dimensiones del motor de contexto. Esta interfaz permite que el dashboard consuma el sistema en tiempo real, sirviendo como herramienta de inspección durante el desarrollo y como evidencia demostrativa para la defensa del trabajo.

```html
<header>
  <div class="logo">QUANTUM<span>BOT</span> LAB ...</div>
  <div class="header-tags">
    <span class="tag tag-c" id="tag-params">α=0.3 c=0.5 θ=0.5</span>
    <span class="tag tag-p" id="tag-mode">RAPIDO</span>
    <span class="tag tag-g" id="tag-qubits">12Q</span>
    <span class="tag tag-a" id="tag-llm">LLM OFF</span>
    <span class="tag tag-g" id="tag-green">🌿 0.000 gCO₂</span>
  </div>
  <div class="header-status">
    <div class="dot" id="conn-dot"></div>
    <span id="conn-label">DESCONECTADO</span>
  </div>
</header>
<div class="main">
  <div class="panel"> <!-- izquierda: métricas cuánticas --> </div>
  <div class="chat-panel"> <!-- centro: conversación --> </div>
  <div class="panel"> <!-- derecha: estado del sistema --> </div>
</div>
```

Backend que orquesta todo el turno (lo que se ejecuta una vez por mensaje y llena los tres paneles), líneas 723-877:

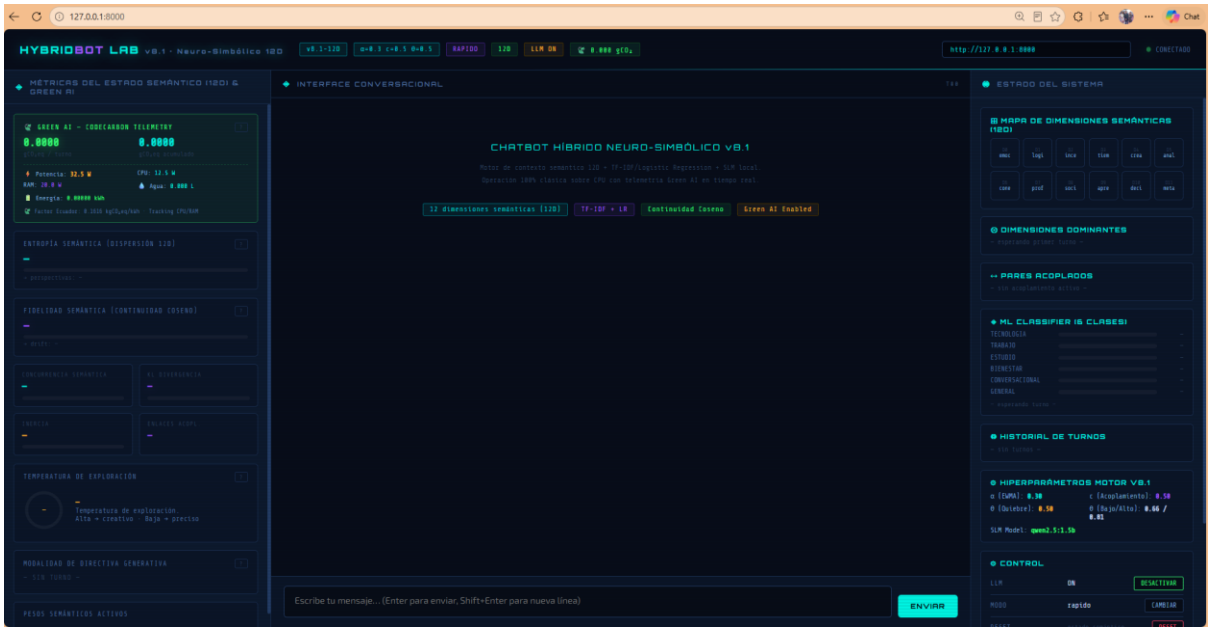


Figura 3.1. Vista general del panel de control (Lab Dashboard) en su estado inicial, con los tres paneles y la configuración nominal de hiperparámetros visible en el encabezado.

3.1.3 Componente 1: clasificador de intención

El primer componente que procesa cada mensaje del usuario es el clasificador de intención, cuya función es determinar la categoría temática general del mensaje. El clasificador puede asignar seis categorías —tecnología, trabajo, estudio, bienestar, conversacional y general—, que cubren las principales temáticas de una conversación típica con un asistente personal. La técnica empleada combina la vectorización TF-IDF, considerando tanto palabras individuales como pares de palabras consecutivas, con un clasificador de Regresión Logística ponderado para compensar el desbalance entre categorías.

La elección de esta combinación, en lugar de un clasificador basado en la arquitectura Transformer (Vaswani et al., 2017), responde a tres criterios alineados con los objetivos del trabajo: eficiencia computacional, dado que la técnica entrena en milisegundos y predice con latencia despreciable, dejando el presupuesto computacional disponible para la generación del modelo de lenguaje; reducido tamaño en memoria, ya que el modelo entrenado ocupa un espacio considerablemente menor al de un transformer pequeño, lo cual resulta relevante para un sistema pensado para hardware modesto; e interpretabilidad, puesto que la Regresión Logística permite auditar los pesos asignados a cada término, propiedad coherente con el paradigma neuro-simbólico adoptado en el diseño general (Garcez & Lamb, 2023).

El clasificador se entrena con un corpus reducido de ejemplos etiquetados manualmente. Este tamaño deliberadamente pequeño responde a que la señal del clasificador se combina con otras fuentes de información —las palabras clave léxicas— dentro del motor de contexto, de modo que un error ocasional de clasificación no compromete la coherencia global del sistema; además, un corpus reducido permite un reentrenamiento prácticamente instantáneo, coherente con lo que Belcak et al. (2025) describen como una de las ventajas distintivas de los sistemas basados en Small Language Models frente a los de gran escala: la rapidez de adaptación.

3.1.4 Componente 2: motor de contexto conversacional

Este componente constituye el aporte original más relevante del trabajo. Su función es mantener un registro acumulado del contexto conversacional y extraer de él las métricas que guían la generación de la respuesta. El motor representa el contexto como un vector numérico de doce componentes, cada uno correspondiente a una dimensión semántica con interpretación específica —emoción, lógica, incertidumbre, tiempo, creación, análisis, conexión, profundidad, social, aprendizaje, decisión y meta—, y cada una asociada a un pequeño vocabulario léxico de palabras clave en español que la activan.

El uso de un espacio vectorial de baja dimensionalidad para representar significado es una técnica establecida en el procesamiento de lenguaje natural. Trabajos recientes han explorado representaciones del significado basadas en vectores normalizados, tanto sobre hardware cuántico (Widdows et al., 2024) como en representaciones clásicas inspiradas en el entrelazamiento (Panahi et al., 2020). El sistema aquí presentado toma de ellos únicamente la analogía formal, simplificándola mediante tres mecanismos matemáticos concretos: la actualización del vector de estado mediante un promedio móvil exponencial, que combina el mensaje actual con el contexto acumulado; una matriz de acoplamiento de doce por doce, que modela correlaciones entre dimensiones semánticas relacionadas; y una entropía de Shannon normalizada, que cuantifica la dispersión semántica del mensaje de cada turno y orienta la selección de la estrategia de respuesta. La derivación matemática completa de estos tres mecanismos, junto con el cálculo de la continuidad conversacional mediante similitud coseno y la selección de la dimensión dominante, se presenta en el Anexo A.

La elección de doce dimensiones responde a tres criterios. Primero, cobertura: las doce categorías se derivaron inductivamente del análisis de los ejes semánticos recurrentes en consultas conversacionales de dominio general, agrupando aquellos con vocabulario de activación diferenciable. Segundo, interpretabilidad: una dimensionalidad reducida permite que cada componente del vector de estado conserve una etiqueta semántica legible por un operador humano, condición necesaria para el objetivo de trazabilidad del artefacto; los espacios de alta

dimensionalidad de los embeddings densos no admiten esta lectura directa. Tercero, costo computacional: la matriz de acoplamiento es de orden n², de modo que doce dimensiones implican 144 coeficientes, volumen que permite su especificación manual y auditable, y cuyo cómputo por turno resulta despreciable frente a la generación. Los resultados de la sección 4.1.4c muestran que la dimensionalidad efectiva sobre el corpus experimental fue inferior a la nominal, limitación atribuible al vocabulario de activación y no al número de dimensiones en sí, que se discute en 4.4.7.

Tanto los coeficientes de la matriz de acoplamiento consignados en la Tabla 3.2 como los términos de activación de la Tabla 3.1 fueron especificados manualmente por el autor sobre la base de relaciones semánticas de sentido común entre las categorías, sin procedimiento de estimación a partir de datos. Los coeficientes no nulos se asignaron a pares de dimensiones con afinidad conceptual reconocible —emoción con social, incertidumbre con aprendizaje, lógica con análisis— con un valor uniforme, y la diagonal principal se fijó en la unidad. Esta especificación responde al criterio de interpretabilidad que orienta el diseño: una matriz derivada de datos habría requerido un corpus etiquetado de coocurrencias semánticas del que no se dispone, y habría sacrificado la auditabilidad de cada coeficiente. Se reconoce como limitación que dicha especificación no fue validada empíricamente de forma independiente, y su derivación a partir de datos se identifica como línea de trabajo futuro en 5.2.

3.1.5 Componente 3: capa de generación

El tercer componente recibe la directiva de generación calculada por el motor de contexto —una instrucción textual que orienta al modelo hacia una postura directa, equilibrada o exploratoria— y la incorpora, junto con el mensaje del usuario y la dimensión dominante detectada, en el mensaje enviado al modelo de lenguaje a través de la interfaz local de Ollama. De esta manera, el comportamiento del modelo se ajusta dinámicamente turno a turno sin requerir reentrenamiento ni ajuste fino (fine-tuning) del modelo de lenguaje subyacente, lo que constituye una forma de

ingeniería de instrucciones (prompt engineering) guiada por señales cuantitativas del propio motor de contexto.

La elección de qwen2.5:1.5b respondió a cuatro criterios verificables en el momento del diseño: disponibilidad en el repositorio de Ollama sin requerir conversión ni cuantización adicional; tamaño compatible con la memoria del equipo descrito en 4.1.1 sin unidad de procesamiento gráfico; soporte documentado de español en su corpus de entrenamiento, condición no satisfecha por varios modelos de escala comparable; y licencia abierta que permite su uso y redistribución en un trabajo académico. Se reconoce que el ecosistema de Small Language Models evoluciona con rapidez y que modelos posteriores podrían ofrecer mejor desempeño a igual escala; la replicación del estudio sobre otros modelos se identifica como línea de trabajo futuro en 5.2.

3.1.6 Consideraciones de diseño para la sostenibilidad

El conjunto de decisiones de arquitectura descritas —clasificador clásico en lugar de un modelo de aprendizaje profundo, motor de contexto basado en álgebra lineal simple en lugar de un modelo neuronal adicional, y Small Language Model en lugar de un modelo de lenguaje de gran escala— responde a un criterio explícito de sostenibilidad computacional. Cada elección técnica minimiza el consumo de recursos en la etapa donde su aporte marginal a la calidad de la respuesta es menor, concentrando el presupuesto computacional disponible en la única etapa —la generación de lenguaje natural— donde un modelo más capaz aporta un beneficio directo y perceptible para el usuario final.

3.1.7 Aporte diferencial del motor de contexto frente al SLM directo

Conviene precisar en qué consiste, específicamente, la mejora que el motor de contexto se propone aportar frente al mismo modelo de lenguaje operando sin dicho componente. Sin el motor de contexto, el SLM local (qwen2.5:1.5b) dispone únicamente del historial conversacional reciente que se le transmite en bruto —los dos turnos anteriores, conforme a la implementación documentada en el Anexo C—, sin ninguna representación explícita del tema dominante, de la dispersión del

diálogo ni del régimen de respuesta apropiado. La ausencia de esa representación explícita permite anticipar dos efectos: primero, una pérdida progresiva de coherencia temática a medida que avanza la conversación, en la medida en que el modelo carece de una señal explícita que le indique si un tema ya fue tratado; y segundo, una invariabilidad en el estilo de respuesta, al no recibir el prompt ninguna indicación sobre si la conversación requiere una postura directa, equilibrada o exploratoria. Estos efectos constituyen la motivación de diseño del motor de contexto y se formulan como la hipótesis H_1 del protocolo experimental (sección 1.5); su magnitud real se determinará empíricamente durante la fase experimental descrita en el Capítulo 4, y no se da por demostrada en esta etapa de diseño.

Frente a esa ausencia de representación explícita del estado, el motor de contexto se propone aportar tres mejoras específicas y verificables: (a) una memoria simbólica de baja dimensionalidad (doce dimensiones) que detecta quiebres temáticos mediante la similitud coseno entre turnos consecutivos; (b) una calibración dinámica del estilo de respuesta —directo, equilibrado o exploratorio— en función de la entropía del vector de estado; y (c) la inyección de una directiva semántica explícita en el prompt del modelo de lenguaje, que orienta su atención hacia el tema dominante sin requerir ajuste fino (fine-tuning) ni reentrenamiento del modelo. Estas tres mejoras se contrastan directamente contra la línea base —definida en la sección 4.2.1 como el mismo modelo sin este componente— mediante el protocolo de evaluación descrito en el Capítulo 4.

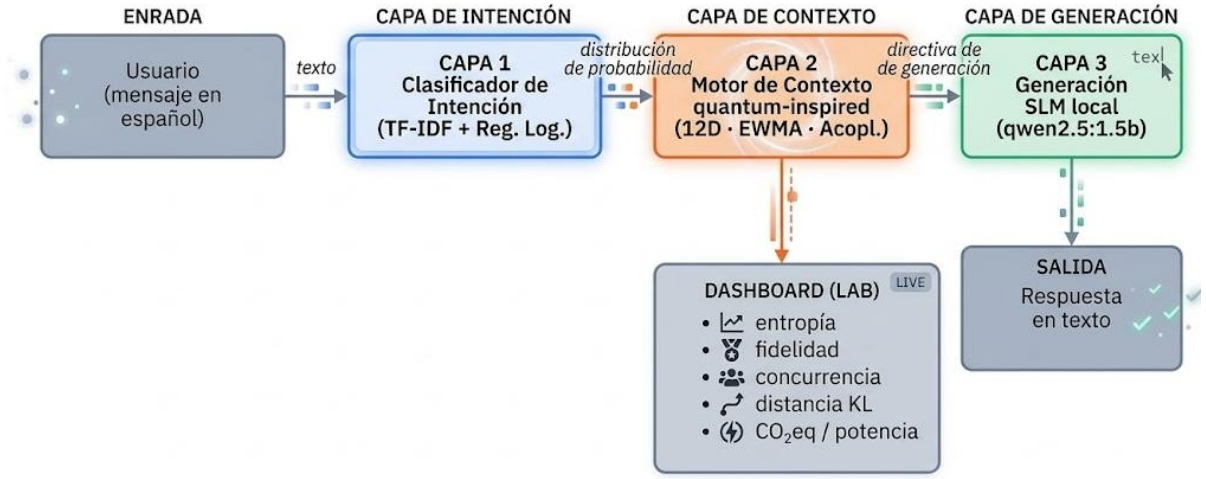


Figura 3.2. Arquitectura de tres capas del artefacto y flujo de datos por turno conversacional.

3.1.8 Protección de datos personales y cumplimiento normativo

El carácter local del procesamiento constituye una medida técnica favorable a la privacidad, pero no equivale por sí mismo al cumplimiento del régimen ecuatoriano de protección de datos personales. Este apartado documenta de forma expresa el ciclo de tratamiento aplicado durante el diseño y la experimentación del artefacto.

Marco jurídico aplicable. El tratamiento de datos personales en el Ecuador se rige por la Ley Orgánica de Protección de Datos Personales (LOPDP; Asamblea Nacional del Ecuador, 2021), publicada en el Quinto Suplemento del Registro Oficial N.º 459 de 26 de mayo de 2021, y por su Reglamento General (Presidencia de la República del Ecuador, 2023), expedido mediante Decreto Ejecutivo N.º 904 y publicado en el Tercer Suplemento del Registro Oficial N.º 435 de 13 de noviembre de 2023. Ambos desarrollan el derecho reconocido en la Constitución de la República del Ecuador (2008, art. 66, num. 19). El régimen sancionatorio administrativo se encuentra en vigencia desde el 26 de mayo de 2023 y la autoridad de control es la Superintendencia de Protección de Datos Personales.

Naturaleza de la información procesada. El corpus experimental está constituido por 57 consultas redactadas por el autor del presente trabajo con fines exclusivamente de prueba, diseñadas para cubrir de forma balanceada las seis categorías temáticas de referencia. Dichas consultas no proceden de usuarios

reales, no fueron obtenidas de conversaciones preexistentes y no contienen datos que permitan identificar, directa o indirectamente, a persona natural alguna. En consecuencia, el corpus no constituye una base de datos personales y su tratamiento queda fuera del ámbito material de aplicación de la ley. La misma consideración aplica al corpus de entrenamiento de 201 ejemplos.

Participación de evaluadores humanos. La evaluación de coherencia se realizó mediante un panel de jueces ciegos a la condición experimental. La participación fue voluntaria e informada respecto de la finalidad académica del ejercicio. De los evaluadores no se recabaron datos identificativos: sus valoraciones se registraron bajo identificadores no reversibles, de modo que ninguna puntuación resulta atribuible a una persona determinada. No se trataron datos de categorías especiales.

Finalidad y base de licitud. La finalidad del tratamiento es exclusivamente la evaluación experimental de un artefacto de software en el marco de un trabajo de titulación. En la medida en que el corpus no contiene datos personales, no se configura un supuesto sujeto a las bases de licitud del artículo 7 de la LOPDP. Respecto de la participación de los evaluadores, la base invocada es el consentimiento libre, específico e informado, otorgado con carácter previo.

Almacenamiento, retención y eliminación. La totalidad del procesamiento se ejecuta en la máquina local descrita en 4.1.1, sin transmisión a servicios de terceros ni a infraestructura de nube. El modelo de lenguaje se invoca sobre la interfaz de bucle local, por lo que ningún contenido conversacional abandona el equipo. Los artefactos generados durante la campaña —registros de turnos, métricas del motor y telemetría de CodeCarbon— se conservan y se ponen a disposición como material de reproducibilidad (Anexo D), al no contener datos personales; las calificaciones de los jueces se publican bajo identificadores no reversibles. El sistema no implementa persistencia de historiales más allá de la ventana de contexto activa.

Controles de seguridad. Se aplicaron medidas técnicas y organizativas proporcionales al riesgo: ejecución confinada al equipo del investigador con control de acceso por credenciales del sistema operativo; ausencia de exposición de

servicios en interfaces de red distintas del bucle local; y ausencia de terceros encargados del tratamiento.

Protección desde el diseño. La arquitectura incorpora el principio de protección de datos desde el diseño y por defecto en dos decisiones estructurales: la ejecución íntegramente local del modelo generativo, que suprime la transferencia de contenido a terceros, y la representación del estado conversacional como un vector semántico agregado de doce dimensiones, que no almacena el texto de los turnos previos sino su proyección numérica sobre el espacio definido en 3.1.4.

Consideración prospectiva. Las conclusiones anteriores se circunscriben al escenario experimental descrito. Un despliegue con usuarios reales —particularmente en dominios sensibles como salud o educación— situaría el tratamiento dentro del ámbito de aplicación de la LOPDP y exigiría la designación de un responsable de tratamiento, la implementación de mecanismos para el ejercicio de los derechos de acceso, rectificación, eliminación, oposición y portabilidad, y la evaluación de impacto correspondiente. Esta extensión se identifica como línea de trabajo futuro en 5.2.

CAPÍTULO 4

RESULTADOS Y DISCUSIÓN

Nota metodológica sobre el estado del protocolo. *La campaña experimental comparativa se ejecutó conforme al protocolo del plan de trabajo aprobado y se cerró el 6 de septiembre de 2026, comprendiendo 57 consultas y 80 turnos conversacionales evaluados bajo dos condiciones. Las secciones 4.1 y 4.2 se sustentan íntegramente en datos instrumentados. La totalidad de los componentes del protocolo fue ejecutada: la evaluación de coherencia mediante panel de jueces ciegos se completó y se reporta en 4.2.4, y el estudio de sensibilidad de hiperparámetros se ejecutó y se reporta en 4.3. No se incorporan valores de relleno: toda tabla presentada corresponde a datos registrados durante la campaña. Las cifras de energía y emisiones son estimaciones instrumentadas de CodeCarbon, con el alcance que se precisa en 4.2.3.*

El problema de investigación, las cuatro preguntas de sistematización y las hipótesis H_1, H_2 y H_3 se formularon en la sección 1.5. Este capítulo presenta la evidencia experimental con la que se contrastan.

4.1 Pruebas de campo y pruebas individuales por componente

Esta sección verifica el funcionamiento de cada una de las tres capas de la arquitectura de forma aislada, antes de la comparación integrada frente al baseline. Corresponde al objetivo específico 1 (arquitectura modular) y al objetivo específico 2 (motor de contexto).

4.1.1 Entorno y condiciones de ejecución

Las pruebas se ejecutaron sobre el equipo especificado en la delimitación geográfica del plan de trabajo: laptop con CPU Intel Core i7 de 11.ª generación y 28 GB de RAM, sin unidad de procesamiento gráfico. El modelo de lenguaje qwen2.5:1.5b se sirvió localmente mediante Ollama en 127.0.0.1:11434. La instrumentación energética se realizó con CodeCarbon empleando el factor de

emisión de la red eléctrica ecuatoriana de 0,1616 $kgCO_2eq/kWh$. Los hiperparámetros del motor se mantuvieron en su configuración nominal (α = 0,30; c = 0,50; θ_quiebre = 0,50) durante toda la campaña, condición necesaria para aislar el efecto del motor de la variabilidad paramétrica.

La campaña completa consumió 47,7 minutos de cómputo agregado entre ambas condiciones y produjo emisiones totales de 1,7142 gCO_2eq. Las emisiones de cada turno se obtienen como el incremento de las emisiones acumuladas que reporta CodeCarbon; cuando ese incremento es nulo, el código estima la energía como el producto de la potencia por el tiempo de ejecución y aplica el factor de emisión nacional (Código C.12). La versión empleada no registra cuál de las dos rutas se utilizó en cada turno. El texto literal de las directivas y los parámetros de generación se documentan en el Anexo D.

4.1.2 Composición del corpus de prueba

Tabla 4.1. Distribución del corpus experimental por categoría y tipo

Categoría	Consultas simples	Diálogos multiturno	Consultas	Turnos
Tecnología	6	3	9	14
Trabajo	6	3	9	13
Estudio	6	3	9	13
Bienestar	6	3	9	13
Conversacional	6	3	9	13
General	10	2	12	14
Total	**40**	**17**	**57**	**80**

De los 17 diálogos multiturno, 11 constan de dos turnos y 6 de tres turnos. Esta estructura permite evaluar la propiedad de retención de contexto, que por definición no puede observarse en consultas de turno único.

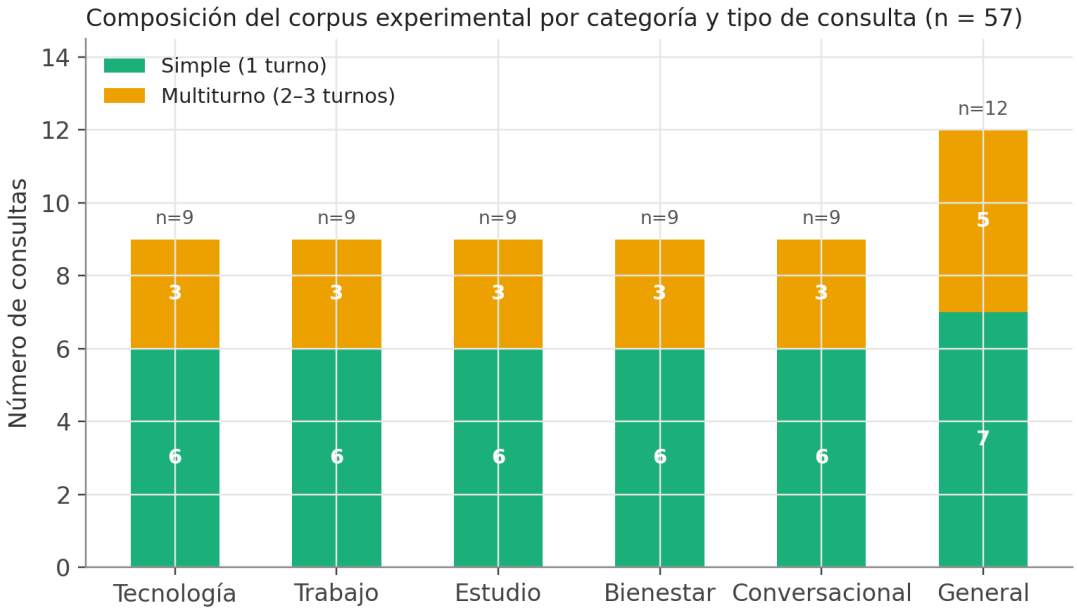


Figura 4.1. Composición del corpus experimental por categoría temática y tipo de consulta (n = 57).

4.1.3 Prueba de la capa 1: clasificador de intención

Construcción y etiquetado del corpus de entrenamiento. El corpus de entrenamiento del clasificador está integrado en el código fuente del artefacto (clase IntentClassifier, Anexo C) y se compone de consultas breves en español redactadas por el autor del presente trabajo, con una extensión media de 5,7 palabras (rango de 1 a 11), en registro coloquial y sin tildes, en coherencia con la normalización que el clasificador aplica antes de vectorizar. El etiquetado fue realizado por el propio autor en el momento de la redacción, de modo que cada ejemplo se generó ya asignado a su categoría. El criterio aplicado fue el propósito comunicativo predominante de la consulta. Como reglas de desambiguación, observables en el propio corpus, las consultas que combinan un ámbito temático con un estado emocional se asignaron a Bienestar (por ejemplo, «me siento muy estresado con el trabajo») y las que expresan una necesidad de aprendizaje sobre una tecnología se asignaron a Estudio (por ejemplo, «necesito aprender python desde cero»). No intervino un segundo anotador, por lo que no procede el cálculo de concordancia inter-anotador.

Distribución por clase. La versión del código fuente documentada en el Anexo C contiene 202 ejemplos, distribuidos del siguiente modo: Tecnología, 51 (25,2 %);

Conversacional, 35 (17,3 %); Trabajo, Estudio y Bienestar, 30 cada una (14,9 %); y General, 26 (12,9 %). La razón entre la clase más representada y la menos representada es de 1,96. Este desbalance se compensa durante el entrenamiento mediante una ponderación inversa a la frecuencia de cada clase (parámetro class_weight="balanced" de la Regresión Logística).

Duplicados. El corpus no contiene duplicados exactos. Una búsqueda de casi-duplicados mediante similitud coseno sobre vectores TF-IDF de palabras identificó ocho pares con similitud igual o superior a 0,60, todos ellos dentro de una misma clase; el más próximo (0,87) corresponde a «el codigo de hola mundo en python» y «dame el codigo de hola mundo en python». Los casi-duplicados intraclase no introducen contradicciones de etiquetado, pero pueden elevar ligeramente la exactitud estimada por validación cruzada, puesto que un ejemplo y su variante pueden quedar repartidos entre los pliegues de entrenamiento y de prueba.

Solapamiento léxico entre clases. El corpus contiene términos compartidos entre categorías que anticipan las confusiones registradas en la Tabla 4.2. El término «trabajo» aparece en cuatro ejemplos de Bienestar y solo en dos de Trabajo; «python» aparece en trece ejemplos de Tecnología y en dos de Estudio; y «sql», en tres y dos, respectivamente. Esta superposición es coherente con la fuga de Tecnología hacia Estudio y con la dispersión de los errores de Bienestar que se analizan más adelante en este apartado.

Separación entre entrenamiento y evaluación. La validación cruzada estratificada (k = 5) se realizó exclusivamente sobre el corpus de entrenamiento. El corpus experimental de 57 consultas (80 turnos) se construyó de forma independiente y posterior, sin reutilizar ejemplos del corpus de entrenamiento, y no intervino en el ajuste del clasificador ni de sus hiperparámetros. La independencia entre ambos conjuntos descansa en ese procedimiento de construcción, ya que el presente documento no incorpora un contraste automatizado de solapamiento textual entre ellos.

Limitaciones de trazabilidad y efecto sobre la interpretación. La versión del código revisada contiene un ejemplo más que los 201 sobre los que se reportó la

validación cruzada, y con la documentación disponible no es posible identificar el ejemplo incorporado, lo que se declara como limitación de trazabilidad del corpus. La diferencia representa menos del 0,5 % del conjunto y no modifica de forma apreciable las proporciones por clase; una reejecución de la validación cruzada estratificada sobre la versión actual arroja exactitudes de entre 0,76 y 0,80 según la partición de los pliegues, valores consistentes con el 0,775 reportado. En conjunto, las limitaciones descritas —anotador único, casi-duplicados intraclase y ausencia de un contraste automatizado de solapamiento— tienen dos consecuencias para la interpretación de los resultados. Primera, la exactitud obtenida en validación cruzada debe leerse como una estimación optimista del desempeño del clasificador, lo que refuerza la consideración de la brecha de generalización como limitación central (sección 4.4.7). Segunda, la exactitud de 0,600 sobre el corpus experimental conserva su validez como estimación del desempeño sobre datos no vistos, en la medida en que dicho corpus fue construido de forma independiente; su incertidumbre principal deriva del tamaño reducido de la muestra más que de un posible solapamiento con el entrenamiento.

El clasificador TF-IDF con Regresión Logística se evaluó contra la categoría de referencia asignada a cada consulta durante la construcción del corpus.

Tabla 4.2. Exactitud del clasificador de intención por categoría de referencia

Categoría de referencia	n	Aciertos	Exactitud	Confusiones principales
General	14	12	0,857	bienestar (1), estudio (1)
Conversacional	13	9	0,692	tecnología (2), general (1), estudio (1)
Trabajo	13	8	0,615	tecnología (2), otras (3)
Estudio	13	8	0,615	general (2), otras (3)
Tecnología	14	6	0,429	estudio (4), conversacional (2)
Bienestar	13	5	0,385	estudio (3), conversacional (2), trabajo (2)
Global	80	48	0,600	—

La confianza media de predicción fue de 0,4881 (DE = 0,1997).

Este resultado exige una lectura cuidadosa. La validación cruzada estratificada (k = 5) sobre el corpus de entrenamiento de 201 ejemplos había arrojado una exactitud de 0,775 (DE = 0,045). Sobre el corpus experimental —construido de forma independiente y con vocabulario no visto durante el entrenamiento— la exactitud desciende a 0,600. La caída de 17,5 puntos porcentuales constituye una brecha de generalización atribuible al tamaño reducido del corpus de entrenamiento y a la representación por bolsa de n-gramas, que no captura sinonimia ni variación morfológica fuera del vocabulario observado.

Las categorías más afectadas son Bienestar (0,385) y Tecnología (0,429). En Bienestar, las confusiones se dispersan entre cuatro categorías distintas, lo que sugiere que la frontera semántica de esta clase es difusa en el espacio TF-IDF: expresiones sobre estrés laboral, hábitos de estudio y estado de ánimo comparten léxico. En Tecnología, la confusión dominante es con Estudio (4 de 8 errores), coherente con que las consultas técnicas del corpus están formuladas como preguntas de aprendizaje (“cómo se declara”, “qué diferencia hay entre”).

La confianza media de 0,4881 sobre seis clases indica un clasificador conservador: raramente concentra masa de probabilidad en una sola clase. Esta propiedad tiene una consecuencia favorable para la arquitectura, que se analiza en 4.1.4: la distribución de probabilidad completa —no solo la etiqueta ganadora— alimenta el codificador semántico, de modo que la incertidumbre del clasificador se propaga como dispersión del vector de estado en lugar de como error categórico.

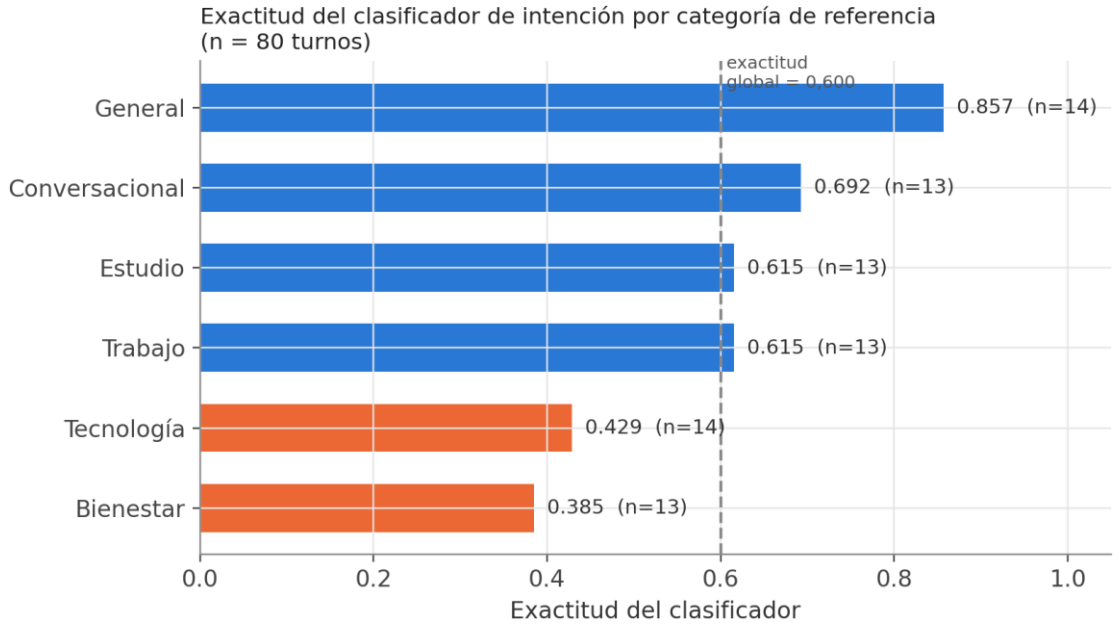


Figura 4.2. Exactitud del clasificador de intención por categoría de referencia, ordenada de mayor a menor, con la exactitud global (0,600) como referencia.

Tabla 4.3. Precisión, exhaustividad y F1 del clasificador de intención por categoría de referencia (n = 80)

Categoría	n	Precisión	Exhaustividad	F1
Tecnología	14	0,545	0,429	0,480
Trabajo	13	0,667	0,615	0,640
Estudio	13	0,444	0,615	0,516
Bienestar	13	0,556	0,385	0,455
Conversacional	13	0,643	0,692	0,667
General	14	0,750	0,857	0,800
Macro-promedio	**80**	**0,601**	**0,599**	**0,593**

Nota. Exactitud global = 0,600; F1 ponderada = 0,594; κ de Cohen = 0,520.

La macro-F1 de 0,593 confirma que la exactitud global no enmascara un desempeño heterogéneo compensado por clases mayoritarias: al ser el corpus prácticamente balanceado, macro-F1 y exactitud convergen.

4.1.4 Prueba de la capa 2: motor de contexto

a) Verificación del mapeo entropía → directiva de respuesta

El motor selecciona la estrategia de generación a partir de la entropía de la distribución de probabilidad derivada del vector semántico del mensaje de cada turno (Anexo A.4). La Tabla 4.4 verifica el comportamiento de los umbrales sobre los 80 turnos.

Tabla 4.4. Distribución de directivas según los rangos de entropía normalizada

Directiva	n	%	Entropía mínima	Entropía máxima	Entropía media
Directa	41	51,3 %	0,4584	0,6562	0,5279
Equilibrada	27	33,8 %	0,6638	0,8016	0,7333
Exploratoria	12	15,0 %	0,8139	0,9246	0,8445

Los tres rangos son disjuntos y no presentan solapamiento en ningún turno, lo que confirma que la regla de decisión opera de forma determinista conforme a la especificación del Anexo A. La distribución observada (51 % directa, 34 % equilibrada, 15 % exploratoria) indica que el motor discrimina efectivamente entre consultas focalizadas y consultas dispersas, sin reducirse a una única estrategia. Este resultado responde en parte a la segunda pregunta de sistematización: las métricas internas de dispersión son suficientes para seleccionar la estrategia de respuesta. Si esa selección mejora la respuesta se contrasta en 4.2.4.

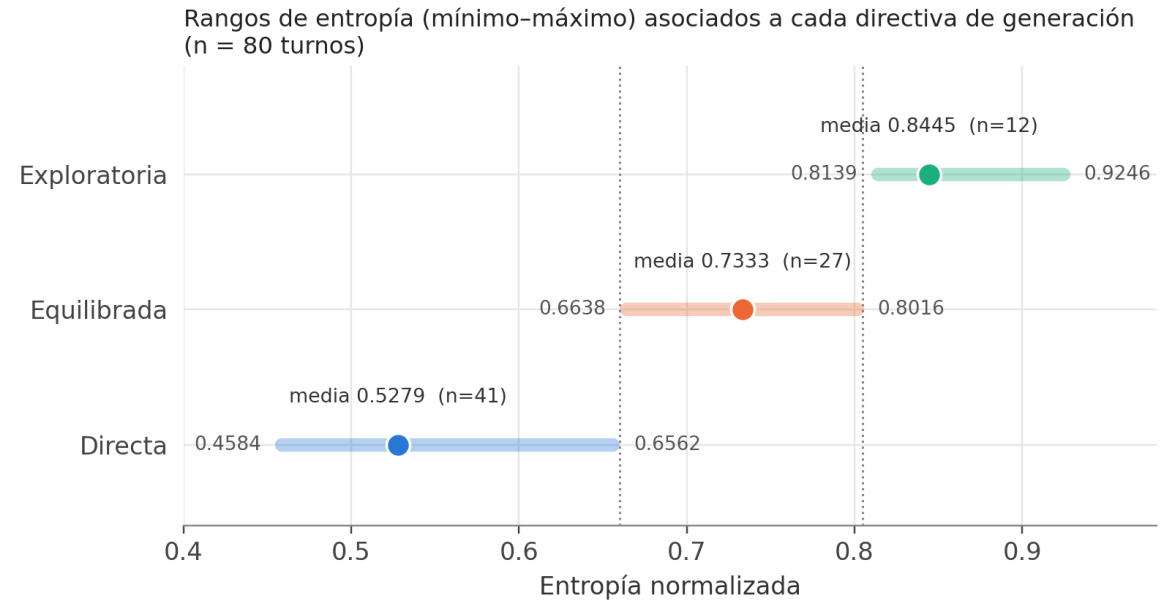


Figura 4.3. Rango (mínimo–máximo) y media de la entropía normalizada dentro de cada una de las tres directivas de generación.

b) Verificación de la retención de contexto en diálogos multiturno

Tabla 4.5. Continuidad, fidelidad y entropía media por turno conversacional

Turno	n	Continuidad media	Fidelidad media	Entropía media
1	17	1,0000	1,0000	0,6297
2	17	0,7944	0,7944	0,7063
3	6	0,8592	0,8592	0,8150

Nota. En la implementación evaluada, la magnitud registrada como continuidad se calcula como φ_t · ψ_(t−1) y coincide con la fidelidad del estímulo; por ello ambas columnas presentan valores idénticos (Anexo A.5).

El comportamiento observado es el esperado por diseño. En el primer turno no existe estado previo y la continuidad es unitaria por definición. En el segundo turno la continuidad desciende a 0,7944, valor que se mantiene por encima del umbral de quiebre (θ = 0,50) e indica que el mensaje del segundo turno mantiene un alineamiento alto con el estado acumulado. En el tercer turno la continuidad se recupera a 0,8592, lo que indica que, en este corpus, los mensajes del tercer turno se alinean aún más con el estado acumulado del diálogo.

El aumento monótono de la entropía a lo largo de los turnos (0,6297 → 0,7063 → 0,8150) indica que, en este corpus, los mensajes de los turnos avanzados activan más dimensiones semánticas; al calcularse sobre el mensaje del turno, no refleja una acumulación en el estado, pero explica que el motor tienda a estrategias más exploratorias en turnos avanzados.

c) Distribución de dimensiones dominantes

Tabla 4.6. Frecuencia de activación dominante de las dimensiones semánticas

Dimensión	Frecuencia	%
Incertidumbre	37	46,3 %
Social	27	33,8 %
Lógica	9	11,3 %
Tiempo	3	3,8 %
Conexión	3	3,8 %
Meta	1	1,3 %
Emoción, Creación, Análisis, Profundidad, Aprendizaje, Decisión	0	0,0 %

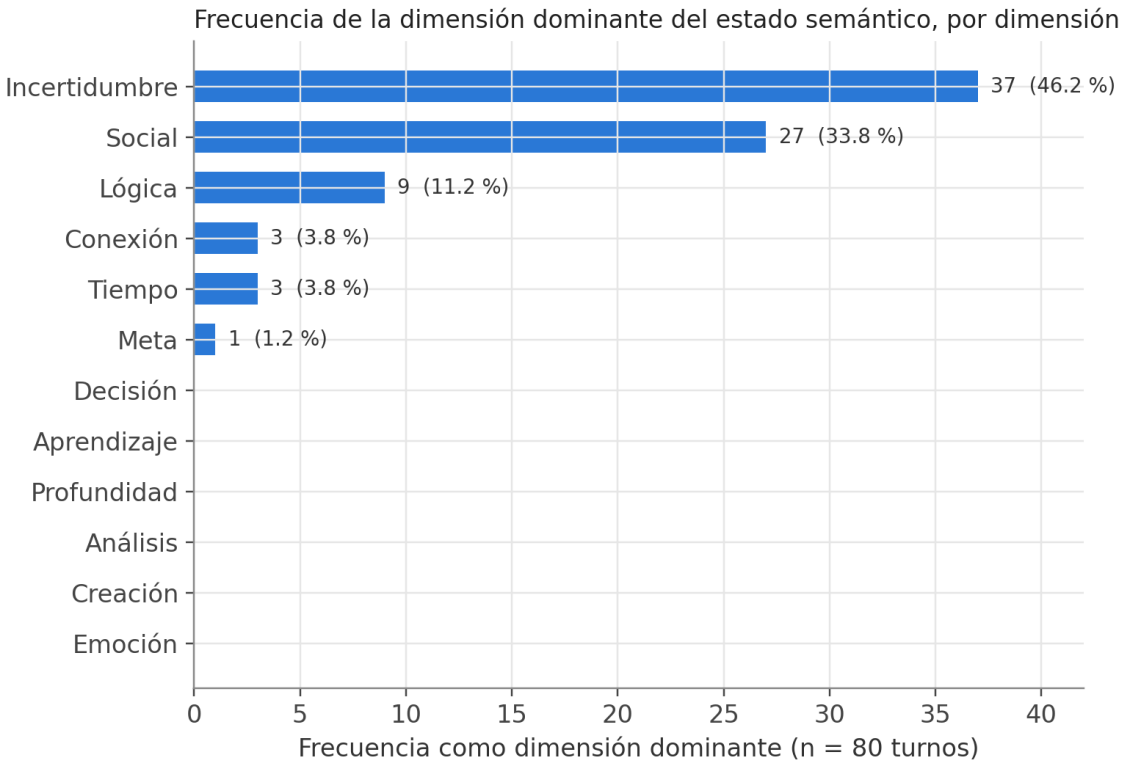


Figura 4.4. Frecuencia con que cada una de las doce dimensiones semánticas resultó dominante en el estado del turno (n = 80).

Este resultado revela una limitación real del codificador semántico que debe consignarse sin atenuación: seis de las doce dimensiones nunca resultaron dominantes, y dos de ellas concentran el 80,1 % de los casos. El predominio de *incertidumbre* se explica por el diseño del corpus, compuesto mayoritariamente por preguntas cuyos marcadores léxicos (“cómo”, “qué”, “cuál”, “por qué”) activan directamente esa dimensión. El predominio de *social* obedece a un efecto de segundo orden: la matriz de acoplamiento vincula *social* con *emoción* (0,65) y con *conexión* (0,50), de modo que recibe activación indirecta desde múltiples orígenes.

La consecuencia práctica es que el espacio de 12 dimensiones opera efectivamente sobre un subespacio de menor rango para este corpus. Esto no invalida el mecanismo —las métricas de dispersión y continuidad siguen siendo informativas, como muestran las Tablas 4.5 y 4.6 pero acota el alcance de la afirmación de riqueza representacional. Se discute en 4.4.4 y se recoge en las limitaciones.

d) Detector de quiebre temático

En los 80 turnos evaluados, el indicador break_detected registró valor falso en el 100 % de los casos. El detector no se activó en ninguna ocasión, incluyendo los turnos en que la continuidad descendió a su valor mínimo observado.

Este resultado es negativo y se reporta como tal. Con el umbral nominal θ = 0,50, la continuidad mínima registrada en la campaña no alcanzó a cruzarlo, de modo que la funcionalidad no pudo ser ejercitada ni, por tanto, validada. El detector de quiebres queda documentado como característica paramétrica no verificada empíricamente, y no se le atribuye contribución alguna a los resultados de las secciones siguientes. Su calibración requiere un corpus con cambios temáticos deliberados y abruptos, diseño que se propone en el trabajo futuro (Capítulo 5).

4.1.5 Prueba de la capa 3: generación con SLM local

Las 160 generaciones (80 turnos × 2 condiciones) se completaron sin fallo de servicio, sin recurso al mecanismo de degradación elegante y sin respuestas vacías. La longitud media de respuesta fue de 141,2 palabras (DE = 93,9) en la condición A y de 134,6 palabras (DE = 92,0) en la condición B.

Se verificó una correlación muy alta entre la longitud de la respuesta generada y la latencia total: r = 0,961 en la condición A y r = 0,904 en la condición B. Asimismo, la correlación entre latencia y emisiones de CO_2 alcanzó r = 0,923 sobre el conjunto de las 160 observaciones. Ambos hallazgos son determinantes para la interpretación de la sección 4.2: el costo computacional del sistema está dominado por la generación autorregresiva del SLM, y cualquier efecto del motor de contexto sobre latencia o energía debe leerse contra esa fuente de variabilidad.

4.2 Comparativa de resultados frente al baseline

Esta sección aborda el objetivo específico 3 y responde a la cuarta pregunta de sistematización y, solo para la configuración nominal de hiperparámetros, a la tercera.

4.2.1 Diseño de la comparación

Se contrastaron dos condiciones sobre los mismos 80 turnos, en diseño de medidas repetidas:

- **Condición A (baseline):** la consulta del usuario se envía al SLM sin directiva de contexto. Corresponde al uso directo de qwen2.5:1.5b.
- **Condición B (sistema completo):** la consulta se envía al SLM acompañada de la directiva generada por el motor de contexto a partir del estado semántico del turno.

Debe precisarse el alcance exacto de la manipulación experimental. En ambas condiciones, la invocación al modelo generador incluye una ventana de historial conversacional compuesta por los dos turnos inmediatamente anteriores —consulta del usuario y respuesta del sistema, truncadas a 200 caracteres cada una— tal como documenta el Código C.11 del Anexo C. La condición A no constituye, por tanto, un modelo sin memoria conversacional, sino un modelo que dispone de contexto reciente en bruto y carece únicamente de la directiva derivada del estado semántico. La variable manipulada es exclusivamente la presencia de dicha directiva en el prompt de sistema; todo lo demás —modelo, parámetros de decodificación, ventana de historial y consulta— permanece idéntico entre condiciones.

Esta precisión es metodológicamente relevante para la interpretación de los resultados. El contraste A–B no mide el aporte del motor de contexto frente a un generador desprovisto de toda memoria, sino su aporte marginal frente a un generador que ya dispone de contexto conversacional reciente. El efecto esperable bajo este diseño es, en consecuencia, de magnitud sustancialmente menor que el postulado en la formulación original de H_1 (sección 3.1.7), consideración que se retoma en 4.4.1 al discutir el resultado nulo.

4.2.2 Latencia

Tabla 4.7. Estadísticos descriptivos de latencia por condición experimental

Estadístico	Condición A	Condición B
Media (ms)	18 200,22	17 545,43

Estadístico	Condición A	Condición B
Desviación estándar (ms)	9 094,26	10 373,91

Tabla 4.8. Contrastes inferenciales de latencia entre condiciones experimentales

Contraste	Valor
Diferencia media Δ (B − A)	−654,80 ms (−3,6 %)
IC 95 % de la diferencia	[−2 438,12 ; +1 128,53] ms
t pareada (gl = 79)	t = −0,731; p = 0,4670
Wilcoxon rangos con signo	p = 0,2805
Tamaño del efecto (d de Cohen, medidas repetidas)	d_z = −0,082

El sistema con motor presenta una latencia media 654,80 ms inferior a la del baseline. La diferencia no es estadísticamente significativa por ninguna de las dos pruebas, y el tamaño del efecto es despreciable (|d_z| < 0,10).

La interpretación correcta de este resultado no es que el motor acelere el sistema, sino que **no se detecta sobrecarga atribuible al motor**. El intervalo de confianza contiene el cero, lo que impide afirmar dirección alguna del efecto. Debe señalarse con precisión que el límite superior del intervalo (+1 128,53 ms) no permite descartar una sobrecarga de hasta aproximadamente 1,1 segundos con los datos disponibles; la potencia estadística de la muestra es insuficiente para acotar el efecto por debajo del umbral de 500 ms fijado en la hipótesis.

Un análisis complementario aporta evidencia adicional. Normalizando la latencia por la longitud de la respuesta generada —métrica que controla la principal fuente de variabilidad identificada en 4.1.5— se obtiene 199,0 ms/palabra (DE = 167,9) en la condición A frente a 183,4 ms/palabra (DE = 155,7) en la condición B, con t = −1,678 y p = 0,0974. La diferencia normalizada de 15,6 ms/palabra (−7,8 %) es marginalmente significativa al nivel de 0,10 y apunta en la misma dirección: la directiva de contexto no encarece la generación por unidad de texto producido.

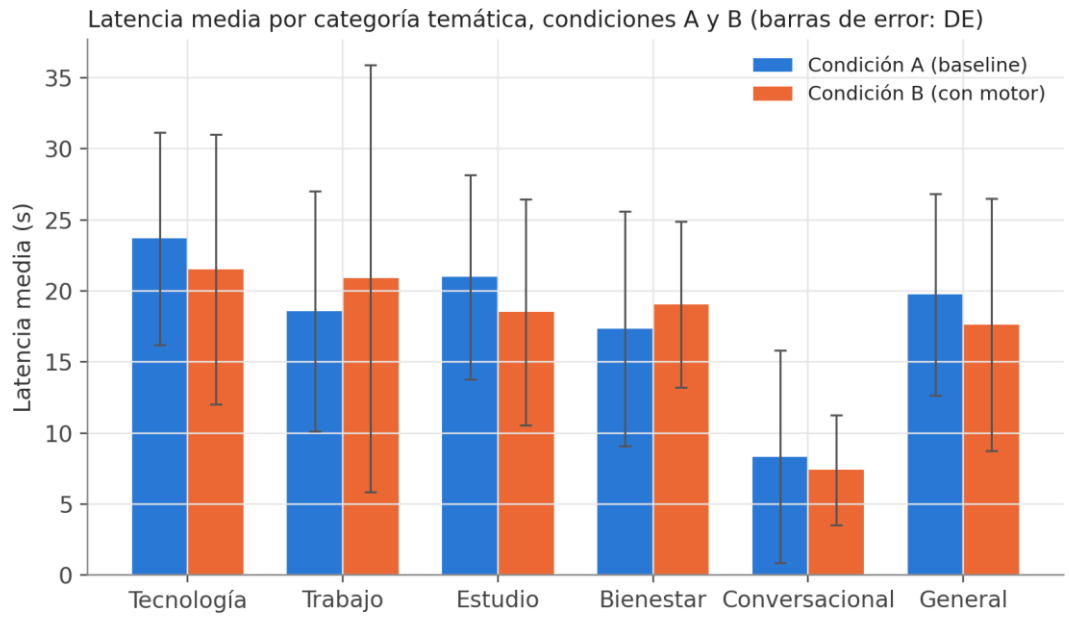


Figura 4.5. Latencia media (en segundos), con barras de error correspondientes a la desviación estándar, por categoría temática y condición experimental.

4.2.3 Huella de carbono

Tabla 4.9. Estadísticos descriptivos de emisiones por condición experimental

Estadístico	Condición A	Condición B
Media por turno (gCO_2eq)	0,010891	0,010537
Desviación estándar (gCO_2eq)	0,006393	0,006794
Total de la campaña (gCO_2eq)	0,8713	0,8429

Tabla 4.10. Contrastes inferenciales de emisiones entre condiciones experimentales

Contraste	Valor
Diferencia media Δ (B − A)	−0,000355 gCO_2eq (−3,3 %)
IC 95 % de la diferencia	[−0,001422 ; +0,000712] gCO_2eq
t pareada (gl = 79)	t = −0,657; p = 0,5128
Wilcoxon rangos con signo	p = 0,3386
Tamaño del efecto (d_z)	−0,074

El resultado replica el patrón de la latencia, como era previsible dada la correlación r = 0,923 entre ambas magnitudes: CodeCarbon estima el consumo como producto

de potencia por tiempo, de modo que las emisiones heredan la estructura de la latencia. La diferencia no es significativa y el intervalo contiene el cero. En términos absolutos, la magnitud del efecto —355 microgramos de CO_2 por turno— carece de relevancia práctica.

El dato de mayor interés para la pregunta de investigación no es la comparación A–B, sino la magnitud absoluta del consumo. El sistema completo emite 0,010537 gCO_2eq por turno conversacional. La campaña experimental íntegra, con 160 generaciones, produjo 1,7142 gCO_2eq, cantidad equivalente a la que genera un automóvil de combustión en aproximadamente 11 metros de recorrido.

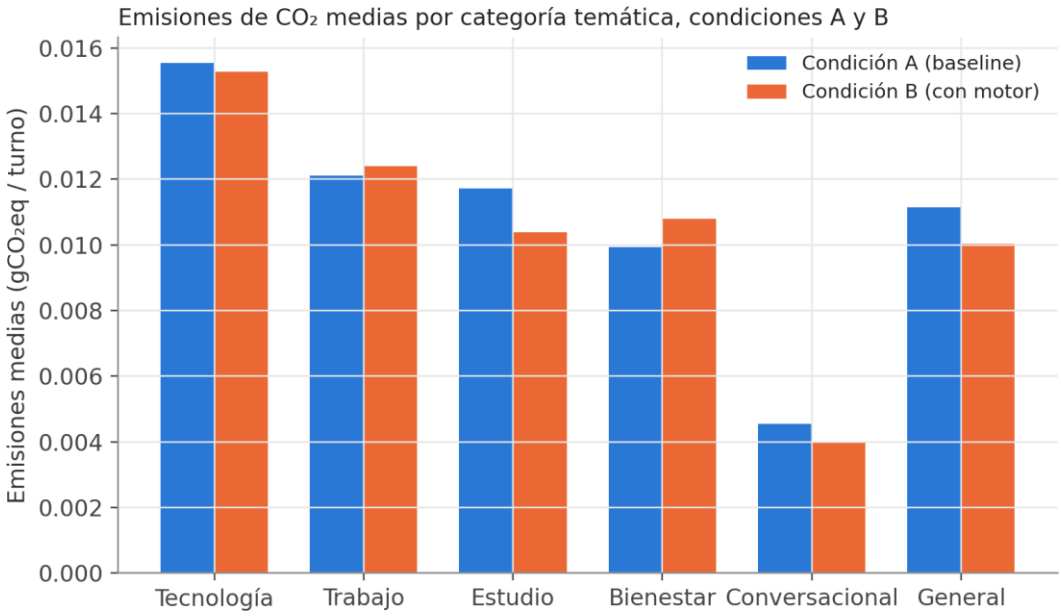


Figura 4.6. Emisiones medias de CO_2 equivalente por turno (gCO_2eq/turno), por categoría temática y condición experimental.

Tabla 4.11. Comparación del consumo energético con sistemas de referencia

Sistema	Consumo reportado	Fuente
Entrenamiento de GPT-3	1 287 MWh; 552 t CO_2eq	Patterson et al. (2021)
Servicio mensual de asistente comercial (alta demanda)	23 364 MWh mensuales (peor	Jiang et al. (2024)

Sistema	Consumo reportado	Fuente
	escenario estimado)	
Sistema propuesto (inferencia local, por turno)	0,010537 gCO_2eq	Estimación instrumentada (CodeCarbon), esta campaña

La comparación debe leerse con la salvedad de que las cifras de referencia corresponden a entrenamiento y a operación a escala de servicio, mientras que la medición propia corresponde exclusivamente a inferencia en un único dispositivo. No son magnitudes directamente conmensurables. Lo que la comparación sí sustenta es la afirmación central del trabajo: **operar un asistente conversacional en español sobre hardware doméstico tiene una huella de carbono cuantificable, reproducible y de orden de milésimas de gramo por turno**, sin transferencia de datos a terceros y sin infraestructura de GPU.

Debe precisarse el alcance del instrumento empleado. CodeCarbon no realiza una medición eléctrica directa en la fuente de alimentación, sino que estima el consumo a partir de la utilización de procesador y memoria mediante modelos de potencia parametrizados por componente, multiplicando el resultado por el factor de emisión de la red eléctrica. Las cifras reportadas en este apartado constituyen, por tanto, estimaciones instrumentadas y no mediciones de potencia en sentido estricto. Su validez para los fines de este trabajo descansa en la consistencia del método entre condiciones experimentales ambas fueron medidas con el mismo instrumento sobre el mismo equipo, lo que preserva la comparabilidad relativa aun cuando el valor absoluto esté sujeto al error del modelo de potencia subyacente.

4.2.4 Coherencia conversacional

La evaluación de coherencia mediante panel de jueces ciegos se completó sobre la totalidad de los ochenta pares de respuestas de la campaña. El protocolo aplicado fue el siguiente:

- **Instrumento:** rúbrica de cinco criterios (pertinencia, continuidad temática, corrección del contenido, adecuación de la extensión, naturalidad del registro), cada uno en escala Likert de 1 a 5.
- **Unidades de evaluación:** los 80 pares de respuestas (A, B) correspondientes a los mismos turnos.
- **Cegado:** presentación aleatorizada del orden dentro de cada par y ocultamiento de la condición de origen; contrabalanceo 50/50.
- **Jueces:** tres evaluadores independientes con formación en Ciencias de la Computación.
- **Concordancia:** kappa ponderado con pesos cuadráticos. El coeficiente kappa original fue formulado para escalas nominales, en las que todo desacuerdo se considera igualmente grave (Cohen, 1960); dado que la rúbrica empleada es una escala ordinal de cinco niveles, en la que un desacuerdo entre 4 y 5 es menos severo que uno entre 1 y 5, corresponde aplicar la generalización ponderada que incorpora la gravedad relativa de cada discrepancia (Cohen, 1968). Se fija $\kappa_w > 0{,}60$ como criterio de aceptación, umbral a partir del cual la escala de interpretación de Landis y Koch (1977) califica la concordancia como sustancial.
- **Contraste:** prueba de Wilcoxon de rangos con signo sobre la puntuación agregada, con tamaño del efecto d_z e intervalo de confianza al 95 %.

La evaluación de coherencia se completó sobre los ochenta pares de respuestas, mediante tres evaluadores independientes con formación en Ciencias de la Computación que aplicaron la rúbrica de cinco criterios del Anexo B bajo el protocolo de cegado declarado. Para cada respuesta se calculó una puntuación agregada como el promedio de sus cinco calificaciones; la puntuación de cada condición, en cada uno de los ochenta pares, se obtuvo promediando a su vez esa puntuación entre los tres jueces. La puntuación agregada media fue de 3,91 (DE = 0,43; mediana = 3,87; rango 2,67-4,80) en la condición A y de 3,93 (DE = 0,40;

mediana = 3,87; rango 2,47-4,67) en la condición B.

La concordancia entre evaluadores se midió mediante kappa ponderado con pesos cuadráticos (Cohen, 1968) sobre las ciento sesenta respuestas calificadas por cada juez (ochenta pares × dos condiciones), calculando el coeficiente para cada uno de los tres pares de evaluadores y promediando los tres valores. Las concordancias por pares fueron de 0,66 (jueces 1-2), 0,60 (jueces 1-3) y 0,65 (jueces 2-3), con un promedio de κ_w = 0,64, que supera el umbral de aceptación de 0,60 fijado a priori y se clasifica como concordancia sustancial conforme a la escala de Landis y Koch (1977).

El contraste entre condiciones mediante la prueba de Wilcoxon de rangos con signo sobre la puntuación agregada de los ochenta pares arrojó W = 1255,5, p = 0,8974, sin significancia estadística al nivel convencional de 0,05, con un tamaño del efecto d_z = 0,04. El intervalo de confianza al 95 % para la diferencia media (condición B menos condición A, de 0,02) fue de [−0,10; 0,14], un intervalo estrecho y centrado en cero. En consecuencia, la hipótesis H_1 no recibe respaldo empírico en la presente muestra: no se observa una mejora de la coherencia conversacional percibida atribuible al motor de contexto, y la estrechez del intervalo de confianza sugiere que, de existir un efecto real, su magnitud sería mínima y sin relevancia práctica. Los resultados se presentan en las Tablas 4.17 y 4.18.

Tabla 4.12. Latencia y emisiones por categoría temática y condición experimental

Categoría	n	Latencia A (ms)	Latencia B (ms)	Δ Latencia (ms)	CO_2 A (g)	CO_2 B (g)	Δ CO_2 (g)
Tecnología	14	23 681	21 500	−2 181	0,015542	0,015279	−0,000263
Trabajo	13	18 565	20 884	+2 318	0,012101	0,012405	+0,000305
Estudio	13	20 986	18 507	−2 479	0,011712	0,010380	−0,001332

Categoría	n	Latencia A (ms)	Latencia B (ms)	Δ Latencia (ms)	CO_2 A (g)	CO_2 B (g)	Δ CO_2 (g)
Bienestar	13	17 345	19 053	+1 708	0,009936	0,010788	+0,000852
Conversacional	13	8 338	7 399	−940	0,004544	0,004003	−0,000541
General	14	19 745	17 620	−2 125	0,011136	0,010037	−0,001099

Se observan dos comportamientos diferenciados. En cuatro categorías —Tecnología, Estudio, Conversacional y General— la condición con motor presenta menor latencia y menores emisiones. En las dos restantes —Trabajo y Bienestar— la relación se invierte. La magnitud de las diferencias, entre 940 y 2 479 milisegundos, es en todos los casos superior a la diferencia global de 654,8 milisegundos, lo que indica que la comparación agregada resulta de la cancelación parcial entre efectos de signo opuesto.

Explicación del patrón

La Tabla 4.12 aporta el mecanismo que subyace a esta divergencia.

Tabla 4.13. Longitud de respuesta y variación de latencia por categoría temática

Categoría	Condición A	Condición B	Δ (B − A)	Δ Latencia (ms)
Tecnología	194,6	166,7	−27,9	−2 181
Trabajo	127,8	139,5	+11,7	+2 318
Estudio	179,8	152,6	−27,2	−2 479
Bienestar	127,5	160,2	+32,6	+1 708
Conversacional	40,6	34,5	−6,1	−940
General	170,3	150,4	−19,9	−2 125

El signo de la variación en longitud coincide con el signo de la variación en latencia en las seis categorías, sin excepción. Este resultado, leído junto con la correlación r = 0,961 documentada en la sección 4.1.5, sustenta una interpretación precisa del

efecto del motor: **la directiva de contexto no modifica la velocidad de generación, sino la extensión del texto generado, y la latencia sigue a esa extensión.**

El comportamiento resulta consistente con el diseño del sistema. En las categorías de contenido técnico o factual, el motor tiende a seleccionar la directiva directa, que instruye al modelo hacia respuestas concisas; el texto se acorta y, en consecuencia, el tiempo y la energía disminuyen. En las categorías de contenido abierto o subjetivo, el motor selecciona con mayor frecuencia directivas equilibradas o exploratorias, el texto se extiende y los costos aumentan proporcionalmente.

Tabla 4.14. Entropía y distribución de directivas por categoría temática

Categoría	Entropía media	DE	Directa	Equilibrada	Exploratoria
Tecnología	0,6887	0,1064	4	8	2
Trabajo	0,6764	0,1398	6	4	3
Estudio	0,6626	0,1377	6	6	1
Bienestar	0,6682	0,1331	6	4	3
Conversacional	0,6098	0,1670	9	1	3
General	0,5653	0,1159	10	4	0

La categoría Conversacional presenta el caso más nítido: nueve de trece turnos recibieron directiva directa, la longitud media de respuesta fue la menor del corpus con 34,5 palabras y la latencia resultó menos de la mitad que la de cualquier otra categoría. La categoría General muestra la entropía media más baja del corpus, 0,5653, coherente con consultas factuales de tema único.

Consecuencias para la interpretación de los resultados

Este análisis obliga a matizar la conclusión de transparencia computacional formulada en 4.2.2. La afirmación de que el motor no introduce sobrecarga es correcta en el agregado, pero incompleta: el motor sí modifica el costo por turno, en magnitudes de hasta 2,5 segundos, y lo hace en direcciones opuestas según el dominio temático. Que la diferencia global resulte próxima a cero es consecuencia

de la composición equilibrada del corpus y no de una neutralidad intrínseca del mecanismo.

La implicación práctica es que el costo computacional del sistema depende de la distribución temática de las consultas que reciba en operación. Un despliegue orientado a soporte técnico obtendría respuestas más breves y económicas que la línea base; uno orientado a acompañamiento o consulta abierta obtendría lo contrario. Esta observación se retoma en las limitaciones de la sección 4.4.7

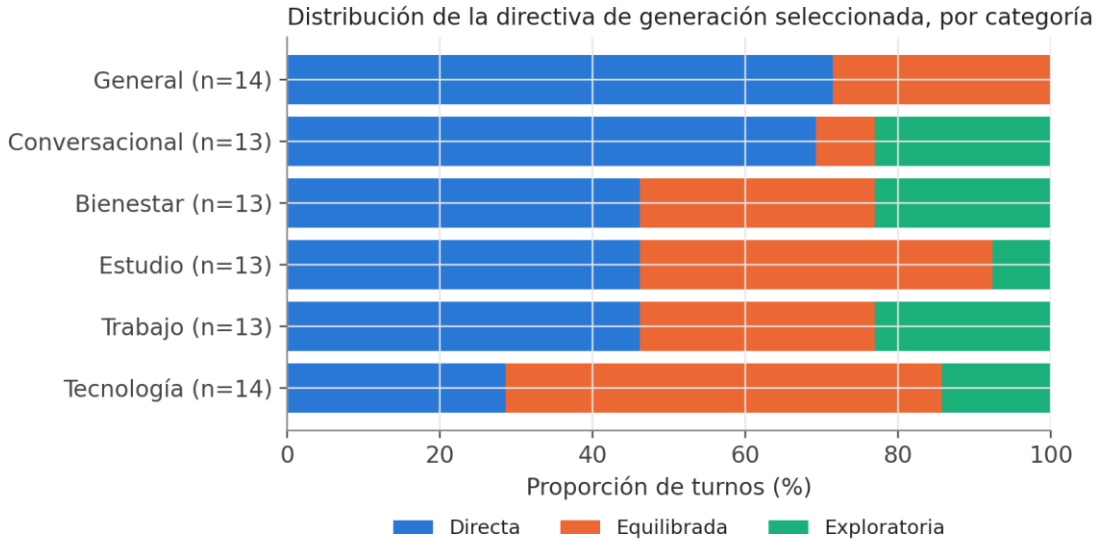


Figura 4.7. Distribución porcentual de la directiva de generación seleccionada (directa, equilibrada, exploratoria) por categoría temática.

4.3 Estudios de sensibilidad de hiperparámetros

4.3.1 Diseño y alcance del estudio

La campaña comparativa reportada en 4.2 se ejecutó con los hiperparámetros fijados en su configuración nominal (α = 0,30; c = 0,50; θ_quiebre = 0,50). La verificación de los registros confirma que estos tres valores permanecieron constantes en la totalidad de los 160 registros. Fijar los hiperparámetros era condición necesaria para aislar el efecto del motor en la comparación A–B; el

estudio de sensibilidad constituye, por tanto, un experimento distinto, con su propio diseño y su propia unidad de análisis.

El estudio que se reporta a continuación evalúa la sensibilidad del **motor de contexto**, no del sistema completo. Las variables de respuesta son magnitudes internas del estado semántico —continuidad del estado y fidelidad del estímulo—, cuyo cálculo no requiere invocar la capa de generación. Esta decisión de diseño permite recorrer la malla completa a partir de los vectores semánticos precodificados una sola vez, sin repetir las inferencias del Small Language Model. En consecuencia, el costo computacional efectivo del experimento es de segundos y su huella asociada resulta despreciable frente a la campaña de 4.2, lo que sustituye la estimación preliminar de 1 000 generaciones declarada en la versión anterior del protocolo. La contrapartida es que este diseño no permite reportar latencia ni distribución de directivas por celda, variables que quedan identificadas como extensión del estudio en 5.2.

En consecuencia, la tercera pregunta de sistematización queda respondida solo en parte. El comportamiento del sistema en coherencia y latencia frente a la línea base se evaluó en la configuración nominal (sección 4.2). La variación de hiperparámetros se evaluó únicamente sobre magnitudes internas del motor. No se dispone de evidencia sobre cómo varían la coherencia y la latencia a lo largo de la malla.

4.3.2 Protocolo del estudio de sensibilidad

Se define una malla factorial completa sobre los dos hiperparámetros de mayor incidencia en el estado semántico:

- α (factor del promedio móvil exponencial): {0,20; 0,25; 0,30; 0,35; 0,40}
- c (fuerza de acoplamiento): {0,30; 0,40; 0,50; 0,60; 0,70}

Las 25 configuraciones resultantes se evalúan sobre el subconjunto del corpus compuesto por los 17 diálogos multiturno (40 turnos, 23 transiciones inter-turno), por

ser los únicos en que α tiene efecto observable: en consultas de turno único no existe estado previo con el que promediar.

Se registran dos variables de respuesta. La primera es la **continuidad del estado**, ψ_t · ψ_{t−1}, que mide la persistencia del vector semántico entre turnos consecutivos. La segunda es la **fidelidad del estímulo**, φ_t · ψ_{t−1}, que mide el grado de alineamiento entre el vector de entrada del turno actual y el estado acumulado hasta el turno anterior. La distinción entre ambas es metodológicamente central y se desarrolla en 4.3.4.

La exploración del parámetro θ_quiebre sobre el rango {0,40; 0,50; 0,60; 0,70; 0,80}, orientada a determinar si algún umbral activaría el detector de quiebre sobre datos reales (resultado negativo documentado en 4.1.4d), no forma parte de esta malla y se identifica como trabajo pendiente en 5.2.

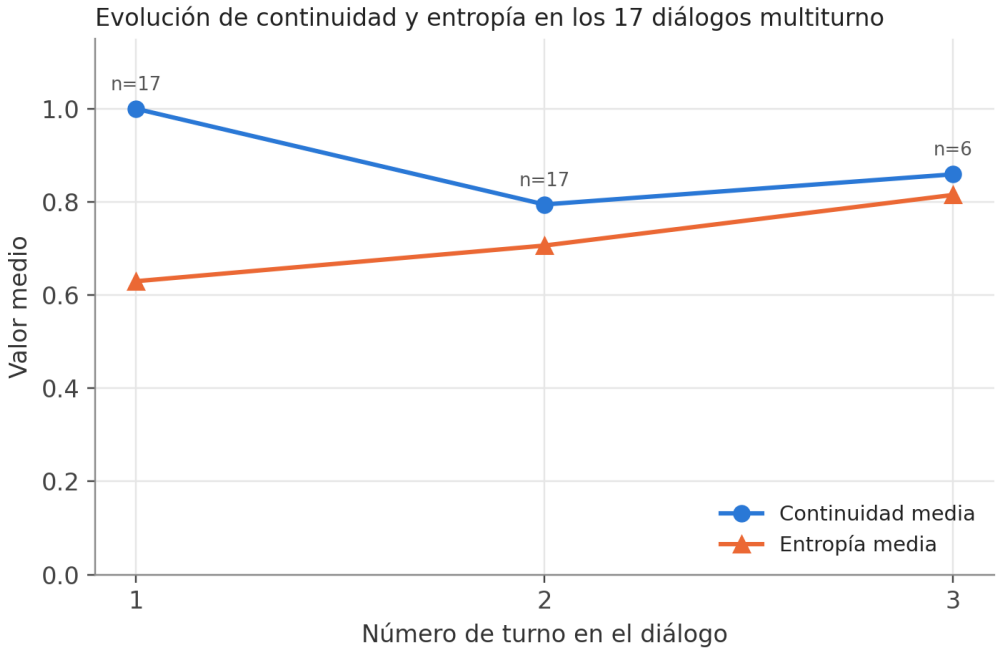


Figura 4.8. Evolución de la continuidad media y de la entropía media a lo largo de los tres primeros turnos, restringida a los 17 diálogos multiturno de la campaña.

4.3.3 Criterio de identificación de la región de operación

Se considerará confirmada la existencia de una región compacta de operación si se cumplen dos condiciones: que la continuidad media se mantenga dentro de una banda de ±0,05 respecto de su máximo en al menos el 20 % de las celdas de la malla, y que dichas celdas sean contiguas en el espacio (α, c). El resultado se representa mediante el mapa de calor bidimensional de la Figura 4.9.

Ambas condiciones fueron declaradas con anterioridad a la ejecución de la malla, de modo que el criterio de aceptación opera como preregistro y no como racionalización posterior de los datos observados. La interpretación de su cumplimiento se discute en 4.3.4 a la luz de la magnitud efectiva de la variación observada.

4.3.4 Resultados de la malla y contraste de H_3

La Tabla 4.15 y la Figura 4.9 presentan la continuidad media del estado para las 25 configuraciones. El valor máximo se localiza en (α = 0,20; c = 0,30) con C = 0,9872 y el mínimo en (α = 0,40; c = 0,70) con C = 0,9667, lo que arroja una variación absoluta de Δ = 0,0206 sobre la totalidad del plano paramétrico. El 100 % de las celdas (25 de 25) queda dentro de la banda de ±0,05 respecto del máximo, y el análisis de componentes conexas con vecindad de cuatro confirma que forman una única región contigua (k = 1). Los dos criterios preregistrados en 4.3.3 se cumplen.

Ahora bien, ese cumplimiento debe leerse con precisión. La banda de tolerancia de ±0,05 es más amplia que la variación total de la superficie (0,0206), de modo que el resultado de 25 celdas sobre 25 estaba determinado por la escasa dispersión del fenómeno y no constituye por sí solo una demostración de robustez. Se requiere, por tanto, una lectura que distinga entre el efecto forzado por la construcción del estimador y el comportamiento efectivamente informativo del motor.

Componente determinado por construcción. La actualización del estado mediante promedio móvil exponencial implica que ψ_t conserva una fracción (1 − α) de ψ_{t−1}. En consecuencia, existe un piso analítico de continuidad que se alcanzaría incluso si el vector de entrada no guardase relación alguna con el estado previo: 0,9701 para α = 0,20 y 0,8321 para α = 0,40. El descenso monótono de la

continuidad a lo largo del eje α, así como la posición del máximo en el valor más bajo de α, son consecuencias algebraicas de esta construcción y no hallazgos empíricos.

Componente informativo. El comportamiento del motor se manifiesta en el margen que separa los valores observados de dicho piso analítico y, de forma más directa, en la fidelidad del estímulo. Esta segunda métrica no está sujeta al efecto de suavizado, puesto que compara el vector de entrada sin procesar contra el estado acumulado. Su valor medio sobre la malla completa es de 0,8083, con un rango de apenas 0,0146 entre la configuración más alta y la más baja. Dos lecturas se desprenden de ello. Primera: las entradas de los diálogos multiturno presentan un alineamiento semántico sustancial con el estado que el motor ha acumulado, lo que indica que la representación de doce dimensiones retiene información con valor predictivo sobre la dirección del turno siguiente y no se limita a arrastrar inercia. Segunda: dicho alineamiento resulta prácticamente invariante frente a (α, c), lo que significa que la capacidad representacional del motor no depende de la calibración fina de estos dos hiperparámetros.

Magnitud relativa de la variación. La dispersión interna de una sola configuración es del mismo orden que la variación entre configuraciones: la desviación estándar media de la continuidad dentro de cada celda es de 0,0137, frente a un rango de 0,0206 en toda la malla. Desplazarse de un extremo al otro del espacio paramétrico produce, por tanto, un efecto comparable a la variabilidad natural entre transiciones de un mismo ajuste. Esta comparación constituye el argumento de estabilidad más defendible del estudio, por ser independiente de la anchura de la banda preregistrada.

Efecto diferenciado de los dos factores. La variación a lo largo del eje α alcanza 0,0200 puntos, mientras que a lo largo del eje c no supera las tres milésimas en ninguna fila. La fuerza de acoplamiento resulta, en consecuencia, casi inerte sobre la continuidad del estado. Este hallazgo es coherente con la limitación documentada en 4.4.7 respecto de la dimensionalidad efectiva del motor: si solo seis de las doce dimensiones llegaron a ser dominantes y dos concentraron el 80,1 % de los casos,

la matriz de acoplamiento dispone de un margen reducido para redistribuir activación entre dimensiones afines. El resultado no invalida el mecanismo de acoplamiento, pero sí acota su contribución observable sobre este corpus.

Validación de la configuración nominal. Los valores por defecto del sistema (α = 0,30; c = 0,50) arrojan una continuidad media de 0,9774 con desviación estándar de 0,0132 y un mínimo de 0,9542 en la peor transición registrada, junto con una fidelidad de estímulo de 0,8091. La configuración nominal se sitúa en el interior de la región estable y a 0,0098 puntos del máximo de la malla, distancia inferior a la propia desviación estándar intra-configuración. La parametrización empleada en la campaña de 4.2 queda así justificada a posteriori, sin que los resultados de dicha campaña dependan de una calibración particular.

Estado de H_3. La hipótesis postulaba la existencia de una región compacta y contigua del espacio (α, c) en la que el desempeño del motor se mantiene estable. Los dos criterios preregistrados se cumplen y la evidencia adicional —fidelidad de estímulo invariante, variación entre celdas comparable a la dispersión intra-celda— apunta en la misma dirección. H_3 se cumple en los términos en que fue preregistrada, pero ese cumplimiento estaba determinado en buena medida por el diseño. La banda de ±0,05 resultó más ancha que la variación total de la superficie (0,0206), en parte por el piso algebraico que el promedio móvil impone a la continuidad del estado; el criterio quedó así satisfecho por construcción y no constituye una prueba exigente. La evidencia con valor propio es más limitada: la fidelidad del estímulo es prácticamente invariante en la malla (rango de 0,0146) y la variación entre celdas es comparable a la dispersión intra-celda. La formulación defendible es que no se observó fragilidad paramétrica en el rango explorado, sobre métricas internas del motor y sin medir coherencia ni latencia.

Tabla 4.15. Continuidad media del estado (ψ_t · ψ_{t−1}) por configuración de la malla factorial (n = 23 transiciones por celda)

α \ c	0,30	0,40	0,50	0,60	0,70
0,20	0,9872	0,9860	0,9850	0,9843	0,9838
0,25	0,9834	0,9823	0,9816	0,9810	0,9805

α \ c	0,30	0,40	0,50	0,60	0,70
0,30	0,9788	0,9780	0,9774	0,9769	0,9766
0,35	0,9734	0,9728	0,9724	0,9722	0,9720
0,40	0,9672	0,9669	0,9668	0,9667	0,9667

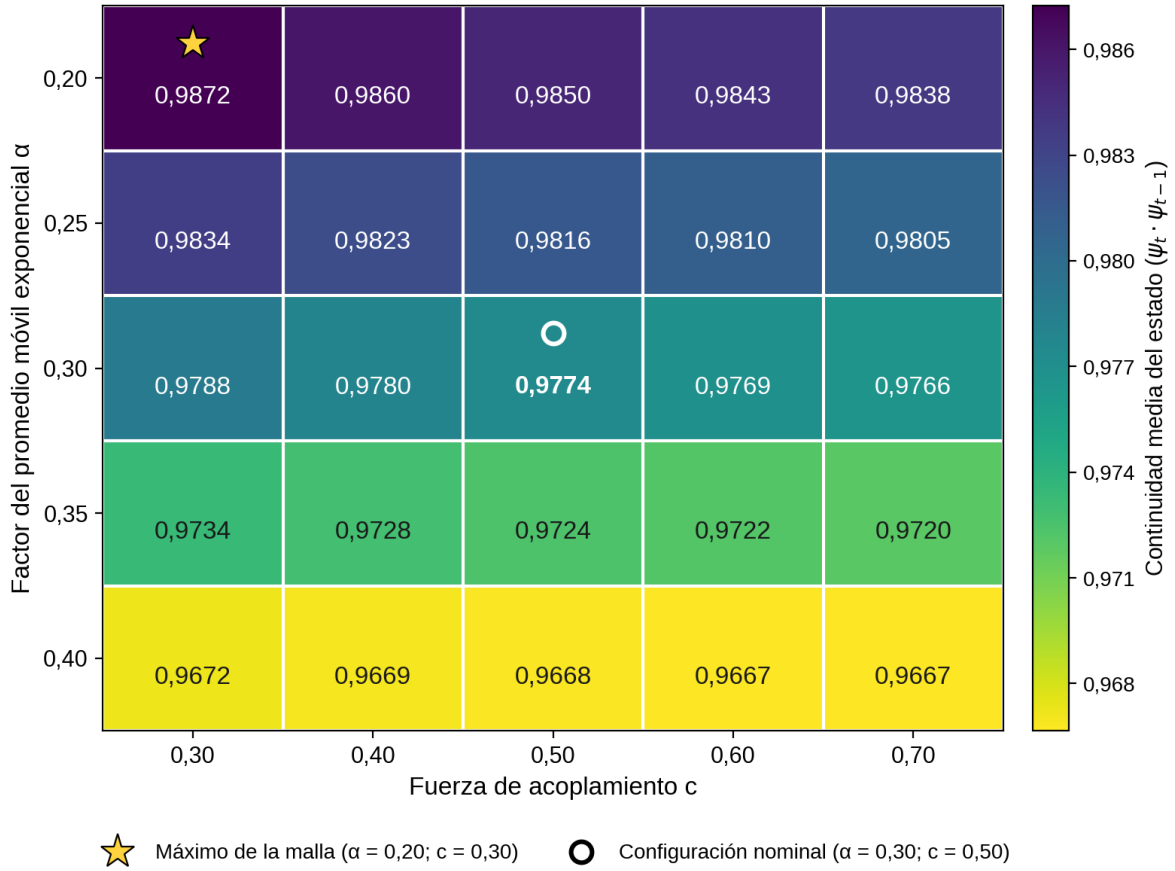


Figura 4.9. *Mapa de sensibilidad factorial de la continuidad media del estado en el plano (α, c) sobre el subconjunto multiturno (23 transiciones por configuración). La banda de estabilidad corresponde al criterio de ±0,05 preregistrado en 4.3.3.*

4.4 Discusión según variables

4.4.1 Variable coherencia

El contraste directo sobre coherencia (sección 4.2.4) no encontró una diferencia estadísticamente significativa entre condiciones (Wilcoxon, p = 0,8974; d_z = 0,04; IC 95 % [−0,10; 0,14]): la hipótesis H_1 no recibe respaldo empírico. La evidencia

instrumental disponible permite interpretar este resultado nulo. La Tabla 4.4 muestra que el motor discrimina de forma determinista tres regímenes de dispersión y les asigna estrategias de generación diferenciadas; la Tabla 4.5 muestra que los mensajes de los diálogos multiturno mantienen un alineamiento alto con el estado acumulado. Que estos mecanismos operen correctamente a nivel interno y, aun así, no se traduzcan en una mejora perceptible de la coherencia para un interlocutor humano admite una explicación en dos niveles. En el plano del diseño experimental, y conforme a la precisión introducida en 4.2.1, ambas condiciones compartían una ventana de historial de dos turnos: el contraste mide el aporte marginal de la directiva sobre un generador que ya disponía de contexto reciente, no su aporte frente a un generador sin memoria alguna. Parte del efecto que H_1 postulaba se encontraba, por tanto, ya presente en la línea base. En el plano del mecanismo, la evidencia sugiere además que la directiva textual inyectada en el prompt no constituye un canal suficientemente potente para transmitir la información de estado al SLM, dado que su contribución no resulta distinguible del ruido una vez que el contexto reciente está disponible. Ambas lecturas son compatibles entre sí y convergen en la misma orientación para el trabajo futuro (sección 5.2.3): mecanismos de condicionamiento más directos que la inyección textual.

Deben distinguirse dos capacidades que el resultado nulo no afecta por igual. La primera es la capacidad del motor para representar y mantener el contexto, respaldada por la evidencia instrumental: el estado acumulado mantiene un alineamiento alto con los mensajes sucesivos de cada diálogo (Tabla 4.5) y sus magnitudes se traducen en decisiones deterministas y auditables (Tabla 4.4). La segunda es la efectividad con que ese contexto se transmite al Small Language Model mediante una directiva textual, que no está respaldada, puesto que la coherencia percibida no difirió entre condiciones. El resultado nulo de H_1 recae sobre el canal de transmisión y sobre el diseño del contraste, y no permite concluir, ni a favor ni en contra, sobre la utilidad de la representación con un canal distinto.

La implementación evaluada acota además ese canal. El estilo de la directiva se decide con la entropía del mensaje actual, y el contexto acumulado por el motor

llega al modelo únicamente como el nombre de la dimensión dominante, en la forma «Foco central: incertidumbre» (Anexo D). Dado que incertidumbre y social fueron dominantes en el 80,1 % de los turnos, esa señal aportó poca información diferencial. El resultado nulo de H_1 es coherente con un canal tan estrecho y orienta el trabajo futuro hacia formas más ricas de transmitir el estado.

4.4.2 Variable latencia

La tercera pregunta de sistematización interroga sobre el comportamiento del sistema en latencia frente al baseline. La respuesta que emerge de 4.2.2, para la configuración nominal, es que el motor de contexto es computacionalmente transparente. Este hallazgo es coherente con la naturaleza de las operaciones involucradas: el motor realiza un producto matriz-vector de dimensión 12 × 12, una combinación lineal y el cálculo de una entropía sobre 12 componentes, todo ello con un costo medido de aproximadamente veinticuatro microsegundos por turno, frente a una generación autorregresiva cuya latencia media fue de 18,2 s en la condición A y de 17,5 s en la condición B (Tabla 4.7): una razón cercana a los seis órdenes de magnitud.

La correlación r = 0,961 entre longitud de respuesta y latencia (4.1.5) cuantifica esta asimetría: prácticamente toda la varianza de la latencia se explica por cuántos tokens genera el modelo. Un motor de contexto de coste marginal nulo, insertado en un sistema cuyo cuello de botella es seis órdenes de magnitud mayor, no puede producir un efecto detectable. Este es, en sí mismo, un argumento arquitectónico a favor del diseño: la funcionalidad añadida se obtiene sin coste temporal.

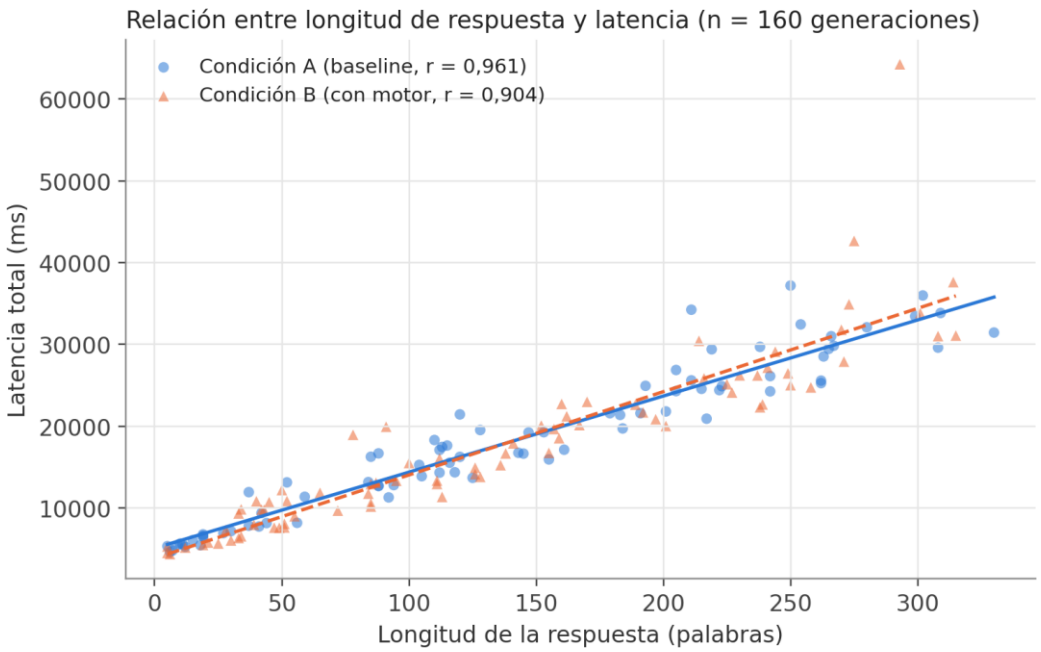


Figura 4.10. Dispersión de la longitud de la respuesta (en palabras) frente a la latencia total (en milisegundos), condiciones A y B, n = 160 generaciones.

4.4.3 Variable huella de carbono

La huella estimada de 0,010537 gCO_2eq por turno responde a la cuarta pregunta de sistematización con una estimación instrumentada en tiempo de ejecución sobre el propio equipo, y no con cifras tomadas de la literatura. Tres consideraciones cualifican el resultado.

Primera: la medición corresponde a inferencia, no a entrenamiento. El sistema no entrena el SLM; lo consume. La comparación honesta con Jiang et al. (2024) es, por tanto, parcial.

Segunda: el factor de emisión ecuatoriano (0,1616 $kgCO_2eq/kWh$) es comparativamente bajo por el peso de la generación hidroeléctrica en la matriz nacional. La misma carga computacional en una red con matriz de mayor intensidad carbónica produciría emisiones proporcionalmente superiores. La cifra reportada es específica del contexto geográfico declarado en la delimitación del estudio.

Tercera: CodeCarbon estima el consumo a partir de la utilización de CPU y RAM mediante modelos de potencia, no mediante medición eléctrica directa en el

enchufe. La precisión absoluta está acotada por ese método; la precisión relativa entre condiciones, que es lo que interesa a la comparación, no lo está.

Con esas salvedades, la conclusión se sostiene: el sistema opera en el orden de milésimas de gramo de CO_2eq por interacción, con la totalidad del cómputo confinada al dispositivo del usuario.

4.4.4 Variable representación semántica

Los resultados de 4.1.4(c) obligan a matizar una de las afirmaciones del planteamiento. El motor representa el diálogo en un espacio de 12 dimensiones, pero sobre este corpus solo 6 dimensiones llegaron a ser dominantes y dos concentraron el 80 % de los casos. La dimensionalidad efectiva es inferior a la nominal. La justificación de las doce dimensiones es de diseño y no empírica: se definieron por criterio del autor (sección 3.1.4) y no fueron validadas mediante anotación independiente ni mediante un análisis de la varianza que explican. Debe señalarse además un sesgo del vocabulario de activación. Los marcadores interrogativos del español —cómo, qué, cuál, por qué— pertenecen al vocabulario de la dimensión incertidumbre y aparecen en la mayoría de las consultas con independencia de su tema, de modo que esa dimensión se activa por la forma interrogativa del mensaje y no por su contenido. El 46,3 % de dominancia de incertidumbre refleja, en parte, ese artefacto léxico y no incertidumbre semántica; su corrección se recoge en 5.2.2.

Esto tiene dos lecturas. La restrictiva: el codificador basado en coincidencia de palabras clave es sensible al vocabulario del corpus y no generaliza a formulaciones que no contienen los términos ancla. La constructiva: aun operando sobre un subespacio, las métricas derivadas del estado —entropía y continuidad— resultaron informativas y discriminantes, como acreditan las Tablas 4.5 y 4.6 La utilidad del mecanismo no depende de que las doce dimensiones se activen uniformemente, sino de que la distribución resultante varíe de forma sistemática con el contenido de la consulta, lo cual sí se verifica.

La brecha del clasificador (0,775 en validación cruzada frente a 0,600 en el corpus experimental, sección 4.1.3) apunta a la misma raíz: un corpus de entrenamiento de 201 ejemplos es insuficiente para que una representación TF-IDF generalice a vocabulario nuevo. Ambas limitaciones —codificador y clasificador— comparten diagnóstico y comparten remedio: ampliación del corpus o sustitución de la representación léxica por embeddings densos, opción esta última que debe ponderarse contra el objetivo de sostenibilidad, pues introduce un modelo adicional en la ruta de inferencia.

4.4.5 Contraste con la literatura

Los resultados dialogan con las tres corrientes que fundamentan el trabajo.

Respecto de Green AI (Jiang et al., 2024), la medición aporta un dato empírico al debate: la inferencia conversacional local en hardware sin GPU es cuantificable y de baja magnitud. La contribución no es teórica sino instrumental, al documentar una cifra reproducible obtenida con herramienta estándar.

Respecto de los Small Language Models (Belcak et al., 2025), la campaña confirma la premisa de operatividad: qwen2.5:1.5b completó 160 generaciones sin fallo sobre CPU doméstica, con una latencia media de 18,2 s en la condición A y de 17,5 s en la condición B, desviaciones estándar de 9,1 y 10,4 s y respuestas que superan los 20 s. Este régimen limita el uso del sistema como chatbot interactivo: es viable para consulta asíncrona, pero no para una conversación fluida en tiempo real sobre este hardware. Esta observación delimita el ámbito de aplicabilidad del sistema.

Respecto del paradigma neuro-simbólico (Garcez & Lamb, 2023), la arquitectura ejemplifica la combinación propuesta: un componente simbólico el motor de contexto, cuyo estado es inspeccionable en su totalidad y cuyas reglas de decisión son deterministas y auditables, según acredita la Tabla 4.4 gobierna a un componente neuronal opaco. El dashboard desarrollado (objetivo específico 4) materializa esa interpretabilidad al exponer cada métrica interna por turno. La transparencia del componente simbólico es, en este trabajo, una propiedad verificada y no una aspiración.

Respecto del aporte diferencial del motor frente a un mecanismo convencional de gestión de contexto (sección 2.1.6), los resultados confirman las características de interpretabilidad y bajo costo —estado inspeccionable, reglas deterministas, costo del orden de microsegundos— y no confirman una ventaja en calidad conversacional frente a la ventana de historial de dos turnos que ya incluía la línea base. No se comparó contra otros mecanismos convencionales, como el resumen del historial o la recuperación de fragmentos, por lo que no se afirma superioridad frente a ellos.

4.4.6 Estado de las hipótesis

Tabla 4.16. Estado de las hipótesis y síntesis de la evidencia experimental

Hipótesis	Enunciado	Estado	Evidencia
H_1	El motor mejora la coherencia conversacional	No sostenida	A = 3,91, B = 3,93; Wilcoxon W = 1255,5, p = 0,8974; d_z = 0,04; IC 95 % [−0,10; 0,14]. Sin efecto detectable en ninguna dirección (Tablas 4.17–4.18)
H_2	El costo computacional del motor es aceptable	Compatible con la evidencia, no demostrada	Δlatencia = −654,8 ms (p = 0,28); ΔCO_2 = −0,000355 g (p = 0,34); ambos IC contienen el cero
H_3	Existe una región compacta de hiperparámetros	Cumple el criterio preregistrado; criterio poco exigente por diseño	Malla 5×5 (n = 23 transiciones/celda): máximo C = 0,9872 en (0,20; 0,30); 25/25 celdas dentro de la banda preregistrada de ±0,05; una única componente conexa (k = 1); Δ = 0,0206, comparable a la DE intra-celda (0,0137)

La reserva sobre H_2 merece explicitarse. La hipótesis postulaba una sobrecarga inferior a 500 ms; lo observado es una diferencia negativa y no significativa, lo que es compatible con la hipótesis pero no la demuestra en sentido estricto: la ausencia de efecto detectable y la ausencia de efecto no son equivalentes. Con n = 80 y una desviación de las diferencias de 7 963 ms, el estudio carece de potencia para acotar el efecto por debajo de 500 ms. La afirmación defendible es que **no se observa sobrecarga**, no que se haya demostrado que la sobrecarga sea menor a 500 ms.

4.4.7 Limitaciones del estudio

El clasificador de intención presenta una limitación de alcance que condiciona la interpretación del conjunto del sistema. La brecha entre la validación cruzada sobre el corpus de entrenamiento (exactitud 0,775) y el desempeño sobre el corpus experimental independiente (0,600; macro-F1 0,593; κ = 0,520) asciende a 17,5 puntos porcentuales y revela una capacidad de generalización insuficiente. Dos categorías operan por debajo de una exhaustividad de 0,45 —Bienestar (0,385) y Tecnología (0,429)—, lo que implica que aproximadamente seis de cada diez turnos de esas categorías reciben una etiqueta incorrecta. El mecanismo de propagación debe precisarse. El clasificador no entrega al motor una etiqueta, sino su distribución completa de probabilidad: cada probabilidad se proyecta, con un peso fijo de 0,40, sobre las tres dimensiones semánticas asociadas a su categoría y se suma a la activación léxica por palabras clave (Código C.8). De ello se siguen tres consecuencias. En primer lugar, un error de clasificación no sustituye el vector de entrada, sino que desplaza parte de su masa hacia dimensiones asociadas a una categoría incorrecta; el efecto es gradual y está acotado por el peso de proyección. En segundo lugar, la baja confianza media (0,4881) reparte esa masa entre varias categorías y tiende a elevar la entropía del vector de entrada, que es la magnitud con la que se selecciona la directiva; la incertidumbre del clasificador puede, por tanto, desplazar turnos hacia directivas equilibradas o exploratorias por razones ajenas al contenido de la consulta. Por último, el efecto es mayor en Bienestar y Tecnología. En cuanto a la interpretación de los resultados, el error del clasificador solo actúa sobre la condición B, a través de la directiva: no puede producir una

ventaja espuria del motor, pero sí atenuar su efecto observable. El resultado nulo de H_1 es, por tanto, compatible con dos explicaciones que este diseño no permite separar: que la directiva textual no mejore la coherencia, o que su efecto haya quedado diluido por el error del clasificador. El origen de esa brecha de generalización es doble: un corpus de entrenamiento de 201 ejemplos, insuficiente para la granularidad de seis clases, y una representación por bolsa de n-gramas incapaz de capturar sinonimia o variación morfológica fuera del vocabulario observado.

Primera, el diseño de la comparación A–B aísla el efecto de la directiva de contexto, pero no el del motor en su conjunto: ambas condiciones compartían la ventana de historial conversacional de dos turnos descrita en 4.2.1. Un contraste frente a una línea base estrictamente sin memoria habría permitido estimar el aporte total del componente de gestión de contexto, y se identifica como extensión necesaria del protocolo. El resultado nulo de H_1 debe interpretarse, en consecuencia, como ausencia de aporte marginal de la directiva y no como ausencia de aporte del mecanismo de memoria conversacional en su totalidad.

Segunda, la dimensionalidad efectiva del motor de contexto resultó inferior a la nominal: de las doce dimensiones semánticas definidas, solo seis llegaron a ser dominantes sobre el corpus experimental, y dos de ellas concentraron el 80,1 % de los casos (sección 4.1.4c). Esta limitación es atribuible al vocabulario léxico de activación empleado por el codificador semántico, y no al mecanismo de acoplamiento ni a las métricas derivadas de él, que se mantuvieron informativas incluso operando sobre este subespacio reducido.

Tercera, el detector de quiebre temático no pudo verificarse empíricamente: el indicador break_detected permaneció en falso durante la totalidad de los ochenta turnos evaluados, dado que la continuidad mínima observada en la campaña no llegó a cruzar el umbral nominal θ_quiebre = 0,50 (sección 4.1.4d). Esta

funcionalidad queda documentada como característica paramétrica pendiente de calibración.

Cuarta, la potencia estadística del diseño experimental es limitada. Con ochenta turnos pareados y una desviación estándar de las diferencias de 7 963 ms, el estudio no permite acotar la sobrecarga de latencia por debajo del umbral de 500 ms postulado en la hipótesis H_2; la afirmación defendible es la ausencia de un efecto detectable, no la demostración de que dicho efecto sea inferior al umbral (sección 4.4.6).

Quinta, el corpus experimental incluye únicamente diecisiete diálogos multiturno de un total de cincuenta y siete consultas, lo que restringe la robustez estadística de las conclusiones sobre retención de contexto a una submuestra reducida (sección 4.1.2). El corpus tampoco contiene consultas con cambios temáticos deliberados y abruptos, condición necesaria para ejercitar el detector de quiebre temático.

Sexta, la generación no fijó semilla y empleó una temperatura decreciente por turno, entre 0,90 y 0,83 (Anexo D); repetir una consulta puede producir una respuesta distinta, de modo que la reproducibilidad del estudio es estadística y no exacta.

Estas limitaciones no invalidan los resultados obtenidos, pero acotan su alcance: los hallazgos de este capítulo son válidos para el corpus, el hardware y la configuración de hiperparámetros empleados, y su generalización a otros contextos de despliegue debe entenderse como una hipótesis de trabajo para investigaciones futuras, no como una conclusión establecida.

Tabla 4.17. Estadísticos descriptivos de la puntuación agregada de coherencia por condición

Estadístico	Condición A	Condición B
Media	3,91	3,93
Desviación estándar	0,43	0,40
Mediana	3,87	3,87

Estadístico	Condición A	Condición B
Mínimo	2,67	2,47
Máximo	4,80	4,67
n	80	80

Tabla 4.18. Contrastes inferenciales de la puntuación agregada de coherencia entre condiciones

Contraste	Valor
Diferencia media (B − A)	0,02
IC 95 % de la diferencia	[−0,10; 0,14]
Estadístico W (Wilcoxon)	1255,5
Valor p	0,8974
Tamaño del efecto d_z	0,04
Concordancia interjueces κ_w	0,64

4.5 Síntesis del capítulo

La fase experimental permite establecer seis resultados verificados.

Primero, **el sistema es operativo en su configuración objetivo**: 160 generaciones completadas sin fallo sobre hardware doméstico sin GPU, con las tres capas funcionando de forma integrada y con instrumentación completa por turno.

Segundo, **el motor de contexto es computacionalmente transparente**: no se detecta sobrecarga de latencia (Δ = −654,8 ms; p = 0,28) ni de emisiones (Δ = −0,000355 gCO_2eq; p = 0,34) atribuible a su operación. La latencia del sistema está dominada por la generación del SLM (r = 0,961 con la longitud de respuesta).

Tercero, **la huella de carbono operativa es de 0,010537 gCO_2eq por turno**, estimada con CodeCarbon y con el factor de emisión de la red ecuatoriana. La campaña completa emitió 1,7142 gCO_2eq.

Cuarto, **los mecanismos internos del motor operan conforme a especificación**: los umbrales de entropía discriminan tres regímenes de respuesta sin solapamiento, y la continuidad del estado evidencia retención de contexto en diálogos multiturno.

Quinto, **la hipótesis principal H_1 no recibe respaldo empírico**: la evaluación de coherencia mediante panel de jueces (4.2.4) no encontró una diferencia estadísticamente significativa entre condiciones.

Sexto, **la hipótesis H_3 cumple el criterio preregistrado, con la salvedad de que este quedó satisfecho en buena medida por construcción (sección 4.3.4):** en el estudio de sensibilidad de hiperparámetros reportado en 4.3, las veinticinco configuraciones de la malla factorial sobre α y c se mantienen dentro de la banda de estabilidad preregistrada y conforman una única región contigua, de modo que el motor de contexto no exhibe fragilidad paramétrica sobre el rango explorado.

CAPÍTULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

La presente investigación se planteó determinar si es posible diseñar un chatbot conversacional coherente en español que opere íntegramente sobre hardware doméstico —sin unidad de procesamiento gráfico ni dependencia de la nube— integrando un Small Language Model local con un motor de contexto quantum-inspired y un clasificador clásico, estimando su huella ambiental de operación mediante instrumentación en tiempo de ejecución. Las conclusiones se organizan en torno a los cuatro objetivos específicos formulados en el Capítulo 1 y se sustentan en la evidencia experimental presentada en el Capítulo 4.

5.1.1 Sobre el objetivo específico 1: arquitectura de software de tres capas

Se concluye que la arquitectura modular de tres capas es viable y estable en la configuración objetivo. La campaña experimental completó ciento sesenta generaciones —ochenta turnos bajo dos condiciones— sin fallo de servicio, sin respuestas vacías y sin necesidad de recurrir al mecanismo de degradación elegante previsto para la indisponibilidad del modelo de lenguaje. El sistema operó sobre un equipo Intel Core i7 de undécima generación con veintiocho gigabytes de memoria y sin unidad de procesamiento gráfico, consumiendo cuarenta y siete minutos de cómputo agregado.

El principio de responsabilidad única aplicado al diseño se verificó en la práctica durante la propia fase experimental: la condición de control se implementó desconectando exclusivamente la inyección de la directiva de contexto en la capa de generación, sin modificar el clasificador ni el motor. Esta separación es la que hizo metodológicamente posible el contraste A–B.

5.1.2 Sobre el objetivo específico 2: motor de contexto de doce dimensiones

Se concluye que el motor de contexto opera conforme a su especificación formal y que sus métricas internas son informativas. La verificación por componente arrojó tres resultados concluyentes.

En primer lugar, la regla de decisión que asigna estrategias de generación según la dispersión semántica del mensaje de cada turno funciona de manera determinista y sin ambigüedad: los tres regímenes de entropía observados sobre los ochenta turnos —directa, equilibrada y exploratoria— ocuparon rangos disjuntos, sin un solo caso de solapamiento. La distribución resultante (51,3 %, 33,8 % y 15,0 % respectivamente) evidencia que el mecanismo discrimina entre consultas focalizadas y dispersas en lugar de reducirse a una estrategia única. Esto responde solo en parte a la segunda pregunta de sistematización. Las métricas internas de dispersión resultan suficientes para seleccionar una estrategia de respuesta de forma determinista y diferenciada, sin recurrir a modelos de mayor tamaño. No se demostró, en cambio, que guiar la respuesta con esas métricas mejore su calidad: la evaluación por jueces no encontró diferencia de coherencia entre condiciones (sección 4.2.4).

En segundo lugar, los mensajes de los diálogos multiturno mantienen un alineamiento alto con el estado acumulado por el motor: la fidelidad del estímulo, registrada en la campaña como continuidad, fue de 0,7944 en el segundo turno y de 0,8592 en el tercero. El estado acumulado conserva, por tanto, información sobre la dirección del diálogo, aunque esa información llega al modelo de lenguaje por un canal estrecho (sección 4.4.1).

En tercer lugar, y con igual peso conclusivo, se identificaron dos limitaciones reales del mecanismo. La dimensionalidad efectiva del espacio semántico resultó inferior a la nominal: seis de las doce dimensiones nunca fueron dominantes sobre este corpus, y dos de ellas —incertidumbre y social— concentraron el 80,1 % de los casos. Asimismo, el detector de quiebre temático no se activó en ninguno de los ochenta turnos, por lo que esta funcionalidad queda documentada como no verificada empíricamente y no se le atribuye contribución alguna a los resultados obtenidos.

5.1.3 Sobre el objetivo específico 3: evaluación experimental

Se concluye que el motor de contexto es computacionalmente transparente. No se detectó sobrecarga de latencia atribuible a su operación (diferencia media de −654,80 ms; prueba de Wilcoxon, p = 0,2805; tamaño del efecto d_z = −0,082) ni de emisiones de dióxido de carbono (diferencia media de −0,000355 gCO_2eq; p = 0,3386; d_z = −0,074). Ambos intervalos de confianza al 95 % contienen el cero.

Este resultado admite una explicación mecánica y no meramente estadística. La latencia del sistema está dominada por la generación autorregresiva del modelo de lenguaje: la correlación entre la longitud de la respuesta producida y el tiempo total de respuesta alcanzó r = 0,961 en la condición de control y r = 0,904 en la condición experimental. El motor de contexto, cuyo costo se reduce a un producto matriz-vector de dimensión doce por doce y al cálculo de una entropía sobre doce componentes, opera en una escala temporal varios órdenes de magnitud inferior a la del cuello de botella del sistema.

Debe reconocerse, no obstante, una limitación de uso. La latencia media fue de 18,2 s por turno en la línea base y de 17,5 s con el motor, con respuestas que superaron los 20 s. Este tiempo de respuesta limita la utilidad del sistema como chatbot interactivo sobre el hardware evaluado: resulta adecuado para consulta asíncrona, no para una conversación fluida. La viabilidad que este trabajo acredita es técnica y de costo, no de experiencia de uso.

Debe formularse esta conclusión con la precisión que la evidencia permite. La hipótesis H_2 postulaba una sobrecarga inferior a quinientos milisegundos; lo observado es una diferencia negativa y no significativa. Ausencia de efecto detectable y ausencia de efecto no son equivalentes: con una desviación de las diferencias de 7 963 ms, la potencia estadística del estudio no permite acotar el efecto por debajo del umbral postulado, y el límite superior del intervalo de confianza no descarta una sobrecarga de hasta 1 128 ms. La afirmación defendible es que no se observa sobrecarga, no que se haya demostrado que esta sea inferior a quinientos milisegundos.

Respecto de la cuarta pregunta de sistematización, se concluye que la huella de carbono del sistema operando localmente, estimada mediante CodeCarbon, es de 0,010537 gCO_2eq por turno conversacional. La campaña experimental completa emitió 1,7142 gCO_2eq. El factor de emisión aplicado, 0,1616 t CO_2-eq/MWh, corresponde al valor oficial publicado por el Operador Nacional de Electricidad para el Ecuador continental en el ejercicio 2024, calculado bajo el Alcance 2 del Protocolo de Gases de Efecto Invernadero. La magnitud de este factor se explica porque la generación de bajo costo —hidráulica y renovable no convencional— representó el 87,11 % promedio de la generación del Sistema Nacional Interconectado en el quinquenio analizado; en consecuencia, la cifra reportada es específica del contexto geográfico declarado en la delimitación del estudio y no extrapolable a redes eléctricas de mayor intensidad carbónica.

Esta cifra corresponde exclusivamente a la huella operacional de la inferencia local bajo las condiciones del experimento. No constituye una evaluación integral de sostenibilidad: no incluye la fabricación ni el fin de vida del hardware, el entrenamiento del modelo qwen2.5:1.5b realizado por terceros, ni el consumo del equipo en reposo. Las conclusiones ambientales de este trabajo se limitan, por tanto, al consumo y a las emisiones durante la inferencia.

Se concluye asimismo que el clasificador de intención presenta una brecha de generalización que debe consignarse sin atenuación. La exactitud descendió de 0,775 en validación cruzada estratificada sobre el corpus de entrenamiento a 0,600 sobre el corpus experimental independiente, con una confianza media de predicción de 0,4881. Las categorías más afectadas fueron Bienestar (0,385) y Tecnología (0,429). El diagnóstico coincide con el de la limitación dimensional del codificador: una representación por bolsa de n-gramas entrenada sobre doscientos un ejemplos no generaliza a vocabulario no observado.

Finalmente, se concluye que la hipótesis principal del trabajo no recibe respaldo empírico. La evaluación de coherencia conversacional mediante panel de tres jueces ciegos sobre los ochenta pares de respuestas (sección 4.2.4) no encontró una diferencia estadísticamente significativa entre condiciones (Wilcoxon, p =

0,8974; d_z = 0,04), con un intervalo de confianza al 95 % estrecho y centrado en cero. Los resultados obtenidos acreditan la viabilidad técnica y el costo del sistema, y descartan con razonable certeza un beneficio perceptible sobre la calidad conversacional en las condiciones evaluadas.

La campaña comparativa se ejecutó deliberadamente con los hiperparámetros fijados en su configuración nominal, a fin de aislar el efecto del motor; la hipótesis H_3, relativa a la existencia de una región compacta de hiperparámetros, se contrastó por tanto mediante un experimento independiente, cuyos resultados se sintetizan a continuación.

Se concluye, por último, que el motor de contexto no exhibe fragilidad paramétrica sobre el rango explorado. El estudio de sensibilidad factorial de veinticinco configuraciones sobre los factores α y c, reportado en 4.3, verifica los dos criterios preregistrados: la totalidad de las celdas se mantiene dentro de la banda de tolerancia de ±0,05 y conforma una única región contigua. La variación de continuidad entre configuraciones extremas es de 0,0206 puntos, magnitud comparable a la dispersión interna de una sola configuración (DE = 0,0137), y la fidelidad del estímulo permanece prácticamente invariante a lo largo de la malla (media de 0,8083, rango de 0,0146). La configuración nominal empleada en la campaña se sitúa en el interior de la región estable, a una distancia del óptimo inferior a la propia desviación intra-configuración, lo que justifica a posteriori la parametrización adoptada. La hipótesis H_3 cumple, en consecuencia, el criterio preregistrado. Ese cumplimiento estaba determinado en buena medida por el diseño del criterio, por lo que debe leerse como ausencia de fragilidad observada y no como demostración de robustez.

.

5.1.4 Sobre el objetivo específico 4: dashboard de visualización

Se concluye que el dashboard cumple la doble función prevista. Como herramienta de inspección durante el desarrollo, permitió verificar en tiempo real la coherencia entre el estado interno del motor y las respuestas generadas, facilitando la

depuración del sistema durante la fase de implementación. Como evidencia demostrativa, expone en tiempo real la totalidad del estado interno de cada turno: entropía, continuidad, dimensión dominante, distribución del clasificador, temperatura de recocido, potencia instantánea y emisiones acumuladas.

Esta transparencia constituye la materialización concreta del argumento neuro-simbólico que fundamenta la arquitectura: un componente simbólico enteramente inspeccionable y de reglas deterministas gobierna a un componente neuronal opaco. En este trabajo la interpretabilidad no es una aspiración de diseño sino una propiedad verificada, acreditada por la correspondencia sin solapamiento entre estados internos y estrategias de respuesta.

5.1.5 Conclusión general

Se concluye que es posible construir y operar un asistente conversacional en español sobre hardware doméstico sin unidad de procesamiento gráfico y sin transferencia de datos a terceros, con una huella de carbono operacional estimada del orden de milésimas de gramo de CO_2 equivalente por interacción, e incorporando un motor de contexto interpretable cuyo costo computacional resulta indetectable frente al de la generación de lenguaje. Su latencia media, cercana a los 18 segundos por turno, limita no obstante su uso como chatbot interactivo sobre el hardware evaluado. El beneficio de dicho motor sobre la coherencia conversacional, que constituye la pregunta central de la investigación, no encontró respaldo empírico conforme al contraste declarado en el Capítulo 4: la evaluación por panel de jueces no detectó una diferencia perceptible entre condiciones, dentro de un intervalo de confianza estrecho y centrado en cero. La estabilidad paramétrica del motor, postulada en H_3, se sostuvo en el estudio de sensibilidad de 4.3 solo en el sentido de que no se observó fragilidad, con un alcance limitado a métricas internas del motor y a un criterio poco exigente por diseño.

5.2 Recomendaciones

5.2.1 Recomendaciones para la extensión del estudio de sensibilidad

El estudio reportado en 4.3 cubre la malla factorial sobre α y c, pero deja abiertas tres extensiones que se derivan de su propio diseño.

Se recomienda ejecutar la exploración del parámetro θ_quiebre sobre el rango {0,40; 0,50; 0,60; 0,70; 0,80}, no abordada en la malla reportada. El detector de quiebre temático no se activó en ninguno de los 80 turnos de la campaña (sección 4.1.4d). Dado que el detector opera sobre la fidelidad del estímulo φ_t · ψ_(t−1) (Anexo A.5), cuyos valores medios en la campaña fueron de 0,7944 en el segundo turno y de 0,8592 en el tercero, el umbral nominal de 0,50 quedó lejos de los valores observados en un corpus sin cambios temáticos deliberados. Se recomienda, por tanto, que la exploración del umbral se realice sobre un corpus que incluya cambios de tema abruptos (sección 5.2.2).

Se recomienda completar el conjunto de variables de respuesta previsto en el protocolo. El diseño ejecutado, al operar sobre vectores semánticos precodificados sin invocar la capa de generación, no permitió registrar la distribución de directivas, la entropía media ni la latencia por configuración. La primera de ellas reviste particular interés: dado que la directiva se selecciona por comparación de la entropía contra umbrales deterministas, la estabilidad del reparto entre directiva directa, equilibrada y exploratoria a lo largo de la malla constituiría una verificación de estabilidad operativa complementaria y de naturaleza distinta a la reportada en 4.3.4. Su ejecución requiere las 1 000 generaciones originalmente estimadas y un costo aproximado de 4,9 horas de cómputo.

Se recomienda ampliar el rango explorado de α más allá de {0,20; 0,40}. La totalidad de las 25 configuraciones evaluadas satisfizo el criterio de estabilidad, lo que indica que la malla no alcanzó los límites de la región de operación del motor. Un rango extendido permitiría localizar la frontera en que el comportamiento deja de ser estable, información que el presente estudio, por construcción, no puede aportar: la evidencia disponible establece que existe una región compacta, pero no delimita su extensión.

5.2.2 Recomendaciones sobre las limitaciones identificadas

Se recomienda ampliar el corpus de entrenamiento del clasificador de intención más allá de los doscientos un ejemplos actuales, dado que la brecha de generalización documentada en 5.1.3 constituye la limitación de mayor impacto sobre el desempeño global del sistema. Como alternativa complementaria debe evaluarse la sustitución de la representación por n-gramas por embeddings densos de bajo costo, ponderando este beneficio contra el objetivo de sostenibilidad, por cuanto tal sustitución introduce un modelo adicional en la ruta de inferencia.

Se recomienda revisar el codificador semántico a fin de corregir la concentración de la activación en dos de las doce dimensiones. El sesgo hacia la dimensión de incertidumbre obedece a que los marcadores interrogativos del español —cómo, qué, cuál, por qué— aparecen en la práctica totalidad de las consultas, por lo que dichos marcadores deberían ponderarse o excluirse del vocabulario de activación de esa dimensión.

Se recomienda ampliar el corpus experimental incrementando de forma sustancial la cantidad de diálogos multiturno —actualmente diecisiete de cincuenta y siete consultas— y su longitud, que en este corpus no supera los tres turnos. La capacidad de retención de contexto es la propiedad distintiva del motor y solo resulta observable en secuencias de más de un turno; su evaluación con robustez estadística exige un número considerablemente mayor de estas secuencias.

Se recomienda incorporar al corpus consultas con cambios temáticos deliberados y abruptos, condición necesaria para ejercitar y calibrar el detector de quiebre temático que no pudo verificarse en la presente campaña.

5.2.3 Recomendaciones para trabajos futuros

Se recomienda investigar mecanismos de condicionamiento del modelo de lenguaje más directos que la inyección de una directiva textual en el prompt. La evaluación de coherencia (sección 4.2.4) arrojó un efecto nulo, lo que es consistente con la interpretación de que el canal textual resulta insuficiente para transmitir la información de estado al modelo generador; se recomienda orientar la investigación

futura hacia el ajuste dinámico de parámetros de decodificación en función del estado semántico.

Se recomienda evaluar la calibración adaptativa de los hiperparámetros del motor en función del tipo de conversación, en lugar de emplear valores fijos. Los resultados por categoría temática del Capítulo 4 sugieren que el comportamiento del sistema difiere entre dominios técnicos y dominios abiertos, lo que justifica explorar una configuración dependiente del contexto.

Se recomienda replicar el estudio con otros Small Language Models y sobre configuraciones de hardware distintas, dado que la totalidad de los resultados corresponde a una única combinación de modelo, equipo e idioma.

Se recomienda evaluar la extensión del sistema a otras lenguas, con particular atención a las lenguas originarias del Ecuador. La operación local y de bajo costo resulta especialmente pertinente para comunidades lingüísticas cuyo volumen de hablantes no justifica comercialmente el desarrollo de asistentes conversacionales por parte de los proveedores de servicios en la nube. Debe advertirse, no obstante, que el codificador semántico actual depende de listas de palabras clave en español y requeriría una reconstrucción completa para cada lengua adicional.

Se recomienda comparar el sistema con arquitecturas de generación aumentada por recuperación operando igualmente en local, a fin de determinar si el motor de contexto vectorial aporta ventajas frente a la recuperación explícita de fragmentos de conversación previos.

5.3 Presupuesto ejecutado

La selección de herramientas respondió a un criterio explícito de sostenibilidad, ejecución local y reproducibilidad: se priorizó software de código abierto ejecutable en hardware ya disponible, con el objeto de eliminar costos de nube y de unidad de procesamiento gráfico, reducir la huella ambiental y garantizar la privacidad de los

datos. El presupuesto ejecutado, expresado en dólares de los Estados Unidos, es el siguiente:

Tabla 5.1. Presupuesto de recursos para el desarrollo y la experimentación

Descripción	Detalle	Costo (USD)
Software	Python, scikit-learn, Ollama, qwen2.5, FastAPI, CodeCarbon (código abierto)	0,00
Equipo de cómputo	Equipo doméstico propio (CPU, sin GPU)	0,00
Energía eléctrica	Consumo durante el desarrollo y los experimentos (3 meses)	15,00
Acceso a bibliografía y conectividad	Consulta de artículos indexados y recursos	20,00
Impresión y empastado	Documento final para entrega	40,00
TOTAL		75,00

A título ilustrativo del orden de magnitud de los costos operativos, la campaña experimental completa —ciento sesenta generaciones y cuarenta y siete minutos de cómputo— consumió aproximadamente 0,0106 kWh, cifra derivada de las emisiones estimadas y del factor de emisión nacional. El costo eléctrico directo de la fase experimental resulta, por tanto, despreciable frente a los demás rubros; el valor consignado en la tabla corresponde al consumo del equipo durante los tres meses completos de desarrollo.

5.4 Referencias bibliográficas

Referencias bajo normas APA 7.ª edición, con identificador digital de objeto o enlace oficial del editor en cada caso.

Asamblea Nacional del Ecuador. (2021, 26 de mayo). *Ley Orgánica de Protección de Datos Personales*. Registro Oficial Suplemento N.º 459.

Belcak, P., Heinrich, G., Diao, S., Fu, Y., Dong, X., Muralidharan, S., Lin, Y. C., & Molchanov, P. (2025). *Small language models are the future of agentic AI*. arXiv. https://arxiv.org/abs/2506.02153

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 610–623. https://doi.org/10.1145/3442188.3445922

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*(1), 37–46. https://doi.org/10.1177/001316446002000104

Cohen, J. (1968). Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological Bulletin, 70*(4), 213–220. https://doi.org/10.1037/h0026256

Constitución de la República del Ecuador. (2008, 20 de octubre). Registro Oficial N.º 449.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT 2019*, 4171–4186. https://doi.org/10.18653/v1/N19-1423

Garcez, A. d'A., & Lamb, L. C. (2023). Neurosymbolic AI: The 3rd wave. *Artificial Intelligence Review, 56*(11), 12387–12406. https://doi.org/10.1007/s10462-023-10448-w

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly, 28*(1), 75–105. https://doi.org/10.2307/25148625

Hinton, G., Vinyals, O., & Dean, J. (2015). *Distilling the knowledge in a neural network*. arXiv. https://arxiv.org/abs/1503.02531

Jiang, P., Sonne, C., Li, W., You, F., & You, S. (2024). Preventing the immense increase in the life-cycle energy and carbon footprints of LLM-powered

intelligent chatbots. *Engineering, 40*, 202–210. https://doi.org/10.1016/j.eng.2024.04.002

Lacoste, A., Luccioni, A., Schmidt, V., & Dandres, T. (2019). *Quantifying the carbon emissions of machine learning*. arXiv. https://arxiv.org/abs/1910.09700

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*(1), 159–174. https://doi.org/10.2307/2529310

Lu, Z., Li, X., Cai, D., Yi, R., Liu, F., Zhang, X., Lane, N. D., & Xu, M. (2024). *Small language models: Survey, measurements, and insights*. arXiv. https://arxiv.org/abs/2409.15790

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space*. Proceedings of the International Conference on Learning Representations (ICLR) Workshop. https://arxiv.org/abs/1301.3781

Ministerio de Ambiente y Energía del Ecuador & Operador Nacional de Electricidad [CENACE]. (2025). Factor de emisión de CO_2 del Sistema Nacional Interconectado de Ecuador – Informe 2024. https://arconel.gob.ec/wp-content/uploads/downloads/2025/09/Factor-de-emision-de-CO2-del-Sistema-Nacional-Interconectado-de-Ecuador-Informe-2024_compressed.pdf

Panahi, A., Saeedi, S., & Arodz, T. (2020). *word2ket: Space-efficient word embeddings inspired by quantum entanglement*. International Conference on Learning Representations (ICLR). https://arxiv.org/abs/1911.04975

Patterson, D., Gonzalez, J., Le, Q., Liang, C., Munguia, L. M., Rothchild, D., So, D., Texier, M., & Dean, J. (2021). *Carbon emissions and large neural network training*. arXiv. https://arxiv.org/abs/2104.10350

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-

learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825–2830. https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems, 24*(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. *Proceedings of EMNLP 2014*, 1532–1543. https://doi.org/10.3115/v1/D14-1162

Presidencia de la República del Ecuador. (2023, 13 de noviembre). *Reglamento General a la Ley Orgánica de Protección de Datos Personales* (Decreto Ejecutivo N.º 904). Registro Oficial Tercer Suplemento N.º 435.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. *Proceedings of EMNLP-IJCNLP 2019*, 3982–3992. https://doi.org/10.18653/v1/D19-1410

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). *DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter*. arXiv. https://arxiv.org/abs/1910.01108

Scherer, M., Macan, L., Jung, V. J. B., Wiese, P., Bompani, L., Burrello, A., Conti, F., & Benini, L. (2024). Deeploy: Enabling energy-efficient deployment of small language models on heterogeneous microcontrollers. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 43*(11), 4009–4020. https://doi.org/10.1109/TCAD.2024.3443718

Schmidt, V., Goyal, K., Joshi, A., Feld, B., Conell, L., Laskaris, N., Blank, D., Wilson, J., Friedler, S., & Luccioni, S. (2021). CodeCarbon: Estimate and track carbon emissions from machine learning computing [Software]. Zenodo. https://doi.org/10.5281/zenodo.4658424

Schöbel, S., Schmitt, A., Benner, D., Saqr, M., Janson, A., & Leimeister, J. M. (2024). Charting the evolution and future of conversational agents: A research agenda along five waves and new frontiers. *Information Systems Frontiers, 26*(2), 729–754. https://doi.org/10.1007/s10796-023-10375-9

Schwartz, R., Dodge, J., Smith, N. A., & Etzioni, O. (2020). Green AI. *Communications of the ACM, 63*(12), 54–63. https://doi.org/10.1145/3381831

Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3645–3650. https://doi.org/10.18653/v1/P19-1355

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30*. https://doi.org/10.48550/arXiv.1706.03762

Wang, F., Zhang, Z., Zhang, X., Wu, Z., Mo, T., Lu, Q., Wang, W., Li, R., Xu, J., Tang, X., He, Q., Ma, Y., Huang, M., & Wang, S. (2025). A comprehensive survey of small language models in the era of large language models: Techniques, enhancements, applications, collaboration with LLMs, and trustworthiness. *ACM Transactions on Intelligent Systems and Technology, 16*(6), 1–87. https://doi.org/10.1145/3768165

Weidinger, L., Mellor, J., Rauh, M., Griffin, C., Uesato, J., Huang, P.-S., Cheng, M., Glaese, M., Balle, B., Kasirzadeh, A., Kenton, Z., Brown, S., Hawkins, W., Stepleton, T., Biles, C., Birhane, A., Haas, J., Rimell, L., Hendricks, L. A., … Gabriel, I. (2021). *Ethical and social risks of harm from language models*. arXiv. https://arxiv.org/abs/2112.04359

Widdows, D., Alexander, A., Zhu, D., Zimmerman, C., & Majumder, A. (2024). Near-term advances in quantum natural language processing. *Annals of Mathematics and Artificial Intelligence, 92*(5), 1249–1272. https://doi.org/10.1007/s10472-024-09940-y

Yu, Y. (2025). *Hybrid-Code: A privacy-preserving, redundant multi-agent framework for reliable local clinical coding*. arXiv. https://doi.org/10.48550/arXiv.2512.23743

Zhang, P., Zeng, G., Wang, T., & Lu, W. (2024). *TinyLlama: An open-source small language model*. arXiv. https://arxiv.org/abs/2401.02385

5.5 Anexos

Anexo A. Formalización matemática del motor de contexto. Se desarrolla íntegramente a continuación.

Anexo B. Rúbrica de evaluación de coherencia conversacional. Instrumento de cinco criterios en escala Likert de uno a cinco, con descripción de cada nivel: pertinencia respecto de la consulta, continuidad temática con los turnos previos, corrección del contenido, adecuación de la extensión y naturalidad del registro. El instrumento se presenta íntegramente en el Anexo B; su aplicación sobre los ochenta pares de respuestas se completó conforme al protocolo declarado en la sección 4.2.4, con los resultados reportados en las Tablas 4.17 y 4.18.

Anexo C. Documentación complementaria. Código fuente completo del sistema, capturas del dashboard de visualización en tiempo real y registro de trazas de conversación empleadas durante el desarrollo.

Anexo D. Material de reproducibilidad. Texto literal de las directivas de generación, parámetros de decodificación del modelo de lenguaje y enlace al material experimental complementario.

ANEXO A. FORMALIZACIÓN MATEMÁTICA DEL MOTOR DE CONTEXTO

Este anexo presenta la formalización matemática completa del motor de contexto quantum-inspired descrito conceptualmente en el Capítulo 2 e implementado según lo expuesto en el Capítulo 3. Los valores de los parámetros consignados corresponden a la configuración nominal empleada en la campaña experimental del Capítulo 4.

A.1 Vector de estado semántico

Cada mensaje del usuario se codifica como un vector real no negativo de doce dimensiones, una por cada categoría semántica descrita en el Capítulo 2. Sea $\psi_raw \in \mathbb{R}^{12}$ el vector de activaciones brutas, obtenido a partir de coincidencias de palabras clave y de la distribución completa de probabilidad del clasificador de intención. El vector se normaliza a norma euclidiana unitaria:

$$\psi = \psi_raw / \|\psi_raw\|_2 \text{ , con } \|\psi\|_2 = 1$$

Se emplea la distribución completa del clasificador y no únicamente la etiqueta de máxima probabilidad. Esta decisión de diseño tiene una consecuencia relevante a la luz de los resultados del Capítulo 4: dado que la confianza media del clasificador fue de 0,4881, la incertidumbre de la clasificación se propaga como dispersión del vector de estado en lugar de traducirse en un error categórico.

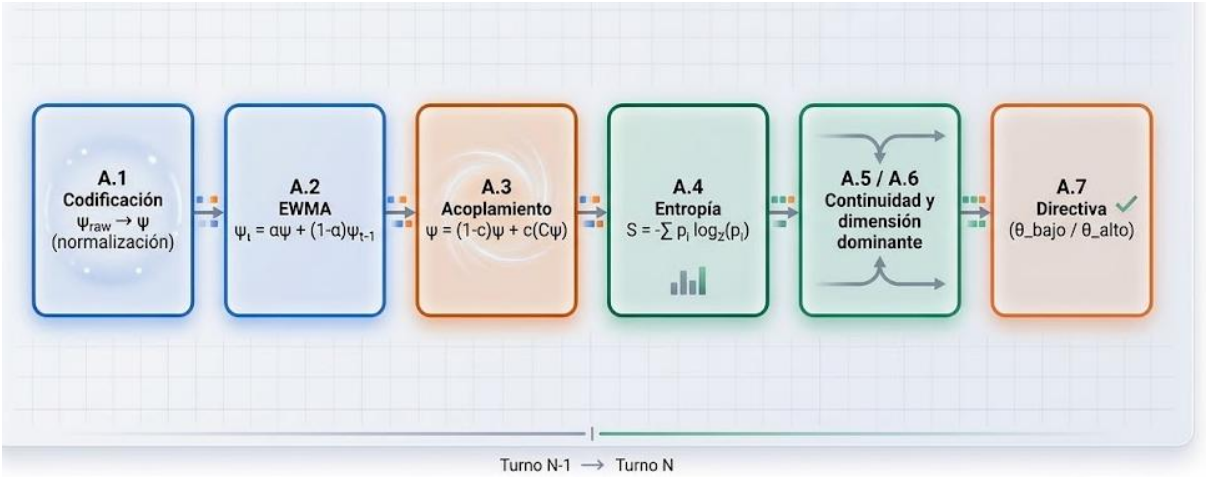


Figura A.1. Secuencia de cómputo del motor de contexto en cada turno, correspondiente a las ecuaciones A.1 a A.7 del Anexo A.

A.2 Actualización por promedio móvil exponencial

El contexto se actualiza turno a turno combinando el vector del mensaje actual con el estado previo mediante un promedio móvil exponencial de factor α:

ψ_t = α·ψ_input + (1 − α)·ψ_(t−1) , α = 0,30

El resultado se renormaliza a norma unitaria tras la combinación. En el primer turno de cada sesión no existe estado previo, por lo que ψ_t = ψ_input por definición.

A.3 Matriz de acoplamiento semántico

Para modelar correlaciones lingüísticas entre categorías —por ejemplo, entre lógica y análisis— se aplica una matriz de acoplamiento simétrica C ∈ ℝ^(12×12), con C_ii = 1 y coeficientes C_ij ∈ [0,1] en las posiciones correspondientes a dimensiones relacionadas. El estado se combina con su proyección acoplada mediante un coeficiente de fuerza de acoplamiento c:

ψ = (1 − c)·ψ + c·(C·ψ) , c = 0,50

seguido de una renormalización a norma unitaria.

El acoplamiento produce activación indirecta de dimensiones no invocadas directamente por el léxico del mensaje. Este efecto explica un resultado documentado en el Capítulo 4: la dimensión social resultó dominante en el 33,8 % de los turnos pese a no ser activada léxicamente en esa proporción, por recibir activación indirecta desde emoción (C = 0,65) y desde conexión (C = 0,50).

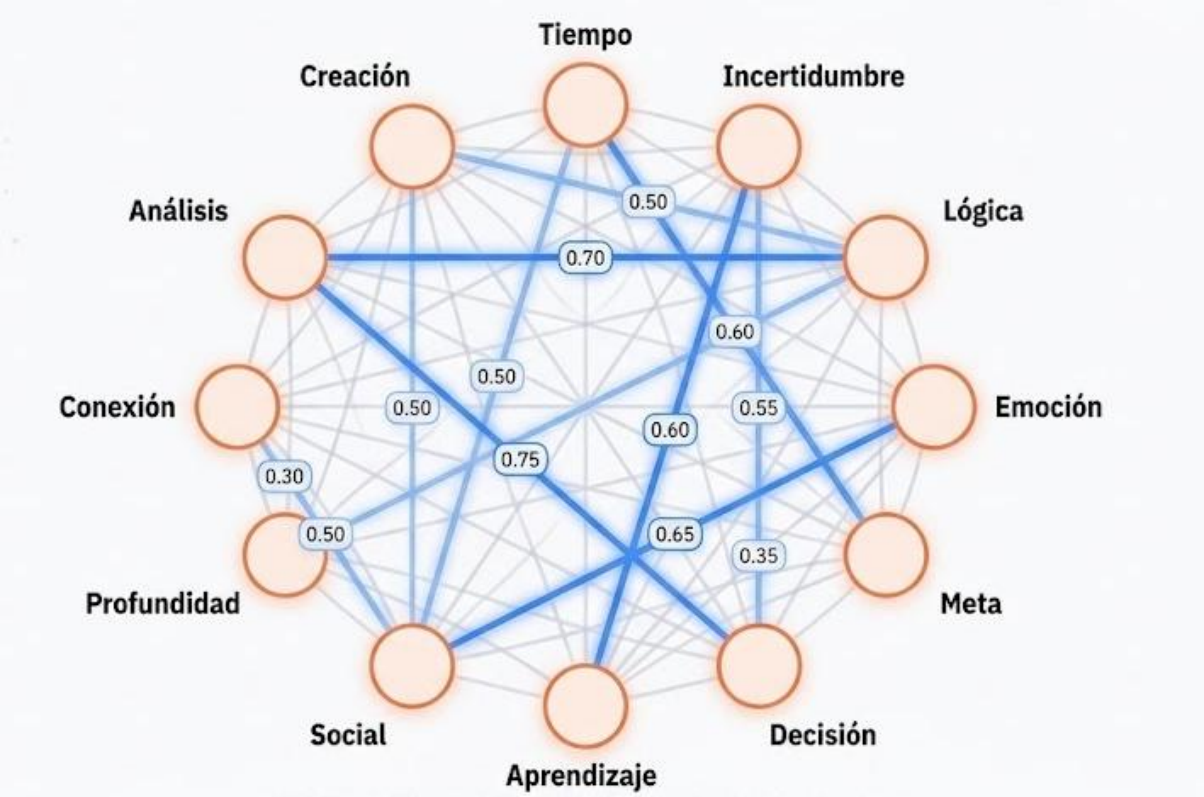


Figura A.2. Grafo de la matriz de acoplamiento semántico C entre las doce dimensiones del vector de estado, con el peso de cada uno de los diez enlaces definidos en COUPLING_LINKS

A.4 Distribución de probabilidad y entropía de dispersión

A partir del vector de estado normalizado se deriva una distribución de probabilidad sobre las doce categorías:

$$p_i = \psi_i^2 / \Sigma_j \psi_j^2 \,, \; i = 1,\ldots,12$$

La dispersión del contexto se cuantifica mediante una entropía de Shannon normalizada al intervalo [0,1] por división entre $\log_2(12)$:

$$S = -\Sigma_i \, q_i \cdot \log_2(q_i) \,/\, \log_2(12)$$

La entropía se calcula sobre la distribución q derivada del vector de entrada normalizado del turno, $\varphi = \psi_{raw} / \|\psi_{raw}\|_2$, con $q_i = \varphi_i^2 / \Sigma_j \varphi_j^2$, antes de las operaciones A.2 y A.3, tal como implementa el Código C.4. La magnitud reportada describe, por tanto, la dispersión del mensaje actual y no la del contexto acumulado. La distribución p del estado se emplea en la selección de la dimensión dominante (A.6).

A.5 Continuidad conversacional

La continuidad entre turnos se mide mediante la similitud coseno entre el vector de entrada del turno actual, φ_t, y el vector de estado acumulado hasta el turno anterior, ψ_(t−1). Al estar ambos normalizados a norma unitaria, esta se reduce al producto interno:

$$\text{continuidad}_t = \varphi_t \cdot \psi_{(t-1)} = \sum_i \varphi_{t,i} \cdot \psi_{(t-1),i}$$

Valores altos indican continuidad temática; valores bajos, cambio de tema. Esta magnitud corresponde a la que la sección 4.3.2 denomina fidelidad del estímulo; por ello las columnas de continuidad y fidelidad de la Tabla 4.5 son idénticas. La continuidad del estado en sentido estricto, ψ_t · ψ_(t−1), se calcula en el código (variable state_continuity), pero no interviene en ninguna decisión del motor y solo se analiza en el estudio de sensibilidad de la sección 4.3.

A.6 Selección de la dimensión dominante

La categoría dominante del turno se selecciona de forma determinista como la de mayor probabilidad en la distribución p derivada en A.4, y su nombre se incorpora a la directiva de generación como foco central de la respuesta:

$$\text{categoría_dominante} = \operatorname{argmax}_i \, p_i$$

Esta operación es determinista: dos ejecuciones sobre la misma entrada seleccionan la misma dimensión dominante. Las magnitudes empleadas en las decisiones de la sección A.7 —entropía y continuidad— son igualmente deterministas. El único componente estocástico del sistema es la decodificación del modelo de lenguaje, cuyos parámetros se documentan en el Anexo D.

A.7 Selección de la estrategia de respuesta

La directiva que se transmite a la capa de generación se determina de forma determinista a partir de la entropía calculada en A.4, mediante dos umbrales:

Tabla A.1. Reglas de selección de directiva según la entropía normalizada del estado

Condición	Directiva	Estructura del estado
S < θ_bajo	Directa	Resolución directa
θ_bajo ≤ S < θ_alto	Equilibrada	Perspectiva dual
S ≥ θ_alto	Exploratoria	Perspectiva múltiple

con **θ_bajo = 0,66** y **θ_alto = 0,81**.

La verificación empírica de esta regla sobre los ochenta turnos de la campaña experimental arrojó rangos disjuntos y sin solapamiento: directa en [0,4584; 0,6562], equilibrada en [0,6638; 0,8016] y exploratoria en [0,8139; 0,9246].

A.8 Detección de quiebre temático

Se declara un quiebre temático cuando la continuidad calculada en A.5 desciende por debajo de un umbral θ_quiebre:

quiebre = 1 si continuidad_t < θ_quiebre , con θ_quiebre = 0,50

Esta funcionalidad no se activó en ninguno de los ochenta turnos de la campaña experimental, por lo que permanece sin verificación empírica y se documenta como característica paramétrica pendiente de calibración, conforme se consigna en las secciones 4.1.4 y 5.1.2.

A.9 Nota sobre la terminología

Todas las operaciones descritas en este anexo se implementan mediante álgebra lineal clásica sobre vectores reales, utilizando la biblioteca numpy. Ninguna requiere hardware cuántico ni simulación de circuitos cuánticos. La denominación quantum-inspired alude exclusivamente a la analogía conceptual explicada en el Capítulo 2: la representación del contexto como vector normalizado, la derivación de una distribución de probabilidad a partir del cuadrado de sus componentes y el uso de la entropía como medida de dispersión guardan correspondencia formal con el formalismo de los estados cuánticos, pero no implican fenómeno cuántico alguno. Esta distinción es sostenida de manera consistente en la totalidad del documento.

ANEXO B. RÚBRICA DE EVALUACIÓN DE COHERENCIA CONVERSACIONAL

Instrucciones para el evaluador. Se presentan pares de respuestas producidas por dos configuraciones distintas de un mismo sistema conversacional ante idéntica consulta. El orden de presentación dentro de cada par es aleatorio y la configuración de origen permanece oculta. Debe puntuarse cada respuesta de forma independiente en los cinco criterios que siguen, empleando una escala de uno a cinco puntos. No debe compararse una respuesta con la otra: cada una se evalúa contra los descriptores de la rúbrica. En los turnos pertenecientes a diálogos de más de un turno se proporciona el historial previo, necesario para valorar el criterio de continuidad temática.

Criterio 1. Pertinencia respecto de la consulta

Tabla B.1. Rúbrica de pertinencia respecto de la consulta

Puntos	Descriptor
5	Responde de forma completa y directa a lo preguntado, sin desviaciones
4	Responde a lo preguntado, con alguna digresión menor que no interfiere
3	Responde parcialmente; omite algún aspecto de la consulta o añade contenido no solicitado
2	Aborda el tema general pero no responde a la pregunta concreta formulada
1	No guarda relación identificable con la consulta

Criterio 2. Continuidad temática con los turnos previos

Tabla B.2. Rúbrica de continuidad temática con el contexto previo

Puntos	Descriptor
5	Integra el contexto de turnos anteriores de forma explícita y pertinente
4	Mantiene el hilo temático sin referencia explícita a lo previo
3	Es compatible con el contexto previo pero podría leerse como turno aislado
2	Ignora el contexto previo y provoca una discontinuidad perceptible
1	Contradice o desconoce información establecida en turnos anteriores

Puntos	Descriptor
N/A	Aplíquese en consultas de turno único, donde no existe contexto previo

Criterio 3. Corrección del contenido

Tabla B.3. Rúbrica de corrección factual de la respuesta

Puntos	Descriptor
5	La información es verificablemente correcta en su totalidad
4	Correcta en lo sustancial, con alguna imprecisión menor
3	Mayoritariamente correcta, con un error identificable que no invalida la respuesta
2	Contiene errores relevantes que comprometen su utilidad
1	Contiene afirmaciones falsas o información fabricada
N/A	Aplíquese cuando la consulta no admite verificación factual

Criterio 4. Adecuación de la extensión

Tabla B.4. Rúbrica de proporcionalidad de la extensión de la respuesta

Puntos	Descriptor
5	La extensión es proporcional a la complejidad de la consulta
4	Ligeramente extensa o breve, sin afectar a la comprensión
3	Perceptiblemente desproporcionada, aunque la respuesta sigue siendo utilizable
2	Excesivamente extensa con contenido redundante, o insuficiente para responder
1	La extensión impide el uso de la respuesta

Criterio 5. Naturalidad del registro

Tabla B.5. Rúbrica de naturalidad y adecuación lingüística en español

Puntos	Descriptor
5	Fluida y natural en español, con registro apropiado a la consulta
4	Natural, con alguna construcción rígida o artificiosa

Puntos	Descriptor
3	Comprensible pero con formulaciones perceptiblemente mecánicas
2	Construcciones forzadas, calcos de otra lengua o registro inadecuado
1	Deficiente en gramática o léxico hasta dificultar la lectura

Procesamiento de las puntuaciones

La puntuación agregada de cada respuesta es la media aritmética de los criterios aplicables, excluyendo los marcados como no aplicables. La concordancia entre los tres evaluadores se calcula mediante kappa ponderado con pesos cuadráticos (Cohen, 1968), apropiado para escalas ordinales, fijándose $\kappa_w > 0{,}60$ como criterio de aceptación conforme a la escala de Landis y Koch (1977). El contraste entre condiciones se realiza mediante la prueba de Wilcoxon de rangos con signo sobre las puntuaciones agregadas pareadas.

ANEXO C. DOCUMENTACIÓN COMPLEMENTARIA

Este anexo documenta, mediante capturas de pantalla del Lab Dashboard en ejecución y los fragmentos de código fuente que las generan, el funcionamiento del panel de visualización desarrollado como cuarto objetivo específico del proyecto. El dashboard se sirve sobre la interfaz de bucle local (127.0.0.1) con selección automática de puerto disponible, iniciando por el 8080, por lo que el puerto visible en cada captura puede variar entre sesiones sin que ello afecte al funcionamiento. Las capturas corresponden a turnos ilustrativos ejecutados en vivo con la versión final del artefacto y tienen carácter demostrativo: no forman parte de los 80 turnos de la campaña experimental reportada en el Capítulo 4 ni de sus estadísticos. Los fragmentos de código se reproducen literalmente desde el archivo fuente, por lo que conservan algunos identificadores internos heredados de la etapa de desarrollo —por ejemplo, colapso_directo, superposicion_alta, entangled_names o multiverse_states—. Se trata de nombres de variables y claves sin efecto sobre el cálculo; la interfaz los presenta al usuario con la terminología adoptada en este documento: las claves colapso_directo y superposicion_alta se muestran como modalidades «resolución directa» y «exploración multidimensional», mientras que entangled_names y multiverse_states alimentan los bloques «pares acoplados» y «dimensiones dominantes». Las denominaciones «perspectiva dual» y «perspectiva múltiple» de la Tabla A.1 corresponden, en cambio, a la estructura de la directiva generativa (variable structure_label), conforme a la nota del apartado A.9.

C.1 Vista general del panel de control

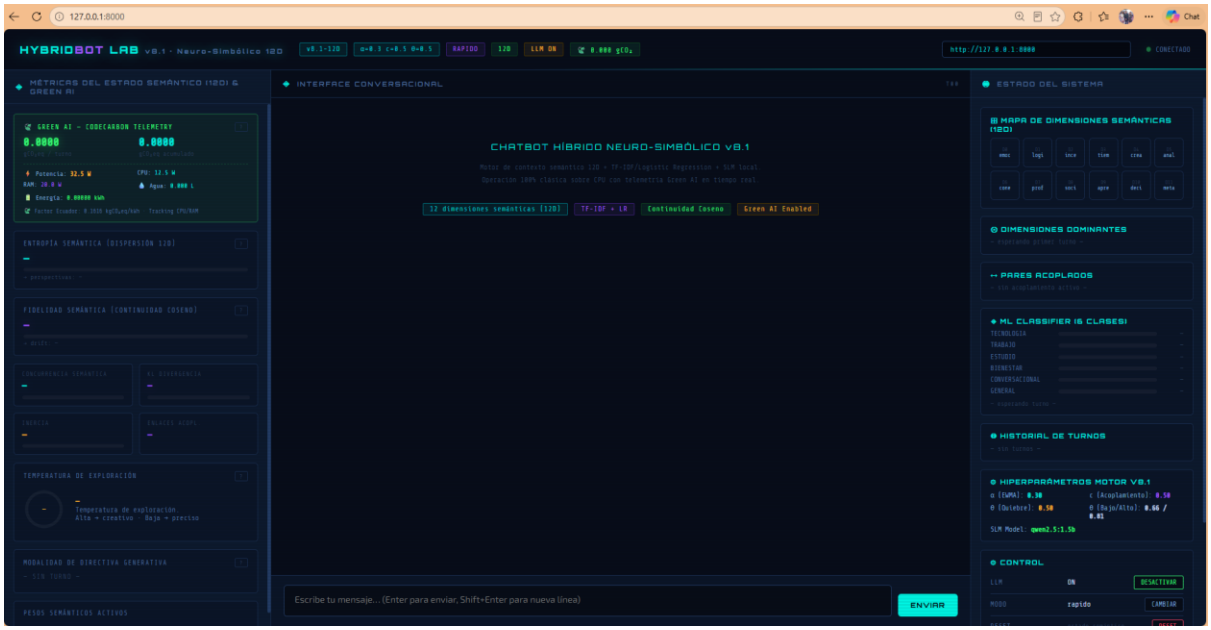


Figura C.1. Vista general del Lab Dashboard mostrando los tres paneles —métricas del estado semántico, interfaz conversacional y estado del sistema— en su estado inicial, previo al primer turno.

Código C.1 — Esqueleto HTML del dashboard (quantum.py, líneas 1658–1679, 1935)

```
<header>
  <div class="logo">HYBRID<span>BOT</span> LAB ...</div>
  <div class="header-tags">
    <span class="tag tag-c" id="tag-params">α=0.3 c=0.5 θ=0.5</span>
    <span class="tag tag-p" id="tag-mode">RAPIDO</span>
    <span class="tag tag-g" id="tag-dims">12D</span>
    <span class="tag tag-a" id="tag-llm">LLM OFF</span>
    <span class="tag tag-g" id="tag-green">gCO2</span>
  </div>
  <div class="header-status">
    <div class="dot" id="conn-dot"></div>
    <span id="conn-label">DESCONECTADO</span>
  </div>
</header>
<div class="main">
  <div class="panel">      <!-- izquierda: metricas del estado semantico --> </div>
  <div class="chat-panel"> <!-- centro: conversacion       --> </div>
```

```
  <div class="panel">      <!-- derecha: estado del sistema --> </div>
</div>
```

Código C.2 — Orquestador que produce el estado mostrado en los tres paneles (quantum.py, líneas 723–877)

```
def respond(self, user_input: str) -> dict:
    label, conf, probs = self.classifier.predict(user_input)        # Capa 1
    psi_raw, coverage = self.encoder.encode(user_input, probs)      # Capa 2A
    res = self.engine.step(psi_raw)                                 # Capa 2B
    cs = ContextSignal(turn_number=self.engine.turn, ...)           # empaqueta metricas
    cs = self.director.direct(cs)                                   # directiva
generativa
    text = self._call_ollama(messages, options) or \
           self._generate_classical_fallback(user_input, cs)        # Capa 3
    ...
    return {"response": text, ..., "quantum_signal": asdict(cs)}
```

C.2 Panel izquierdo — Métricas del estado semántico 12D & Green AI



Figura C.2. Panel de métricas del estado semántico y telemetría CodeCarbon: entropía de Shannon normalizada, fidelidad semántica, concurrencia, divergencia KL, inercia, temperatura de exploración y modalidad de directiva del turno.

Código C.3 — Marcado HTML del panel (quantum.py, líneas 1681–1809)

```
<div class="panel-header"><span class="ph-icon">◈</span> MÉTRICAS DEL ESTADO SEMÁNTICO
(12D) & GREEN AI</div>
<div class="metric-card"><div class="mc-title">GREEN AI - CODECARBON TELEMETRY</div>
  <div class="gv-green gauge-value" id="val-co2-turn">0.0000</div>
  <div class="gauge-value" id="val-co2-total">0.0000</div>
```

```
  Potencia: <b id="val-watts-total"></b>  CPU: <span id="val-watts-cpu"></span>
  RAM: <span id="val-watts-ram"></span>   Agua: <span id="val-water-l"></span>
  Energia: <span id="val-kwh-total"></span>
</div>
<div class="metric-card"><div class="mc-title">ENTROPÍA SEMÁNTICA (Dispersión 12D)</div>
  <div class="gv-cyan gauge-value" id="val-entropy">-</div></div>
<div class="metric-card"><div class="mc-title">FIDELIDAD SEMÁNTICA (Continuidad
Coseno)</div>
  <div class="gv-purple gauge-value" id="val-fidelity">-</div></div>
<div class="mini-metric">CONCURRENCIA SEMÁNTICA <span id="val-conc"></span></div>
<div class="mini-metric">KL DIVERGENCIA <span id="val-kl"></span></div>
<div class="mini-metric">INERCIA <span id="val-inertia"></span></div>
<div class="mini-metric">ENLACES ACOPL. <span id="val-gates"></span></div>
<div class="metric-card"><div class="mc-title">TEMPERATURA DE EXPLORACIÓN</div>
  <svg><circle id="anneal-circle" .../></svg></div>
<div class="collapse-display" id="collapse-display">
  <div class="mc-title">MODALIDAD DE DIRECTIVA GENERATIVA</div>
  <div class="cd-type" id="collapse-type">- sin turno -</div></div>
```

Código C.4 — Cálculo de entropía, fidelidad, concurrencia, KL, inercia y temperatura de exploración (ContextEngine.step, líneas 451–552)

```
def step(self, psi_raw: np.ndarray) -> dict:
    phi = self._norm(psi_raw)
    continuity = float(np.clip(phi @ self.psi_prev, 0.0, 1.0))       # Fidelidad
    p_input = phi ** 2; p_input = p_input / max(p_input.sum(), 1e-12)
    entropy = float(-(p_input * np.log2(p_input + 1e-12)).sum() / np.log2(N_DIMS))
    psi = self.alpha * phi + (1.0 - self.alpha) * self.psi_prev      # EWMA
    psi = (1.0 - self.coupling) * psi + self.coupling * (self.C @ psi)  # Acoplamiento
    conc = 0.0
    for i, j, w in COUPLING_LINKS:
        prod = np.sqrt(p_state[i] * p_state[j])
```

```
        if prod > 0.02: conc += w * prod
    concurrence = float(np.clip(conc * 2.0, 0.0, 1.0))
    kl = float(np.sum(p_state * np.log2((p_state + 1e-12) / (1.0/N_DIMS))))
    annealing_temp = float(0.95 * np.exp(-0.07 * (self.turn - 1)))
    inertia = min(1.0, self.same_dim_turns / 5.0)
    return {"entropy": entropy, "continuity": continuity, "concurrence": concurrence,
            "kl_divergence": kl, "inertia": inertia, "annealing_temp": annealing_temp,
...}
```

Código C.5 — Selección de la modalidad de directiva mostrada en el panel (ResponseDirector.direct, líneas 573–596)

```
def direct(self, cs: ContextSignal) -> ContextSignal:
    if cs.turn_number == 1: ct = "primer_turno"
    elif cs.break_detected: ct = "ruptura_semantica"
    elif cs.inertia >= 0.8: ct = "decoherencia"
    elif cs.concurrence >= 0.45: ct = "acoplamiento_detectado"
    elif len(cs.constructive_pairs) > 0: ct = "interferencia_constructiva"
    elif h < THETA_BAJO: ct = "colapso_directo"
    elif h > THETA_ALTO: ct = "superposicion_alta"
    else: ct = "colapso_directo"
    cs.collapse_type = ct
```

C.3 Panel derecho — Estado del sistema



Figura C.3. Panel de estado del sistema: mapa de las doce dimensiones semánticas, dimensiones dominantes, pares acoplados, salida del clasificador de intención (6 clases) e historial de turnos.

Código C.6 — Marcado HTML del panel (quantum.py, líneas 1848–1906)

```
<div class="panel-header"><span class="ph-icon">O</span> ESTADO DEL SISTEMA</div>

<div class="edu-title">Mapa de Dimensiones Semánticas (12D)</div>

<div class="qubit-grid" id="qubit-grid"></div>

<div class="edu-title">Dimensiones Dominantes</div>

<div id="mv-states"></div>
```

```
<div class="edu-title">Pares Acoplados</div>

<div id="ent-pairs"></div>

<div class="edu-title">ML Classifier (6 clases)</div>

<div class="ml-bar-wrap" id="ml-bars"></div>

<div id="ml-verdict"></div>

<div class="edu-title">Historial de Turnos</div>

<div class="turn-history" id="turn-history"></div>

<div class="edu-title">Hiperparametros Motor v8.1</div>

  alfa (EWMA): <b id="val-alpha">0.30</b>  c: <b id="val-coupling">0.50</b>  theta: <b

id="val-theta">0.50</b>
```

Código C.7 — Clasificador de intención que llena el bloque ML Classifier (IntentClassifier.predict, líneas 379–392)

```
def predict(self, text: str) -> Tuple[str, float, Dict[str, float]]:

    probs = self.pipeline.predict_proba([self.normalize(text)])[0]

    classes = list(self.pipeline.classes_)

    idx = int(np.argmax(probs))

    return classes[idx], float(probs[idx]), {c: float(p) for c, p in zip(classes, probs)}
```

Código C.8 — Codificación semántica que activa el mapa de dimensiones semánticas (SemanticEncoder.encode, líneas 403–419)

```
def encode(self, text: str, intent_probs: Dict[str, float]):

    clean = IntentClassifier.normalize(text)

    words = set(re.findall(r"[a-z0-9_+#.-]{2,}", clean))

    for i, kws in self._kw.items():

        hits = sum(1 for k in kws if (" " in k and k in clean) or (" " not in k and k in

words))

        coverage[i] = hits / max(len(kws), 1)                      # activacion lexica

        raw[i] = coverage[i]

    for label, p in (intent_probs or {}).items():

        for d in INTENT_TO_DIMS.get(label, []):

            raw[d] += p * ML_PROJECTION_WEIGHT                     # proyeccion del
```

```
clasificador

    return raw, coverage
```

Código C.9 — Dimensiones dominantes y pares acoplados (dentro de respond() y ContextEngine.step, líneas 514–527, 770–772)

```
# Pares acoplados (ContextEngine.step):

for i, j, w in COUPLING_LINKS:

    if p_state[i] > 0.03 and p_state[j] > 0.03:

        entangled_names.append([DIM_SEMANTICS[i][0], DIM_SEMANTICS[j][0]])


# Dimensiones dominantes del panel (variable interna multiverse_states, respond()):

order = np.argsort(-res["probs"])

cs.multiverse_states = [[DIM_SEMANTICS[int(i)][0], round(float(res["probs"][int(i)]), 4)]

                        for i in order[:3]]
```

C.4 Panel central — Interfaz conversacional

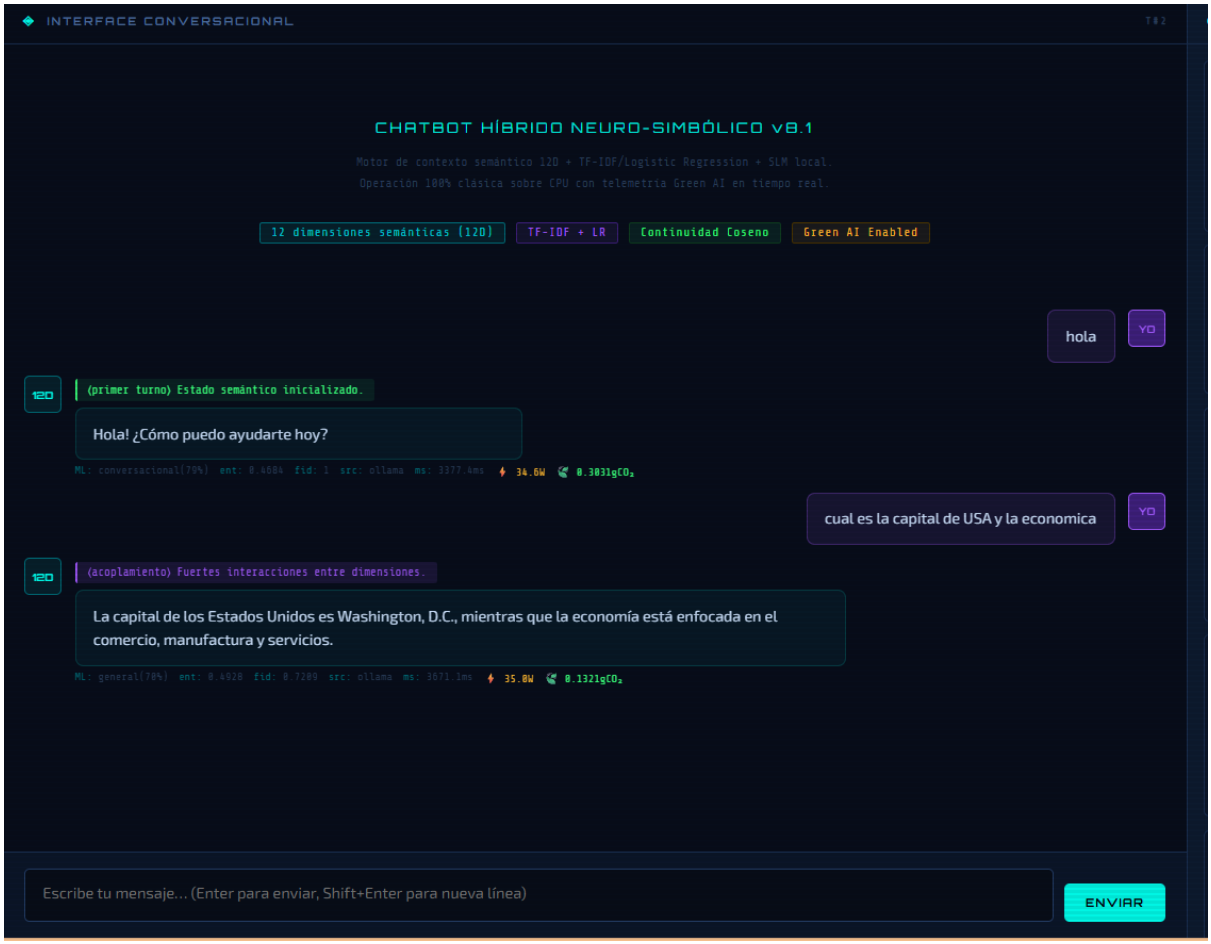


Figura C.4. Interfaz conversacional del dashboard, mostrando el intercambio con el usuario, la etiqueta de la modalidad de directiva del turno y la barra de telemetría (modelo, entropía, fidelidad, fuente de la respuesta, latencia, emisiones).

Código C.10 — Marcado HTML del panel (quantum.py, líneas 1815–1843)

```
<div class="panel-header"><span class="ph-icon">D</span> INTERFACE CONVERSACIONAL
  <span id="turn-counter">T#0</span></div>
<div class="chat-messages" id="chat-messages"></div>
<div class="chat-input-area">
  <textarea class="chat-input" id="chat-input" placeholder="Escribe tu
mensaje..."></textarea>
  <button class="send-btn" id="send-btn" onclick="sendMessage()">ENVIAR</button>
</div>
```

Código C.11 — Generación de la respuesta y memoria de turnos previos, extracto (quantum.py, líneas 684–721, 778–792)

```
def _call_ollama(self, messages, options):
    r = c.post(self.ollama_base_url + "/api/chat", json={"model": self.ollama_model,
        "messages": messages, "stream": False, "options": options})
    return r.json().get("message", {}).get("content", "").strip()


def _generate_classical_fallback(self, user_input, cs):
    # respuestas de respaldo cuando Ollama no esta disponible
    ...


# Historial usado para dar contexto al SLM:
history = []
for u, b in list(self.turn_memory)[-2:]:
    history.append({"role": "user", "content": u[:200]})
    history.append({"role": "assistant", "content": b[:200]})
system = f"Eres un asistente conversacional en espanol. {cs.directive} ..."
```

C.5 Detalle — Telemetría CodeCarbon (Green AI)



Figura C.5. Detalle de la tarjeta de telemetría CodeCarbon del panel de métricas del estado semántico, correspondiente a las variables reportadas en la sección 4.2.3.

Código C.12 — Cálculo de potencia, energía, agua y emisiones por turno (quantum.py, líneas 803–845)

```
ram_w = 20.0

cpu_w = round(min(40.0, max(4.0, 9.5 + min(25.0, exec_sec * 1.5))), 2)

if self.tracker:

    pow_obj = self.tracker._last_measured_power

    if pow_obj.cpu_power: cpu_w = round(float(pow_obj.cpu_power), 2)

    if pow_obj.ram_power: ram_w = round(float(pow_obj.ram_power), 2)

total_w = round(ram_w + cpu_w, 2)

ram_kwh = round((ram_w * exec_sec) / (3600.0 * 1000.0), 8)

cpu_kwh = round((cpu_w * exec_sec) / (3600.0 * 1000.0), 8)

total_kwh = round(ram_kwh + cpu_kwh, 8)

water_l = round(total_kwh * 0.25, 8)

if self.tracker:

    curr_emissions_kg = self.tracker.flush() or 0.0

    emissions_gco2 = round(max(0.0, curr_emissions_kg - self._last_tracker_emissions) *

1000.0, 5)

if emissions_gco2 <= 0:

    emissions_gco2 = round(total_kwh * 161.6, 5)   # fallback con factor Ecuador
```

ANEXO D. MATERIAL DE REPRODUCIBILIDAD

Este anexo documenta los elementos necesarios para reproducir la campaña experimental del Capítulo 4: el texto literal de las directivas que recibió el modelo de lenguaje, los parámetros de generación y el material experimental complementario. Los valores corresponden a la versión 8.1 del archivo quantum.py, documentada en el Anexo C.

D.1 Texto literal de las directivas de generación

En la condición B (sistema completo), el prompt de sistema enviado al modelo tiene la forma «Eres un asistente conversacional en español. [directiva] No inventes librerías ni datos ficticios.». La directiva se compone de una instrucción de estilo, un foco y, cuando corresponde, una indicación de quiebre temático. La condición A (línea base) se describe en la sección 4.2.1.

Directiva directa (entropía normalizada inferior a 0,66): «Responde de forma técnica, directa y concisa.»

Directiva equilibrada (entropía entre 0,66 y 0,81): «Responde de forma clara y bien fundamentada.»

Directiva exploratoria (entropía superior a 0,81): «Responde de forma reflexiva, explorando varios aspectos.»

Foco, añadido en todos los turnos: «Foco central: [dimensión dominante].», donde la dimensión dominante es la de mayor probabilidad en el estado acumulado (Anexo A.6).

Quiebre temático, añadido solo cuando se detecta: «El usuario cambió de tema: concéntrate solo en la nueva pregunta.» Esta indicación no se activó en ningún turno de la campaña.

Historial: los dos turnos previos, con la consulta y la respuesta truncadas a 200 caracteres cada una (Código C.11).

D.2 Parámetros de generación

El modelo qwen2.5:1.5b se invocó mediante la interfaz /api/chat de Ollama con los parámetros de decodificación que fija el código del sistema (Código C.11):

Temperatura: 0,95·e^(−0,07·(t−1)), acotada al intervalo [0,1; 0,9], donde t es el número de turno dentro de cada diálogo; resulta 0,900 en el primer turno, 0,886 en el segundo y 0,826 en el tercero.

top_p: 0,9.

num_predict (longitud máxima de la respuesta): 512 tokens.

num_ctx (ventana de contexto): 2048 tokens.

Semilla: no fijada. La decodificación es, por tanto, estocástica, y una misma consulta puede producir respuestas distintas entre ejecuciones.

D.3 Material experimental complementario

El corpus experimental de 57 consultas y 80 turnos, con su categoría de referencia e identificador de diálogo; las 160 respuestas generadas en ambas condiciones; las calificaciones de los tres jueces por criterio, registradas bajo los identificadores J1, J2 y J3; los registros por turno de latencia, emisiones, entropía y fidelidad; y el código fuente completo se ponen a disposición en el repositorio público del proyecto: https://github.com/leogerm/chatbot-quantum-inspired-slm-sostenible .